



HAL
open science

Amélioration des simulations Monte Carlo en physique des réacteurs par échantillonnage adaptatif des histoires de neutrons

Kévin Fröhlicher

► **To cite this version:**

Kévin Fröhlicher. Amélioration des simulations Monte Carlo en physique des réacteurs par échantillonnage adaptatif des histoires de neutrons. Statistical Mechanics [cond-mat.stat-mech]. Université Paris-Saclay, 2023. English. NNT : 2023UPASP011 . tel-04082033

HAL Id: tel-04082033

<https://theses.hal.science/tel-04082033v1>

Submitted on 26 Apr 2023

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

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

Improving Monte Carlo reactor physics simulations using adaptive sampling of neutron histories

*Amélioration des simulations Monte Carlo en physique
des réacteurs par échantillonnage adaptatif des
histoires de neutrons*

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 576, Particules Hadrons Énergie et Noyau : Instrumentation,
Image, Cosmos et Simulation (PHENIICS)
Spécialité de doctorat : Sciences de l'énergie nucléaire
Graduate School : Physique, Référent : Faculté des sciences d'Orsay

Thèse préparée dans le Laboratoire de Neutronique (LN), Institut de Radioprotection et de
Sûreté Nucléaire (IRSN), sous la direction de Eric DUMONTEIL, directeur de recherche, la
co-direction de Alain HÉBERT, professeur, le co-encadrement de Julien Taforeau,
ingénieur-chercheur, et Mariya Brovchenko, ingénieure-chercheure

Thèse soutenue à Paris-Saclay, le 01 février 2023, par

Kévin FRÖHLICHER

Composition du jury

Membres du jury avec voix délibérative

Tony LELIÈVRE Professeur, CERMICS, Ecole des Ponts ParisTech	Président
Elsa MERLE Professeure des Universités, Grenoble INP/PHELMA	Rapporteuse & Examinatrice
Eugene SHWAGERAUS Professeur, University of Cambridge	Rapporteur & Examineur
Cheikh M'BACKÉ DIOP Directeur de recherche, CEA/ISAS	Examineur
Margaux FAUCHER Ingénieure-chercheure, EDF Lab Paris-Saclay	Examinatrice

Titre : Amélioration des simulations Monte Carlo en physique des réacteurs par échantillonnage adaptatif des histoires de neutrons

Mots clés : neutronique, équation de Boltzmann dépendante du temps, Monte Carlo, réduction de variance, transitoire de puissance, flux adjoint temporel

Résumé : L'Adaptive Multilevel Splitting (AMS) est une méthode de réduction de variance précédemment mise en œuvre pour le transport des neutrons dans les calculs de radioprotection afin d'améliorer l'efficacité des calculs pour la modélisation d'événements rares. Cette thèse vise à présenter une extension du champ d'application de cette technique aux problèmes de physique des réacteurs en régime permanent et en cinétique/dynamique. En ce qui concerne les calculs en régime permanent ou de criticité, l'algorithme inspiré de la méthode d'itération sur la puissance et sur lequel ces calculs sont généralement basés peut présenter des problèmes. En effet, la normalisation de la population effectuée entre les générations amplifie les fortes corrélations spatiales et générationnelles déjà induites par le processus de naissance et de mort (processus de Galton-Watson) dans le transport des neutrons en milieu multiplicatif. Dans la plupart des situations pathologiques, ces corrélations peuvent provoquer la formation de clusters de neutrons, dépeuplant ainsi certaines parties du système et entraînant des problèmes de convergence. Dans cette thèse, l'itération sur la puissance classique a été refondue en un problème de réduction de variance pour lequel l'AMS a été utilisé afin de résoudre ce problème de clustering. Dans le but de réaliser une preuve de principe, l'AMS a été implémentée dans un code Monte Carlo maquette entièrement développé pendant ce doctorat, pouvant modéliser des problèmes de criticité/statiques avec une modélisation physique simplifiée. La méthode a été caractérisée sur un problème homogène

avec des neutrons mono-énergétiques, et les résultats obtenus se sont avérés encourageants pour ce qui est de la réduction des corrélations indésirables dans les calculs de criticité. Une fois testée avec succès dans la maquette, l'AMS a donc été implémentée dans le code Monte Carlo SERPENT2 afin d'étendre son utilisation aux problèmes de cinétique et de dynamique. En effet, malgré les récents développements en matière de réduction de variance, les coûts de calcul associés à ce type de calculs demeurent très élevés. A cet égard, il a été envisagé de s'appuyer sur l'utilisation d'une carte d'importance par l'AMS pour réduire la variance associée à l'estimation des scores locaux lors des transitoires, améliorant ainsi l'efficacité des calculs de cinétique. Pour ce faire, différentes combinaisons de méthodes ont été testées lors de l'implémentation de l'AMS dans SERPENT2. La caractérisation de l'AMS en cinétique a été principalement faite sur des cas simples tels que des compositions homogènes représentatives d'assemblages de Réacteur à Eau sous Pression (REP), ou des problèmes hétérogènes avec neutrons mono-énergétiques. La méthode a ensuite été appliquée à un transitoire opéré sur un groupe de 3x3 assemblages. Une carte d'importance dépendante du temps a été extraite des équations de cinétique ponctuelle adjointes à cette fin. L'objectif étant, avec le temps, de s'appuyer sur des cartes d'importance dépendantes de l'espace et du temps obtenues à partir de la cinétique espace-temps adjointe pour les neutrons et les précurseurs via des méthodes déterministes.

Title : Improving Monte Carlo reactor physics simulations using adaptive sampling of neutron histories
Keywords : neutron transport, kinetic Boltzmann equation, Monte Carlo, variance reduction, power transient, time-dependent adjoint flux

Abstract : Adaptive Multilevel Splitting (AMS) is a variance reduction method previously implemented for neutron transport in shielding computation applications to perform variance reduction for rare events. This thesis aims to present an extension of the scope of this technique to steady state and kinetic/dynamic reactor physics problems. Regarding steady state or criticality calculations, the power iteration on which these calculations are generally based can present problems. The population normalization performed between generations amplifies the strong spatial and generational correlations already induced by the birth and death process in neutron transport in multiplicative media. In most pathological situations, these correlations can cause neutrons to form clusters, thus depopulating parts of the system and leading to convergence problems. In this Ph.D., the classical power iteration was recast as a variance reduction problem for which the AMS was used to solve this clustering problem. To realize a proof of concept, the AMS has been implemented in a toy-model Monte Carlo code entirely developed during this Ph.D., which can model criticality/static problems with simplified physics modeling. The method was characterized on a one-speed homoge-

neous problem, and the obtained results proved to be encouraging regarding the reduction of unwanted correlations in criticality calculations. Once successfully tested on a toy model, the AMS was therefore implemented in SERPENT2 Monte Carlo code to extend its use to kinetics and dynamics problems. Indeed, despite recent developments in variance reduction, the computational costs associated with this type of calculation remain very high. In this regard, we planned to rely on AMS's use of an importance map to reduce the variance on local tallies during transients, therefore improving kinetics calculations efficiency. To do so, different combinations of methods were tested while implementing the AMS into SERPENT2. While the characterization of the AMS in kinetics was mainly done on simple test cases such as homogeneous PWR-like compositions, or heterogeneous one-speed problems, the method was finally applied to a transient operated on an 3x3 assemblies cluster. A time-dependent importance map was retrieved from adjoint point kinetics equations for that purpose. The goal would be, with time, to rely on space and time-dependent importance maps retrieved from deterministic space-time adjoint kinetics for both neutrons and precursors.

Wisdom comes from experience. Experience is often a result of lack of wisdom.

- Sir Terry Pratchett

Acknowledgments

The first words of this thesis are also the last words I wrote during my Ph.D., before taking a vacation. After defending my thesis, and taking into account the suggestions of the jury in the last version of this document, it is time to give due thanks to all the people who made this possible.

I would like to start by thanking all the members of the jury, for your dedication in the evaluation of my scientific work. Thank you to Elsa Merle and Eugene Shwageraus, for having reviewed my thesis, especially during the holiday season... Thanks also to Margaux Faucher and Cheikh M'Backé Diop for complementing this jury as examiners. Finally, I thank Tony Lelièvre, for having chaired the jury and brought an external view of neutronics on this work. I am very grateful for your work, and the improvements made to my work following your comments.

Je souhaiterais maintenant remercier les personnes ayant encadré ce travail de thèse. Tout d'abord, merci, Alain, de m'avoir encouragé à poursuivre en thèse après la maîtrise, en particulier sur ce sujet. Je te remercie chaleureusement, Éric, pour ton aide incommensurable tout au long de la thèse. Merci pour ta patience, ta disponibilité, ton temps, passé à m'expliquer les arcanes du calcul stochastique et du clustering. Les nombreuses digressions scientifiques (dont tu étais souvent à l'origine, rendons à César ce qui est dans sa salade) ont été parmi les meilleurs moments du travail de thèse, et encore bien des questions que nous nous sommes posées restent encore ouvertes... Mille mercis à Julien, pour avoir encadré la thèse, pour avoir passé un temps monstre à relire mes productions (papiers, présentations, manuscrit ...), et pour avoir accepté (au moins pour un temps) d'utiliser Overleaf pour relire mon travail. Merci aussi d'avoir contrebalancé la propension d'Éric à digresser et toujours vouloir rajouter plus de questions à un sujet de thèse déjà bien ambitieux (même si, encore une fois, c'était extrêmement stimulant dans mon travail de recherche). Enfin, un immense merci à Mariya. Je te suis reconnaissant d'avoir toujours laissé ta porte ouverte pour m'écouter raconter mes histoires, émettre des hypothèses à voix haute, et me rassurer quand mon code tardait à fonctionner, ou que j'avais l'impression d'avancer dans le flou. Ce fut un honneur et un plaisir de travailler sur un sujet aussi fascinant à vos côtés, c'est pour travailler avec des personnes comme vous que je me suis engagé en recherche.

Je pense aussi à Aurélie et Sophie. Merci d'avoir tout fait pour que je puisse me concentrer sur mon travail de recherche, et ce, dans un cadre confortable. Je vous suis reconnaissant pour la confiance que vous m'avez accordée en me permettant de rejoindre le LN.

Mes pensées vont aussi aux membres du SNC, passés et présents. Je ne citerai pas tout le monde (ne serait-ce que pour n'oublier personne), mais sachez que je garde de très bons souvenirs avec chacun et chacune d'entre vous, que ce soit en pause café, en conférence, autour d'un verre après le travail, ou entre deux portes. Je tiens tout de même à citer certaines personnes qui m'ont particulièrement aidé pendant ces trois années. Merci, Benjamin, pour nos discussions sur les processus stochastiques. Merci, Wilfried et Greg, pour votre aide en programmation et avec les outils informatiques (je vous ai bien embêtés avec ça, hein...). Merci, Fausto, pour les discussions enrichissantes sur la physique des réacteurs. Merci, Jean-Baptiste, pour les multiples covoiturages vers ou depuis l'IRSN. Merci, Vivian, pour ton aide précieuse sur TeX (entre autres). Merci à Luiz pour les relectures et les commentaires constructifs sur mes articles. Un grand merci à Catherine et Florence pour votre travail ô combien précieux !

Une pensée aux autres thésard · e · s que j'ai pu côtoyer au SNC: Léa, Clément et Romain. Merci pour les templates, les bons plans formations, les conseils, et le soutien moral. Je te souhaite bon courage, Pierre, pour les trois années (un peu moins maintenant) qui s'annoncent :) J'en viens maintenant à remercier d'autres collègues pour leur aide. Un grand merci à Ludyvine et Xavier, pour avoir accepté la charge du comité de suivi organisé en express en début de quatrième année. Merci aussi à Davide et Jean-Charles, pour avoir fait partie de la réunion de suivi en début de deuxième année. Merci, Loïc, pour ton travail et nos échanges sur l'AMS, ils ont clairement diminué le coût d'entrée sur la méthode. Une pensée pour Guy, qui m'avait aussi encouragé à poursuivre sur ce sujet lorsque j'étais à Montréal.

Je remercie aussi mes ami · e · s, qui ont été présent · e · s pendant ces trois années, et particulièrement mes colocataires pour les bons moments partagés, et leur compréhension pendant les périodes stressantes. Bon courage à toi, Guillaume, pour ta rédaction, c'est bientôt fini !

Un immense merci à mes parents, grâce à qui j'ai pu poursuivre mes études, d'abord à Nantes puis à Montréal, sans jamais être dans le besoin. C'est grâce à votre soutien que j'en suis arrivé là, que j'ai pu rencontrer tous ces gens, et que je peux maintenant continuer à travailler dans un domaine qui me plait.

Finalement, merci à celles et ceux qui ont assisté à ma soutenance, pour votre soutien et l'intérêt que vous avez porté à mes travaux.

Contents

List of figures	vii
Liste of tables	xi
Introduction	1
1 Neutron transport : physical aspects	7
1.1 Nuclear reactions	8
1.1.1 Neutron-nucleus reactions	9
1.1.2 Neutron-induced fission	10
1.1.3 Cross sections	11
1.2 Transport theory	12
1.3 Stochastic transport theory : birth-death process	15
1.3.1 A simple neutron birth and death process	15
1.3.2 Brownian motion	22
<i>Conclusion</i>	24
2 Modeling of a neutron transport problem	27
2.1 Deterministic methods	28
2.1.1 Key tenets of deterministic methods	28
2.1.2 Drawbacks of deterministic methods	29
2.2 Monte Carlo methods	29
2.2.1 General principle	29
2.2.2 Estimators	30
2.2.3 Markov chains	32
2.2.4 Variance reduction and population control	32
2.3 Nuclear data	39
<i>Conclusion</i>	40
3 Reactor physics with Monte Carlo methods	41
3.1 Steady-state criticality calculations	42
3.1.1 The k -eigenvalue equation	43
3.1.2 Power Iteration method	44
3.1.3 Issues regarding the convergence of the solution	45
3.2 Time-dependent neutron transport	49
3.2.1 A brief history of Monte Carlo neutron kinetics	50
3.2.2 Coupled neutron-precursor simulations with stochastic methods	50
3.2.3 Current limitations to industrial applications of dynamics calculations	54

<i>Conclusion</i>	54
4 Adaptive Multilevel Splitting	57
4.1 Mathematical background	58
4.1.1 Description of the algorithm	58
4.1.2 Sampling replicas for duplication	61
4.1.3 Estimators	63
4.2 Application to particle transport	63
4.2.1 Structure of a particle history in the AMS	63
4.2.2 Iterations execution	65
4.2.3 Handling branches when sampling new branching tracks	65
4.2.4 On-the-fly scoring	66
4.2.5 About the importance function	68
<i>Conclusion</i>	68
5 Criticality calculations with the AMS	69
5.1 AMS in criticality calculations	70
5.1.1 Turning criticality calculations into attenuation problems	70
5.1.2 Defining a detector	71
5.1.3 Population control mechanism	71
5.1.4 Scoring over successive generations	72
5.2 Application to a one dimensional slab reactor	75
5.2.1 Korrigan : a <i>toy-model</i> Monte Carlo code for neutron transport	75
5.2.2 Description of the system properties	75
5.2.3 Homogeneous bare slab reactor : numerical results	76
5.3 Application to heterogeneous slab	89
5.3.1 Description of the geometry, materials and simulation parameters	89
5.3.2 Estimations of the fundamental mode	90
5.3.3 Analyze of the particle weights over space and generations	91
<i>Conclusion</i>	96
6 Application of the AMS to kinetics calculations	99
6.1 Kinetics calculations with SERPENT2	100
6.1.1 Time structure	101
6.1.2 Particles storage structure	102
6.1.3 Variance reduction and population control	104
6.2 Implementation of the AMS in SERPENT2	109
6.2.1 Handling delayed neutron precursors	109
6.2.2 Recasting the population control step	112
6.2.3 Distribution of particle weights over time	113
6.3 Computing the α fundamental mode of a one-speed 1D heterogeneous slab reactor	118
6.3.1 Modeling of a dynamic α mode with SERPENT2 in one-speed theory	118
6.3.2 Test case	119
6.3.3 Results	121
<i>Conclusion</i>	127
7 Transient simulation with the Adaptive Multilevel Splitting	129

7.1	The adjoint flux as neutron importance	130
7.1.1	Relation between neutron importance and adjoint flux	130
7.1.2	Setting the "initial" condition in the backward approach	133
7.1.3	Formulation of the adjoint point kinetics	134
7.2	Control rods drop on a 3x3 UOX assembly cluster	137
7.2.1	Case description	137
7.2.2	Adjoint flux calculation for the rod drop transient	139
7.3	Numerical results	142
7.3.1	Impact of the AMS on the power over time estimate	143
7.3.2	Impact of the multi-parameter importance map on the FoM degradation	144
7.3.3	FoM gains regarding the power distribution in the last time bin	146
	<i>Conclusion</i>	149
8	Prospects regarding consequences of neutron transport's stochasticity on computational schemes	151
8.1	Application of the AMS to neutron noise	152
8.1.1	Zero-power neutron noise	152
8.1.2	AMS and correlations	154
8.2	Stochastic fluctuations in time-dependent neutronics	154
8.2.1	Distribution of the number of particles at time t and power law	154
8.2.2	Detecting a power law in time-dependent neutronics calculations	155
8.3	Thoughts on the relevance of coupled calculations in the case of dynamics Monte Carlo calculations	156
	Conclusion	159
	Appendix A Data analysis tools	165
A.1	Variance for correlated random variables	165
A.2	Exponentially Weighted Moving Average	166
	Appendix B Derivation of the adjoint equations	169
B.1	Adjoint transport equation	169
B.1.1	Derivation of the importance equation	169
B.1.2	Multigroup adjoint diffusion equation	172
B.2	Time discretization of the adjoint point kinetics equations	172
	Appendix C SERPENT2 input data for the geometry and materials of the assembly cluster transient	175
	Appendix D BPLUS energy mesh	181
	Résumé en français	183
	References	190

List of Figures

1.1	Representation of a neutron-induced fission	10
1.2	Total microscopic cross sections at 300 K for ^{235}U , ^{238}U and ^1H	12
1.3	Critical Galton-Watson process for 100 independent fission chains	18
1.4	Subcritical Galton-Watson process for 100 independent fission chains	18
1.5	Supercritical Galton-Watson process for 100 independent fission chains	19
1.6	Illustration of the Gambler's ruin for a critical Galton-Watson process	22
2.1	Combing method	34
3.1	Monte Carlo power iteration algorithm	45
3.2	Distribution of prompt fission chain's lifetime and delayed neutrons precursors decay time	51
3.3	Monte Carlo kinetics algorithm for one independent simulation	53
4.1	AMS iterations for non branching tracks with a detector defined in the (x,y) plane	60
4.2	AMS track/branch/point structure	64
4.3	AMS track with two branches.	65
4.4	AMS algorithm	66
4.5	Handling branches during the re-sampling step.	67
5.1	Re-sampling particles in criticality calculations with the AMS.	72
5.2	AMS population control effects on the number of collisions scored	73
5.3	AMS sampling zones for cases where $f(\mathbf{r}) > 1$	73
5.4	Comparison scheme between the Power Iteration (PI) and AMS used for criticality.	74
5.5	Convergence of the average k_{eff}	78
5.6	Evolution of the mean entropy over generations	78
5.7	Mean entropy and number of collisions for different K in the AMS branchless case.	79
5.8	Distribution of the k_{eff} after convergence	81
5.9	Spatial flux profile with 3σ confidence interval	82
5.10	Spatial correlations for the homogeneous 1D rod (scale from -1 to 1).	83
5.11	Spatial correlations for the homogeneous 1D rod (scale from -0.1 to 0.1).	83
5.12	Mean number of families over generations.	84
5.13	Absolute (top) and relative (bottom) number of independent neutron families killed by birth/death process	85
5.14	Absolute (top) and relative (bottom) number of independent neutron families killed by population control	86

5.15	Generational correlations over space for the homogeneous 1D bare slab reactor (PI analog case).	87
5.16	Generational correlations at $x = 23.5$ cm for the 1D bare slab reactor.	87
5.17	Figure of Merit for the flux estimation over x for the 1D bare slab reactor.	88
5.18	Heterogeneous one energy group slab geometry	90
5.19	k_{eff} distribution for the heterogeneous one-speed slab problem.	91
5.20	Average k_{eff} (computed over independent calculations) between generations 200 and 250 for the heterogeneous one-speed slab problem	92
5.21	Flux distribution in the heterogeneous case.	93
5.22	Flux FoM in the heterogeneous case.	93
5.23	Weight VS position along x -axis for collision points in the case of the PI combing branchless heterogeneous case	94
5.24	Weight VS position along x -axis for collision points in the case of the AMS branchless heterogeneous case	94
5.25	Weight VS position along x -axis for collision points in the case of the PI combing branchless homogeneous case	94
5.26	Weight VS position along x -axis for collision points in the case of the AMS branchless homogeneous case	95
5.27	Statistical moments of the weights distribution over generations (EWMA)	96
6.1	SERPENT2 particle storage structure	103
6.2	Population control sequence in SERPENT2	105
6.3	Creation of a delayed neutron precursor in SERPENT2	109
6.4	Spatially integrated power over 5 ms with 1σ confidence intervals for a subcritical (-0.23%) axially infinite cuboid	115
6.5	Variance of spatially integrated power over 5 ms for a subcritical (-0.23%) axially infinite cuboid for different AMS re-sampling kernels	116
6.6	Figure of Merit for the spatially integrated power over 5 ms for a subcritical (-0.23%) axially infinite cuboid for different AMS re-sampling kernels	117
6.7	Box plots of the spatial FoM distribution for the local power evaluation with time for a subcritical (-0.23%) axially infinite cuboid for different AMS re-sampling kernels	117
6.8	Heterogeneous slab geometry in SERPENT2 for the α -like kinetics calculation	120
6.9	Flux over space for the heterogeneous slab case in kinetics	122
6.10	Deviations from a reference solution for the heterogeneous slab in kinetics	123
6.11	Weight VS position along x -axis for collision points in the kinetics heterogeneous slab geometry (<i>case ref branchless</i>) for different times	124
6.12	Weight VS position along x -axis for collision points in the kinetics heterogeneous slab geometry when no weight window is used in the AMS for different times	124
6.13	Weight VS position along x -axis for collision points in the kinetics heterogeneous slab geometry when the weight window technique is used in the AMS for different times	124
6.14	Weight VS position along x -axis for collision points in the kinetics heterogeneous slab geometry when the weight window technique is used in the AMS and weighted sampling is used for different times	125

6.15	Statistical moments of the weights distribution over time (EWMA) . . .	125
6.16	Figure of Merit over space for the heterogeneous slab case in kinetics. . .	126
7.1	Cluster geometry for the rod drop transient case	138
7.2	Power distribution for the assembly cluster case retrieved from a k - eigenvalue calculation with SERPENT2 before and after the rod drop . .	139
7.3	Spatial ADVANTG importance distribution in energy group 7 ($37.3\text{eV} <$ $E < 101\text{eV}$) for the assembly cluster with AIC control rods	141
7.4	Spatial ADVANTG importance distribution in energy group 32 ($1.92\text{MeV} <$ $E < 2.23\text{MeV}$) for the assembly cluster with AIC control rods	141
7.5	Spatially averaged ADVANTG importance in each energy group for the assembly cluster with AIC control rods	142
7.6	Solution of the adjoint point kinetics neutron equation for the control rods drop subcritical transient with and without precursors	143
7.7	Power over time for the UOX cluster rod drop	145
7.8	Distribution of the relative increase of the spatial FoM at $t = 10^{-2}$ s for the cluster rod drop simulation for the 3 AMS calculations	147
7.9	Ratio of the FoM obtained with 2 different spatial importance maps for the cluster rod drop simulation (detector is either central assembly fission rate or surroundings assemblies fission rate)	148
8.1	Illustration of neutron noise in a critical reactor	153
A.1	Statistical moments of the weights distribution over generations.	166
A.2	Statistical moments of the weights distribution over generations.	167

List of Tables

1.1	Delayed neutron precursors group constants for ^{235}U	11
5.1	Physical properties for homogeneous 1D rods	76
5.2	Description of calculations parameters for the 1D bare slab reactor . . .	76
5.3	Averaged k_{eff} values with their standard error (SEM) and relative Figure of Merit (FoM).	81
5.4	Computation time for each k -eigenvalue calculation on the homogeneous bare slab	89
6.1	Precursor group constants for ^{235}U ($\beta = 0.00585$) for a 1 eV incident neutron	111
6.2	SERPENT2 AMS simulation in the case of a homogeneous box for different Weight Window parameters	114
6.3	Materials composition for the homogeneous PWR like box.	114
6.4	SERPENT2 simulation cases for the one-speed heterogeneous slab geometry in kinetics.	121
7.1	Simulation parameters for the cluster rod drop simulation	139
7.2	Computation times and fraction of sampled flights leading to real collisions and scored collisions over the whole transient for the cluster rod drop simulation	146
D.1	BPLUS library groups	181

Introduction

Context

There are currently 422 operating nuclear power reactors dedicated to the production of electricity in the world [1]. In France, nuclear power accounted for 62 to 82 % of the electricity produced between 2010 and 2021 with 58 reactors [2]. There are currently a total of 57 reactors under construction for the sole purpose of electricity production in the world. But the use of neutron physics does not stop there and also includes medicine or research.

The operation of nuclear reactors is based on a chain reaction phenomenon involving a tremendous number of neutron-induced fissions of nuclei. But designing reactors, predicting the behavior of fissile systems, evaluating nuclear accidents fallout or designing experiments and deepening the understanding of physical phenomena, are activities that cannot rely solely on experiments, and this for several reasons. While being eventually extremely costly, it also happens that experiments are limited by strong physical constraints, in particular the difficulty concerning nuclear instrumentation and detectors (it is for example not possible to design an in-core detection chain allowing to give access to a fine description of the spatial or spectral neutron distributions within the core). Also, conducting experiments associated to the study of accidental transients might imply to partially reproduce them while being representative of the accidents that may occur in a real core, which might reveal to be both a technical and a safety challenge. The nuclear industry therefore heavily relies on simulation tools for many applications.

In return, in order for these tools to fulfill their role, it is necessary that the physical models that are implemented in numerical codes faithfully represent reality or alternatively, that the discrepancies with reality are well known and under control. Hence, the validation and the qualification of numerical codes is a prerequisite of their use whether in an industrial or research context. Although the use of experimental data to validate the simulation codes seems to be natural, numerical simulation tools may also be used for the validation of new or less accurate tools. Regarding this specific point, the french nuclear safety authority provides guidelines regarding the need to validate any calculation tool using reference calculations [3]:

“ Article 3.8 of the decree in reference [3] requires the use of "validated" methods and qualified Scientific Computation Tools (SCTs) to carry out these studies. To this end, the operator must have formalized procedures establishing the validation of methods and the qualification of SCTs. [...] This validation is based on the comparison of the results of calculations on validation cases with [...] results of reference scientific calculation tools. [...] Thus, the validation can be performed on the basis of comparisons with

a reference SCT, [...] ¹ ”

By reference calculations, this guide means numerical simulations capable of producing high-fidelity results regarding the physics that is modeled. In that sense, Monte Carlo simulations are usually considered to be reference tools in the field of neutronics [3]:

“ Reference scientific computing tool: Scientific calculation tool whose predictive performance is judged superior to that of the expected from the scientific calculation tool to be validated. For example, in neutronics, "Monte-Carlo" or deterministic SCT with a very high number of energy groups and spatial mesh can, in some cases, be considered as reference SCTs.¹ ”

Monte Carlo simulations have thus been extensively used to produce reference calculations for criticality-safety, or radiation protection applications, for example. It is not the case, however, for all reactor physics applications, especially regarding dynamics simulations. This research work is motivated by the desire to develop such reference tools for these types of application.

Although formalized (and named) for numerical calculation in the middle of the 20th century, the principle of these methods goes back several centuries, and is based on an estimation of the result based on a large number of random experiments, in the manner of a statistical survey. The interpretation of Monte Carlo results is based on statistical laws such as the law of large numbers, which states that the statistical uncertainty associated with the estimate of the result decreases according to $1/\sqrt{N}$, where N is the number of simulated particles in the case of particle transport. Hence, the more particles simulated, the more accurate the result, and better precision can be reached through increasing the number of simulated particles. However, Monte Carlo simulations are generally more expensive than deterministic calculations in the context of neutron transport.

To reduce the costs of Monte Carlo simulations, mathematical properties of random sampling have been used to implement so-called biasing methods. These methods consist in modifying the natural rules of *analog* Monte Carlo simulations. The idea is that, for the same number of simulated particles, the statistical uncertainty of the score would be lower. These methods are thus called *variance reduction* methods. They have been widely used in particle transport calculations, especially for attenuation problems such as radiation shielding applications.

The Adaptive Multilevel Splitting (AMS) is one of those variance reduction methods. It was developed as a mathematical method to increase performances of rare event simulations [4, 5], and later implemented in neutral particle transport precisely for shielding applications [6, 7]. It is based on the splitting of particles that have more chances to reach a given detector, thus increasing the number of particles contributing to the measure. It has proven to be quite robust in strong spatial attenuation configurations in which only a few particles (if any) may reach the location of interest [7].

¹Translated from the ASN Guide 28 [3].

Nevertheless, neutron transport simulations are used for several types of applications (radiation protection, criticality safety, ...), and this thesis focuses mainly on two of them which are less prone to variance reduction schemes, namely steady state reactor physics and transient behavior (e.g., accidents) modeling.

Steady-state neutronics calculations consist in characterizing the *criticality*; i.e., the long-term growth, of a system. A system whose fission chain reaction is stable over time is said to be *critical*. It is *subcritical* when the chain reaction tends to die out, and *supercritical* when it gets out of control. They are also commonly used to evaluate the long-term state of the system (e.g., its power distribution) if it were alive². This long-term state can mathematically be called *the system asymptotic state conditioned to its survival*. These simulations have been used since the beginning of the civil nuclear industry, and benefit from decades of refinement. However, there is still room for improvement. In particular, the study of temporal and spatial correlations affecting Monte Carlo simulations has raised interest in the past few years, in particular the so-called *clustering* phenomenon [8]. It was shown that, due to the inner nature of the branching mechanism coupled to the transport of the particles, strong fluctuations might develop in loosely coupled systems, leading to the formation of patchy spatial patterns called clusters [9]. Population control algorithms used in Monte Carlo simulations can indeed amplify these phenomena, thus affecting even the estimation of mean results [10].

As previously said, Monte Carlo simulations have served as reference tools for many years now. For a long time, however, the limited computing power of the computing units has held Monte Carlo kinetic calculations out of reach, and transients calculations have been handled using deterministic solvers [11, 12, 13]. Nevertheless, there has been a renewed interest in these simulations since the 2010s, following the groundbreaking work of Hoogenboom and Sjenitzer [14, 15]. Since then, several Monte Carlo codes implemented algorithms to perform transient simulations [16, 17, 18, 19, 20], and coupled calculations have also been performed [21, 22, 23, 24]. However, the costs associated with such calculations are still far too high to turn to industrial applications.

Two terms may be used to describe the simulation of a transient behavior from the neutron physicist point of view: *kinetics* and *dynamics*. Although both words are used to identify the study of a system short-term behavior following a perturbation (let us say from around 10^{-4} seconds to minutes at most), *kinetics* calculations only consider the neutron transport aspects, whereas *dynamics* simulations encompass neutron transport and feedback phenomena due to other physical processes.

The aim of this Ph.D. thesis was to propose a new variance reduction scheme to improve kinetics calculations' performances (in view of improving dynamics calculations). The present document summarizes three years of work conducted on the implementation of the AMS for steady-state and time-dependent nuclear reactor physics.

²For a subcritical system which tends to die out, the real system might never reach the "long-term."

Approach adopted in this thesis

To address the above-mentioned issues of steady-state and kinetics calculations, the general approach of this thesis is to interpret reactor physics calculations as variance reduction problems, and use the Adaptive Multilevel Splitting algorithm to address these problems. For subcritical systems, the comparison with a spatial attenuation problem such as radiation shielding is quite straightforward, as particles tend to die out over time. For supercritical systems, however, it was necessary to recast the initial simulation into an attenuation problem.

For steady-state simulations, the idea was to focus on the disappearance of independent lineages of particles over generations, which eventually leads to particle clustering problems. The AMS was implemented in a simplified Monte Carlo code to test the effect of such a method on the specific issue of particle clustering.

Regarding kinetics calculations, the following work presents the implementation of the AMS into the SERPENT2 Monte Carlo code to improve local tallies estimation performances in kinetics calculations. Improving the accuracy of results in a specific time-defined detector could prove to be worthwhile in cases of coupled calculations. Indeed, multi-physics couplings always involve an exchange of information between codes, and estimating more accurate power distribution in the Monte Carlo calculation would improve the following thermal hydraulics or thermal mechanics' calculation (and thus the following time steps).

Structure of this thesis

This manuscript presents a summary of the work carried out on the implementation of the approach described above.

Chapter 1 outlines a brief introduction of neutron transport's underlying physics. It aims at introducing basic concepts regarding the random nature of neutron transport, which will be used later in the document. It goes from the introduction of neutron-nucleus reactions to the macroscopic and stochastic neutron transport theory in fissile media.

Chapter 2 presents basic concepts regarding numerical simulations. Although presenting both deterministic and Monte Carlo methods, it focuses on the latter, as deterministic methods have been used only marginally in this work.

Chapter 3 focuses on steady-state eigenvalue calculations and kinetics calculations using Monte Carlo methods. It aims at presenting the features that motivate the use of a variance reduction method in these types of simulations. It introduces the state of the art regarding the clustering phenomenon in steady-state calculations, along with the current state of the art of dynamics calculations. The specific points on which variance reduction methods may lead to improvements are highlighted.

Chapter 4 is intended to the presentation of the Adaptive Multilevel Splitting (AMS) as it was previously implemented for neutral particle transport by Louvin in TRIPOLI4 [7]. First, the mathematical framework is succinctly presented in order to introduce the notions then applied to particle transport. The AMS as implemented in the TRIPOLI4 Monte Carlo code is then summarized for multiplicative media.

Chapter 5 compiles results obtained in steady-state calculations using the AMS. The first part aims at describing how the AMS was extended from fixed-source calculations to steady-state reactor physics problems. The method was then applied to a homogeneous bare slab reactor with mono-energetic neutrons, and results were analyzed with special attention to clustering. Calculations were performed using an in-house simplified ad hoc Monte Carlo code which is also briefly described. Finally, calculations were performed on a heterogeneous slab reactor using the same code with the objective to progressively increase the complexity of the study.

Chapter 6 is dedicated to the implementation of the AMS in the SERPENT2 Monte Carlo code. Due to the complexity of the code (which is composed of around 350 000 lines of code), an extensive study of SERPENT2 kinetics mode has been done. The interactions between the AMS and the already implemented algorithm have been analyzed to perform the integration of the method in the current algorithm. A comprehensive description of issues related to the implementation of the AMS is given, along with the choices that were made in the current implementation of the method. Calculations presented in this chapter were destined to illustrate choices of implementations and introduce phenomena that may impact the AMS results compared to state-of-the-art calculations. The last part of this chapter is devoted to reproduce the heterogeneous case modeled with the ad hoc Monte Carlo code, but this time with SERPENT2.

Chapter 7 gives a proof of concept for variance reduction in kinetics simulations with the AMS, as it presents the application of the AMS method for the simulation of a subcritical transient in a 3x3 fuel assembly cluster. In this chapter, the concept of neutron importance and its link to the adjoint transport equation are first presented. The corresponding adjoint equations are derived for the time-dependent case, and an approximation based on the mathematical adjoint of the point kinetics equations are developed. Using the adjoint point kinetics just introduced, multiple AMS calculations were performed on an assembly cluster using different importance maps. The purpose of this part is twofold, showing the AMS capability to model a transient in a rather realistic system, hence proving the possibility for it to be used as a variance reduction method in a time-dependent case, and studying the behavior of estimates for different importance functions. Three different importance functions were used for this objective:

- a pure time-dependent function aiming at pushing neutrons to the end of the transient,
- an importance function based on the product of a static adjoint flux and the solution of the adjoint point kinetics equations,
- the same type of function as the previous one but for different definitions of the detector when the static map was computed.

The results are compared to two reference calculations performed with the SERPENT2 code in its 2.1.32 version.

The final chapter, namely Chapter 8, aims at outlining some prospects regarding both the AMS method and neutron dynamics calculations with Monte Carlo methods. Thoughts are presented regarding zero-power neutron noise application of the AMS

algorithm, along with open questions on the significance of stochastic fluctuations in dynamics simulations.

Chapter 1

Neutron transport : physical aspects

God does not play dice with the universe; He plays an ineffable game of His own devising, which might be compared, from the perspective of any of the other players [i.e. everybody], to being involved in an obscure and complex variant of poker in a pitch-dark room, with blank cards, for infinite stakes, with a Dealer who won't tell you the rules, and who smiles all the time.

- Sir Terry Pratchett,
Good Omens: The Nice and Accurate Prophecies of Agnes Nutter, Witch

Contents

1.1	Nuclear reactions	8
1.1.1	Neutron-nucleus reactions	9
1.1.2	Neutron-induced fission	10
1.1.3	Cross sections	11
1.2	Transport theory	12
1.3	Stochastic transport theory : birth-death process	15
1.3.1	A simple neutron birth and death process	15
1.3.2	Brownian motion	22
	<i>Conclusion</i>	24

In 1932, Sir James Chadwick discovered the existence of neutrons as constituents of the atom nucleus. Later in 1938, Lise Meitner, Otto Hahn and Fritz Strassmann brought to light the neutron-induced fission reaction in which a neutron hitting a fissile nucleus might split it in half, releasing kinetic energy in the process. More than energy, several neutrons are also produced during this reaction as shown by Frédéric Joliot-Curie in 1939. These three observations laid the foundations of the fission based reactors. Indeed, Leó Szilárd, who had thought of a way to produce energy through nuclear chain reactions already, proposed to use the neutron-induced fission as a chain

reaction to produce energy. While the first uses of these reactions were oriented towards military applications, civilian reactors were then put into service to produce electricity from the heat released by numerous fissions.

To model the nuclear reactor behavior regarding fission, neutron physicists thus need to characterize the behavior of a population of neutrons inside the reactor core. While different types of physics are necessary to fully describe nuclear reactors, this thesis essentially focuses on the transport and interaction of neutrons through matter. To describe these interactions at the reactor scale, two fields are of interest. Nuclear physics is used to explain interaction mechanisms between a neutron and a nucleus during a collision, but is not sufficient to model the full behavior of the fission chain reaction in an entire reactor. To do so, neutron physicists also need to describe the motion of neutrons inside matter, and take into account the progeny that arises from fission. While the flight of a neutron between two collisions is as simple as a straight line (because a neutron particle does not bear any electric charge), accounting for all the particles present in a reactor is a daunting task. A description inherited from statistical mechanics was therefore applied to neutron physics to overcome this issue. The theory arising from this description is commonly called neutron *transport theory*. The field that studies neutrons' motion in matter and their interactions with the nuclei that constitute it is generally called *neutronics*.

This chapter presents some basics of neutron transport theory useful for understanding neutronics problems addressed in this thesis. The first section covers some notions of neutron-nucleus interactions without going further into details regarding nuclear physics. While it is generally possible to describe the transport of neutrons within matter using diffusion theory with fairly good accuracy, the most faithful representation lies on transport theory. Thus, the main characteristics of neutron transport theory are presented in the second section of this chapter. However, in practice, the results obtained from transport (or diffusion) calculations usually represent the average behavior of a system, while the underlying physics is purely stochastic. The fluctuations of the neutron population due to random processes and their effects are presented in the third section of this chapter.

1.1 Nuclear reactions

As neutrons travel through matter, they might interact with nuclei constituting matter. When a neutron interacts with a nucleus, we refer to it as a *collision*. Different types of collisions can happen, some leading to the absorption of the neutron thus forming a compound nucleus, for others the neutron directly interacts with a nucleon composing the target nucleus [25]. While the details of how and why neutrons preferably interact directly or through the formation of a compound nucleus are of great interest for the nuclear physicist, only the overall result of such interaction is useful for the modeling of neutron transport. For this reason, we shall not go into details regarding nuclear physics of neutron-nucleus interactions. For more details, the reader can, for example, consult existing literature on the subject [25]. However, basic mechanisms governing the course of neutrons inside a reactor are presented in the current section.

1.1.1 Neutron-nucleus reactions

In a nuclear system, collisions between a neutron and a nucleus can result in various outcomes. The main reactions taken into account in most neutronics model, even the simplest ones, are absorption and scattering.

Regarding the absorption, it regroups two channels, sometimes called *sterile capture*, and *fission*. The sterile capture, usually noted (n,γ) , happens when a neutron is absorbed into the target nucleus, leading to an excited state which then induces a gamma emission by decay. It is called *sterile* because a neutron is lost due to the absorption, and no new neutron is produced by the reaction. This is how actinides such as plutonium, neptunium or americium are produced in a nuclear reactor. It is also called *radiative capture* since a gamma ray is emitted in the process when the excited nucleus decays. The fission, however, leads to the release of energy, which is extracted and then converted into electricity in power reactors. One to several neutrons are also emitted, thus maintaining the chain reaction. It is this mechanism that gives its name to *fission reactors*. It is further detailed in the following section.

In the case of a scattering, a unique neutron exits the collision event, and two outcomes are possible. When the kinetic energy is conserved in the center of mass of the nucleus-neutron system, it is called an *elastic scattering*. As neutrons scatter on nuclei, they loss kinetic energy (and change direction), progressively slowing them down. The closer the weight of the nucleus is to that of the neutron, the higher the energy loss can be, similarly to the collision between two masses. Imagine two billiard balls in the case of a neutron colliding with a light nucleus, and a billiard ball hitting the border of the billiard table in the case of a heavy nucleus. In the first case, the neutron can be stopped and transmit up to all of its kinetic energy to the nucleus it hits if the two particles are of the same mass, this is the main mechanism of slowing down in a thermal reactor. In the second case, the neutron will be scattered without losing much of its energy. As neutrons slow down, their probability to interact with matter changes, which makes some energy range more or less useful depending on the objective. For example, water reactors make extensive use of slow neutrons which have a higher probability to induce the fission of ^{235}U nuclei compared to fast neutrons. They are also called thermal neutrons due to the fact that their energy is close to the thermal energy of the system. In terms of notations, elastic scattering reactions are generally written (n,n) . The second type of scattering reactions is called an *inelastic scattering*, because the kinetic energy is not conserved. The outgoing neutron also leaves the collision with a different direction and speed, but the nucleus hit by the incident particle enters an excited state. It then emits a gamma particle before going back to its ground state. The inelastic scattering is written (n,n') , and can sometimes be ignored in most simple models. In reality, other reactions also occur but are often negligible or lumped with the ones presented above when studying the system of equations governing a neutron population evolution (e.g., $(n,2n)$ reactions are often included in fission reactions).

Due to quantum mechanisms happening inside the nucleus, the outcome of a collision is purely random. Depending on the energy of the incident neutron, the nature of the target nucleus or its motion due to the ambient temperature, the probability for a given outcome to happen changes.

1.1.2 Neutron-induced fission

Fission, whose global process from the neutron transport perspective is represented in Figure 1.1, is the main source of energy, but also of neutron reproduction in fissile media. Immediately after fission ($\sim 10^{-19}$ s), one or several neutrons are emitted. They are called *prompt neutrons*. Fission fragments, which carry most of the released kinetic energy, are also created by splitting of the initial nucleus. Some fission fragments can be in an excited state following the fission event, and release neutrons by decay at a later time t following the probability density

$$P(t) = \lambda e^{-\lambda t} \quad (1.1)$$

where λ is the decay constant for a given isotope. These neutrons are called *delayed neutrons*, as they appear later compared to prompt neutrons ($\sim 10^{-2}$ to 10^2 s after the fission), and fission fragments emitting these neutrons are called *delayed neutron precursors* (or precursors for short). The number of neutrons that are emitted by fission is also a random quantity [26].

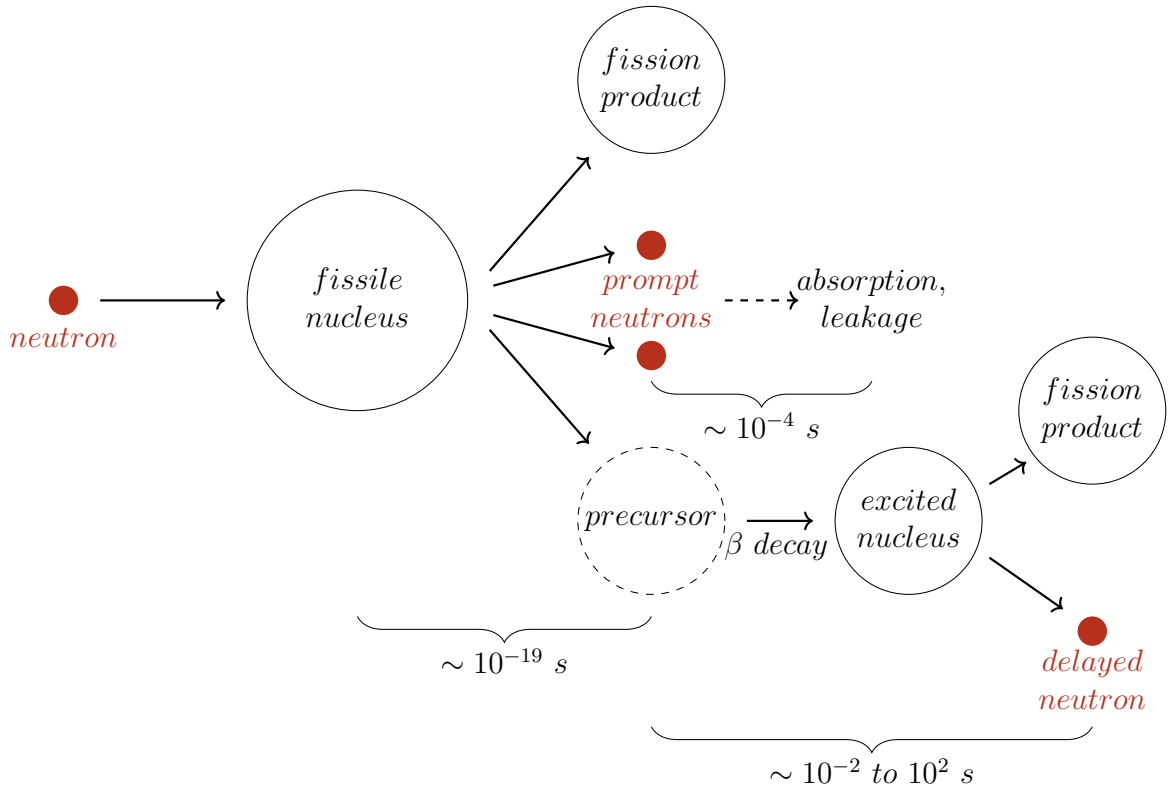


Figure 1.1: Representation of a neutron-induced fission

The delayed neutron fraction denoted β (i.e., the fraction of all fission neutrons that are emitted at a later time) represents the average number of delayed neutrons per fission neutron. These delayed neutrons can be emitted by many different nuclei (~ 40 different isotopes can lead to delayed neutrons [27]) which are usually collapsed into groups, or families, depending on their decay constant. For example, the delayed neutron fractions and decay constants for delayed neutrons precursors of ^{235}U fission are presented in Table 1.1. Regrouping precursors into groups (often six or eight in numbers) allow simplifying the system of equations that describe the neutron evolution

over time without much impact on the results. While delayed neutrons represent only a small fraction of all neutrons emitted by a fission event, operating a stable chain reaction without them would be impossible as they greatly modify kinetics properties of a system [28].

Table 1.1: Delayed neutron precursors group constants for ^{235}U ($\beta = 0.0065$). β_i is the delayed neutron fraction for precursor group i , and λ_i is the decay constant characterizing that group.

Precursor group i	β_i	λ_i [s^{-1}]	$1/\lambda_i$ [s]
1	2.150×10^{-4}	1.24×10^{-2}	8.065×10^1
2	1.424×10^{-3}	3.05×10^{-2}	3.279×10^1
3	1.274×10^{-3}	1.11×10^{-1}	9.009×10^0
4	2.568×10^{-3}	3.01×10^{-1}	3.322×10^0
5	7.480×10^{-4}	1.14×10^0	8.772×10^{-1}
6	2.730×10^{-4}	3.01×10^0	3.322×10^{-1}

1.1.3 Cross sections

Microscopic cross sections are used to quantify the likelihood of a reaction to happen when a neutron is in the vicinity of a nucleus. Here is a simple comparison to better understand this concept. Let us imagine that the nucleus is a target, and the neutron is a projectile launched randomly in the direction of the target. The bigger the target area as seen by the projectile, the more likely they are to interact. Defined as such, this area is called the geometric cross section. This concept also applies to nuclear interactions, except that it is not the geometric cross section that is used since nuclear forces leading to nuclear reactions operate on the neutron in the vicinity of a nucleus. It is possible to define a microscopic cross section, noted σ_ρ for reaction ρ and often expressed in barn ($1\text{barn} = 10^{-24} \text{ cm}^2$), for every neutron-nucleus reaction, for every isotope. The total microscopic cross section being the sum of partial cross sections over all possible reactions. The probability for reaction ρ to occur during a collision can thus be estimated by

$$P = \frac{\sigma_\rho}{\sigma_{tot}} \quad (1.2)$$

where σ_{tot} is the total microscopic cross section. Unlike the geometric cross section mentioned earlier as an example, nuclear cross sections depend on the energy of the incident neutron as displayed in Figure 1.2, meaning that the probability for a neutron to interact with a nucleus depends on its speed.

At this point, it begins to be clear that knowing the behavior of neutrons is of the utmost necessity to predict the behavior of such a complex system that is a nuclear reactor. While nuclear physics describes how neutrons interact with a nucleus, studying the whole neutron population of a reactor cannot be done solely based on nuclear physics rules. The main mechanisms of neutron transport are presented in the following sections.

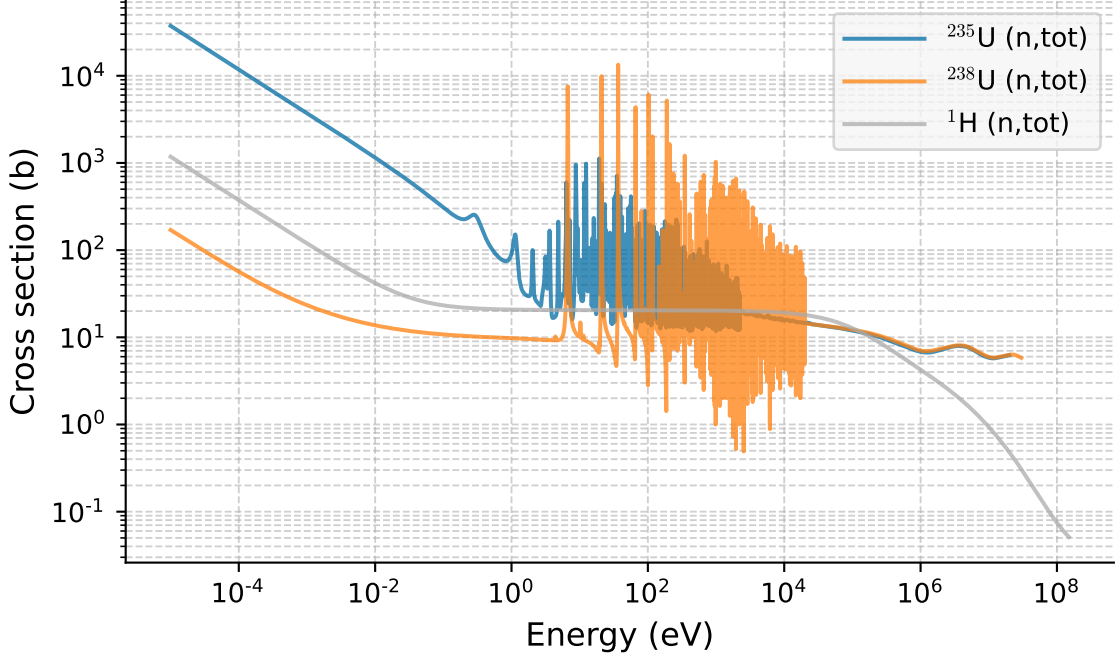


Figure 1.2: Total microscopic cross sections at 300 K for ^{235}U , ^{238}U and ^1H .

1.2 Transport theory

To fully characterize a medium and thus study how a neutron population interacts with it, atoms microscopic cross sections along with their density N (expressed in cm^{-3}) are necessary. The product of the atom density and a microscopic cross section is called the macroscopic cross section noted Σ_ρ for reaction ρ , expressed in cm^{-1} . The total macroscopic cross section for a medium is the sum over all its isotopes i and all reactions ρ of the macroscopic cross section Σ_ρ^i

$$\Sigma_{tot} = \sum_i \sum_\rho N_i \sigma_\rho = \sum_i \sum_\rho \Sigma_\rho^i. \quad (1.3)$$

The mean free path is defined as

$$\lambda = \frac{1}{\Sigma_{tot}} \quad (1.4)$$

and represents the average distance traveled by a neutron between two collisions in a medium with a total macroscopic cross section equals to Σ_{tot} . Indeed, neutron flights are straight lines between two collisions, and since the Probability Density Function (PDF) for the path length l is equal to

$$P(l) = \Sigma_{tot} e^{-\Sigma_{tot} l} \quad (1.5)$$

the average value of l is equal to

$$\bar{l} = \int_0^\infty l \Sigma_{tot} e^{-\Sigma_{tot} l} dl = \frac{1}{\Sigma_{tot}} = \lambda. \quad (1.6)$$

Since path length are exponentially distributed, the motion of neutrons between two collisions is sometimes referred to as *exponential flight*.

The rate of collisions happening inside a medium is defined by three quantities:

- the number of neutrons in the medium (noted n),
- the probability to interact with nuclei (i.e., the macroscopic cross sections),
- the neutron speed v .

The resulting quantity, named *reaction rate*, is equal to

$$\tau = vn\Sigma_{tot} \quad (1.7)$$

and describes the number of collisions happening by unit of time and unit of volume (i.e., the number of collisions occurring in a volume d^3r during dt is equal to $\tau d^3r dt$). Reaction rates can be defined for any nuclear reaction as

$$\tau_\rho = vn\Sigma_\rho \quad (1.8)$$

where τ_ρ is the reaction rate for reaction ρ . Being able to compute τ_ρ allows to quantify the number of reactions ρ occurring per unit of time. In the case of fission for example, it allows to quantify the fission power of a system like a reactor (i.e., the energy released by fission in the reactor by units of time). Predicting reaction rates (like the power released by a system) is the basis of neutronics studies, including design applications, safety studies, design of experiments.

To simplify equations, the quantity vn describing the neutron population interacting with the medium is called the *angular* or *phase flux*, and is noted ϕ . As explained before, neutrons may have different energies E (i.e., speed), position \mathbf{r} and direction Ω . In result, the angular flux is a distribution depending on these variables.

In a reactor, the number of neutrons is too high to follow them individually¹. Systems are thus usually described using statistical mechanics tools. In 1872, so before the discovery of the neutron particle, Ludwig Boltzmann established an equation to describe thermodynamic systems not at equilibrium based on their statistical behavior. It can be shown that writing the balance equation for a neutron population leads to the same form of equation [28], named the neutron transport equation. Including delayed neutrons dynamics leads to the following system of equations

$$\frac{1}{v} \frac{\partial}{\partial t} \phi(\mathbf{r}, E, \Omega, t) + \Omega \cdot \nabla \phi(\mathbf{r}, E, \Omega, t) + \Sigma_{tot} \phi(\mathbf{r}, E, \Omega, t) = Q(\mathbf{r}, E, \Omega, t) \quad (1.9)$$

$$\begin{aligned} Q(\mathbf{r}, E, \Omega, t) = & \int_0^\infty \int_{4\pi} \phi(\mathbf{r}, E', \Omega', t) \Sigma_s(E', \Omega' \rightarrow E, \Omega) dE' d^2\Omega' \\ & + \frac{1}{4\pi} \left\{ \chi^p(E) (1 - \beta) \int_0^\infty \nu \Sigma_f(\mathbf{r}, E') \phi(\mathbf{r}, E', \Omega', t) dE' \right. \\ & \left. + \sum_k \chi_k^d(E) \lambda_k C_k(\mathbf{r}, t) \right\} + Q_{ext}(\mathbf{r}, E, \Omega, t) \end{aligned} \quad (1.10)$$

$$\frac{\partial}{\partial t} C_k(\mathbf{r}, t) = \int_{4\pi} \int_0^\infty \beta_k \nu \Sigma_f(\mathbf{r}, E) \phi(\mathbf{r}, E, \Omega, t) dE d^2\Omega - \lambda_k C_k(\mathbf{r}, t) \quad (1.11)$$

where the following quantities are used:

¹The neutron density in a power reactor is about 10^8 neutrons.cm⁻³ [28].

- $\Sigma_s(E', \mathbf{\Omega}' \rightarrow E, \mathbf{\Omega})$ is the macroscopic cross section describing the probability of scattering from energy E' and direction $\mathbf{\Omega}'$ to energy E and direction $\mathbf{\Omega}$,
- $\chi^p(E)$ is the energy distribution for prompt neutrons,
- $\chi_k^d(E)$ is the energy distribution for delayed neutrons coming from precursors belonging to group k ,
- ν is the average number of neutrons emitted by fission, which here is assumed not to depend on E' ,
- β is the total delayed neutron fraction and β_k is the delayed neutron fraction for precursors belonging to the group k ,
- Σ_f is the macroscopic fission cross section,
- λ_k is the decay constant for precursors of group k ,
- C_k is the concentration of precursors from group k ,
- Q_{ext} is the external neutron source (i.e., independent of the neutron flux ϕ , such as spontaneous fissions).

Technically, $\chi^p(E)$ and $\chi_k^d(E)$ also depend on the incident neutron energy. However, this dependence is omitted for simplicity purposes. In addition, only one fissile isotope is considered in Equation 1.10 for the sake of simplicity. For more rigor, it would be necessary to consider the contribution of all fissile isotopes composing the medium in the expression of the fission source.

Equations 1.9, 1.10 and 1.11 describe the average evolution of a neutron population over time, as if the system had an infinite number of neutrons. While this system of equations neglects statistical fluctuations inherent to the very nature of neutron-matter interactions, it allows to describe almost all reactors accurately enough, including all power reactors, due to the high number of neutrons in these systems. When the average number of neutrons remains constant over time, the system is said to be *critical*. Whereas it is *subcritical* or *supercritical* when the neutron population decreases or increases over time.

It is also technically possible to simplify the neutron transport problem to an equation that is easier to solve in some simple cases, namely the diffusion equation. The system described by this diffusion equation, however, deviates from the real system it represents. Indeed, exponential flights are replaced by Brownian motion of neutrons, which leads to discrepancies in highly absorbing systems or near interfaces between two media with different properties. Nevertheless, the diffusion equation is widely used, for example, to establish the theoretical framework of certain processes taking place inside a reactor, or to approximate the behavior of a system in short computing times.

While the average behavior allows for characterizing large systems such as power reactors, stochastic fluctuations may still remain large enough to impact the operation of smaller systems [29]. In these cases, it is therefore necessary to study fission chain reactions as stochastic processes in order to capture their variable nature. Like in ecology, the neutron population can be described as sets of individuals undergoing birth and death processes, as presented in the following section.

1.3 Stochastic transport theory : birth-death process

In 1873, Sir Francis Galton took an interest in the progressive disappearance of patronyms amongst Victorian aristocrats over time. Together with the Reverend Henry William Watson, they came with a mathematical interpretation of the process now called a Galton-Watson process in probability theory². Using a probabilistic description of human birth and death at a population scale, they were able to lay the foundation to a stochastic theory that is now used to study population dynamics. Galton-Watson processes are now a branch of branching stochastic processes, and are extensively used in various fields of applications such as ecological dynamics, genetics, and nuclear chain reactions. Indeed, the neutron population, due to absorption and branching processes such as fission events, is subject to stochastic birth and death processes. Unlike the preceding section which standpoint was to present equations accounting for the average behavior of systems, this one aims at outlining a different paradigm in which random fluctuations of the neutron population are not ignored.

To describe neutron birth and death processes, terminology borrowed from population dynamics will henceforth be used. An event leading to the disappearance of a particle (absorption, leakage out of the system) might be called a *death* (or death event) from now on. A branching event giving rise to new particles, like fission, will be called a *birth event*. Similarly to genealogy, an incident neutron inducing a fission leading to the birth of new neutrons can be called a mother particle belonging to a given *generation*, whereas neutrons born from the fission are daughter particles belonging to the next generation.

1.3.1 A simple neutron birth and death process

The aim of this section is to provide a simple example to illustrate the underlying phenomena of neutron branching processes.

Let us consider a simple stochastic system to describe a neutron population dynamics with no spatial or energy dependence, driven by fission and sterile capture only. Furthermore, no different time scale is taken into account for delayed neutron emissions, i.e., all neutrons are prompt.

Each neutron can give rise to new neutrons through fission, symbolizing a birth event. Sterile captures, on the other hand, cause the total population to decrease as they represent death events. Neutrons can thus give rise to i new neutrons upon absorption, where $i = 0$ for a sterile capture, and $i \geq 1$ for fission. The probability for a neutron to give rise to i new neutrons per unit of time is written g_i . For a time interval δt , small enough so that only one neutron of the entire population may undergo absorption, transition probabilities from N neutrons are therefore equal to

$$P(N \longrightarrow N + i - 1) = N \times g_i \times \delta t. \quad (1.12)$$

The probability of having N neutrons at $t + \delta t$ is therefore equal to the sum of all possible transitions from time t plus the probability of no transition (i.e., no absorption

²Actually, Bienaymé also developed a probabilistic explanation for the disappearance of surnames about 30 years earlier. It would seem that Galton and Watson derived their interpretation independently from Bienaymé, years later.

for any of the N neutrons). It is written $P(N, t + \delta t)$ and is equal to

$$P(N, t + \delta t) = \sum_i P(N+1-i \rightarrow N)P(N+1-i, t) + \left(1 - \sum_i P(N \rightarrow N+i-1)\right) P(N, t) \quad (1.13)$$

$$P(N, t + \delta t) = \sum_i g_i(N+1-i)P(N+1-i, t)\delta t + \left(1 - \sum_i Ng_i\delta t\right) P(N, t). \quad (1.14)$$

Thus

$$\sum_i g_i = 1. \quad (1.15)$$

Consequently

$$P(N, t + \delta t) = \sum_i g_i(N+1-i)P(N+1-i, t)\delta t + (1 - N\delta t) P(N, t). \quad (1.16)$$

The time derivative of $P(N, t)$ is thereby

$$\frac{\partial P(N, t)}{\partial t} = \sum_i g_i(N+1-i)P(N+1-i, t) - NP(N, t). \quad (1.17)$$

Equation 1.17 is called a master equation.

Average number of individuals over time

The average number of neutrons at time t is written $\langle N(t) \rangle$ and is equal to

$$\langle N(t) \rangle = \sum_n NP(n, t). \quad (1.18)$$

Its time derivative can then be derived from Equation 1.16, giving

$$\frac{\partial \langle N \rangle}{\partial t} = \sum_n \sum_i ng_i(n+1-i)P(n+1-i, t) - n^2P(n, t). \quad (1.19)$$

For each i , it is possible to replace variable n by $n' = n - i + 1$. Since it is not possible to have a negative population

$$P(n+1-i, t) = 0, \forall i > n+1. \quad (1.20)$$

The sum over n can thereby be switched with the sum over i which leads to

$$\begin{aligned} \sum_n \sum_i ng_i(n+1-i)P(n+1-i, t) &= \sum_i g_i \sum_{n'} n'(n'+i-1)P(n', t) \\ &= \sum_i g_i \sum_{n'} n'^2 P(n', t) + \sum_i g_i(i-1) \sum_{n'} n' P(n', t) \\ &= 1 \times \left(\sum_{n'} n'^2 P(n', t) \right) + \sum_i g_i(i-1) \langle N \rangle \end{aligned} \quad (1.21)$$

Reintegrating Equation 1.21 into Equation 1.19 finally leads to

$$\frac{\partial \langle N \rangle}{\partial t} = \sum_i [(i-1)g_i] \langle N \rangle = (\bar{g} - 1) \langle N \rangle \quad (1.22)$$

$$\langle N(t) \rangle = N_0 e^{(\bar{g}-1)t}. \quad (1.23)$$

To illustrate this process, a numerical example was performed. In this example, the probability for fission to happen during an infinitesimal interval dt is noted P_f and is equal to

$$P_f = \lambda_f dt \quad (1.24)$$

where λ_f is the fission rate per unit of time. Similarly, the probability for a sterile capture is defined according to the capture rate λ_c as

$$P_c = \lambda_c dt. \quad (1.25)$$

Each fission gives rise to i new particles, distributed according to the following modified Poisson distribution³ of parameter $\nu - 1$

$$p_i = \frac{(\nu - 1)^{i-1} e^{-\nu}}{(i - 1)!}, \quad i \geq 1 \quad (1.26)$$

where p_i is the probability that i new neutrons are born from fission, and

$$p_0 = 0. \quad (1.27)$$

The average number of neutrons emitted by fission is thus

$$\sum_{i=0}^{\infty} i p_i = \sum_{i=1}^{\infty} i \frac{(\nu - 1)^{i-1} e^{-\nu}}{(i - 1)!} = \nu. \quad (1.28)$$

Using the notations introduced earlier, one finds

$$g_0 = \lambda_c \quad (1.29)$$

$$g_i = \lambda_f \frac{(\nu - 1)^{i-1} e^{-\nu}}{(i - 1)!} \quad (1.30)$$

The evolution of a prompt neutron population was simulated a hundred times with different random number sequences. For the system to be critical, birth should compensate deaths on average, i.e.,

$$\lambda_c + \lambda_f = \lambda_f \nu. \quad (1.31)$$

All independent realizations are plotted in Figure 1.3. It can be observed that each fission chain follows a different evolution in time. All have their population fluctuates over time, with some tending to become extinct while others diverge with more and more individuals. Still, the average behavior represented in red is globally constant over time because the system is critical. Despite being critical, the system presented above does not remain endlessly stable. The concept of *gambler's ruin* presented below illustrates this phenomenon.

Subcritical ($\lambda_c + \lambda_f > \lambda_f \nu$) and supercritical ($\lambda_c + \lambda_f < \lambda_f \nu$) examples are also presented in Figures 1.4 and 1.5 respectively.

³In reality, the number of fission neutrons is distributed according to a Gaussian distribution [26]. For the sake of simplicity, we have used a modified Poisson distribution in this numerical model to illustrate the theoretical description presented above.

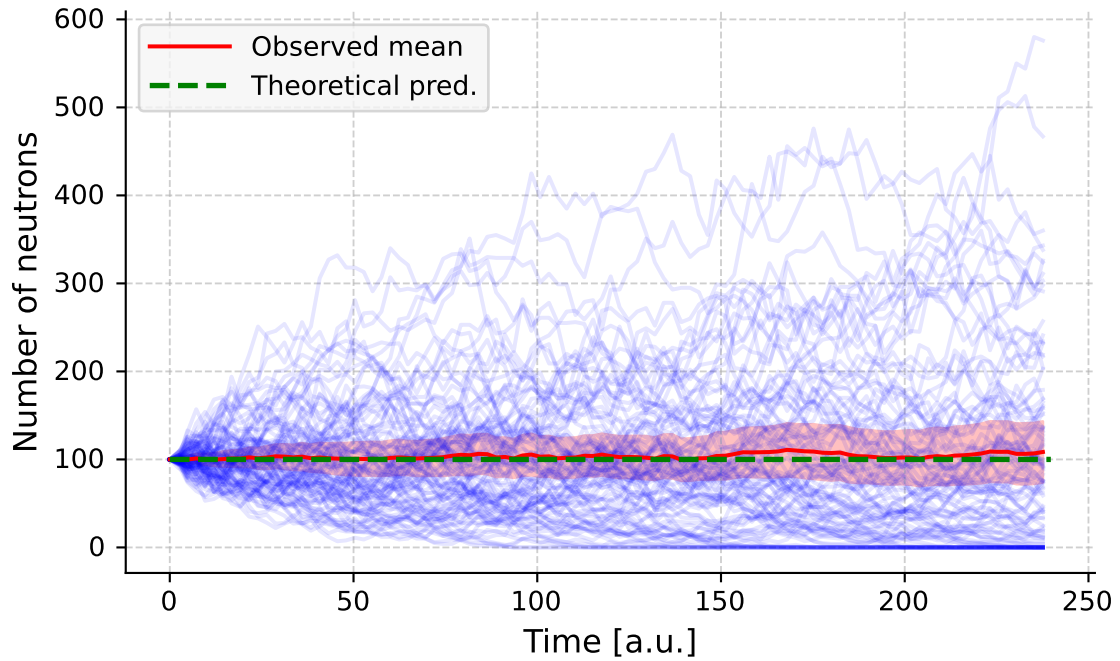


Figure 1.3: Critical Galton-Watson process for 100 independent fission chains. Each chain starts with 100 neutrons and is represented as a blue line. The red line represents the average value with 99.7% confidence intervals. The dashed green line is the theoretical average value.

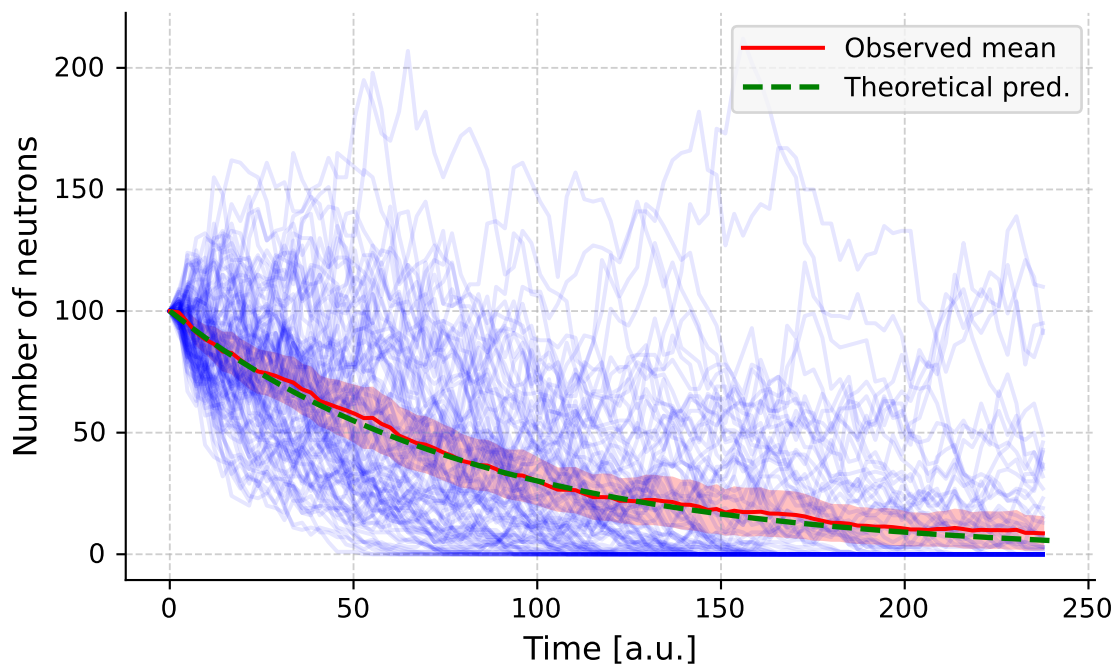


Figure 1.4: Subcritical Galton-Watson process for 100 independent fission chains. Each chain starts with 100 neutrons and is represented as a blue line. The red line represents the average value with 99.7% confidence intervals. The dashed green line is the theoretical average value.

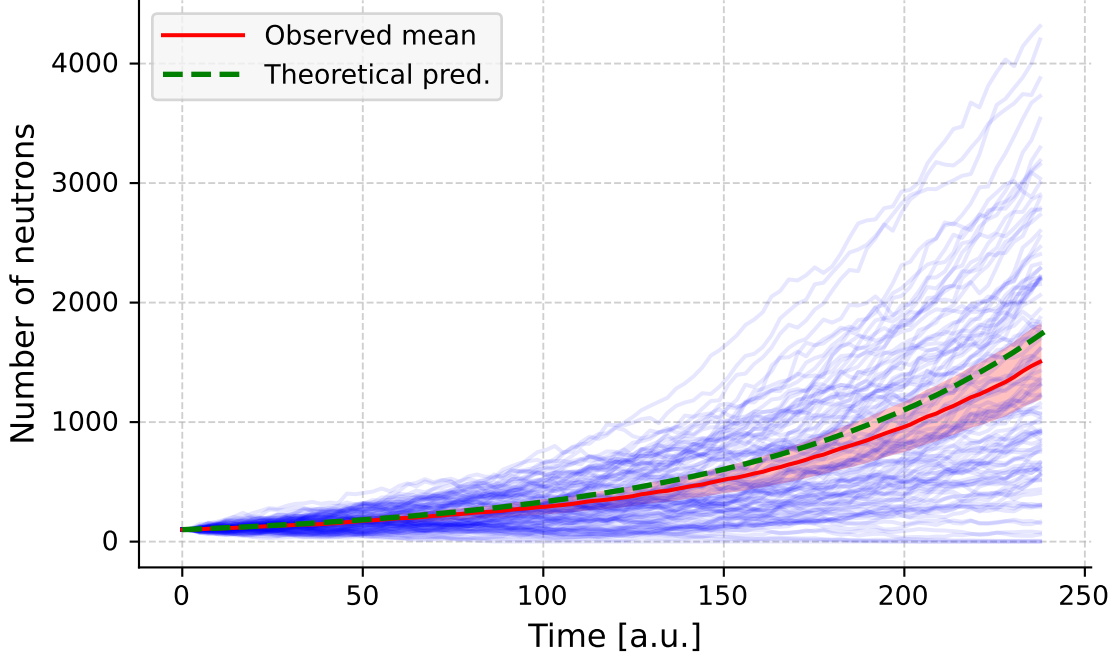


Figure 1.5: Supercritical Galton-Watson process for 100 independent fission chains. Each chain starts with 100 neutrons and is represented as a blue line. The red line represents the average value with 99.7% confidence intervals. The dashed green line is the theoretical average value.

Variance of $N(t)$

Likewise, the variance of the number of neutrons can be derived as such

$$\text{Var}(N) = \sum_n n^2 P(n,t) - \langle N \rangle^2 \quad (1.32)$$

$$\frac{\partial \text{Var}(N)}{\partial t} = \sum_n n^2 \frac{\partial P(n,t)}{\partial t} - 2\langle N \rangle \frac{\partial \langle N \rangle}{\partial t} \quad (1.33)$$

$$\begin{aligned} \frac{\partial \text{Var}(N)}{\partial t} &= \sum_n \sum_i g_i n^2 (n+1-i) P(n+1-i,t) + \sum_n n^3 P(n,t) - 2\langle N \rangle \frac{\partial \langle N \rangle}{\partial t} \\ &= \sum_n \sum_i g_i n (n+i-1)^2 P(n,t) + \sum_n n^3 P(n,t) - 2(\bar{g}-1) \langle N \rangle^2 \\ &= \sum_n \sum_i g_i n (n^2 + 2n(i-1) + (i-1)^2) P(n,t) + \sum_n n^3 P(n,t) - 2(\bar{g}-1) \langle N \rangle^2 \\ &= \sum_n \sum_i g_i (2n^2(i-1) + n(i-1)^2) P(n,t) - 2(\bar{g}-1) \langle N \rangle^2 \\ &= 2(\bar{g}-1) \sum_n n^2 P(n,t) + \sum_n \sum_i n(i-1)^2 P(n,t) - 2(\bar{g}-1) \langle N \rangle^2 \\ &= 2(\bar{g}-1) \text{Var}(N) + \sum_n \sum_i n [i(i-1) - (i-1)] P(n,t) \end{aligned} \quad (1.34)$$

Finally

$$\frac{\partial \text{Var}(N)}{\partial t} = 2(\bar{g}-1) \text{Var}(N) + \langle N \rangle [\overline{g(g-1)} - (\bar{g}-1)] \quad (1.35)$$

with

$$\overline{g(g-1)} = \sum_i i(i-1)g_i. \quad (1.36)$$

For subcritical and supercritical systems (i.e., $\bar{g} \neq 1$) the variance is equal to

$$Var [N(t)] = N_0 \left[1 - \frac{\overline{g(g-1)}}{\bar{g}-1} \right] \left(e^{2(\bar{g}-1)t} - e^{(\bar{g}-1)t} \right). \quad (1.37)$$

In the critical case (i.e., $\bar{g} = 1$), the solution of Equation 1.35 is equal to

$$Var [N(t)] = \overline{g(g-1)} N_0 t. \quad (1.38)$$

The relative standard deviation is thus equal to

$$\sigma_{rel} [N(t)] = \frac{\sqrt{Var [N(t)]}}{N_0} = \sqrt{\frac{\overline{g(g-1)}}{N_0}} t. \quad (1.39)$$

One can observe from Equation 1.38 that the variance grows linearly with time ($\overline{g(g-1)}$ and N_0 are constant). Statistical fluctuations become higher than the average number of neutrons for $t > \frac{N_0}{\overline{g(g-1)}}$ (cf. Equation 1.39).

Extinction probability of a fission chain: the gambler's ruin

Due to stochastic fluctuations, the number of neutrons in a fission chain fluctuates and can eventually reach zero. If so, there is no neutron to maintain the chain reaction and the fission chain dies.

Here, the moment-generating function is introduced to compute the probability $P(n,t)$. This function is written $\psi(z,t)$ and is defined as

$$\psi(z,t) = \sum_n z^n P(n,t) \text{ with } z \in [0,1]. \quad (1.40)$$

The k -th moment of N is thereby equal to

$$\langle N^k \rangle = \left(z \frac{\partial}{\partial z} \right)^k \psi(z,t) \Big|_{z=1}. \quad (1.41)$$

The time derivative of ψ can be deduced from Equation 1.17

$$\begin{aligned} \frac{\partial \psi}{\partial t} &= \sum_n z^n \frac{\partial P(n,t)}{\partial t} \\ &= \sum_n z^n \left[\left(\sum_i g_i (n+1-i) P(n+1-i,t) \right) - n P(n,t) \right] \\ &= \left[\sum_i g_i z^{i-1} \sum_n z^{n+1-i} (n+1-i) P(n+1-i,t) \right] - z \frac{\partial \psi}{\partial z} \end{aligned} \quad (1.42)$$

Again, replacing $n+1-i$ by n' leads to

$$\begin{aligned} \frac{\partial \psi}{\partial t} &= \sum_i g_i z^{i-1} \sum_n z^n P(n,t) - z \frac{\partial \psi}{\partial z} \\ &= \sum_i g_i z^i \frac{\partial \psi}{\partial z} - z \frac{\partial \psi}{\partial z}. \end{aligned} \quad (1.43)$$

Finally

$$\frac{\partial \psi}{\partial t} = g(z) \frac{\partial \psi}{\partial z} - z \frac{\partial \psi}{\partial z} \quad (1.44)$$

with $g(z)$ the moment-generating function for the progeny, defined as

$$g(z) = \sum_i g_i z^i. \quad (1.45)$$

$$\frac{\partial \psi}{\partial t} - (g(z) - z) \frac{\partial \psi}{\partial z} = 0 \quad (1.46)$$

The initial population is composed of N_0 neutrons. The initial condition is accordingly equal to

$$P(n, t = 0) = \delta_{N, N_0} \quad (1.47)$$

whith δ as the delta Kronecker function. This condition implies that

$$\psi(z, t = 0) = z^{N_0}. \quad (1.48)$$

From now on, a few hypotheses are made to simplify the problem and compute an analytical solution for $P(N, t)$ in the particular following case:

- $g(z) = g_0 + g_1 z + g_2 z^2$, i.e., a maximum of two neutrons can arise from a birth event,
- $g_0 = g_2 = g$, thus $g_1 = (1 - 2g)$,
- $\bar{g} = 1$, i.e., the system is critical.

$g(z)$ is now equal to

$$g(z) = g + (1 - 2g)z + gz^2 = g(z - 1)^2 + z. \quad (1.49)$$

Equation 1.46 then becomes

$$\frac{\partial \psi}{\partial t} - g(z - 1)^2 \frac{\partial \psi}{\partial z} = 0. \quad (1.50)$$

The solution of Equation 1.50 is equal to

$$\psi(z, t) = \left(1 - \frac{z - 1}{gt(z - 1) - 1} \right). \quad (1.51)$$

Doing a n -th order Taylor polynomial development of Equation 1.40 around $z = 0$ thus gives

$$\psi(z, t) = \psi(0, t) + z \frac{\partial \psi}{\partial z} + \dots + \frac{z^n}{n!} \frac{\partial^n \psi}{\partial z^n}. \quad (1.52)$$

According to Equation 1.40 (recalled here for clarity purposes)

$$\psi(z, t) = P(0, t) + zP(1, t) + \dots + z^n P(n, t), \quad (1.53)$$

it becomes clear that computing the n -th derivative according to z in $z = 0$ gives access to $P(n, t)$. For $N_0 = 1$, $P(n, t)$ is thus equal to

$$P(0, t) = \frac{gt}{gt + 1} \quad (1.54)$$

$$P(N,t) = \frac{(gt)^{N-1}}{(gt+1)^{N+1}}, \forall N > 0. \quad (1.55)$$

As it can be noted, $P(0,t)$ tends towards 1 as t tends to infinity. As time goes by, the probability of the neutron population's extinction increases.

It is possible to obtain a formulation of $P(N,t)$ for $N_0 > 1$, but it implies to derive a more complex n -th z -derivative of $\psi(z,t)$.

Going back to the numerical example previously introduced, the extinction of fission chains is illustrated in Figure 1.6 in the critical case for $N_0 = 100$, in which only one fission chain survives long times. Even this chain will disappear if one waits long enough.

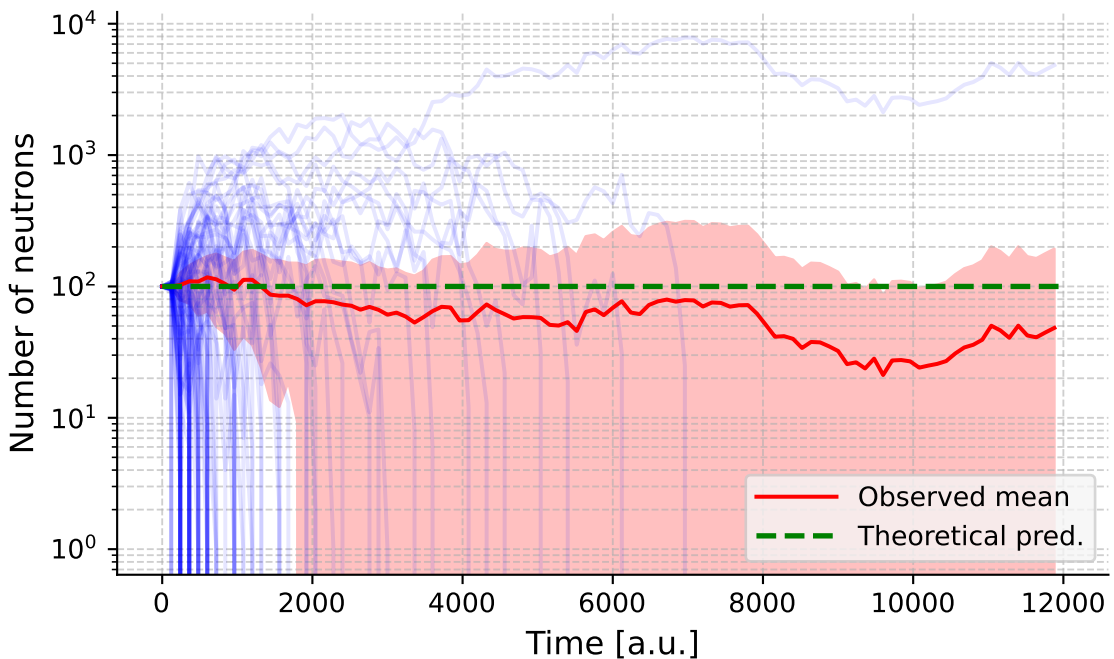


Figure 1.6: Illustration of the Gambler's ruin for a critical Galton-Watson process. The process is critical, i.e., the number of particles remains constant on average, but the asymptotic extinction probability is equal to 1. After some time, only one fission chain survives.

In the critical case, this extinction phenomenon is often called the gambler's ruin. In stochastic applications, this phenomenon can have practical implications and is sometimes being referred to as the *critical catastrophe* [30]. The critical catastrophe needs to be addressed in Monte Carlo applications in which the number of particles may be low.

1.3.2 Brownian motion

Beside the fluctuations of the neutron population over time, the spatial dependence of neutron transport can also be described as a stochastic process over space. Let us first consider a non-branching process in which particles may only move in the medium without inducing birth or death events, i.e., a stationary process. A point of a particle trajectory is noted (x,t) in the phase space, x representing its position in space and t

its position in time. The probability for a whole trajectory $(x_1, t_1); (x_2, t_2); \dots; (x_n, t_n)$ is noted $P(x_n, t_n; \dots; x_1, t_1)$. *Markov processes* are stochastic processes for which the transition probability only depends on the current state, i.e.,

$$P(x_n, t_n; \dots; x_i, t_i | x_{i-1}, t_{i-1}; \dots; x_1, t_1) = P(x_n, t_n; \dots; x_i, t_i | x_{i-1}, t_{i-1}) \quad (1.56)$$

where $P(x|y)$ is the probability to observe x considering that we previously observed y (i.e., the conditional probability). This type of process is said to be memoryless, since knowledge of states prior to the current state does not change the transition probability to the next state. In that regard, successive neutron-nucleus interactions inherently define a discrete Markov process.

The Fokker-Planck equation is a generalized partial differential equation describing variations of the probability $P(x, t)$ for stationary (i.e., $P(x_2, t_2 | x_1, t_1)$ does not depend on the absolute time but only on $\Delta t = t_2 - t_1$) markovian processes [31]. In one spatial dimension, it takes the following form

$$\frac{\partial P}{\partial t} = -\frac{\partial}{\partial x} [A(x)P(x, t)] + \frac{1}{2} \frac{\partial^2}{\partial x^2} [B(x)P(x, t)] \quad (1.57)$$

where $A(x)$ is the drift coefficient (vector in several dimensions) and $B(x)$ is the diffusion coefficient (matrix in several dimensions). In the particular case of a null average spatial drift ($A(x) = 0$) and a constant diffusion coefficient ($B(x) = D$), the Fokker-Planck equation describes *Brownian motion* (i.e., the diffusion of particles). This specific case is widely used to describe the fluctuations of a neutron population over space [32].

Combining a Galton-Watson process with the statistical description of a Brownian motion leads to a *branching Brownian motion*. This type of process can be used to describe the stochastic behavior of neutrons undergoing birth-death events while moving around in a medium according to diffusion mechanisms. This section is merely intended to introduce the concept of population fluctuations in the case of neutron transport in fissile media. Therefore, the theory of branching Brownian motions is not comprehensively described here. A few important results are introduced instead, and more practical considerations regarding the stochastic behavior of neutron transport in Monte Carlo calculations are, however, mentioned in Chapter 3.

The equation that allows computing the average density of particles inside a sub-region of space, noted $\langle n(x, t) \rangle$, in the case of a branching Brownian motion, is the following diffusion equation

$$\left(\frac{\partial}{\partial t} - 2D\vec{\nabla}^2 \right) \langle n(x, t) \rangle = (\bar{g} - 1) \langle n(x, t) \rangle. \quad (1.58)$$

Similarly to the variance equation in Section 1.3.1, one can perform the same type of derivation for the covariance. The covariance of two random variables X and Y is defined as

$$Cov(X, Y) = \langle XY \rangle - \langle X \rangle \langle Y \rangle. \quad (1.59)$$

By considering the following quantity

$$\mathcal{G}(x, y, t) = \frac{\langle n(x, t)n(y, t) \rangle}{\langle n(x, t) \rangle \langle n(y, t) \rangle} - 1 - \frac{\delta(x - y)}{\langle n(x, t) \rangle} \quad (1.60)$$

which is the relative, centered, covariance minus the self-contribution for $x = y$, one finds the following equation

$$\left(\frac{\partial}{\partial t} - 2D\vec{\nabla}^2\right)\mathcal{G}(x,y,t) = (\bar{g} - 1)\langle n(x,t)\rangle. \quad (1.61)$$

\mathcal{G} is commonly called the pair-correlation function and represents a measure of the correlations of the neutron density between two points of space at time t . Systems with naturally high spatial correlations may be subject to large fluctuations that can lead to the formation of particle clusters in extreme cases [9, 33]. While particle clustering is a naturally occurring phenomenon, particular methods used in Monte Carlo simulations may amplify the spatial correlations, and thus amplify the clustering effect. This particular issue can cause serious problems when estimating the average spatial distribution of neutrons (or the flux), and will be further discussed in Chapter 3.

As seen hereinabove, random processes governing neutron population dynamics in fissile systems can lead to fluctuations over time. In most extreme cases, fluctuations may overwhelm the average behavior of the system, the critical catastrophe being the most striking example. A more complex description of stochastic processes involved in a nuclear reactor is left aside for the moment, but can be found in References [31, 32, 34]. Further thoughts on the implications of random processes in reactor physics are, however, discussed in Chapters 3 and Chapter 3 will highlight potential issues due to random processes that may arise in stochastic numerical simulations.

Conclusion

It is clear that neutron transport is inherently stochastic. The fact that a collision occurs, leading to this or that nuclear reaction, producing a certain number of neutrons, with defined energies is purely random due to the quantum nature of the interactions taking place. Moreover, the distance between two collisions is also random, because of the lack of information about the system (i.e., too many nuclei and their unknown arrangement in space). For a large enough number of neutrons evolving in a strongly coupled medium⁴, however, it is possible to rely on average values to describe a system. Average estimates of some quantities such as the number of neutrons produced by fission, or the delayed neutron fraction, are most often used in equations and practical applications than their true distribution. In that context, the neutron transport equation, which is built upon statistical mechanics tools, is used to describe systems based on their average behavior. While allowing to simply describe complex systems in which tracking every single neutron would be impossible (which encompasses most of nuclear reactors), it fails to capture the stochastic behavior in cases where population fluctuations may overwhelm the average number of neutrons (which is particularly true when the number of particles is low). In the next chapter, numerical methods used to compute the solution of neutron transport problems are introduced. The focus

⁴*Strongly coupled medium* here refers to a medium in which the information of a local perturbation is easily transmitted to the rest of the system, as opposed to a *loosely coupled medium*.

is exclusively on characterizing the average behavior of neutron populations for now. Emphasis is made on Monte Carlo methods as they are the main topic of this thesis.

Chapter 2

Modeling of a neutron transport problem

The definition of insanity is doing the same thing over and over and expecting different results.

- Narcotics Anonymous

Witness me.

- Anonymous Monte Carlo user

Contents

2.1	Deterministic methods	28
2.1.1	Key tenets of deterministic methods	28
2.1.2	Drawbacks of deterministic methods	29
2.2	Monte Carlo methods	29
2.2.1	General principle	29
2.2.2	Estimators	30
2.2.3	Markov chains	32
2.2.4	Variance reduction and population control	32
2.2.4.1	Population control	33
2.2.4.2	Variance reduction	34
2.2.4.3	Measuring the efficiency of a variance reduction method	38
2.3	Nuclear data	39
	<i>Conclusion</i>	40

Analytically solving transport Equations 1.9 and 1.11 is either a daunting task or impossible, depending on the complexity of the geometry. Except for simple cases,

such as one or two energy group homogeneous steady-state systems in diffusion theory, solving neutronics problems generally involves numerical calculations.

For this purpose, two main families of numerical methods, namely *deterministic* and *stochastic* methods, are used. Stochastic methods are also called *Monte Carlo* methods, and are the main focus of this thesis. Both methods may be applied to a wide range of problems. In neutronics, these problems include, for instance, radiation protection or shielding problems, criticality-safety problems, depletion calculations, and steady-state or time-dependent reactor physics problems. This chapter raises some general aspects of neutron transport regarding methods used to numerically simulate it, with a particular focus on Monte Carlo methods for particle transport.

2.1 Deterministic methods

Deterministic neutron transport methods are used to solve the neutron transport (or diffusion for that matter) equation directly. They are generally described as fast but rather inaccurate in some cases, particularly due to the intricate treatment of the neutron energy and the self-shielding effect. It is also true that they are widely used in the nuclear industry for their relative speed compared to Monte Carlo methods.

Regarding transport problem solving, deterministic methods include the following methods

- the spherical harmonics (P_n) method and the simplified P_n method,
- the collision probability method and its variant the interface current method,
- the discrete ordinates (S_n) method,
- the method of characteristics (MOC).

These methods will not be detailed here, but further information can be found in the following references [35, 36]. Codes based on these methods include DRAGON [37], CASMO [38], APOLLO [39, 40], WIMS [41], DENOVO [42].

2.1.1 Key tenets of deterministic methods

To solve the transport equation, deterministic methods rely on multiple approximations that aim to simplify the problem from a numerical point of view. The main approximation, which affects all methods cited above, is the phase-space discretization. The phase space is discretized into subdomains over which physical quantities are considered constant. Regarding space for example, the real geometry is described by multiple regions in which cross sections, the flux, the temperature, etc. are homogeneous. Regarding the energy dimension, discretizing the continuous energy spectrum into energy groups is called the *multigroup* approximation. Time is also discretized when dealing with time-dependent problems. Moreover, it is the diffusion approximation, and not the transport operator, that is generally used to handle the space-dependent operators in time-dependent problems. The Boltzmann transport equation is then solved over every subdomain by accounting for the boundary conditions continuity. The overall solution is constant over each subdomain. Hence, the more subdomains, the more detailed the solution.

2.1.2 Drawbacks of deterministic methods

Among all approximations, the multigroup discretization is probably the most debated one as cross sections must then undergo the *resonance self-shielding* procedure (self-shielding for short) [35]. Without resonance self-shielding, the multigroup cross-section approximation would lead to errors in the solution produced.

Theoretically, it would be possible to bring discretized deterministic resolutions closer and closer to the real continuous system by increasing the number of subdomains in the discretization. However, deterministic methods suffer from the *curse of dimensionality*. Indeed, since there are as many equations to solve as subdomains, the more subdomains the more time is required to solve the entire problem. For steady-state systems, the most detailed deterministic calculations are comparable to Monte Carlo calculations both in terms of accuracy and computation time.

2.2 Monte Carlo methods

Considerations on Monte Carlo methods date back to a few centuries (although mentioned under the designation *statistical sampling* back then) with the so-called Buffon's needle problem. Later, in the 1940s, Monte Carlo based algorithms were among the first ones to be tested on the ENIAC (Electronic Numerical Integrator and Computer) [43], in the case of neutron transport. Since then, they have been one of the main motivations for the development of extensive numerical calculations, and have been used in many different fields from physics to chemistry, biology or finance.

Monte Carlo methods rely on multiple random sampling to estimate the average behavior of the observable that is sought. These methods allow representing physical problems with high fidelity, which makes them particularly appropriate to perform high accuracy quantitative studies on complex systems. Indeed, their use does not rely on coarse mesh approximations unlike deterministic methods. Monte Carlo methods do not explicitly solve equations such as the transport equation presented earlier (cf. Equations 1.9, 1.10 and 1.11), but rather aim at giving an estimation of the solutions of these equations based on multiple random sampling.

Due to the random nature of particle transport and interaction with matter, Monte Carlo methods naturally fit the simulation of neutronics systems. For particle transport, it can intuitively be represented as modeling the natural behavior of many particles and deducing how the studied system should behave in reality. This section aims at presenting general tenets regarding Monte Carlo simulations, especially for particle transport, before diving further into neutron criticality and kinetics calculations in the next chapters.

2.2.1 General principle

In its simplest form, the problem boils down to estimating an integral value R defined by

$$R = \mathbb{E}[z(X)] = \int z(x)f(x)dx \quad (2.1)$$

where $z(X)$ is a function of the random variable X , and f is the probability density for variable X . The integral is defined for all possible values of X . For example, z could be the response function of a detector depending on the neutron distribution over the

phase space, and f the distribution of neutrons over the same phase space. In that case, R would be the reaction rate associated with the detector response function for a system in which neutrons would be distributed according to f .

To estimate the value of R with a Monte Carlo method, a sample of size n is first randomly drawn according to $f(x)$. Its average value is computed using the following estimator

$$\bar{z}(n) = \frac{1}{n} \sum_{i=1}^n z(X_i) \quad (2.2)$$

where $z(X_i)$ is the reaction rate observed in the i -th realization of the sample. The estimation of R is then based on two fundamental theorems of probability theory, the *law of large numbers* and the *central-limit theorem* (CLT). According to the law of large numbers, the estimator $\bar{z}(n)$ gets closer to R as the sample size increases, i.e.,

$$\lim_{n \rightarrow \infty} \bar{z}(n) = R. \quad (2.3)$$

The central-limit theorem allows accessing the statistical uncertainty of the estimate $\bar{z}(n)$ by interpreting the statistical fluctuations of the random sampling, i.e., the variance of $z(n)$. Thus, without having access to the true value of R , it is possible to have an estimation of the precision of its estimate $\bar{z}(n)$. The classical CLT states that for a sample size n large enough, the distribution of $\bar{z}(n)$ estimates tends to a Gaussian distribution of mean R and standard deviation $\sigma(z(n))/\sqrt{n}$. If all variables are sampled from the same distribution of variance σ^2 , the variance of the mean is thus equal to

$$\text{Var} [\bar{z}(n)] = \frac{\sigma^2}{n}. \quad (2.4)$$

2.2.2 Estimators

Since the aim is to extrapolate the behavior of a small population of particles to the behavior of a system with a larger population, interpreting the results of a Monte Carlo simulation falls into the domain of inferential statistics. Monte Carlo methods require the use of statistical estimators to interpret the results of a calculation. Estimators can be biased or unbiased. Let us consider a quantity θ and an estimator T used to estimate the real value of θ . The estimator T is unbiased if its expected value is equal to the real quantity θ , i.e.,

$$\mathbb{E}[T] = \theta. \quad (2.5)$$

On the other hand, T is said to be biased if its expected value does not converge towards θ . The main estimators used are those of the mean and variance, which are presented below.

Let X be a random variable, with n realizations $(X_i)_{i \in [1;n]}$, identically and independently distributed according to a given distribution with mean μ and variance σ^2 . The following estimators are presented in a more general context that of Monte Carlo methods. Therefore, the function z will be omitted here for the sake of simplicity, as it would be rather cumbersome. The average is usually computed using the following unbiased estimator

$$\bar{X}(n) = \frac{1}{n} \sum_{i=1}^n X_i, \text{ with } \mathbb{E}[\bar{X}] = \mu. \quad (2.6)$$

For a random variable X , the variance of the mean computed from n observations is thus computed as such

$$\text{Var} [\bar{X}(n)] = \frac{1}{n^2} \text{Var} \left[\sum_{i=1}^n X_i \right]. \quad (2.7)$$

If the X_i are independent, $\text{Var} [\sum_{i=1}^n X_i] = \sum_{i=1}^n \text{Var} [X_i]$ and since the X_i are supposed to be identically distributed, we have

$$\text{Var} [\bar{X}(n)] = \frac{\sigma^2}{n}, \quad (2.8)$$

The estimation of the variance¹, here noted σ^2 , is empirically estimated using the following unbiased estimator

$$\sigma^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}(n))^2. \quad (2.9)$$

$\frac{\sigma}{\sqrt{n}}$ is often referred to as the *standard error of the mean*, or *error of the mean*, and represents the statistical uncertainty on the average score computed using Monte Carlo methods.

Therefore, supposing that the CLT can be applied to the sample $(X_i)_{i \in \llbracket 1; n \rrbracket}$, i.e.,

$$\bar{X} \sim \mathcal{N}(\bar{X}(n), \sigma^2(n)) \quad (2.10)$$

where $\mathcal{N}(\mu, \sigma^2)$ denotes a normal law of mean μ and variance σ^2 , the estimation of confidence intervals for $\mathbb{E} [\bar{X}(n)]$ are based on those of $\mathcal{N}(\bar{X}(n), \sigma^2(n))$. For example

$$\begin{aligned} \mathbb{P}(\bar{X}(n) - \sigma(n) \leq \mu \leq \bar{X}(n) + \sigma(n)) &\approx 67\% \\ \mathbb{P}(\bar{X}(n) - 2\sigma(n) \leq \mu \leq \bar{X}(n) + 2\sigma(n)) &\approx 95\% \\ \mathbb{P}(\bar{X}(n) - 3\sigma(n) \leq \mu \leq \bar{X}(n) + 3\sigma(n)) &\approx 99.7\%. \end{aligned} \quad (2.11)$$

These are the so-called *error bars* sometimes mentioned. They represent the statistical uncertainty of the average estimation.

For correlated variables, however, Equations 2.8 is not true anymore and correlations between variables must be accounted for. Thus with ρ_{ij} being the correlation factor between the variable X_i and X_j , the unbiased estimation of $\text{Var} [\bar{X}(n)]$ is

$$\text{Var} [\bar{X}(n)] = \frac{\sigma^2}{n^2} \left[n + 2 \sum_{i=1}^{n-1} \rho_{ij} \right]. \quad (2.12)$$

For more details on its derivation, see Appendix A.1.

¹Usually in mathematics, the notation " σ^2 " refers to the true variance of the random variable X , and " s^2 " is often used to identify the estimate of σ^2 . Here, we will use σ^2 to refer to the true variance or its unbiased estimate indifferently for the sake of simplicity.

2.2.3 Markov chains

Having direct access to the probability density $f(x)$ in Equation 2.1 might prove to be difficult for multidimensional integrals or when complex physical processes are involved. It is, however, possible to use Markov chains properties to build a process leading to f [44]. As explained in the previous chapter, neutron transport is a natural Markov process because the probability distribution of possible events only depends on the current state of the particle. As a matter of fact, Monte Carlo particle transport simulations rely on this specific feature to estimate the physical response of a system to a particle population. A succession of events, e.g., neutron-nucleus collisions is called a Markov chain (which is discrete with regard to collisions). By simulating the natural random path of neutrons inside a reactor, one can build a Markov chain and access the distribution f without much mathematical effort. The natural simulation of neutrons can be rather difficult, though. Indeed, the precise estimation of the value of R may require a large number of random samples, and thus a heavy computational workload. To increase numerical simulation performances, mathematical methods have been introduced to reduce the variance of $\bar{z}(n)$ compared to a natural simulation, for the same sample size n .

2.2.4 Variance reduction and population control

The natural simulation of particle transport with Monte Carlo methods is commonly referred to as an *analog* simulation. In many cases, it can be troublesome to perform with regard to the computational capacity available. It is possible to improve analog simulations by changing natural laws according to which events are sampled during a Monte Carlo calculation, to reduce the variance of the estimations compared to the natural variance of the process. Most often, one is concerned with just an accurate estimate of \bar{g} , and preserving higher-order moments is not necessary. In some cases, however, the preservation of higher order statistical moments may also be of interest. Particles must carry a statistical weight that is modified according to mathematical rules so that the *non-analog* alterations coupled with particle weight modifications result in an unbiased estimation of \bar{g} . These simulations are sometimes called *non-analog* simulations, *weighted* Monte Carlo game or rarely *biased*² simulation.

Three distinct categories are used in this thesis to classify these methods. In other works, different definitions may be used to categorize methods [45].

- *Population control methods*: these methods are used to avoid numerical problems due to large fluctuations in the particle population that could prevent the calculation from completing properly. Unlike variance reduction methods, the gain of efficiency regarding variance estimates is not the main motivation and may not be important.
- *Weight control methods*: they control particle weights to limit variance jumps due to high disparities among particle weights. Several population control methods also control the weight of particles (e.g., the combing method presented below).

²This designation is rather unfortunate as it may cause ambiguity regarding the term *biased*. Indeed, this refers to the fact that the natural probability distributions are modified, biased, and not that there is an error, a bias, in the estimation of the result.

- *Variance reduction methods*: their goal is to improve simulation performances by reducing the variance of \bar{g} without impacting the calculation time (or decreasing the calculation time without increasing $Var[\bar{g}(n)]$).

Depending on how a method is implemented and used, it can belong to one or several categories at the same time.

2.2.4.1 Population control

Population control techniques aim at preventing the simulation from dying out of particles when the population decreases, or the calculation time and memory from diverging due to too many particles. To do so, it either deletes or samples particles to bring the total population from N particles to M particles without changing the total weight of the population (but taken separately, the weight of each particle can change). Two methods that are extensively used in Monte Carlo calculation for particle transport are presented here, the Russian Roulette (RR) / splitting combination, and the combing method.

Russian Roulette / splitting

The Russian Roulette and splitting methods are some of the fundamental building block of many methods used in Monte Carlo simulations (including population control, variance reduction or weight control methods). Their mechanism for population control is as follows. Let W_{TOT} be the total weight of a population of N particles, defined by

$$W_{TOT} = \sum_{i=1}^N w_i, \quad (2.13)$$

where w_i is the weight of particle i . The mean weight of the M particles after population control is noted \bar{w} and is equal to

$$\bar{w} = \frac{W_{TOT}}{M}. \quad (2.14)$$

To control the population, the algorithm loops over the N initial particles to either split or delete it depending on their weight w_i .

- If $w_i < \bar{w}$, particle i is either discarded or survives with a survival probability $P_{\text{surv}} = \frac{w_i}{\bar{w}}$. If it survives, its weight is set to \bar{w} . This operation is called Russian Roulette (RR).
- If $w_i > \bar{w}$, particle i is split in $\lfloor \frac{w_i}{\bar{w}} \rfloor \pm 1$ particles of weight \bar{w} . The ± 1 is due to the remainder of w_i/\bar{w} which undergoes RR with survival probability $P_{\text{surv}} = \frac{w_i - \lfloor w_i/\bar{w} \rfloor \bar{w}}{\bar{w}}$. Here, $\lfloor x \rfloor$ is the floor operator and returns the integer part of x .

Due to statistical fluctuations, the number of particles after population control might not be strictly equal to M . Therefore, W'_{TOT} may also not be strictly equal to W_{TOT} . However, its expected value is equal to W_{TOT} .

$$\mathbb{E}[W'_{TOT}] = M \times \bar{w} = W_{TOT}. \quad (2.15)$$

A variant of this method called the Weight Window (WW) technique is often used to limit the spread of particles weights in weighted Monte Carlo simulations.

- When the weight of a particle i gets below a threshold noted w_{min} , the particle undergoes RR with a probability of survival $P_{surv} = \frac{w_i}{w_{min}}$.
- On the other hand, when its weight gets above a threshold noted w_{max} , it is split in $\lfloor \frac{w_i}{w_{max}} \rfloor$ particles while the remainder undergoes RR if it is lower than w_{min} .

Following this definition, the WW technique can be classified among the weight control methods.

The combing method

The combing method has been used for decades, but the older reference to this technique in literature dates back to the 1990s with the work of Booth [46]. The idea is also to split or discard particles but the process is slightly different from the RR/splitting technique. In its original version, a comb is built with evenly spaced teeth and is passed through the particle population (hence the name of the method) as illustrated in Figure 2.1. Particles are lined up according to their weight to form a segment of

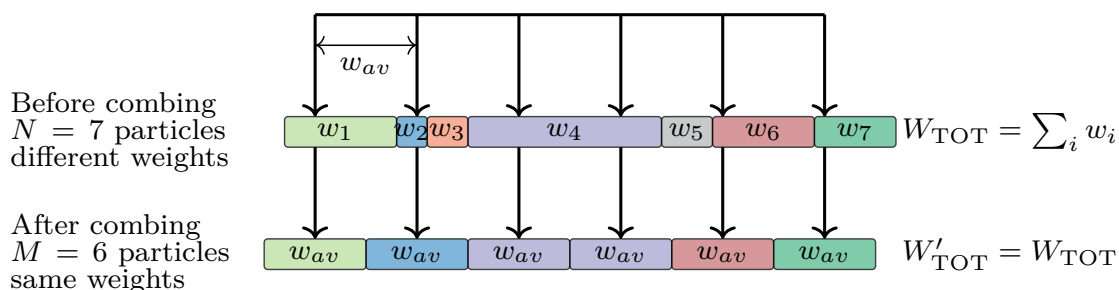


Figure 2.1: Combing method

length W_{TOT} , and teeth of the comb hit a position every $w_{av} = \bar{w}$, where \bar{w} is equal to W_{TOT}/M as in the RR/splitting method. Each time a particle is hit by a tooth, it is sampled to survive the population control and given a weight equal to \bar{w} . Unlike the RR/splitting method, exactly M particles are selected for a total weight of W_{TOT} . The position of the first tooth is set in $[0; \bar{w}]$ by uniform random sampling to ensure randomness of the method.

Variations of this method exist to improve its efficiency regarding correlated particles (if particles were to be ordered for some reason when lined up, e.g., randomly or because of the way the method is specifically implemented). For example, it is possible to introduce particle importance to favor some particles over others to do variance reduction [46, 47] (see the following section for hints on importance and variance reduction).

2.2.4.2 Variance reduction

In Monte Carlo simulations, one is interested in obtaining an average score representative of the average behavior of the system that is modeled with a given statistical precision. In theory, the more precise the result, the better. Surely, it is possible to reduce the variance of a Monte Carlo score by increasing the number of independent observations, i.e., the number of particles or independent simulations as stated by the Central-Limit Theorem (CLT). However, this implies an increase of computational costs (memory or computation time, but mostly computation time) because of

the extra particles that are simulated. Algorithmic methods have therefore been implemented in Monte Carlo codes to increase the score precision. These methods aim at reducing the variance without much impact on computational time (compared to simply increasing the number of simulated particles) by modifying the rules by which particles are sampled during the simulation.

Non-analog collisions

Implicit capture The implicit capture method consists in replacing capture reactions by scattering reactions. This way, particles that should have been terminated are allowed to survive to contribute to the score later. To account for the decrease in the number of particles that would normally be caused by captures, and still compute unbiased results, neutron weights are modified at each collision in the following manner

$$w' = w \times \left(1 - \frac{\Sigma_c}{\Sigma_{tot}}\right) \quad (2.16)$$

where Σ_c and Σ_{tot} are the total and capture macroscopic cross section of the medium in which the collision happened.

Branchless collision The branchless collision method is used to reduce the variance between fission chain lengths. It is usually implemented as a combination of implicit capture and full forced fission so that exactly one outgoing neutron is sampled at each collision.

The modified probabilities to induce fission or undergo scattering during a collision are respectively noted P_{fiss} and P_{scat} and are equal to

$$P_{\text{fiss}} = \frac{\nu\Sigma_f}{\nu\Sigma_f + \Sigma_s} \quad (2.17)$$

$$P_{\text{scat}} = \frac{\Sigma_s}{\nu\Sigma_f + \Sigma_s} \quad (2.18)$$

where Σ_s and Σ_g are the macroscopic cross sections for scattering and fission reactions, and ν is the mean number of neutrons produced per fission. To ensure unbiased estimates of tallies, the outgoing particle weight must carry the multiplicity of particles that should have been produced/absorbed. The weight of the particle is thus modified at each collision as such

$$w' = \frac{\nu\Sigma_f + \Sigma_s}{\Sigma_{tot}} w \quad (2.19)$$

where w and w' are the particle weight before and after the collision and Σ_{tot} is the total macroscopic cross section of the material in which the collision occurred. For non-fissile materials, $\Sigma_f = 0$ and the branchless collisions behave as implicit captures.

Splitting based methods

A criterion can be defined to differentiate particles based on their chances to contribute to the score. Based on this criterion, usually called *importance*, some particles may be favored over others to improve estimates. Indeed, particles that have a low probability to contribute most often vanish before reaching the detector and were thus simulated

for nothing. The idea is, therefore, to simulate as many important particles as possible, and as few uninteresting particles as possible.

Unlike the statistical weight, which is tethered to a unique particle (each particle carries its own weight), the importance is bound to a specific point of the phase space. All particles located at the same point of the phase space have thus the same importance.

The following techniques are examples of methods based on the use of an importance map to discriminate particles and favor the ones most likely to contribute to the score of interest.

Importance based RR / splitting Like the RR and splitting techniques presented above, the idea is to duplicate or remove particles, but this time based on their importance regarding the score in a detector.

For this method, the whole domain has to be subdivided into sub-regions, each of which is assigned an importance value. When a particle goes from a region of importance I_1 to a region of importance I_2 , there are two possible outcomes.

- If $I_1 < I_2$, the particle moves from a less important region to a more important region, and it is split into $n = \lfloor \frac{I_2}{I_1} \rfloor$ particles of weight w/n (where w is the particle weight prior to splitting). One extra particle may be created with probability $\frac{I_2}{I_1} - \lfloor \frac{I_2}{I_1} \rfloor$.
- If $I_1 > I_2$, the particle undergoes a Russian Roulette with a survival probability equal to $P_{\text{surv}} = \frac{I_2}{I_1}$. If the particle survives, its weight is multiplied by I_1/I_2 .

Adaptive Multilevel Splitting The Adaptive Multilevel Splitting (AMS) is a variance reduction technique based on the splitting of particle histories. It consists of the analog simulation of the process for all particles of the simulation until they all disappeared. Particles are then ranked according to an importance criterion, and a part of the most important ones may be split at some point to start a new analog simulation process with the newly split particles. The above procedure is conducted over successive iterations until enough particles have reached the detector. It is adaptive in the sense that levels at which particles will be split are not defined beforehand, but along the simulation course. This technique has been widely used in the context of particle transport in this thesis, and is more detailed in Chapter 4.

Importance sampling

Suppose that one seeks to compute the value of R (cf. Equation 2.1) using estimator of Equation 2.2. It is first necessary to sample the X_i from the probability density function (PDF) defined by $f(x)$. In the case where only a few X_i induce $z(X_i) > 0$, i.e., only a few samples contribute to the score, a lot of time is spent to simulate useless particles. Importance sampling consists in modifying the PDF $f(x)$ to increase the number of particles contributing to the score estimation, for the same number of total particles simulated. The computation of R thus becomes

$$R = \int z(x)f^*(x)dx \tag{2.20}$$

where $f^*(x)$ is the modified PDF. Though, this equation is wrong, and the estimation of R is biased if the z is not modified according to the following

$$z^*(x) = z(x) \frac{f(x)}{f^*(x)}. \quad (2.21)$$

Equation 2.20 thus becomes

$$R = \int z^*(x) f^*(x) dx \quad (2.22)$$

$$R = \int z(x) \frac{f(x)}{f^*(x)} f^*(x) dx = \int z(x) f(x) dx. \quad (2.23)$$

The factor $f(x)/f^*(x)$ is used as a weight that is applied to a particle when sampled from the modified PDF $f^*(x)$ so that the result of the Monte Carlo simulation remains unbiased. This way, it is possible to sample the values of x from a modified PDF which increases the number of sampled values in a region of interest (i.e., where $z(x) > 0$), artificially reducing the variance on the estimate of R without increasing the global size of the sample. This method is called importance sampling, because it gives more importance than natural kernels to certain regions when randomly sampling.

Exponential biasing The probability distribution according to which the distance traveled by a neutral particle between two collisions is of the form

$$p(d) = \Sigma_{tot} e^{-\Sigma_{tot} d} \quad (2.24)$$

where Σ_{tot} is the total macroscopic cross-section of the medium and d is the traveled distance. In some cases, it is more interesting to sample more or fewer collisions along the path of a particle (e.g., when a particle is heading towards a detector, one might wish to sample fewer collisions that might prevent it from reaching the target). It is suitable to modify the flight length kernel $p(d)$ as follows to achieve this goal

$$p^*(d) = (1 - \beta\omega) \Sigma_{tot} e^{(1-\beta\omega)\Sigma_{tot} d}. \quad (2.25)$$

ω is the cosine of the angle between the preferred direction and the current direction of the particle flight, and β is a biasing parameter. To avoid negative values, β must be taken in $[0; 1]$. This strategy is commonly called *exponential biasing* or *path stretching* and was initially proposed by Kahn [48]. Once again, a weighting function must be applied to the particle weight for an unbiased estimation of the result

$$w' = w \times \frac{e^{\beta\omega\Sigma_{tot} d}}{1 - \beta\omega} \quad (2.26)$$

where w is the weight of the particle before the modified flight and w' is the weight of the particle after the flight.

Zero-variance schemes

Seeking optimal schemes that would lead to the minimum variance has been a topic of interest for decades [49, 50, 51, 52, 53, 54], and still are [55, 56, 57]. These theoretical schemes were designed to lead to zero-variance estimators.

In the case where $z(X_i) = \bar{z}(n)$, $\forall i \in \llbracket 1; n \rrbracket$, the variance is equal to 0

$$Var [\bar{z}(n)] = \frac{1}{n} \sum_i (z(X_i) - \bar{z}(n))^2 = 0. \quad (2.27)$$

Equation 2.27 implies that all random realization of the process gives the exact same estimation of the score, which is equal to the solution that is sought. Zero-variance schemes thus require knowing the result that is sought beforehand to optimally sample particles, which does not make sense in practice. Nonetheless, these theoretical considerations aim at presenting optimal strategies so to inspire more practical Monte Carlo calculation schemes, which can rely on approximations of the solution to accelerate the main Monte Carlo calculation.

Other variance reduction methods exist but are not listed here as they will not be mentioned later in this thesis. For more information, the reader can consult existing literature on the matter, such as Refs. [54, 44, 45].

2.2.4.3 Measuring the efficiency of a variance reduction method

The purpose of variance reduction technique is to reduce the variance of an estimate. Measuring the efficiency of a method is therefore equivalent to measuring the reduction in variance resulting from its use. However, these methods can impact the calculation time depending on their implementation. For example, the AMS resamples particles iteration after iteration, which increases the total number of particles that are simulated. Therefore, how to differentiate the effect of the method from the effect of simulating more particles ? The Figure of Merit (FoM) is an indicator that aims at measuring the efficiency of a method irrespective of the number of particles that are sampled. It is defined by

$$FoM = \frac{1}{\sigma^2 T_{\text{calc}}} \quad (2.28)$$

where σ is the standard error of the mean for the score of interest and T_{calc} is the total simulation CPU time. From this definition, it appears that decreasing the variance without impacting the calculation time increases the FoM, as does decreasing the calculation time without changing the variance. In that sense, the FoM measures how optimal the compromise is between calculation time and precision for a simulation.

Besides, since the computation time T_{calc} can be considered proportional to the number of particles that are sampled N , and σ is proportional to $1/\sqrt{N}$ according to the CLT, the FoM is therefore independent of N . This way, it is possible to compare the efficiency of two methods even when the simulations were performed with different numbers of sampled particles. Nonetheless, the FoM does not indicate if a method is efficient in absolute since the variance and the simulation time both depend on the system modeled, the computer features, etc., but only allow evaluating a method relatively to another method. For this reason, its absolute value is never used on its own, and it must always be compared to the FoM of another calculation. The ratio FoM_1/FoM_2 thus gives the relative efficiency of method 1 compared to method 2.

Compared to deterministic methods which give a macroscopic result of neutron transport through the resolution of the Boltzmann transport equation, Monte Carlo methods simulate the microscopic interactions of neutrons to statistically estimate the macroscopic behavior of a system. This approach aims at reproducing "reality" by using probability distributions to sample the outcome of random events induced by neutrons. These methods thereby have several advantages over deterministic approaches. Since they model the microscopic motion of neutrons, there are no requirements regarding the discretization of the phase space. For this reason, Monte Carlo calculations do not suffer from the curse of dimensionality and allow to obtain accurate results compared to the real system. However, since the accuracy of the result is based on the central-limit theorem and decreases according to $\frac{1}{\sqrt{N}}$ (where N is the number of particles), getting a more precise result implies simulating more particles. And since simulating more particles implies sampling more random events, calculation costs (memory footprint and CPU time) also increase. To get an accurate result in regions with a low neutron density may thus require expensive calculations. As a result, Monte Carlo methods are generally used to compute reference calculations, which may in turn be used to validate deterministic methods. It is, however, possible to implement methods, namely variance reduction methods, to improve the efficiency of Monte Carlo methods.

2.3 Nuclear data

Whereas macroscopic cross-sections depend on the density of the material, and therefore on the design of the system being modeled, microscopic energy-dependent cross-sections depend only on the nature of the nucleus being considered. They are therefore the basic building block for characterizing a material in neutronics calculations and are measured in *differential* experiments. But before obtaining effective cross-sections that can be used in input of calculation codes, several steps are necessary [28].

Theoretical nuclear physics models and mathematical tools are used to compute *evaluations* of nuclear data (for example resonance parameters for cross-sections) over the whole energy domain from experimental data. This process (i.e., *evaluation* of nuclear data) is performed to compute a single value from several measurements (for the entire energy spectrum) of the same cross-section, and to overcome the lack of data for energies for which measurements are not available. Several different evaluations are done and shared with the international community, such as JEFF (Joint Evaluated Fission and Fusion nuclear data library - Europe) [58], ENDF (Evaluated Nuclear Data File - USA) [59], JENDL (Japanese Evaluated Nuclear Data Library - Japan) [60], TENDL (TALYS-based Evaluated Nuclear Data Library) [61]. Note that the choice of the evaluation can strongly impact the neutronics calculation results. The evaluation process also allows for nuclear data uncertainties to be calculated based on measurement uncertainties and evaluation methods. These uncertainties also impact neutronics calculation results, and efforts are made to propagate these uncertainties along the whole calculation chain up to the final result [62, 63]. The evaluation of nuclear data is an area of expertise in its own, it is carried out using numerical codes such as CONRAD [64], TALYS [65] or SAMMY [66].

Once neutron nuclear data have been evaluated, they need to be processed to be readable by neutronics solvers, using codes like NJOY [67], FRENDDY [68] or AMPX

[69]. This processing stage aims at providing pointwise or multigroup cross-section values, as well as other data (thermal scattering law data, the average number of fission neutrons, ...) at a specific temperature for example which can then be used by neutronics codes. Both deterministic and Monte Carlo codes generally use the same evaluations of nuclear data, except that deterministic codes may require extra processing due to the multigroup approximation.

This brief description of the nuclear data is only intended to give a rough introduction to the link between reality, and the data used by the simulation codes to represent it. Despite being considered to be the main source of uncertainty regarding the result accuracy, the impact of nuclear data evaluations, or their uncertainties goes beyond the scope of this thesis which focuses on Monte Carlo simulation methods. Henceforth, further reflections on neutron-nucleus cross sections and their implications in neutron simulations will not be included in this thesis.

Conclusion

Solving realistic neutronics problems involves using numerical methods to analyze complex systems. Two different approaches are used to achieve this task, namely deterministic and Monte Carlo methods. While deterministic methods are relatively fast in computing the solution of a neutron transport problem in systems such as nuclear reactors, they may have some shortcomings regarding their accuracy compared to Monte Carlo methods. As a matter of fact, Monte Carlo methods make little use of approximations compared to deterministic solvers. Their interpretation is based on the law of large numbers, which allows their uncertainty to converge to 0 as the number of simulated particles tends to infinity. The only sources of remaining uncertainty are due to input data uncertainties, i.e., mainly nuclear data uncertainties, and technological uncertainties. For this reason, Monte Carlo calculations are generally used to serve as reference for the validation of deterministic calculation schemes. However, Monte Carlo simulations are usually way more expensive than deterministic calculations. For some uses, it may in fact be prohibitive, even with modern technical advances in computing power. This is why it is sought to improve their performances without degrading the result using techniques such as variance reduction methods.

The work that follows is about variance reduction in Monte Carlo reactor physics simulations, with particular focus on the Adaptive Multilevel Splitting (AMS) method. Next chapter will thereby focus on two types of reactor physics calculation, namely criticality calculations and kinetics calculations, and will present characteristics of these calculations that motivate the use of variance reduction methods such as the AMS.

Chapter 3

Reactor physics with Monte Carlo methods

If knowledge can create problems, it is not through ignorance that we can solve them.

- Isaac Asimov

Contents

3.1	Steady-state criticality calculations	42
3.1.1	The k -eigenvalue equation	43
3.1.2	Power Iteration method	44
3.1.3	Issues regarding the convergence of the solution	45
3.1.3.1	Convergence of neutron sources	46
3.1.3.2	Biased estimation of the eigenvector variance	47
3.1.3.3	Biased estimation of the average eigenvector	48
3.2	Time-dependent neutron transport	49
3.2.1	A brief history of Monte Carlo neutron kinetics	50
3.2.2	Coupled neutron-precursor simulations with stochastic methods	50
3.2.2.1	Delayed neutron precursors decay	51
3.2.2.2	Handling population fluctuations over time	52
3.2.3	Current limitations to industrial applications of dynamics calculations	54
	<i>Conclusion</i>	54

Variance reduction methods presented in Chapter 2 were mainly developed in the context of fixed-source calculations. Indeed, in such calculations, one is generally interested in calculating the score produced in a localized detector by a constant source of particles, such as a dose rate at a specific location for radiation protection purposes,

for example. Blindly simulating every particle could prove to be inefficient in cases where a lot of particles would never reach the detector. It would therefore be beneficial not to waste time simulating particles having low probabilities to contribute to the score in that particular detector, in order to focus on more useful ones. Intuitively, using methods based on importance sampling for example, where some particles will be sampled more often based on their position in the phase space to enhance the estimation of the score, seems to be a good option. If variance reduction techniques are predominately implemented for stationary source calculations, it may be because understanding the root issue behind a lack of efficiency of analog simulations can often be straightforward when only a fraction of particles contribute to the detector: some of the simulated particles will never contribute to the detector.

It is much less obvious, on the other hand, to intuit limiting factors in reactor physics calculations, since one is generally interested in the behavior of a fissile system over the whole phase space (or almost), and thus all the simulated particles can potentially contribute to the score. However, some methods might be well adapted to these calculations, such as the branchless collision method (cf. Section 2.2.4.2), especially in kinetics. Nevertheless, if it is less intuitive to set up variance reduction schemes for steady-state and kinetics calculations, the current methods show some limitations calling for more efficient modeling techniques. In this chapter, we will try to highlight features in criticality and kinetics calculations that motivate the use of variance reduction techniques, and more precisely the Adaptive Multilevel Splitting method.

3.1 Steady-state criticality calculations

Neutron transport theory is used to describe neutronics systems in various ways, each way relying on a different variation of the linear transport equation. While Equations 1.9 and 1.11 represent the system evolution through time, the majority of neutronics calculations in reactor physics are done over static systems. For decades, criticality calculations have been used to characterize the proximity of a system to criticality [70, 71, 72, 73, 74], by solving the so-called Boltzmann critical equation, or k -eigenvalue equation. This equation interprets the system in the form of an eigen-problem for which external neutron sources (i.e., which do not depend on the current neutron flux, as for instance spontaneous fissions) are neglected. To ensure that the system is static (and therefore cancel the term $\frac{\partial \phi}{\partial t}$ in Equation 1.9), the fission production term in Equation 1.10 is modified, revealing an eigenvalue equation. In spite of fundamental questions related to the inner nature of the problems that is solved [75]¹, this method has made consensus and is now largely used not only in nuclear criticality-safety but also in reactor physics. An abuse of language, albeit widespread, is to talk about criticality calculations in cases where we are interested in the fundamental flux distribution (or any derived score in reactor physics applications) rather than solely on the k_{eff} (criticality-safety applications), while working with the exact same equation. For the sake of simplicity, the term *criticality calculation* will be used in the remainder of this thesis to refer to calculations done for systems portrayed by the k -eigenvalue equation.

¹In their article, Cullen et al. highlighted the fact that modifying the production term in such a way would make the modeled system diverge from the real one as k_{eff} drifted from 1. Thus, solving the k -eigenvalue equation would be giving "the right answer to the wrong problem".

3.1.1 The k -eigenvalue equation

In criticality calculations, the system is made constant over time by renormalizing the fission production term. This allows to model any type of system regardless of its effective multiplicative factor. The Equation that represents such systems is the k -eigenvalue equation, which in operators form is

$$(L + T) \phi = \left(S + \frac{1}{k} F \right) \phi \quad (3.1)$$

where ϕ is the neutron flux, k is the renormalization factor, and L , T , S and F are respectively the leakage, collision, scatter-in and fission operators presented in Section 1.2. By rearranging the terms, the eigenvalue equation appears

$$\phi = \frac{1}{k} (L + T - S)^{-1} F \phi. \quad (3.2)$$

The flux ϕ can be decomposed in series of modes that make up a base of the solutions of Equation 3.1

$$\phi = \sum_{i=0}^{\infty} a_i \phi_i \quad (3.3)$$

where the ϕ_i are solutions of Equation 3.1 with

$$\langle \phi_i, \phi_j \rangle = 0 \text{ if } i \neq j \quad (3.4)$$

ϕ_i being the i -th modes of the flux, and the operator $\langle \rangle$, the dot product. Among the different solutions of this equation, the one with the highest eigenvalue, which will be named k_0 , is the fundamental mode of the system. All higher order modes are named such that

$$k_0 > k_1 > k_2 > \dots \quad (3.5)$$

The fundamental flux ϕ_0 can be seen as the asymptotic flux distribution where the productions are divided by k_{eff} , which makes k_0 equal to the k_{eff} of the system. Basically, the k_{eff} obtained from Equation 3.1 is considered as the value by which the mean number of neutrons emitted by fission should be divided to make the reactor critical. While this equation is valid, the meaning behind this interpretation of k_{eff} does not exactly fit with reality [75], since it introduces a bias in energy and space (compared to the real system) when $k_{\text{eff}} \neq 1$. Aside from this mathematical interpretation, the k_{eff} can also be physically pictured as the ratio of the number of neutrons in a generation over the number of neutrons in the previous generation

$$k_{\text{eff}} \equiv \frac{N_g}{N_{g-1}} \quad (3.6)$$

where N_g is the number of neutrons in generation g . This stems from the fact that²

$$\frac{N_g}{N_{g+1}} = \frac{\text{productions}}{\text{disappearances}} = \frac{\nu \Sigma_f}{\Sigma_a + \text{Leakage}}. \quad (3.7)$$

²Here, (n, xn) reactions were included into the scattering source for the sake of simplicity, as in [35].

3.1.2 Power Iteration method

In steady-state reactor physics applications, one is most often interested in the fundamental flux shape that persists once the higher order harmonics have vanished (e.g., to compute the steady state power distribution of a reactor). To bring out the fundamental mode of the system with Monte Carlo methods, an iterative scheme based on the *power iteration* algorithm (which is also used in deterministic methods) is used. This algorithm is based on the interpretation of neutron chain reactions as a succession of generations over which it iterates.

Starting from an initial guess $k^{(0)}$ and $\phi^{(0)}$, the eigenvector and eigenvalue are computed in an iterative way such that

$$\phi^{(n+1)} = \frac{1}{k^{(n)}} (L + T - S)^{-1} F \phi^{(n)}, \quad (3.8)$$

where $\phi^{(n)}$ and $k^{(n)}$ are the flux and the eigenvalue at iteration n . In practice, operator $(L + T - S)^{-1}$ is applied by performing a random walk on the source population represented by $\frac{1}{k^{(n)}} F \phi^{(n)}$. The neutron source at iteration $n + 1$ is obtained from a fission bank filled during iteration n , from which N_0 neutrons are sampled. This step is often referred as the population control step since the number of source neutrons is kept constant in each generation³. The random walk is then performed over the N_0 source neutrons which are tracked from their birth to their death (usually either by absorption or leakage), while their progeny arising from fission is stored in the fission bank. Once all the neutrons of a given generation have been simulated, the eigenvalue can be computed based on Equation 3.6, as follows

$$k^{(n+1)} = k^{(n)} \frac{\int F \phi^{(n+1)} d^3r}{\int F \phi^{(n)} d^3r} \quad (3.9)$$

where $\int F \phi^{(n+1)} d^3r$ is the total number of fission neutron produced during iteration $n + 1$, and $\frac{\int F \phi^{(n)} d^3r}{k^{(n)}}$ is the number of fission neutrons produced during iteration n , rescaled by the previous estimation of k (i.e., the number of source neutrons of iteration $n + 1$).

Until now, there were no mention of the fundamental mode in the iterative algorithm, although it has been said that this method returns it once it has converged. Indeed, by substituting the mode decomposition of Equation 3.3 into Equation 3.8, it is possible to express $\phi^{(n+1)}$ and $k^{(n+1)}$ as function of the fundamental and higher modes [76]

$$\phi^{(n+1)} = C_1 \left[\phi_0 + C_2 \left(\frac{k_1}{k_0} \right)^{n+1} \phi_1 + C_3 \left(\frac{k_2}{k_0} \right)^{n+1} \phi_2 + \dots \right], \quad (3.10)$$

$$k^{(n+1)} = k_0 \left[1 + C_3 \left(\frac{k_1}{k_0} \right)^n \left(1 - \frac{k_1}{k_0} \right) + \dots \right]. \quad (3.11)$$

Because $k_0 > k_1 > k_2 > \dots$, the solution of the power iteration converges to the fundamental mode of the eigenvalue equation, but it also implies that estimators $\phi^{(n+1)}$ and $k^{(n+1)}$ of ϕ_0 and k_0 are polluted by higher order harmonics during the first iterations.

³Here the number of neutrons at the beginning of the iteration is kept strictly constant, but variations of the method allow small fluctuations around N_0 [76]

Once the source has converged to the fundamental mode distribution, additional successive cycles (called active cycles) are run in order to tally particles contribution to the score, until a given number of active cycles have been executed, or the error on the score has converged below a target value⁴. These steps are summarized in Figure 3.1.

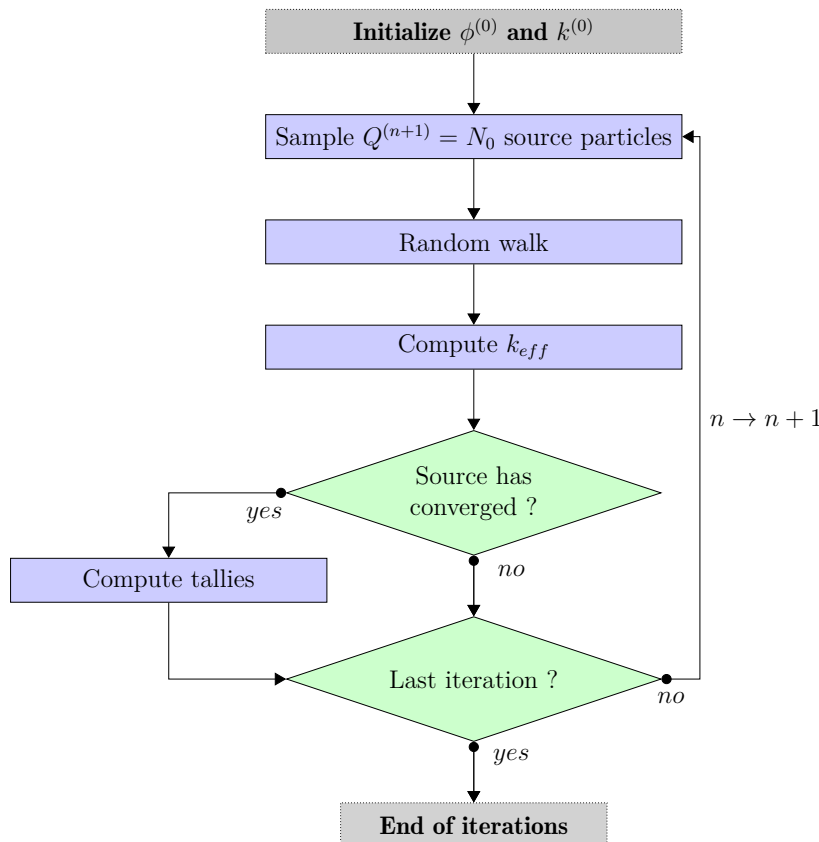


Figure 3.1: Monte Carlo power iteration algorithm

Although it has been used for decades, this approach has its limitations. As a matter of fact, there are known issues to the Monte Carlo power iteration that may jeopardize score estimations.

3.1.3 Issues regarding the convergence of the solution

Regarding the convergence, the whole method is based on the convergence of the flux estimation to the system fundamental mode, and, as seen above, higher order harmonics pollute the solution during the first iterations. Since computing tallies during these iterations would then introduce a bias, it is necessary to discard the first iterations before the source has converged to its fundamental distribution which can be seen as a loss of statistics (and therefore, idle computation time). For that reason, a lot of investigation has been done on how to accelerate this convergence step, especially for loosely coupled systems⁵ (which include large geometries like industrial reactors). While the

⁴Regarding the stopping criterion, some Monte Carlo codes use the number of active cycles provided by the user, while other can use an error threshold below which the estimated error of the mean score has to converge.

⁵Loosely coupled media are defined by a decoupling between the characteristic spatial scale of the media and the typical distance traveled by neutron before capture.

convergence of the source has an effect on the calculation time of the simulation, it should not affect the variance nor the average of the score as long as non-converged cycles are discarded properly. However, another set of issues, which have direct impact on tallies estimations, is linked to spatial and generational correlations that arise during the iterative process. As a matter of fact, successive cycles are correlated due to the filiation between generations. Basically, source points are sampled from fission sites of the previous iteration, which implies that two successive generations, and thus the estimations made from their tallies, are highly correlated. Intuitively, if more particles were to induce fission in a specific area of the geometry compared to other locations, then source neutrons would have higher chances to be born in that area, increasing the probability to contribute to the score in or near that region in the process (the counterpart is true for regions with less fission). Following this logic, a generation should be positively correlated to the previous one. Coping with correlated samples then requires special care when applying the Central-Limit Theorem, if one does not want to bias the error estimation [77, 78].

Additionally, biases on local and global tallies were shown to appear in some cases. Indeed, in the late 60's, biases on the k_{eff} estimation and its variance were highlighted in different works on criticality calculations [79, 80, 81, 82, 83]. These biases are due to the population renormalization process, and decrease as M^{-1} for the average value and $(MN)^{-1}$ for the variance, where M is the number of neutrons per generation and N the number of generations [83]. Later, Ueki et al. [78], as well as Dumonteil et al. [8] respectively shed light on the fact that generational and spatial correlations could also lead to biases on the spatial flux estimation. Since then, a lot of research work has been devoted to characterizing and implementing solutions to limit their impact on calculations. The main underlying mechanisms and implications motivating a different approach are presented here.

3.1.3.1 Convergence of neutron sources

Resuming Equations 3.10 and 3.11, one notes that this is the first highest order harmonic that persists the longest since $k_1 > k_2 > \dots$. In this respect, the ratio $\frac{k_1}{k_0}$ is often used to evaluate the speed of convergence of a system to the fundamental mode, and is called the *dominance ratio*, sometimes written DR . According to Equations 3.10 and 3.11 a higher DR means a slower convergence, moreover, for DR close to one, the flux convergence is slower than the k_{eff} convergence. Typically, higher DR are associated with loosely coupled systems like large reactors through Equation 3.10 (the DR can range from 0.55 to 0.97 for a pincell as the axial length increases [84], and is about 0.996 for the BEAVRS benchmark reactor [85, 86]). This implies that for large systems, looking only at the k_{eff} convergence to evaluate the number of inactive cycles is insufficient. Among the several indicators that exist to assess the convergence of the flux, the Shannon entropy [87] is widely used to check for the stability of the spatial distribution of sources over generations [88, 89].

The Shannon entropy is computed over a spatial mesh as such

$$H = \sum_i -p_i \log(p_i) \quad (3.12)$$

where p_i is the probability to find a neutron in mesh i . In practice, p_i is replaced by its estimation based on the ratio of the number of particles in bin i over the total number of particles in the system.

Nonetheless, biases on the variance and average estimates can be observed even if the neutron source distribution seems to have converged to its fundamental mode.

3.1.3.2 Biased estimation of the eigenvector variance

The k_{eff} estimator is defined as an average value over N observations

$$\overline{k_{\text{eff}}} = \frac{1}{N} \sum_{n=1}^N k^{(n)} \quad (3.13)$$

where $k^{(n)}$ is the n -th estimation of k . The same applies to the flux

$$\overline{\phi} = \frac{1}{N} \sum_{n=1}^N \phi^{(n)} \quad (3.14)$$

where $\phi^{(n)}$ is the n -th estimation of the flux. According to Equations 2.8 and 2.9, the unbiased estimator for the variance of the mean should then be⁶

$$\sigma^2 [\overline{\phi}] = \frac{1}{N(N-1)} \sum_{n=1}^N [\phi^{(n)} - \overline{\phi}]^2. \quad (3.15)$$

However, the Central-Limit Theorem implies to draw estimations from independent realizations of the Markov process, which is generally not the case in criticality calculations since score estimators are usually built with observations drawn in successive generations. As a matter of fact, if the N observations were to be drawn from successive iterations in the same simulation, this would induce a bias on the variance due to correlations between cycles. The bias on the variance estimation can be computed using Equation 2.12. According to the intuitive reasoning presented earlier in this section, it is expected that successive cycles are more or less positively correlated, leading to the underestimation of the variance (since the ρ_{ij} would be positive), and thus at the same time of the size of the confidence interval. In practice, these generational correlations (also called *lag k autocorrelations*) can be empirically estimated using a basic Pearson's correlation coefficient computed from batchwise results

$$\rho_k(x_l) = \frac{\text{Cov}[\phi_g(x_l), \phi_{g+k}(x_l)]}{\sigma[\phi_g(x_l)] \sigma[\phi_{g+k}(x_l)]} \quad (3.16)$$

where $\rho_k(x_l)$ is the correlation coefficient between the flux in two generations k apart (k is sometimes referred as the *lag k*) in spatial bin x_l , $\text{Cov}[\phi_g(x_l), \phi_{g+k}(x_l)]$ is the covariance between the flux estimation at generation g in spatial bin x_l and the flux estimation in the same bin but k generations later, and σ is the standard deviation (see Equation 3.15). To compute these terms, two approaches are possible: one can either compute the variance and covariance terms from different independent calculations in the same generation g , or do a rolling average/variance/covariance over successive cycles in the same simulation. It has been observed that the generational correlations exponentially decrease when the distance between two generations increases [90].

The real variance of the flux in bin x_l would then be estimated as

$$\sigma^2 [\overline{\phi}(x_l)] = \frac{\tilde{\sigma}^2}{N} \left[1 + 2 \sum_{k=1}^{G-1} \left(1 - \frac{k}{G} \right) \rho_k(x_l) \right] \quad (3.17)$$

⁶Mathematically, s^2 is usually used as a notation to refer to the unbiased estimator of σ^2 . However, σ^2 will be used even when speaking of s^2 in the rest of this thesis for the sake of simplicity.

with G the number of active cycles in each simulation and $\tilde{\sigma}^2$ is the variance computed from Equation 2.9 (sometimes called the *apparent variance*).

3.1.3.3 Biased estimation of the average eigenvector

The so-called *clustering* phenomenon is a naturally occurring phenomenon appearing in many different fields [91, 92, 93, 94, 95, 29], including neutron chain reactions. Due to the asymmetry between neutron being captured everywhere, and being born next to their parent, strong spatial correlations may develop in loosely coupled systems. In the worst cases, these spatial correlations can give rise to neutron clustering, which have been extensively studied in the past few years, particularly in the framework of the branching Brownian motion which couples a Galton-Watson birth death process to standard Brownian motion (cf. Section 1.3). Indeed, since particles born from the same ancestor are spatially correlated, high spatial correlations levels are linked to a high probability for neutron clusters to form [8, 33]. As iterations go by, independent families of neutrons go extinct, thus, increasing the number of correlated pairs of particles [96]. In recent work, it has been shown that the sampling method used during the population control step can have a significant impact on the extinction of independent families, therefore on neutron clustering [97]. While, in reality, this is not observed in industrial reactors operating at full power, it is not always the case in Monte Carlo calculations. Indeed, a high density of neutrons combined with stabilizing feedback effects (natural like Doppler or human-made like control rod movements) are supposed to prevent clustering. However, it has been shown that these effects can occur in Monte Carlo criticality calculations over loosely coupled systems when too few particles are modeled and induce a bias on the tallies average estimation [10].

Numerically, they can be characterized through the measurement of spatial correlations between two positions (or spatial bins) using the Pearson's correlation coefficient defined as

$$\rho_{ij} = \frac{Cov[\phi(x_i), \phi(x_j)]}{\sigma[\phi(x_i)] \sigma[\phi(x_j)]} \quad (3.18)$$

where $\phi(x_i)$ and $\phi(x_j)$ are the flux in spatial bins x_i and x_j .

In confined geometry, it has been shown that these correlations also decrease with the distance between two bins, while remaining positive [9]. However, when population control is used -a fortiori during the power iteration-, distant spatial bins can be anti-correlated [33]. This is due to the fact that a constant number of particles is simulated in each generation, thus, if more particles are sampled in a bin due to population control, less have to be sampled in another one.

These spatial correlations can induce a bias on the average local tallies estimation in the case of neutron clustering. In fact, the presence of neutron clusters in the geometry implies the non-convergence of the spatial flux towards its fundamental mode, which may lead to a distorted flux shape [10]. To easily grasp this idea, one can imagine a one-dimensional bare slab with leakage boundary conditions. If the system is loosely coupled and the number of particles simulated is low (i.e., the system is highly correlated), it is possible for the flux shape in a generation to deteriorate into a travelling wave (i.e., all the particles form a unique cluster). When the travelling wave gets closer to a boundary, more particles die out from leakage, but are re-sampled through the population control process, which samples them in the cluster, i.e., near the boundary. The result is that the cluster is reflected at the boundaries, until it moves

away from it. The resulting flux shape, which is the averaged of the travelling wave positions over successive generations, is flattened compared to the analytical cosine shape. On one hand, the number of collisions is overestimated near the boundaries due to the over-sampling of neutrons in these areas, inducing a positive bias. On the other hand, the collisions far from the boundaries (typically at the center of the geometry) are underestimated due to the normalization of the result, leading to a negative bias at the center. This case will be addressed in Chapter 5.

To date, tackling these correlation issues seems to be a relevant way to gain efficiency in criticality calculations. As mentioned earlier, Sutton studied the effect of the population control operated in criticality calculations on clustering [97]. In that context, neutrons were regrouped in families, where a family is defined as the whole progeny descending from a unique particle initially present in the simulation. Resulting from that definition, the N_0 initial particles in a Monte Carlo simulation define the initial ancestors of N_0 families. As generations go by, neutron families go extinct and the number of correlated pairs of particles, and thus spatial correlations, increases. This criticality calculation paradigm can therefore be interpreted as an attenuation problem regarding the decreasing number of families over generations. This paradigm motivated the idea of introducing variance reduction methods into criticality calculations.

Therefore, this thesis proposes an innovative way of replacing power iteration based population control by the Adaptive Multilevel Splitting (cf. Section 2.2.4.2) to tackle this generational attenuation problem, and thus the clustering phenomenon that might arise from it. The detailed methodology in the case of criticality calculations is presented in Chapter 5.

3.2 Time-dependent neutron transport

In dynamics calculations, the goal is to model a transient by coupling a kinetics simulation (i.e., time-dependent neutron transport) with different physics solvers to account for temperature-induced feedback phenomena. In kinetics calculations, the whole time interval is populated with neutrons in order to get a score at almost any time during the transient. Mathematically speaking, kinetics calculations aim at computing the solution of the time-dependent Boltzmann equation (Equation 1.9). In current Monte Carlo kinetics calculations, it is done by running time-dependent random walks and averaging the contribution of each walk in each time bin. Due to stochastic fluctuations, however, the numerical neutron population is prone to either soar or plummet, thus inducing numerical issues regarding calculation time and memory footprint. To avoid such issues, population control are used to contain fluctuations of the number of particles simulated.

These calculations should not be confused with α -eigenvalue calculations [98], which aim at characterizing the asymptotic behavior of a time-dependent system whose population changes over time might be described by

$$N(t) = N_0 e^{\alpha t} \quad (3.19)$$

where $N(t)$ is the number of particles at time t , N_0 is the initial number of particles and α can be seen as an equivalent for k_{eff} in the time dimension (if $\alpha < 0$, the system

is subcritical, if $\alpha = 0$ then the system is critical. It is supercritical for $\alpha > 0$).

While the final step of modeling a transient is a coupled neutronics-multi-physics time-dependent simulation (*dynamics*), the developments presented later will only focus on the time-dependent neutronics step (*kinetics*).

3.2.1 A brief history of Monte Carlo neutron kinetics

Kinetics calculations were motivated by the need of reference methods to model nuclear reactor transients. While power transients have been simulated using deterministic methods in the past, the reliability of their results was never compared to Monte Carlo methods, and the scarcity of experimental data regarding Reactivity Injection Accidents (RIA) made experimental validation difficult. With the increasing power of computing units, a possible implementation of Monte Carlo kinetics calculations has been considered in the late 2000s [99].

Sjenitzer and Hoogenboom then presented the bases of Monte Carlo neutron kinetics [14, 15] that led to the implementation of kinetics modules in several neutron transport codes [17, 100, 16, 19, 20, 18]

Neutronics solvers, now capable of modeling time-dependent transport of prompt and delayed neutrons, were then coupled to other physics solvers to simulate transients with thermal feedback effects [101, 102, 21]. Several couplings schemes have been performed in the last few years to model dynamic transients, including SERPENT2-OpenFOAM [103], SERPENT2-Subchanflow [23] and TRIPOLI4-Subchanflow [24].

3.2.2 Coupled neutron-precursor simulations with stochastic methods

Unlike criticality calculations, kinetics simulations, which general algorithm is presented in Figure 3.3, should not start from an arbitrary neutron distribution. Indeed, the idea is generally to model a transient in a system already in steady state, in one way or another, thus with a particular neutron distribution. Moreover, since the different time scales of prompt and delayed neutrons are taken into account by tracking delayed neutron precursors in addition to neutrons, it is also necessary to initialize the distribution of precursors. To do so, a static eigenvalue calculation is run beforehand to converge to the system steady state distribution. When the initial steady state has a k_{eff} close to unity, it is possible to do a k -eigenvalue simulation to compute the initial distribution. However, it is not the case when the effective multiplication factor is too far from 1, since spectral and spatial discrepancies between the calculation results and the real system distribution arise due to the population control mechanisms [75]. In that case, the k -eigenvalue equation does not represent the real system, and it is necessary to do an α -eigenvalue calculation to get the appropriate distribution. For the sake of convenience, systems modeled in the following chapters will all be considered as initially critical, and the initialization step would thus be realized by k -eigenvalue calculations. To be thorough, one should do one complete k -eigenvalue calculation, from an arbitrary distribution to the converged fundamental mode, for each independent run of the kinetics calculation. However, it can be costly to execute G successive cycles which will be discarded (because they do not have converged to the fundamental mode), to keep only one which will be used for the initialization of the kinetics cal-

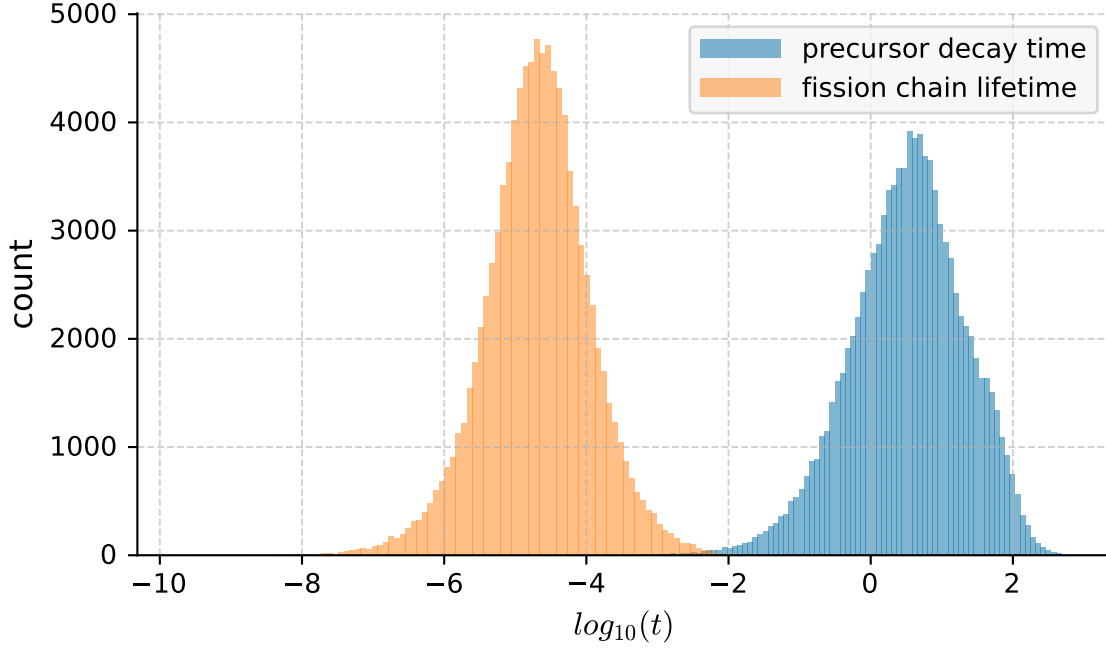


Figure 3.2: Distribution of prompt fission chain's lifetime and delayed neutrons precursors decay time

culation⁷. To improve the initialization stage regarding this issue, different technical solutions have been proposed. For instance, Faucher suggested to only perform one k -eigenvalue calculation until convergence has been reached, initialize the first kinetics calculation, then run a few more cycles in criticality mode starting from the final cycle previously computed, initialize the second kinetics calculation, and so on [21]. This way, only one convergence period is needed in the k -eigenvalue calculation, and running a few cycles between each kinetics calculation allows to partially de-correlate the initial states of each kinetics simulation. This is how TRIPOLI4 initializes the particle distributions at the beginning of a kinetics calculation [21].

3.2.2.1 Delayed neutron precursors decay

One of the major shortcomings of coupled neutron-precursor time-dependent calculations is the difference between prompt and delayed neutrons time scales. The typical duration of a prompt fission chain is about 10^{-2} second, while delayed neutrons are emitted after 10^{-2} to 10^2 seconds.

Most of prompt fission chains will die out before a precursor decays and emits a delayed neutron which will start a new fission chain, as illustrated in Figure 3.2. While in reality, many fission chains will overlap over time due to the high flux in a reactor, it may not be the case in a Monte Carlo simulation⁸. And so, scoring over a fine time

⁷As an example, if 100 cycles are necessary to reach convergence, keeping only the last one to sample initial neutrons and precursors in the kinetics calculation would mean having 99% of the cycles discarded, i.e., having 99% of "useless" calculation time.

⁸The neutron density inside a 2700 MW_{th} nuclear reactor is about 10^{14} neutron.m⁻³ [28], while the number of particles per batch in a Monte Carlo simulation rarely exceeds 10^6 neutrons in the whole geometry.

mesh in a transient simulation might be difficult if there are no neutrons alive (after fission chains have disappeared and before any delayed neutrons have started a new chain) in some time bins. To prevent this issue from happening, Sjenitzer proposed the *forced decay* method [104]. Instead of waiting for precursors to naturally emit delayed neutrons following an exponential decay, each precursor will emit one delayed neutron with probability $P = \frac{1}{\Delta t}$, where Δt is the length of the time bin in which the delayed neutron is emitted. After emission, the precursors remain in the system and might be sampled for other delayed neutron emissions in subsequent time bins. To ensure that the expected number of emitted delayed neutrons remains unbiased, the weight of the delayed neutron that is emitted is modified so that

$$w_d(t) = w_p \Delta t \lambda_i e^{-\lambda_i(t-t_0)} \quad (3.20)$$

where w_d is the weight of the delayed neutron, w_p is the weight of the precursor at creation, λ_i its decay constant and t_0 the time at which the precursor was created. One can observe that the weight of a delayed neutron emitted by forced decay depends on the length of the time interval that is considered, among other things. For short intervals, the weight of delayed neutrons can thus be quite low compared to live neutrons coming from the previous time interval. In that case, delayed neutrons are prone to be deleted by Russian Roulette, as observed by Faucher [21].

Besides being forced to decay, precursors can also undergo population control as described hereafter.

3.2.2.2 Handling population fluctuations over time

Since kinetics simulation aims at modeling the neutron population over time, stochastic fluctuations may cause computational issues. It is easy to grasp the idea for non-critical systems:

- for a subcritical system, the population may plummet, leading to no particles left to score,
- for a supercritical system, the population grows and eventually saturates the computer memory or lead to never-ending simulations.

Even for critical configurations, the population can grow or shrink due to the stochastic nature of the Galton-Watson process introduced in Section 1.3 [30]. To deal with these issues, different population control methods, such as the RR/splitting combination or the combing method presented in Section 2.2.4.1, are regularly used over the time interval that is modeled. As mentioned above, population control techniques are also used on the precursor population since their number is higher than the number of live neutrons by several orders of magnitude in steady-state nuclear reactors⁹.

⁹While delayed neutrons represent only a tiny fraction of all neutrons emitted by fission, the number of precursors in a steady-state reactor is much higher than the number of neutrons at a time as they tend to accumulate in the system due to their slow decay. If we consider a system in which precursors decay with a constant $\lambda = 0.0785 \text{ s}^{-1}$, the fraction of delayed neutrons is $\beta = 0.00685$ and the mean generation time is $\Lambda = 7 \times 10^{-5} \text{ s}$, the concentration of precursors at equilibrium is given by $C = \frac{\beta}{\Lambda \lambda} n \approx 10^3 n$ where n is the number of neutrons in the system at equilibrium.

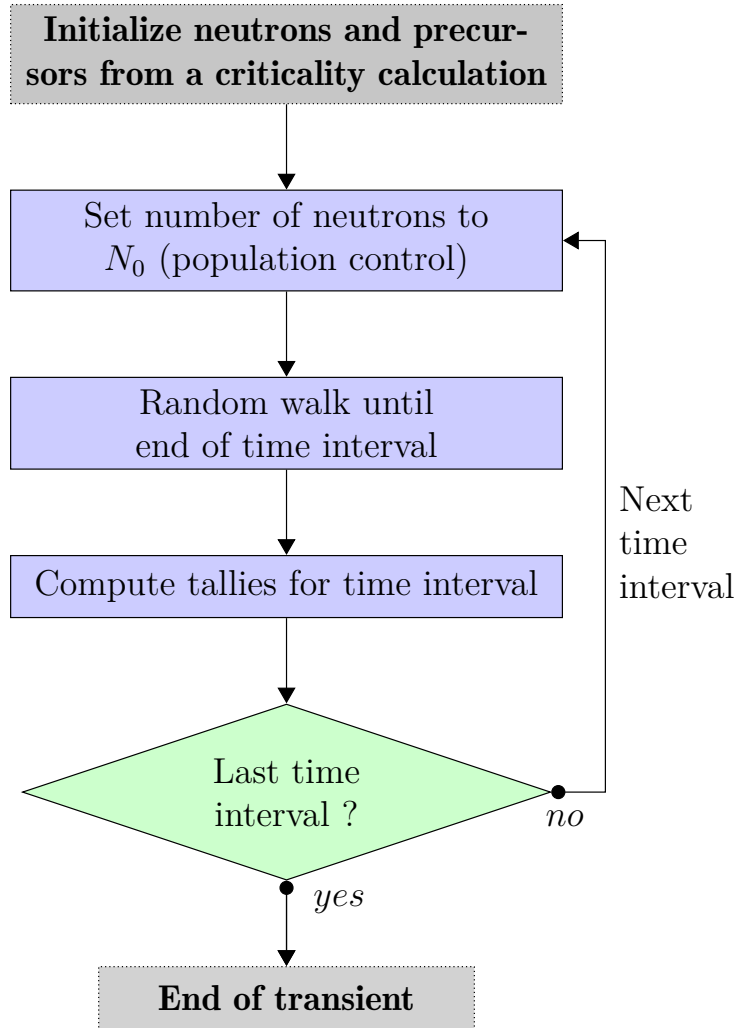


Figure 3.3: Monte Carlo kinetics algorithm for one independent simulation

These two main upgrades allowed to go from fixed source calculations without delayed neutron precursors in subcritical configurations to kinetics simulation of fissile systems. However, modifying the natural time-dependent random walk of neutrons and precursors might change the physics (like in k -eigenvalue calculations as mentioned before). Methods presented above are qualified as *unbiased*, meaning that the average result computed from the modified algorithm has the same expected value as one computed from an analog random walk (i.e., they also allow computing the solution of Equation 1.9). Yet, statistical fluctuations are inherent to the real system as explained in Section 1.3, and basing calculations on modified random walks might introduce differences between the numerical solution and the real system. For now, we will assume that kinetics systems can be described by the time-dependent Boltzmann equation and proceed with methods mentioned above. The issue of natural statistical fluctuations will be addressed later in Chapter 8.

Even if kinetics modeling of whole reactors is now reachable, associated calculation costs, especially regarding computation time, remain high for industrial type applications. This issue is discussed in the following section.

3.2.3 Current limitations to industrial applications of dynamics calculations

The main limiting factor of time-dependent Monte Carlo applications resides with the required computational resources, especially with respect to the CPU time [22]. The high computation cost of coupled time-dependent Monte Carlo calculations is mostly due to the neutronics simulation (i.e., the kinetics) with Monte Carlo methods. Indeed, recent coupled calculations showed negligible computation time for the coupling step compared to the time-dependent Monte Carlo transport step [23, 22]. As for the total computation time, it reached about 1.6×10^5 CPU.hour (around 4000 CPU.day) for the whole PWR core case with an average spatial standard error of the mean around 10 or 14% [22] for the pinwise power distribution. Going under 5% error still seems to be out of reach for the moment.

To overcome this issue, work on new variance reduction scheme applied to time-dependent neutron transport problems have been pursued. For example, Legrady et al. applied importance based population control in GUARDYAN Monte Carlo code [47]. Another class of techniques based on zero-variance schemes was proposed by Mancusi and Zoia for time-dependent problems [57]. While the scope of their article was narrowed to an analytical derivation of biased transport and collision kernels¹⁰ using an exact adjoint flux to achieve zero-variance estimates, it presents principles that could ultimately lead to more practical schemes based on approximate adjoint flux estimations. For example, methods such as the CADIS [105] and FW-CADIS [106] methods could be extended to time-dependent calculations.

While clustering and correlation effects were mainly investigated in the context of criticality calculations, correlations also naturally arise in kinetics simulations [107]. These space and time correlations could eventually lead to untoward effects regarding Monte Carlo estimates of average tallies and their statistical uncertainties [21].

For all reasons mentioned above, transient simulations would certainly benefit from effective variance reduction schemes. Therefore, the impact of a variance reduction method, namely the Adaptive Multilevel Splitting, on time-dependent coupled neutron-precursor calculations will also be investigated in Chapter 6 and 7.

Conclusion

Variance reduction methods are usually used to improve the variance on average tallies estimates regarding localized detectors. In reactor physics calculations, however, one is generally interested in the power distribution over the whole space (and time for kinetics calculations), which makes the aim of variance reduction schemes less obvious to identify. In criticality calculations, major drawbacks revolve around spatial and generational correlations which can negatively impact estimations of scores and their statistical uncertainty. While there are also spatial and time correlations in kinetics calculations, their effect on estimates is not clearly identified yet. The variance of local tallies, however, reaches a cap below which it becomes difficult to go for large reactors

¹⁰Here, *biased* is used to describe transport and collision kernels that were altered compared to their analog form, and does not refer to biased results that might arise from these modifications.

(typically for industrial applications) with current computational means. Yet, variance and correlations are properties of the system that is modeled (physical model, number of histories, methods that are used). This implies that changing the rules by which the physics is represented will also impact variance and correlations.

Based on this observation, we intend to use a variance reduction scheme, namely the AMS, to modify the numerical system behavior regarding variance and correlations. The main goal is here to reduce unwanted correlations in criticality calculations, and reduce the standard error of the mean of local and spatially integrated tallies in kinetics calculations. In the next chapters, the AMS implemented for general particle transport is presented, before being applied to criticality and kinetics calculations.

Chapter 4

Adaptive Multilevel Splitting

It is important to draw wisdom from different places. If you take it from only one place it becomes rigid and stale.

- Iroh,
Avatar The Last Airbender

Contents

4.1	Mathematical background	58
4.1.1	Description of the algorithm	58
4.1.2	Sampling replicas for duplication	61
4.1.3	Estimators	63
4.2	Application to particle transport	63
4.2.1	Structure of a particle history in the AMS	63
4.2.2	Iterations execution	65
4.2.3	Handling branches when sampling new branching tracks	65
4.2.4	On-the-fly scoring	66
4.2.5	About the importance function	68
	<i>Conclusion</i>	68

The Adaptive Multilevel Splitting (AMS) is a variance reduction method initially intended for rare events simulations. It derives from the so-called Multilevel Splitting methods [108, 109], themselves inspired by the splitting methods used in particle transport simulations back in the early 1950s [110, 111]. It was developed in the early 2000s by Cérou and Guyader [4] and adapted for particle transport in the late 2010s by Louvin et al. in the case of fixed source calculations [6]. Since the aim of this thesis is to apply variance reduction patterns to criticality and kinetics calculations, this chapter outlines the state of this method before its scope was extended to these kinds of calculations. Therefore, the AMS original formulation as well as more recent developments made for particle transport in the case of shielding problems are presented.

4.1 Mathematical background

The method was initially developed in the field of applied mathematics for continuous Markov chains [4] to improve pre-existing Multilevel Splitting techniques, which required a prior knowledge of the system behavior to be effective. It was then generalized to discrete Markov chains [5], and more specifically to fit the specific needs of particle transport [6, 7]. This algorithm belongs to the family of Interacting Particle Systems (IPS) [112]. To reduce the cost of sampling a rare event, it indeed aims to push replicas of a Markov process towards an objective by ranking and re-sampling replicas of a Markov process. Iteratively, it will rank the replicas and re-sample the most probable to contribute to the measure events until enough observations of the rare event have been made.

4.1.1 Description of the algorithm

The AMS algorithm is an iterative scheme in which a set of N replicas of a Markov process will be ranked and partially re-sampled in order to evaluate a score associated to a rare event. Considering a discrete-time Markov chain $X = (X_{t_i})_{i \in \mathbb{N}}$ where X_{t_i} is the state of the chain at time t_i (i -th state), with values in \mathbb{R}^d ($d \in \mathbb{N}^*$), we want to evaluate a score over D , a sub-domain of \mathbb{R}^d . If we denote τ_D the time at which X has reached D , and τ_f the stopping time, or time of death, as the time at which the Markov chain reaches its end, reaching D is a rare event if

$$0 < \mathbb{P}(\tau_D < \tau_f) \ll 1. \quad (4.1)$$

Let ϕ_D be an application that maps the set of all possible path P of X , P , into \mathbb{R}

$$\phi_D : P \longrightarrow \mathbb{R} \quad (4.2)$$

with

$$\phi_D(X) = 0 \text{ if } \tau_D > \tau_f. \quad (4.3)$$

From now on, D will be equated to a *detector*. We are here interested in estimating $\mathbb{E}[\phi_D(X)]$ in the case where $\tau_D < \tau_f$ is a rare event. As an example, a shielding problem may fit this description. In such a case, the Markov process X considered would be the history of a particle emitted from a source point and moving in a phase-space of dimension d and the application ϕ_D would correspond to the response function of a detector of interest bounded in a sub-domain D of the phase-space. In the case of a detector placed very far from the source, the event of a particle reaching said detector before the end of its history would be a rare event, and estimating a score ($\mathbb{E}[\phi_D(X)]$) in it could be computationally very expensive.

The iteration scheme is described below in the general case. Although the following description is largely inspired by Louvin's presentation of AMS in [7], it diverges regarding some details, for example it is not assumed here that all replicas are initialized in the exact same way.

Importance function

The importance function is a fundamental parameter of the AMS since it is used as a measure of the proximity to the detector, and is used to rank trajectories and select

the ones to re-sample. This function, noted ξ , must map the phase-space to \mathbb{R} in order to quantify the *distance* of a point in \mathbb{R}^d to the target

$$\xi : \mathbb{R}^d \longrightarrow \mathbb{R}. \quad (4.4)$$

For now, let us assume that ξ is user defined without further conditions except the existence of a value Z_{max} for which

$$\begin{aligned} \xi(x) &= Z_{max} \text{ if } x \in D, \\ \xi(x) &< Z_{max} \text{ if } x \notin D. \end{aligned} \quad (4.5)$$

Additional practical requirements will be detailed in the next section. Finally, the importance of the j -th realization $X^j = (X_{t_i}^j)_{i \in \mathbb{N}}$ ($j \in \llbracket 1; N \rrbracket$) of X is defined as the maximum importance reached by a point of the chain

$$I(X^j) = \sup_{t_i \in [0; \tau_f]} \xi(X_{t_i}^j). \quad (4.6)$$

In applied mathematics, the optimal choice for this function is called the *committor function* and allow to reach the minimum variance possible [113]¹.

First iteration ($q = 0$)

At iteration $q = 0$, N replicas $(X_{0, t_i}^j)_{j \in \llbracket 1; n \rrbracket}$ of the Markov chain are initialized by running the Markov process as it would be without using the AMS. In the context of particle transport, replicas will alternatively be called histories or tracks, in reference to tracks left by particles in bubble or wire chambers. N tracks are thus transported from their birth to their death following an analog scheme² (i.e., without variance reduction). The *iteration weight* of the AMS is initialized to

$$\alpha_{AMS}^0 = 1. \quad (4.7)$$

From there, the iterative scheme continues as described below until the stopping criterion defined below is reached.

Iterative process

At iteration $q \geq 1$, several steps are performed to re-sample K new replicas among the N ones.

1. **Ranking step :** the N replicas are ranked according to the importance function ξ defined by the user and ordered by increasing importance.

For $j \in \llbracket 1; N \rrbracket$, the j -th replica of the Markov chain at iteration q is denoted X_q^j and its importance is noted $I_q(j)$ where

$$I_q(j) = I(X_q^j) = \sup(\xi(X_{q, t_i}^j)). \quad (4.8)$$

¹Unlike methods like exponential biasing, which is often used in zero-variance schemes, the AMS does not make it possible to end up with a variance lower than a theoretical minimum strictly higher than 0.

²As a matter of fact, the initial iteration can be made on the basis of an already existing variance reduction scheme, but we voluntarily simplify the idea for the sake of simplicity.

After being sorted, the replicas are renamed so that

$$I_q(1) < \dots < I_q(K) < \dots < I_q(N). \quad (4.9)$$

At this point a *kill level* is defined as

$$I_q^{kill} = I_q(K). \quad (4.10)$$

If $I_q^{kill} = Z_{max}$, the algorithm stops and the final score is computed as presented in Section 4.1.3, considering that iteration $q-1$ is the last effective iteration. This condition is called the *stopping criterion*. Put differently, if $I_q^{kill} = I_q(K) = Z_{max}$, this means that at least $N-K+1$ replicas have reached the maximum importance. Otherwise, iteration q resumes with the *re-sampling step*.

2. **Re-sampling step :** if the stopping criterion has not been met, the K_q replicas whose importance is lower or equal than I_q^{kill} are deleted and K_q new replicas are sampled by randomly duplicating K_q replicas among the $N - K_q$ that were not deleted, bringing back the total number of replicas to N

$$K_q = \text{card} \left(\left\{ j \in \llbracket 1; N \rrbracket / I_q(j) \leq I_q^{kill} = I_q(K) \right\} \right). \quad (4.11)$$

This step is further detailed in Section 4.1.2 regarding how duplicated tracks are sampled.

3. The iteration weight is then updated to

$$\alpha_{\text{AMS}}^q = \left(1 - \frac{K_q}{N} \right) \times \alpha_{\text{AMS}}^{q-1}. \quad (4.12)$$

4. **Simulation step :** the K_q new replicas of the Markov process are simulated the same way as the N initial ones were in iteration 0. Once all the K_q replicas have reached their time of death, a new iteration starts with the ranking step described above.

The iterative process detailed above is portayed in Figure 4.1. In this example, we are interested in particles scoring in a plane detector perpendicular to the x -axis. The figure displays the first three iterations of a particle transport problem in a non-multiplicative medium with $N = 3$ initial tracks and $K = 1$.

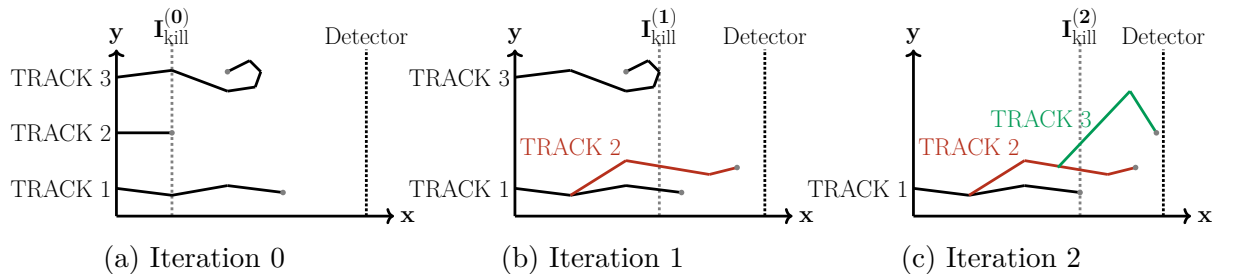


Figure 4.1: AMS iterations for non branching tracks (this implies that 1 track equals 1 particle) with a detector defined in the (x,y) plane, with $N = 3$ and $K = 1$. The importance function is inversely proportional to the distance to the detector.

4.1.2 Sampling replicas for duplication

During the re-sampling step, different strategies are possible to select which replicas to duplicate. In previous implementations of the method, the K_q tracks were sampled uniformly with replacement among the $N - K_q$ remaining ones, copying the characteristics of the selected replica³ into the new one, including a potential statistical weight associated to that replica, e.g., in cases of weighted Monte Carlo games where variance reduction techniques were used in the initial iteration. However, it is possible to consider other approaches as long as the modifications of the sampling kernel are taken into account via the statistical weight of the new replicas. Different possibilities are presented here.

Uniform sampling with replacement

In this paradigm, K_q tracks are selected uniformly among the $N - K_q$ that were not deleted to start K_q new trajectories. Therefore, to conserve an unbiased expectation of the total weight in output of the sampling (i.e., the starting weight of the K_q new trajectories), the statistical weight of the replica selected for duplication is copied into the new trajectory.

Let i be the index of the selected replica and k the index of the new replica/trajectory. The probability \mathbb{P}_i to select trajectory i for duplication uniformly distributed and therefore equal to

$$\mathbb{P}_i = \frac{1}{N - K_q}. \quad (4.13)$$

If w_i is the weight of the i -th replica, then

$$\mathbb{E}[w_k] = \sum_{i=1}^{N-K_q} \frac{1}{N - K_q} w_i \quad (4.14)$$

and the expected total weight of the K_q new replicas is

$$\begin{aligned} \mathbb{E}[W_{K_q}] &= \sum_{k=N-K_q+1}^N \mathbb{E}[w_k] \\ &= \frac{K_q}{N - K_q} \sum_{i=1}^{N-K_q} w_i. \end{aligned} \quad (4.15)$$

This is the sampling method that has been previously used in the field of particle transport [7]. In this thesis, the importance sampling of tracks has also been added for testing purposes.

Importance sampling

There is no impediment to proceed with importance sampling to select the K_q replicas. The idea would be to use an importance criterion (which could be the same as the importance function used to rank tracks, or a completely different one) to favor some tracks over other for duplication. In that case, the probability to select the i -th replica

³E.g., in particle transport, a point representing a particle state at a given time in the AMS can be defined by its position, direction, energy, time and weight.

is denoted \mathbb{P}_i^* and the weight of the resulting trajectory must be modified according to

$$\begin{aligned} w_k &= w_i \times \frac{\mathbb{P}_i}{\mathbb{P}_i^*} \\ &= w_i \times \left(\frac{1}{N - K_q} \times \frac{1}{\mathbb{P}_i^*} \right). \end{aligned} \quad (4.16)$$

Thus, the expected value for w_k is

$$\begin{aligned} \mathbb{E}[w_k] &= \sum_{i=1}^{N-K_q} \mathbb{P}_i^* \frac{w_i}{\mathbb{P}_i^* (N - K_q)} \\ &= \sum_{i=1}^{N-K_q} \frac{1}{N - K_q} w_i \end{aligned} \quad (4.17)$$

and the expected total weight of the K_q new replicas is equal to

$$\mathbb{E}[W_{K_q}] = \frac{K_q}{N - K_q} \sum_{i=1}^{N-K_q} w_i. \quad (4.18)$$

Hence, proceeding with importance sampling does not introduce a bias on the average output weight. In the special case where $\mathbb{P}_i^* = \frac{w_i}{W_{TOT}}$, where

$$W_{TOT} = \sum_{i=1}^{N-K_q} w_i, \quad (4.19)$$

the output weight of all the new replicas is exactly equal to $\frac{W_{TOT}}{N-K_q}$

$$\begin{aligned} w_k &= w_i \times \frac{\mathbb{P}_i}{\mathbb{P}_i^*} \\ &= w_i \times \frac{1}{N - K_q} \times \frac{W_{TOT}}{w_i} \\ &= \frac{W_{TOT}}{N - K_q}. \end{aligned} \quad (4.20)$$

Equation 4.20 implies that all K_q new tracks have the same weight, meaning that the variance $Var[w_k]$ is equal to 0. Proceeding with an importance sampling based on track weights thus reduces the variance between new replicas weight, and is therefore expected to be more efficient than uniform sampling.

Sampling with and without replacement

The two previous strategies implied a sampling with replacement, meaning that track i could be selected up to K_q times for duplication. However, it is possible to limit the number of times a path is selected by sampling without replacement, similar to the combing method. Limiting the number of times a track may be selected for duplication would limit the number of correlated pairs of trajectories re-sampled, which may be valuable if one would prefer to limit correlations in the system.

4.1.3 Estimators

Once the stopping criterion ($I_Q^{kill} = Z_{max}$) is met and the final effective iteration determined, an unbiased estimator $\hat{\phi}_D$ of ϕ_D can be built from the iteration weight and the realizations $(X_Q^j)_{j \in \llbracket 1; N \rrbracket}$ of X of the last iteration Q , according to Brehier et al.'s work in [5]. This unbiased estimator can be written

$$\hat{\phi}_D = \alpha_{AMS}^Q \sum_{i=1}^N \phi_D(X_Q^i) \quad (4.21)$$

where α_{AMS}^Q is the iteration weight defined by Equations 4.7 and 4.12 at last iteration and equal to

$$\alpha_{AMS}^Q = \prod_{q=1}^Q \left(1 - \frac{K_q}{N}\right). \quad (4.22)$$

The iteration weight α_{AMS}^Q can also be interpreted as the probability for a trajectory to reach D , i.e.,

$$\mathbb{E} [\alpha_{AMS}^Q] = \mathbb{P}(\tau_D < \tau_f). \quad (4.23)$$

This section aimed at presenting the general framework of the Adaptive Multi-level Splitting. In the next section, the particular implementation in case of particle transport is presented.

4.2 Application to particle transport

The AMS version for particle transport was initially designed in support of shielding applications, in which the transmission probability of a particle source is extremely low [6, 7], leading to computation issues for the estimation of the score with analog Monte Carlo methods (i.e., very few, if any, particles reach the detector). The method performed quite well compared to exponential biasing, for an easier use [7]. In this section, we present practical requirements and implementation choices that have been made for particle transport simulations. Unless otherwise stated, the following description is based on the end state of the method described above, as it was presented in [7], for the general case of branching neutron histories evolving in multiplicative media.

4.2.1 Structure of a particle history in the AMS

In the AMS framework for particle transport, one simulation is called a *batch*, whose global structure is described in Figure 4.2. A batch regroups N histories, initially independent, named tracks, which corresponds to the N replicas of the Markov process mentioned in the previous section. For multiplicative media (in which neutrons undergo branching events, due to fissions for example), tracks may contain several branches, each one corresponding to the trajectory of a unique particle being part of the whole history (each track beginning with only one branch at initialization). Each branch is composed of several points, each of them representing successive states of the Markov chain (i.e., successive collisions, may they be real or virtual such as the crossing of a boundary). In that way, all the points distributed amongst the branches of the j -th track represents the j -th realization of $(X_{t_i})_{t_i \in \mathbb{N}} : (X_{t_i}^j)_{i \in \mathbb{N}}$.

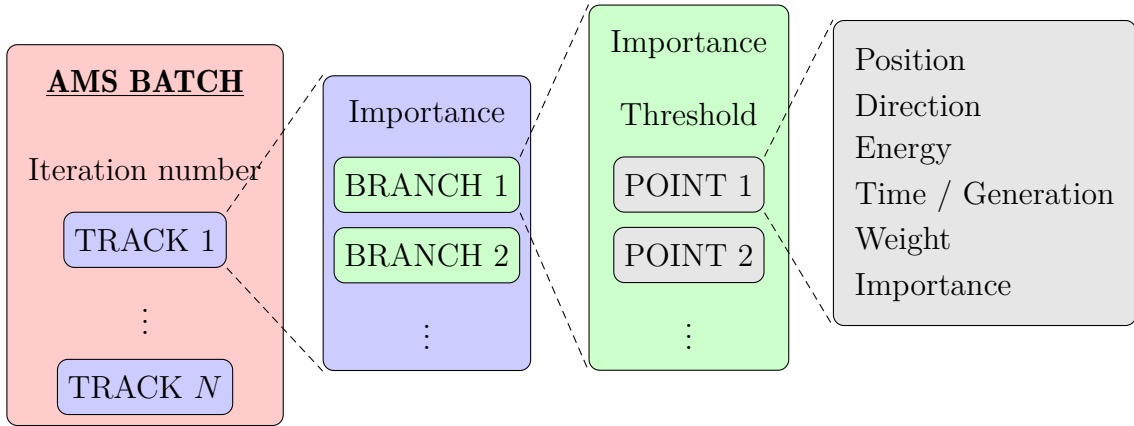


Figure 4.2: AMS track/branch/point structure

During the transport step of the Monte Carlo simulation, histories are transported from their birth to their death, adding points in the batch structure of Figure 4.2 as collisions occur. Currently, only the exit state of collisions (particles outgoing a collision, as opposed to incident particles) are added as points in the structure. Furthermore, only points of increasing importance are added, for memory saving purposes. It is however possible to extend this to any *stopping point* (i.e., when a particle flight is stopped during the simulation), including real collisions, virtual collisions⁴, boundary crossing points, ... to increase the number of points at which the importance is evaluated in the AMS.

When a new particle arises from a splitting event (it can be physical like fission or numerical like splitting in a weighted Monte Carlo game, cf. Section 2.2.4), a new branch is appended to the track to represent the trajectory of the new particle. A branch importance is updated when a point is added to it, if the point importance is higher than the current branch importance, meaning that a branch importance is defined as the maximum importance amongst its points. At any moment, a track importance is defined as the maximum importance amongst its branches, following Equation 4.6. To account for the time of appearance of a branch, a threshold value is affected to each branch of a track, which is defined as the importance of the mother branch prior to the splitting event (except for the initial branch whose threshold is set to 0). For example, let us consider a track with two branches as depicted by Figure 4.3a. Before the point of importance I_1 , only the main branch is present. At some point after the point I_1 , a splitting event occurs leading to the creation of two particles, hence two branches. One of the two particles is arbitrarily considered to belong to the main branch (see Figure 4.3b), while the other starts a new branch (see Figure 4.3c). Since the importance of the mother branch (here the main branch) was I_1 before the splitting event, the new branch threshold is equal to I_1 , while the threshold of the main branch remains unchanged, and equal to 0. This *threshold* parameter is used during the re-sampling step in case of branching tracks, as explained later.

⁴Tracking methods such as *delta-tracking* can produce virtual collision points.

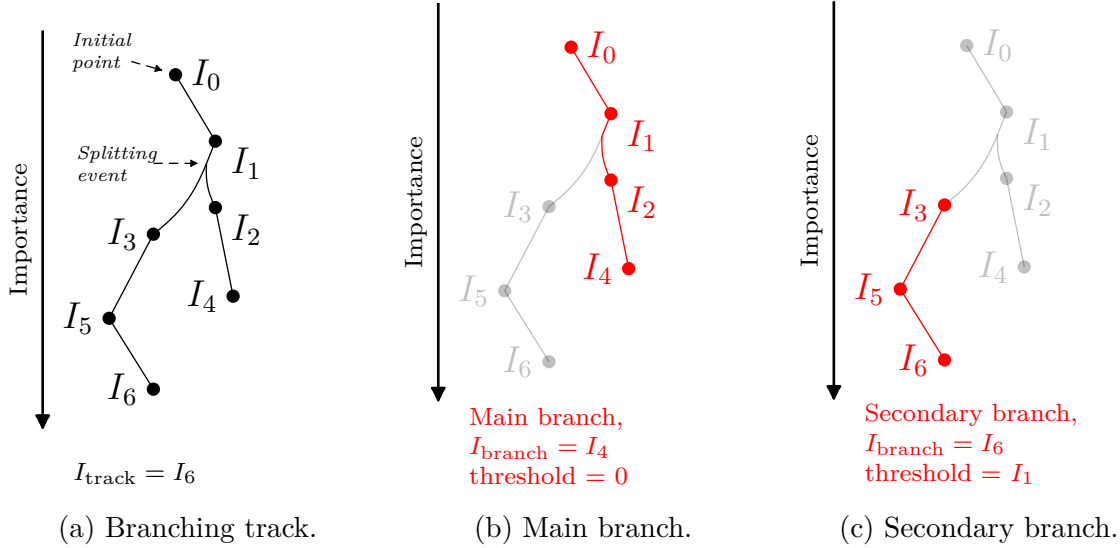


Figure 4.3: AMS track with two branches.

4.2.2 Iterations execution

The iterations are performed as described in Section 4.1.1. First the batch is initialized by running N particle histories⁵ (i.e., the N replicas of the Markov process defined by the transport of a particle history in the system), until their death (i.e., no particle left to continue the history). Once there is no particle left, the N tracks resulting from the transport step are sorted as described in the previous section and a kill level I_{kill} is defined. If the stopping criterion has not been reached, i.e., if less than $N - K + 1$ tracks have reach the target detector, new particles histories are initiated following the re-sampling procedure previously defined, and simulated. During the simulation of the histories, branches and points are added to the structure of Figure 4.2 as detailed in Section 4.2.1. The resulting algorithm looks like the one presented in Figure 4.4.

4.2.3 Handling branches when sampling new branching tracks

In order to initiate new tracks to simulate, different sampling strategies can be used to select which remaining tracks will be duplicated, as explained before (see Section 4.1.2). For branching tracks however, the duplication of sampled tracks is a bit more complex. When a track is selected for duplication, several rules are to be followed to account for branches that may be "alive" near the kill level. These are stated below and illustrated by Figure 4.5.

1. Only branches whose threshold is below the kill level must be considered for duplication, i.e., if the kill level crosses the track before the branching point, only the main branch is to be copied. This refers to Figure 4.5a.
2. Since the duplication is made at the first point with an importance higher than the kill level, if a branch has an importance lower to that kill level it must not be copied, as portrayed by Figure 4.5b.

⁵Also called tracks in the AMS framework.

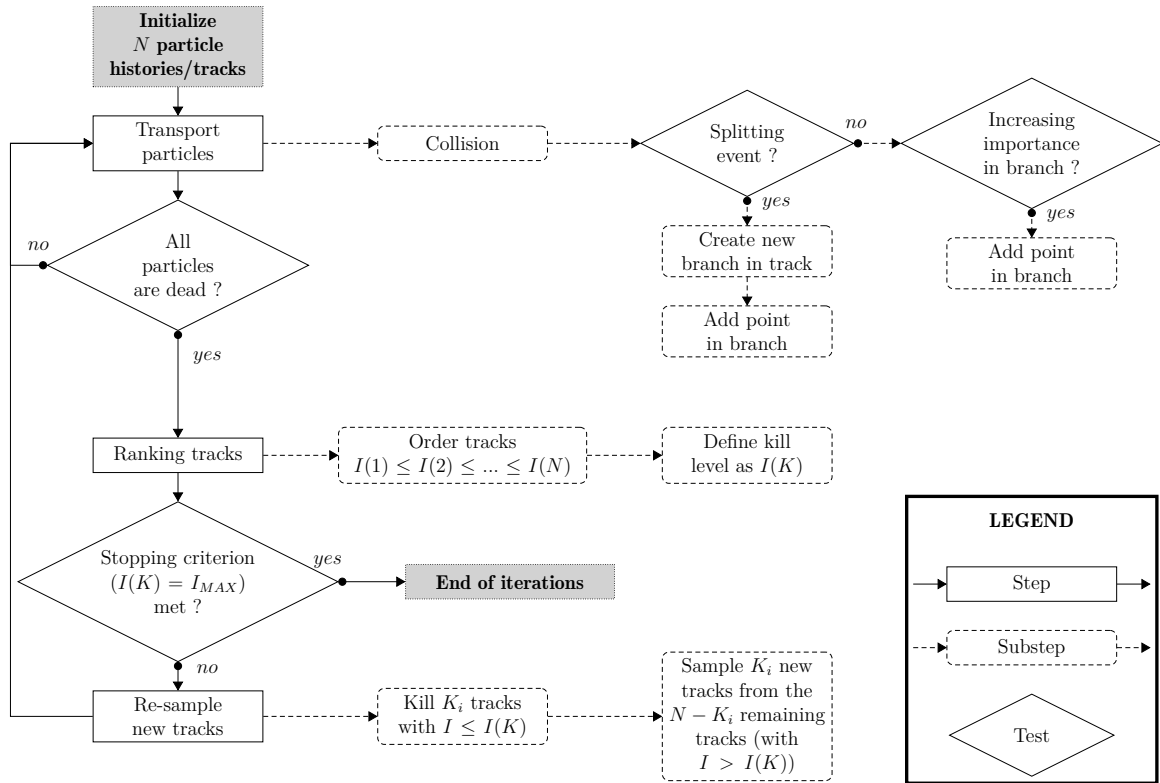


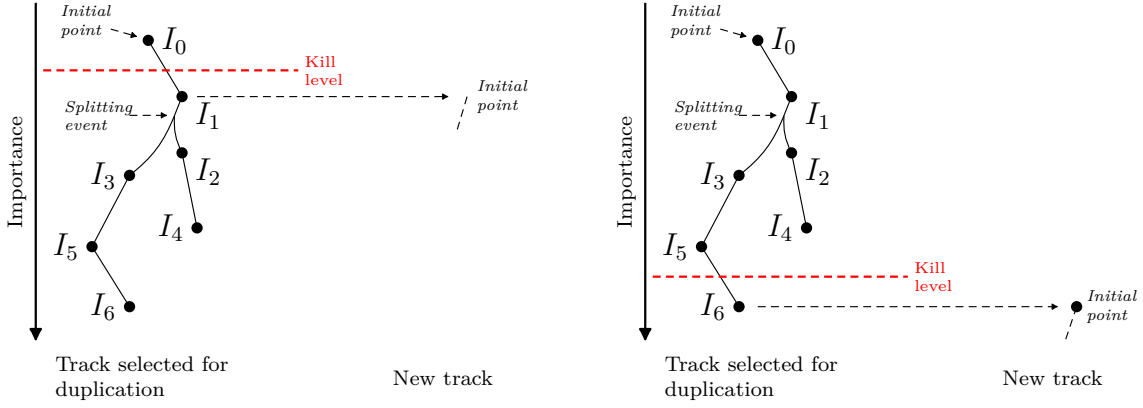
Figure 4.4: AMS algorithm

3. If multiple branches have a threshold below the kill level, and living points (with an importance higher than the kill level), they must be copied into the new track sampled as different branches, as shown by Figure 4.5c.

For uniform sampling, this is quite straightforward since once a track has been selected for duplication, it is sufficient to follow the above-mentioned rules and copy all branches that should be copied into the new track, with their respective weight. To our knowledge, this is the version that has been exclusively used until now in particle transport. If importance sampling is used, an additional step might be necessary to compute the probability distribution \mathbb{P}_i^* at iteration q , since tracks state may change at each iteration.

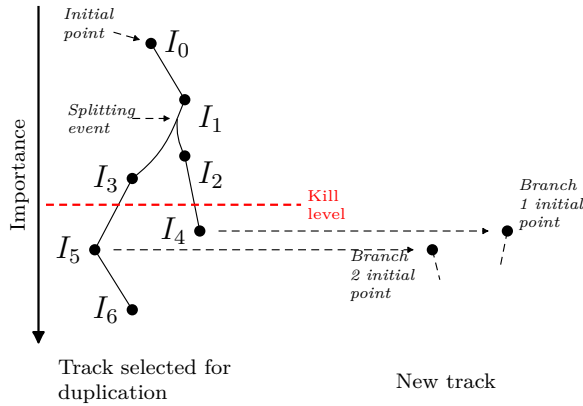
4.2.4 On-the-fly scoring

Until now, the description of the AMS suggests that the method only yields unbiased results for the tallies inside the target detector. However, in practice, it is often desirable to compute a score not only in a specific detector, but also along the path between the source and the detector. For example, one could be interested in the dose rate in intermediate regions between a neutron source and a detector that would define the target D mentioned in Section 4.1.1. For that purpose, a scoring procedure inspired from Bréhier et al.'s work in Ref. [5] was developed by Louvin [7] to estimate tallies in intermediate volumes along the particles path from the source to the detector. This procedure, as described in Ref. [7], states that it is possible to exploit the track history to compute those tallies. In its optimized version, it is however not necessary to keep all the histories in memory and tallies are estimated *on-the-fly* during AMS iterations. To do this, it is necessary to define *scoring volumes*, which define sub-regions of the



(a) The splitting level crosses the main branch before splitting : only one branch is copied into the new track.

(b) The splitting level crosses only one surviving branch : only this branch is copied into the new track.



(c) The splitting level crosses two living branches : the two branches are copied into the new track.

Figure 4.5: Handling branches during the re-sampling step.

phase-space in which we want to compute tallies. These sub-regions may be open to scoring, meaning that events occurring inside those volumes may be tallied, or closed to scoring, meaning that events happening inside must not be taken into account when computing scores. At the beginning of the simulation, all the scoring volumes are open and are assigned an importance level. This importance level is considered to keep a volume open or close. Indeed, at each iteration, when the kill level is determined during the ranking step, all volumes whose importance level is equal or lower than the kill level are closed definitively. Now, every time a particle enters a scoring volume, or collides in it, if the particle importance during crossing/collision (the one defined by function ξ in Equations 4.4 and 4.5) is lower than the volume importance level, the latter is updated to this value. This implies that contributions of all particles from the first iteration to the last one before the volume is closed are accounted for. To keep the result unbiased, the tallies are weighted by the iteration weight as in Equation 4.21, except here that Q is not equal to the last iteration of the AMS, but to the iteration at which the volume was closed. In practice, this importance level is initially set to infinity, and may be updated to lower values during the simulation.

4.2.5 About the importance function

To satisfy the conditions of Equation 4.5, the importance of a track/branch/point is set to infinity when reaching the detector. This means that the track will never be deleted to be re-sampled, given that if $I_{kill} = \infty = Z_{max}$, the algorithm stops.

AMS iterations are supposed to stop when

$$I_{kill} = Z_{max} \quad (4.24)$$

which implies that at least $N - K + 1$ tracks have an importance equal to infinity, therefore have reached the detector. However, the algorithm will also stop if

$$I_{kill} = I(K) = I(1). \quad (4.25)$$

Put another way, the $N - K + 1$ first tracks (at least) have exactly the same importance. This may be highly unlikely for continuous importance functions, but can happen if the function is discrete. This situation corresponds to an accretion of tracks on a discrete level of importance, leading to the algorithm prematurely stopping. To avoid this, the function must not present discrete levels.

Finally, since the importance function is only used to rank particles, only the relative importance between two particles matters, which makes the AMS reasonably robust and easy to use. Indeed, it was observed that even a rough estimation of the ideal importance function would provide an improvement of the variance estimation, as shown by Louvin in Ref. [6, 7].

Conclusion

The AMS is an iterative algorithm intended for score estimations in the case of a rare event. It was first implemented in the Tripoli4 Monte Carlo code for neutron transport and used for shielding calculations. Its principle lies in the resampling of a part of particles histories at each iteration to get more particles closer to a detector, hence assist with the estimation of a rare event.

Besides, allowing for an easier estimation of tallies in a detector, and reducing the variance of this score, it is possible to rely on an *on-the-fly* scoring procedure, giving access to scores in intermediate regions of the phase space between the particles source and the target detector.

The main component of the method is the importance function provided by the user. Although the result has been shown to be unbiased regardless of the function used, it must fulfill some criteria for the algorithm to iterate properly:

- function with discrete levels must be avoided, or the iterations might prematurely stop,
- the value must be maximum inside the detector / target volume,
- in order to get a lower variance, it should reflect the proximity of a particle in the phase space to the target detector.

In the subsequent chapters, we will look at how, starting from this state, the method was used to cope with neutron criticality and kinetics calculations.

Chapter 5

Criticality calculations with the AMS

All models are wrong, but some are useful.

- George Box

Contents

5.1	AMS in criticality calculations	70
5.1.1	Turning criticality calculations into attenuation problems . .	70
5.1.2	Defining a detector	71
5.1.3	Population control mechanism	71
5.1.4	Scoring over successive generations	72
5.2	Application to a one dimensional slab reactor	75
5.2.1	Korrigan : a <i>toy-model</i> Monte Carlo code for neutron trans- port	75
5.2.2	Description of the system properties	75
5.2.3	Homogeneous bare slab reactor : numerical results	76
5.2.3.1	Convergence of inactive cycles	77
5.2.3.2	Averaged fundamental mode	80
5.2.3.3	Clustering	82
5.2.3.4	Variance estimation	85
5.3	Application to heterogeneous slab	89
5.3.1	Description of the geometry, materials and simulation pa- rameters	89
5.3.2	Estimations of the fundamental mode	90
5.3.2.1	Estimation of k_{eff}	90
5.3.2.2	Estimation of the fundamental flux distribution . .	91
5.3.3	Analyze of the particle weights over space and generations .	91
	<i>Conclusion</i>	96

This chapter presents the application of the Adaptive Multilevel Splitting (AMS), previously introduced, to a steady-state problem in the context of criticality calculations. The work featured in the present chapter was initially outlined in Ref. [114] for which a more detailed overview is reported in Ref. [115].

5.1 AMS in criticality calculations

For the AMS to be used in criticality calculations, a recast of the problem is necessary to make it suitable for variance reduction. The AMS for particle transport was typically designed to estimate rare events in the context of shielding problems. In criticality, if $k_{\text{eff}} < 1$, the system tends to go extinct with generations. Therefore, it is clear that the lower the k_{eff} , the less likely a neutron progeny is to survive over generations, and reaching a distant generation eventually is a rare event. In that context, the AMS could be used to re-sample histories and push them across generations, as an alternative to more classical population control technics (some of which are presented in Section 2.2.4.1) already used in the power iteration. It has been shown in Chapter 1 that even for critical (and supercritical) systems, the probability for a fission chain to survive may decrease with time (cf. Section 1.3). As a result, it is possible to apply the AMS to re-sample neutron histories even in those contexts. The probable diverging number of branches inside a history may however require a recast of the problem to enforce a stronger attenuation of the number of particles over time.

5.1.1 Turning criticality calculations into attenuation problems

While it is (very) unlikely that the particle population will decrease drastically in critical and supercritical systems, it is possible for the numerical population to decrease while maintaining the physical population constant or increasing. Let us distinguish the *physical particles*, representing the physical population of the modeled system, from the *numerical particles*, effectively simulated by Monte Carlo methods. Their number might differ in a non-analog calculation, since a numerical particle can represent multiple physical particles through its statistical weight. Thus, by biasing the collision kernel, we can limit the branching events, and thus the number of particles in the simulation, resulting in a decrease of the number of numerical particles even in supercritical systems. The branchless collision method (cf. Section 2.2.4.2) has consequently been used in combination with AMS in order to model systems whatever their k_{eff} . This way, it was possible to generalize an attenuation problem description to critical and supercritical systems provided that there is a way to kill numerical particles (e.g., by leakage or Russian Roulette). There is another advantage to use the branchless collision method, even for subcritical systems. If we consider the way branching tracks are sampled by the AMS (see Section 4.2.3), we notice that the more branches a track contains, the higher the number of re-sampled particles (recall that 1 branch represents 1 particle). This implies that for branching systems, the number of re-sampled particles could increase at each iteration of the AMS, resulting in a slowing

down of iterations. Therefore, limiting the number of branches inside tracks thanks to the branchless collision also helps to keep the calculation time under control without introducing a bias regarding the average values of scores.

5.1.2 Defining a detector

Typically, in the power iteration, neutrons progenies are brought from generation 0 (the initial one sampled from our initial source) to generation G , where G is equal to the sum of inactive and active cycles. Particles go from one generation to the next one through fission, and population control helps keeping their number constant over generations (preventing it from dying or diverging). In a criticality calculation problem recast as a generational attenuation problem, one would not be interested in a specific detector localized in space and/or energy only, but in a neutron generation reached by enough particles (i.e., generation G). Here again the mechanism that brings one neutron from one generation to the next would still be fission, but no population control other than the AMS would be used, leading to a decrease of the particle number over generations. For this purpose, neutrons are assigned a generation parameter, just like energy or position, to monitor the transport of the progeny over generations, and the importance function should reflect the fact that particles are to be pushed over generations. Since the generation parameter is one-dimensional and unidirectional, a simple importance function such as

$$I = g \tag{5.1}$$

should suffice, where g is the neutron generation. However, to satisfy the continuity (as opposed to discrete) condition recommended in Section 4.2.5, an additional term can be added

$$I(\mathbf{r}, g) = g + f(\mathbf{r}), \tag{5.2}$$

where $f(\mathbf{r})$ is a function of space (other variables of the phase-space could also be used, but for the rest of this chapter, we will only consider space to test the method, since the model that has been used, which is described later, only consider single energy neutrons).

5.1.3 Population control mechanism

By re-sampling tracks in order to bring more particles to the target generation G , the AMS would act as a population control method (but only for numerically subcritical systems). Indeed, it is possible to make g the prevailing term by restraining $f(\mathbf{r})$ to $[0; 1]$ this way, particles would always be sampled in the generation g or $g + 1$ because

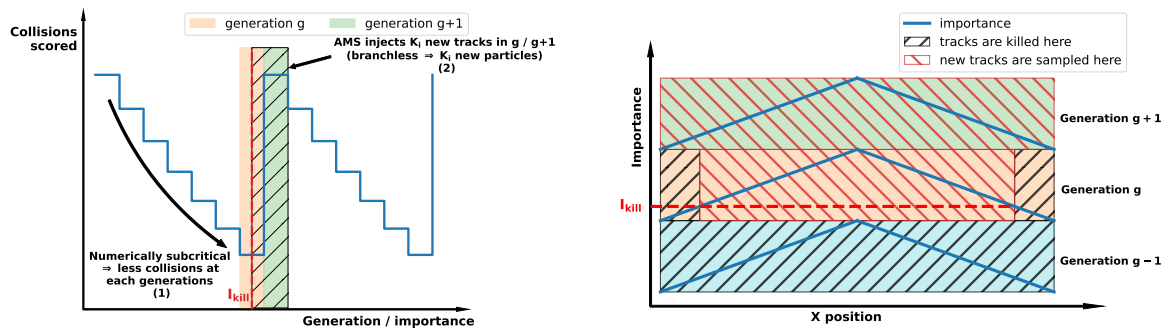
$$g \leq I_{\text{kill}} \leq g + 1. \tag{5.3}$$

This equation implies that at least K tracks (equivalent to particles for non-branching collisions) did not reach generation $g + 1$ (since at least K tracks have an importance lower than I_{kill}), meaning that the population at generation $g + 1$ has fallen below $N_0 - K$ particles. The population control mechanisms of the AMS in the case of such importance function, for which $f(\mathbf{r})$ tends to favor particles in the center of the geometry, are presented in Figure 5.1. As the neutrons disappear over the generations, K new particles are regularly sampled inside some generations as portrayed by Figure 5.1a. Basically, the idea here is to re-inject new tracks inside the system once their

number has fallen below a threshold value ($N_0 - K$). Depending on the number of tracks disappearing at each generation, the re-sampling occurs about every Δg generations, sampling $\Delta N = K_i$ new tracks, where K_i is the number of tracks re-sampled at iteration i (as a reminder, the actual number of tracks that are re-sampled is equal to $K_i \geq K$, cf. Section 4.1.1).

Put simply, controlling the population in a subcritical system with the AMS can be seen as setting the lower limit below which the population should not decrease. Indeed, the user sets the value K , inducing ΔN (which may vary since $K_i \geq K$ particles are re-sampled during an AMS iteration) and therefore Δg (which may consequently also vary). On the contrary, for classical population control techniques, whether for power iteration or kinetics, the user sets the value of Δg (or Δt in kinetics), leading to the value of ΔN (which may fluctuate). Controlling the population with the AMS may thus prove to be more valuable for the user if he/she prefers to set a minimum number of tracks per time bin or generation rather than the frequency at which population control is performed.

Since the branchless collision method is used, track death points are close to the boundaries of the system (in the absence of Russian Roulette, the tracks only die out by leakage). This means that new tracks will be sampled either closer to the center of the geometry in generation g or anywhere in generation $g + 1$ as shown in Figure 5.1b.



(a) Re-sampling particles over generations for $f(\mathbf{r}) \in [0; 1]$.

(b) Re-sampling regions for $f(\mathbf{r}) \in [0; 1]$ maximum at the center of the geometry.

Figure 5.1: Re-sampling particles in criticality calculations with the AMS.

5.1.4 Scoring over successive generations

Since Monte Carlo estimates in criticality calculations are computed over successive cycles, it is necessary to have access to tallies in all generations and not only in the target generation. To achieve this, the on-the-fly scoring process described in Section 4.2.4 is used. Instead of geometrical volumes, generations are set as scoring volumes (since the generation dimension is now included in the phase-space to follow neutrons over generations). If we then look at the population control mechanism described above, if $g \leq I_{\text{kill}} \leq g + 1$, then the scoring volume corresponding to generation g should be closed even if particles are re-sampled in it. Indeed, in Figure 5.1b, each generation defines a scoring volume, and generations marked by a black dashed area must be closed to scoring (see Section 4.2.4 for more details on how scoring volumes

should be closed to scoring to avoid duplicated contributions¹). This affects the scores in the following manner, particles are re-sampled in generations g and $g + 1$, but only contributions in generation $g + 1$ must be scored, as shown in Figure 5.2.

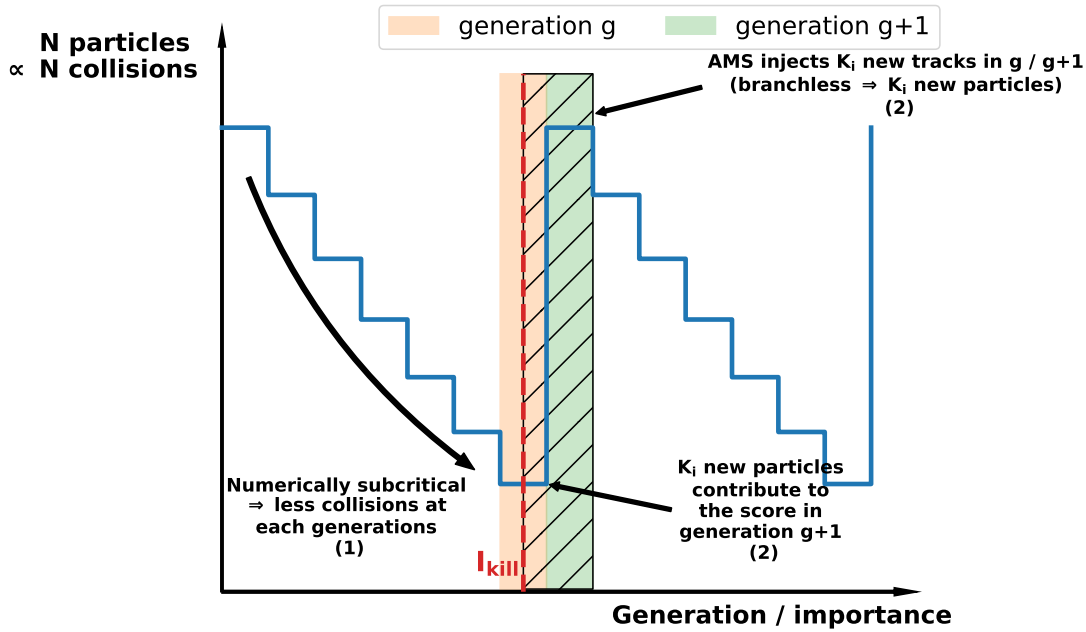


Figure 5.2: AMS population control effects on the number of collisions scored. The hatched area represents the generations in which new particles may be sampled.

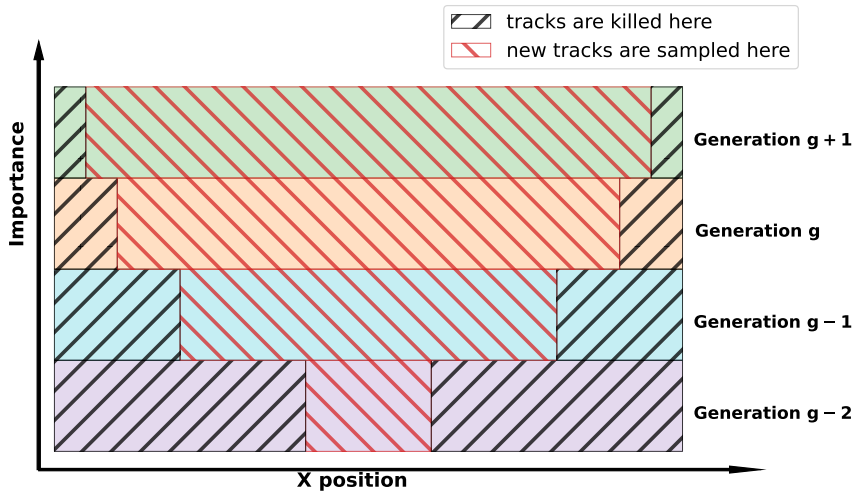


Figure 5.3: AMS sampling zones for cases where $f(\mathbf{r}) > 1$.

While Figure 5.1 displays a case where $f(\mathbf{r})$ has values in $[0; 1]$, the effect just described would not change much if $f(\mathbf{r}) > 1$ (in that case, particles could be sampled "in the past" since some regions in generation $g' < g$ would have an importance higher

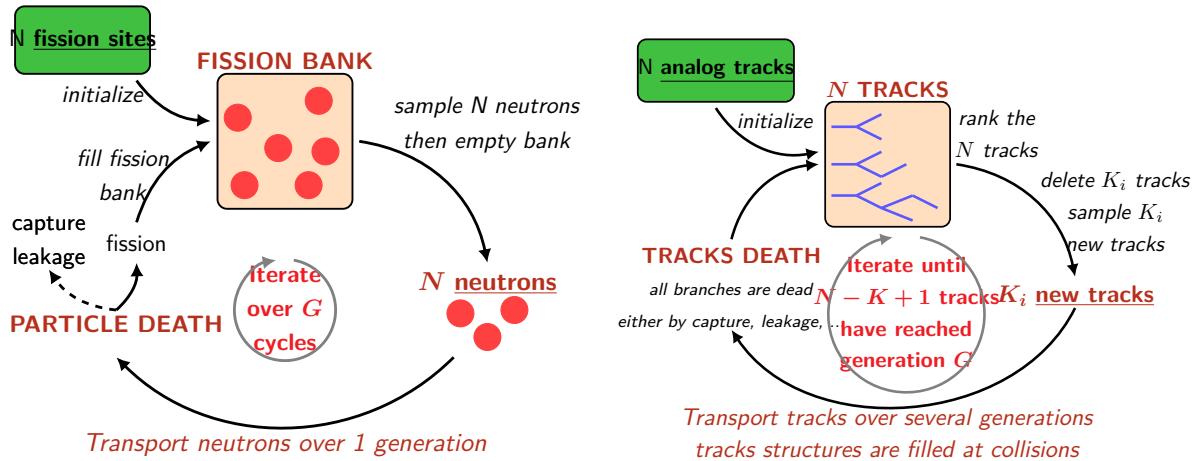
¹Since a volume is closed when its importance level is lower than I_{kill} , and since points with importance lower than I_{kill} were saved in this volume, it means that its importance is also lower than I_{kill} , and therefore it must be closed according to Section 4.2.4

than some others in generation $g : g' + f(\mathbf{r}_1) > g + f(\mathbf{r}_2)$). In that case, generation g and those that precede it, and even generation $g + 1$, would still be closed to scoring as depicted in Figure 5.3.

Either way, the resulting algorithm is compared to the Power Iteration in Figure 5.4.

In a nutshell, the main differences with classical power iteration based algorithm are highlighted here:

- the AMS does not stop neutrons at each generation. Instead, neutrons and their progeny are transported over several generations until the end of the fission chain.
- The AMS algorithm re-samples new particles and thus performs a sort of population control (it cannot kill particles however), which is not set by the frequency at which particles are re-sampled like generation-wise population control of criticality calculations. Instead, the user defines the number of tracks that need to be re-sampled (i.e., the parameter K defined in Chapter 4) which may result in a regular population control given that the importance function mainly pushes particles through generations and that branching events are limited by the branchless collision method.
- While current power iteration based algorithms bring N from generation 0 to generation G , the AMS aims at bringing at least $N - K + 1$ neutron histories at generation G (which can imply more particles in the case of branching events).



(a) PI : iterates over fission neutrons (1 generation per iteration) after being initialized with an arbitrary fission distribution. Neutrons are sampled from fission neutrons of last iteration.

(b) AMS in criticality: iterates over re-sampled tracks (multiple generations per iteration) after being initially fed with N analog tracks.

Figure 5.4: Comparison scheme between the Power Iteration (PI) and AMS used for criticality.

5.2 Application to a one dimensional slab reactor

As a proof of concept, the technique just described in the previous section was applied to a simple case of homogeneous bare slab reactor.

5.2.1 Korrrigan : a *toy-model* Monte Carlo code for neutron transport

The first step was to implement the AMS in a controlled environment to characterize the method. Most of well-known Monte Carlo codes able to perform criticality calculations (sometimes called *k-code*)² have many lines of code³. Besides, interacting functions eventually make experimenting with a new method a daunting task. Therefore, in order to set up a proof of concept for criticality calculations with the AMS, an ad hoc Monte Carlo code, called Korrrigan, has been developed from scratch during this thesis.

Korrrigan is C++ written with about 6 000 lines of code (including the AMS classes). It merely aims at mimicking the main features of Monte Carlo *k*-eigenvalue calculations to conduct a proof of concept regarding the use of the AMS in criticality calculations, and has not been thoroughly validated against complex codes for that specific reason. Neutrons can either be tracked over time or generations depending on the problem (static or time-dependent) in one-speed neutron transport calculations over cartesian geometries. So as to numerically reproduce the main features of the Power Iteration (PI), the combing method and a basic sampling with replacement were implemented to perform population control. In its static mode, the algorithm used is either the AMS as presented in Section 5.1 or the power iteration. The diagram previously given in Figure 3.1 also presents the implementation of the power iteration within Korrrigan, which is typical of the power iteration mechanisms that can be found in main industrial Monte Carlo codes handling criticality calculations [117, 118, 116, 119].

5.2.2 Description of the system properties

The first system that has been modeled in criticality calculations is a one dimensional homogeneous bare slab reactor with absorbing boundary conditions (leakage). The total size of the slab is 100 cm, from $x_{min} = -50.0$ cm to $x_{max} = 50.0$ cm, and the calculations were done with Korrrigan (hence mono-energetic neutrons). The physical properties of the system are described in detail in Table 5.1. These properties were chosen to grasp the behavior of a realistic loosely coupled system in a much more simple benchmark, for which the typical migration length of neutrons ($1/\Sigma_a \approx 1.7$ cm) is much lower than the typical size of the system ($L = 100$ cm).

Several simulations, with calculation options shown in Table 5.2, were performed to investigate the impact of the AMS versus the power iteration on the results.

Starting from a uniform fission distribution, all the calculations presented below were done with $N_0 = 1000$ initial neutrons (this corresponds to 1000 independent initial tracks for the AMS, and 1000 neutrons per cycle for the power iteration) over $G = 1000$ successive generations in $M = 1000$ independent runs. The aim was also to study the effects of the AMS regarding spatial and generational correlations linked to clustering phenomena previously presented in Section 3.1.3.3. The number of neutrons

²E.g., MCNP [116], TRIPOLI4 [117], SERPENT2 [118], MORET [119]

³E.g., about 200 000 for SERPENT2 and TRIPOLI4

Table 5.1: Physical properties for homogeneous 1D rods

Mean number of fission neutrons ($\bar{\nu}$)	2.383
Neutron speed (v)	$2.2 \times 10^4 \text{ cm.s}^{-1}$
Macroscopic cross sections	
Fission (Σ_f)	0.250 cm^{-1}
Absorption (Σ_a)	0.575 cm^{-1}
Scattering (Σ_s)	0.425 cm^{-1}
Total (Σ_{tot})	1.00 cm^{-1}

per cycle was therefore set to 1000, given the characteristic length of the system and the mean free path before absorption (cf. Table 5.1), to have strong spatial correlations [30] (hence a strong clustering).

 Table 5.2: Description of calculations parameters for the 1D bare slab reactor (*w.r.* means *with replacement*).

Case	Population control	Collisions	Importance
PI analog	sampling w.r.	analog	
PI branchless	sampling w.r.	branchless	
PI combing	combing	analog	
PI combing branchless	combing	branchless	
AMS branchless	AMS	branchless	$g + \cos\left(\frac{\pi x}{2 \times 50}\right)$

Since the system presented in Section 5.2.3 can be described by the one-speed theory, an analytical solution can easily be computed to serve as a reference. The fundamental flux in the diffusion theory is of the following form

$$\phi(x) = \phi_0 \cos\left(\frac{\pi}{2(a+z_0)}x\right) \quad (5.4)$$

with ϕ_0 depending on the normalization, a being the half size of our reactor (here $a = 50.0 \text{ cm}$) and z_0 is the linear extrapolated end point of the reactor defined as

$$z_0 = \frac{2}{3\Sigma_{tr}} \quad (5.5)$$

where Σ_{tr} is the transport cross section [120]. Since all collisions are isotropic in the laboratory referential, Σ_{tr} is equal to the total macroscopic cross section Σ_t .

5.2.3 Homogeneous bare slab reactor : numerical results

By using the AMS in the context of criticality calculations, we seek to solve common issues presented in Chapter 3, i.e., the apparition of particle clusters due to spatial and generational correlations. First of all, it is appropriate to focus on the convergence phase. Even if limitations regarding the convergence of inactive cycles have already been addressed in multiple works, the aim here is to verify the behavior of the method on the convergence of the problem. Then, the estimation of the fundamental mode, and

more particularly the clustering phenomenon are characterized in more detail. AMS effects on clustering and particle families disappearance are then studied, as population control methods were in recent works [97]. Finally, the Figure of Merits (FoM) of the different calculations are compared, taking into account the effect of generational correlations on their estimation.

5.2.3.1 Convergence of inactive cycles

To assess the convergence of the iterative schemes towards the fundamental mode of the system, both the convergence of the eigenvalue (k_{eff}) and the eigenvector ($\phi(\mathbf{r})$) are monitored. The convergence of the neutron flux is considered via its Shannon entropy. The convergence of the k_{eff} is shown in Figure 5.5 as the mean k_{eff} per generation as a function of the cycle number, where

$$\overline{k_{\text{eff}}}(g) = \frac{1}{M} \sum_{m=1}^M \frac{N_g^{(m)}}{N_{g,0}}, \quad (5.6)$$

where $N_{g,0}$ is the initial number of neutrons in that cycle (strictly equal to N_0 for the power iteration), and M is the number of independent runs. The variance of the mean in each generation is computed using the following estimator

$$\text{Var} [\overline{k_{\text{eff}}}] (g) = \frac{1}{M(M-1)} \sum_{m=1}^M \left(\frac{N_g^{(m)}}{N_{g,0}} - \overline{k_{\text{eff}}}(g) \right)^2. \quad (5.7)$$

The convergence (to a superior value since the source was initially uniform) is quite fast for all calculations and apart for statistical fluctuations all methods converge towards the same value (which is about 1.034).

Meanwhile, regarding the flux convergence, the mean Shannon entropy plotted in Figure 5.6 was computed as the entropy of the mean flux distribution over the M independent simulations in each generation

$$\overline{H}(g) = \frac{1}{M} \sum_{m=1}^M \left[- \sum_{l=1}^{N_{\text{bins}}} \frac{\phi_g^{(m)}(x_l)}{\phi_{g,\text{tot}}^{(m)}} \log_2 \left(\frac{\phi_g^{(m)}(x_l)}{\phi_{g,\text{tot}}^{(m)}} \right) \right] \quad (5.8)$$

where N_{bins} is the number of spatial bins along the x -axis (here $N_{\text{bins}} = 100$), $\phi_g^{(m)}(x_l)$ is the flux at cycle g in bin x_l for simulation m computed from a collision estimator, and $\phi_{g,\text{tot}}^{(m)}$ is the total normalized flux at cycle g for simulation m so that

$$\phi_{g,\text{tot}}^{(m)} = \sum_{l=1}^{N_{\text{bins}}} \phi_g^{(m)}(x_l). \quad (5.9)$$

As mentioned in Section 3.1.3.1, the entropy is expected to converge slowly in case of loosely coupled systems, compared to the eigenvalue convergence, which is exactly the case here since it takes about 200 generations until the entropy has reached its asymptotic value (compared to less than 100 for the k_{eff}). The value of the asymptotic entropy is lower than the initial one because the initial neutron distribution was uniform, which corresponds to the higher entropy level. If the calculation had started from a point source, the entropy would have increase during the "convergence" phase during which the flux converges to its fundamental distribution.

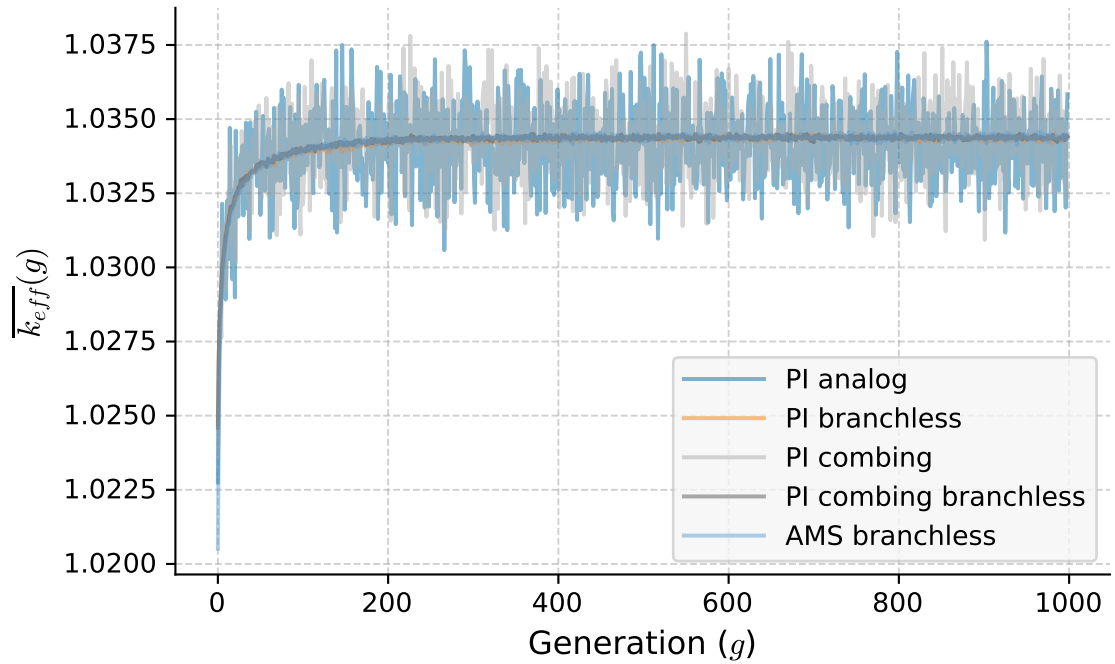


Figure 5.5: Convergence of the average k_{eff} (over 1000 independent runs) with 3σ confidence intervals. Cases PI branchless, PI combing branchless and AMS branchless show the same results, with narrow confidence intervals.

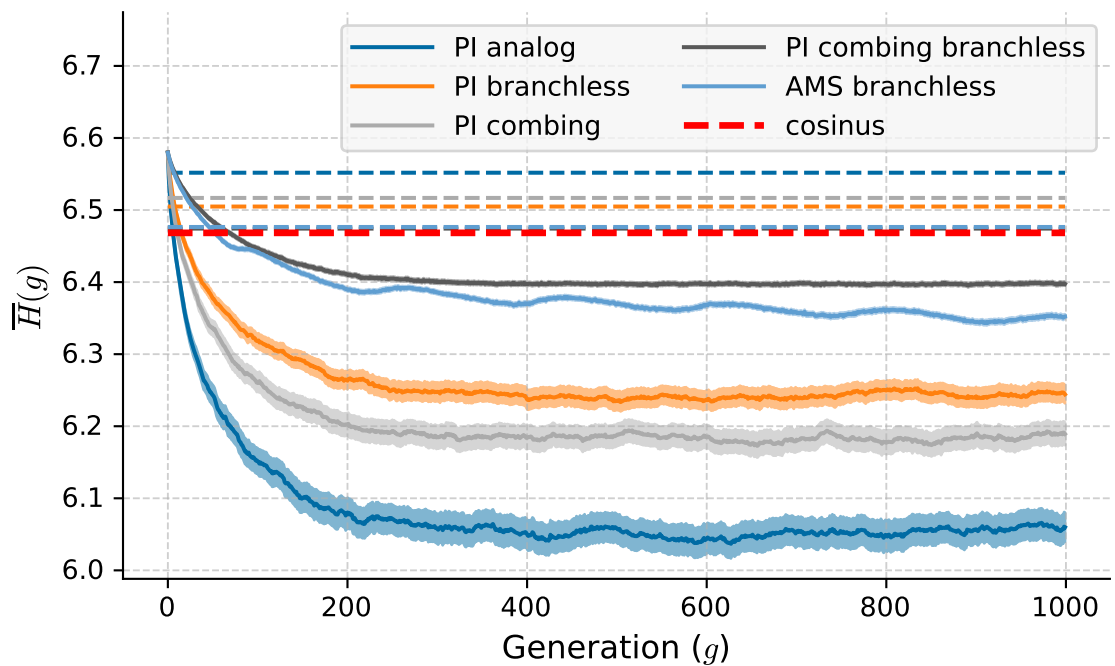


Figure 5.6: Evolution of the mean entropy over generations. The dashed colored lines correspond to the entropy of the flux shape averaged over active cycles for each simulation.

Additionally, the entropy presents oscillations when the AMS is used which is likely due to how the AMS injects particles into the simulation, and can be decomposed into two underlying mechanisms illustrated in Figure 5.1a: the numerical subcriticality of the system (even if k_{eff} might be superior to 1) (1) and the re-sampling of new particles by the AMS only in some generations (2). Indeed, as explained in Section 5.1.3 and 5.1.4, the number of collisions, and therefore the spatial shape of the flux, fluctuate over the generations. Those fluctuations are inversely proportional to the number of particles (qualitatively speaking, meaning that the fluctuations are less pronounced as the number of particles increases). Moreover, the larger the magnitude of the relative fluctuations, the lower the entropy. In a nutshell, the number of particles, hence the number of collisions, declines over generations before being increased once the AMS has sampled new particles, thus inducing growing fluctuations of the flux shape before it is smoothed thanks to the increase of the neutron population after re-sampling. On the other hand, the number of collision is quite stable over generations in the power iteration since population control is operated at each cycle to keep the source population constant.

Based on this observation, it would be possible to reduce these oscillations, would they be unwanted, in two ways. First, reducing K , the number of tracks re-sampled by the AMS, would decrease the variation of the number of particles between one re-sampling step and the next (see ΔN in Figure 5.1a). Doing so would reduce the amplitude of oscillations, but also increase their frequency (according to the mechanism depicted by Figure 5.1a, if ΔN decreases, Δg decreases too). This is what was observed in simulations, as depicted by Figure 5.7.

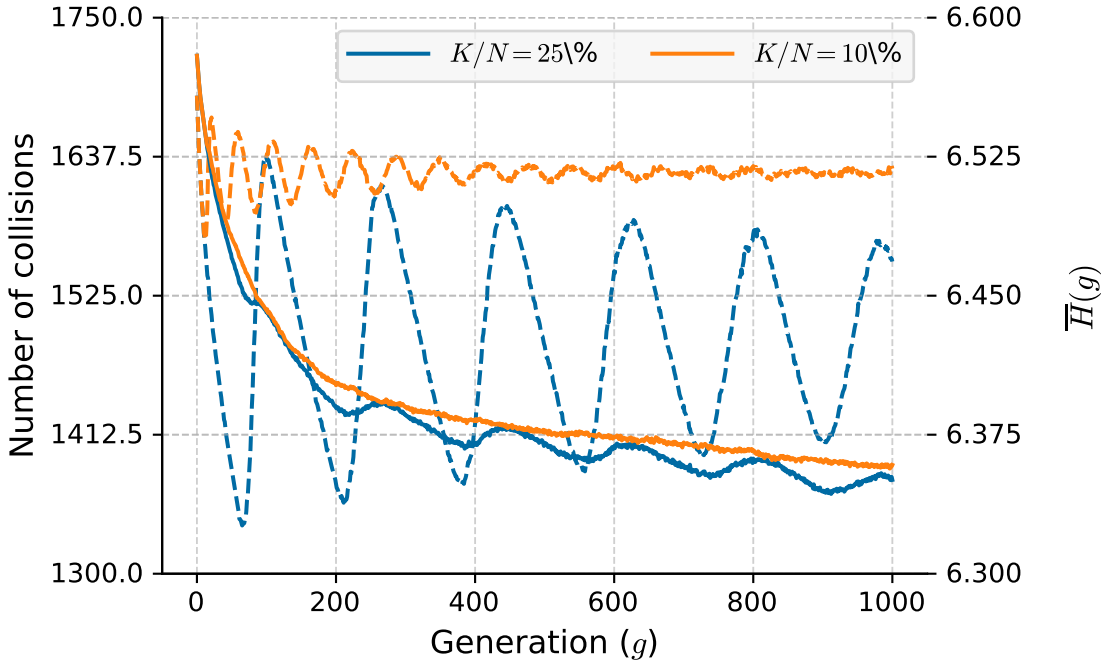


Figure 5.7: Mean entropy (solid lines) and mean number of collision points (dashed lines) per generation for the AMS + branchless case, for $K = 10\%$ and $K = 40\%$.

However, the total computation time will increase if K is smaller since, overall, a larger number of collisions will have to be sampled. The second method consists

in modifying the spatial binning used to score the flux estimator. Indeed, entropy as defined by Equation 5.8 is binning-dependent, and integrating over larger bins would make the spatial fluctuations disappear, thus limiting fluctuations in amplitude. In the end, these oscillations do not appear to be due to poor implementation of the method, nor do they appear to have a significant effect on the average flux shape as noted hereinafter.

On the asymptotic value of the entropy

There is much to say about entropy, at least more than just how many generations are necessary to reach convergence. First and foremost, the absolute value itself carries indications on the spatial shape of the flux. A major drawback of the entropy is that one attempts to assert the flux spatial convergence using a scalar value, yet two different distributions can have the same entropy. While it is useful to have a scalar to monitor the convergence of the flux, it must be therefore used with care since it can lead to false convergence. However, considering Figure 5.6, we notice that all the methods do not converge to the same value of asymptotic entropy, and all of them are different from that of the theoretical fundamental distribution. Besides, entropies of the averaged distributions are different from the ones computed at each generation (these are marked by dashed colored lines in Figure 5.6). This implies that the averaged distribution in a generation is different from the averaged distribution over all active cycles, which could be due to neutron clustering [10]. As an example, a one dimensional bare slab reactor is a system simple enough to illustrate this effect. Let us consider such a system in which neutron clustering leads to a unique travelling wave. This cluster is dense and localized, and consequently the entropy of the spatial distribution it defines is low, in any case lower than the more spread out theoretical fundamental distribution. On the other hand, averaged over successive cycles, the travelling wave tends to flatten the averaged distribution compared to the theoretical one, leading to a higher entropy (because it brings it closer to a uniform distribution). More infos on this topic is provided by Ref. [121].

All things considered, the further the asymptotic entropy reached during convergence is to the average distribution entropy, the further the converged distribution should be from the true fundamental distribution. According to this thought and Figure 5.6, cases PI analog, PI combing and PI branchless are expected to present more clustering than the other two.

All things considered, considering that the asymptotic value is reached after 200 generations, the number of inactive cycles has been set to 200 for all the calculations to compute the results shown below.

5.2.3.2 Averaged fundamental mode

The k_{eff} results over active cycles are presented in Table 5.3 for the average estimation, and in Figure 5.8 for the distribution of the observations made over all active cycles and independent simulations. First of all, all simulated cases converge to k_{eff} values that are consistent in terms of their confidence intervals (according to Table 5.3, all 3σ confidence intervals overlap). The k_{eff} distributions, plotted as violin plots in Figure 5.8 show that the branchless collision method significantly reduces the spread of the

empirical distributions of the k_{eff} values, which results in a decrease of about one order of magnitude for the statistical uncertainties (two to three orders of magnitude for the Figure of Merit) of the mean estimates presented in Table 5.3, compared to the analog collision cases. Hence, regarding the k_{eff} estimation, the branchless collision method seems to be the main contributor to the variance reduction, while the differences between population control methods are not very significant (with slightly better results when combing is used).

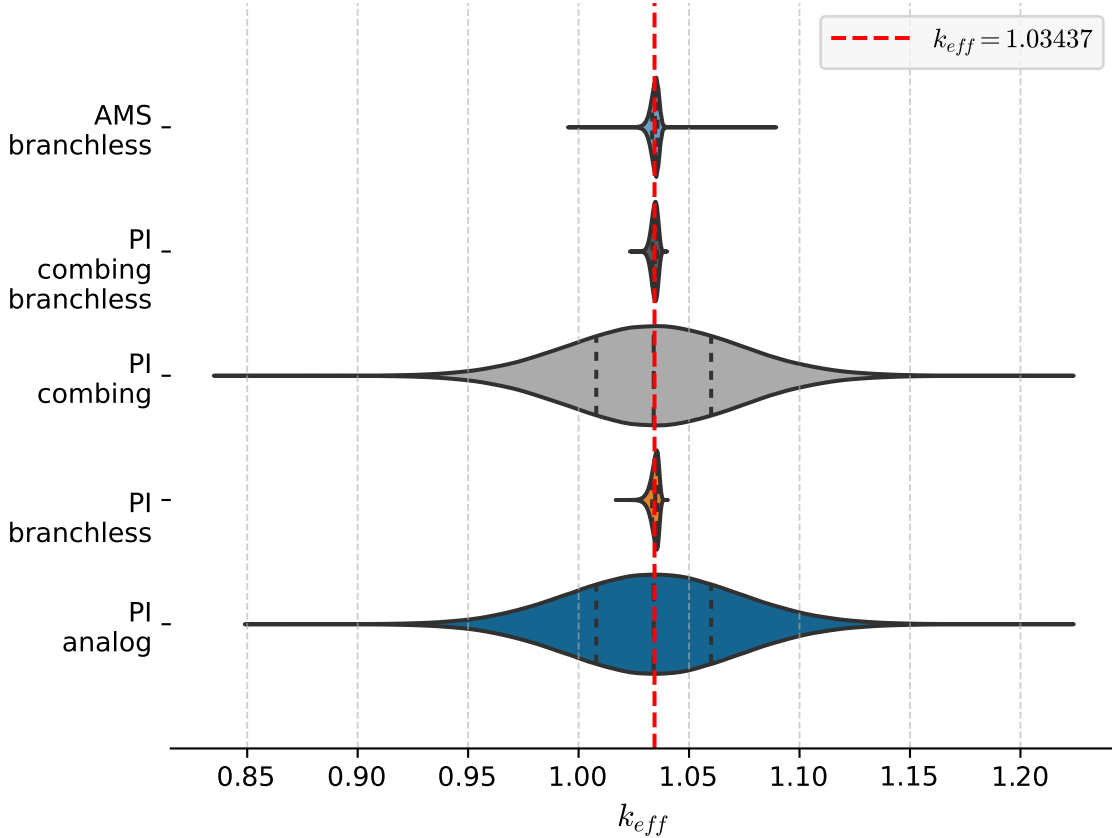


Figure 5.8: Distribution of the k_{eff} after convergence. The bottom plots correspond to the branchless collision (left) and analog collision (right) cases. Violin plots represent kernel density estimations from the empirical distributions, with actual quartiles as dotted lines.

Table 5.3: Averaged k_{eff} values with their standard error (SEM) and relative Figure of Merit (FoM).

Case	$\overline{k_{eff}}$	SEM (pcm)	FoM
PI analog	1.03422	4.3	1.0
PI branchless	1.03433	0.2	3.98×10^2
PI combing	1.03433	4.3	1.22
PI combing branchless	1.03437	0.2	8.36×10^2
AMS branchless	1.03437	0.2	3.00×10^2

The fundamental flux was also computed over 800 successive generations in 1000 independent simulations using the estimator described by Equation 3.14. The results are presented in Figure 5.9, and the first noticeable thing is the lack of consistency between the solutions obtained with the different methods given that the 3σ confidence intervals are far from overlapping. Even though they do not include the analytical solution within their 3σ confidence interval, the combing branchless power iteration, and the AMS branchless show the less discrepancy with the theoretical solution. The deviations between those two solutions and the analytical one are most likely due to the fact that the theoretical solution comes from diffusion theory, which causes a slight underestimation of the flux near the boundaries. The flux in the center is then overestimated due to the normalization effect. As for the deformations of the flux shape in the three other cases, they are probably due to clustering effects mentioned in Section 3.1.3.3, and are discussed below. Indeed, high clustering in the case of absorbing boundary conditions (leakage) may cause a flattening of the flux shape [10].

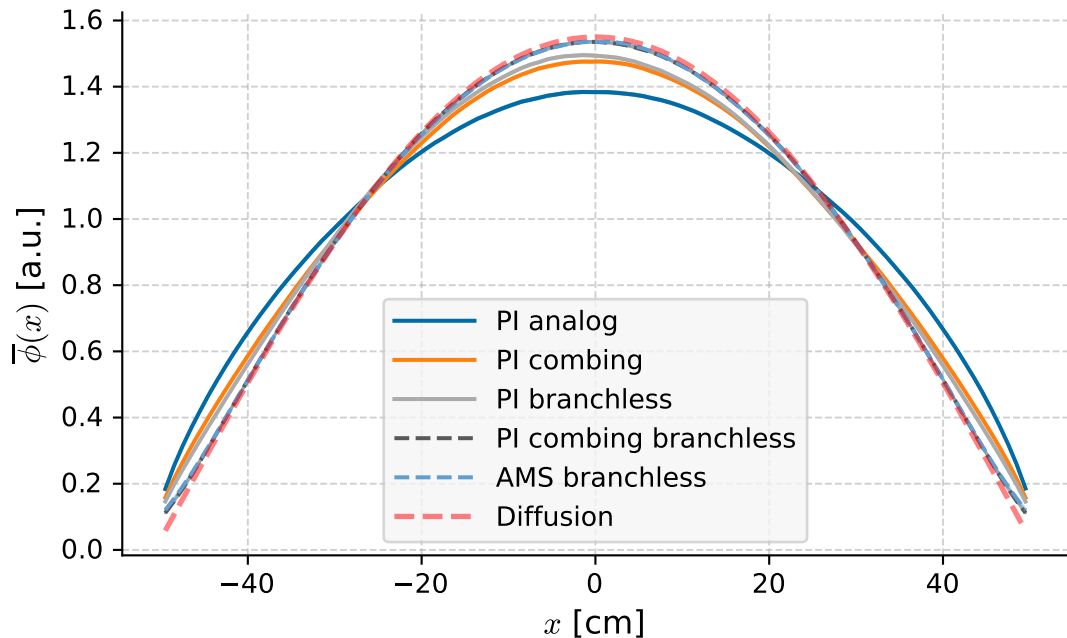


Figure 5.9: Spatial flux profile with 3σ confidence interval (top) with the relative (center) and absolute (bottom) discrepancies against the analytical cosine shaped solution.

5.2.3.3 Clustering

As discussed in Section 3.1.3.3, spatial correlations measurements can help tracking the formation of neutron clusters. These spatial correlations were computed over 100 spatial bins for each case following Equation 3.18, and are presented in Figure 5.10. Based on these results, the PI combing branchless and AMS branchless cases are the less likely to induce clustering due to their almost nonexistent spatial correlations, which corroborates results in Figure 5.9. On the other hand, the three other cases show high correlation levels due to the formation of neutron clusters, which could ultimately lead to biased flux estimates.

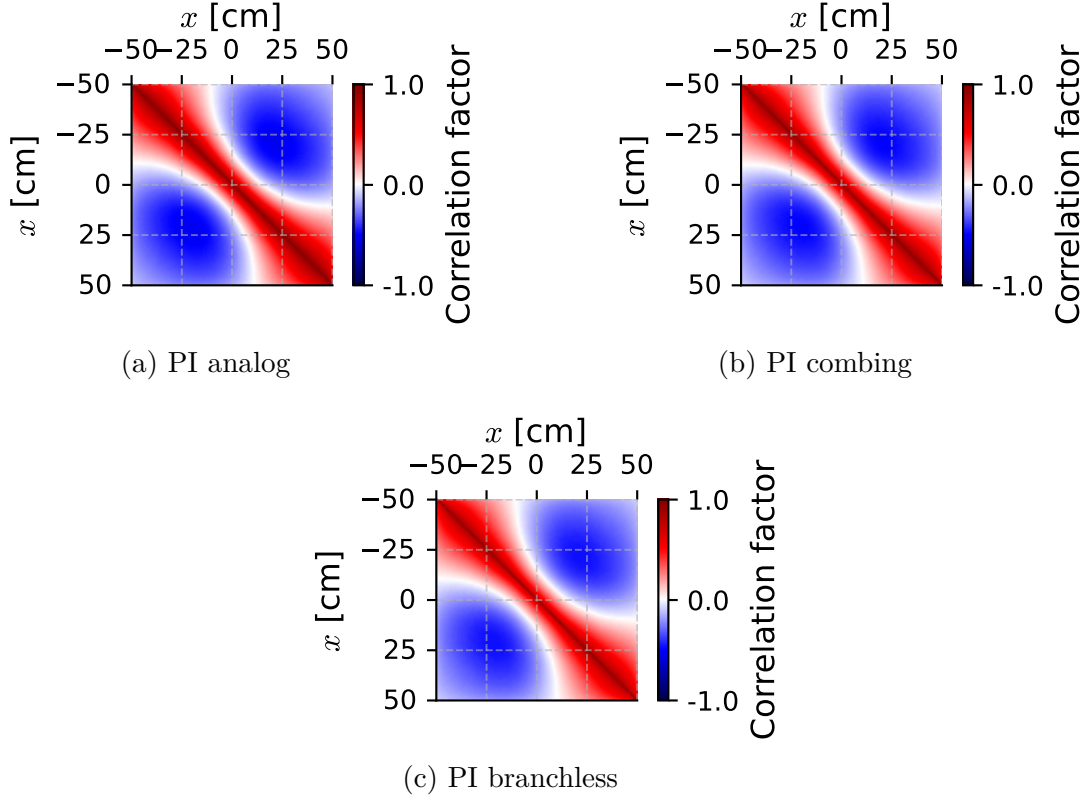


Figure 5.10: Spatial correlations for the homogeneous 1D rod (scale from -1 to 1).

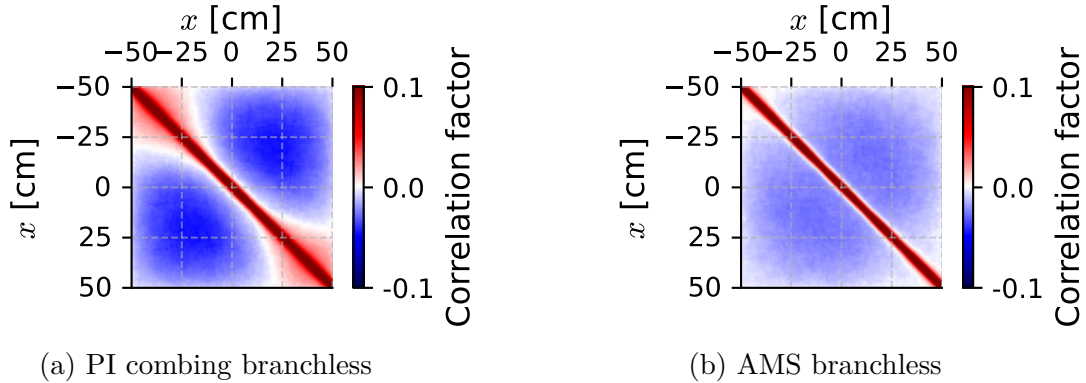


Figure 5.11: Spatial correlations for the homogeneous 1D rod (scale from -0.1 to 0.1).

Sutton, as well as Dumonteil et al. have shown that these strong spatial correlations may be due to the increase of the number of correlated pairs of particles caused by the extinction of independent families of neutrons [97, 10]. To investigate the behavior of the AMS regarding correlated particles, the evolution of the number of independent neutron lineages was monitored over generations. Regarding the extinction of neutron lineage, the average number of families composing the neutron source at the beginning of each generation is plotted in Figure 5.12. All calculations started from 1000 independent families (defined by the first 1000 particles sampled independently) at generation 0 (on the plot of Figure 5.12, the monitoring starts at generation 1, that is why the number of families is different depending on the case). Their number decreases over

generations as physical events or sampling through population control remove neutrons from the batch, while some neutrons are split but do not induce the appearance of a new family. By avoiding the death of particles by sterile capture, a source of lineage extinction, the branchless collision limits the loss of families in a non-negligible way as seen in Figure 5.12. Yet, their number inevitably ends up collapsing if sampling with replacement is used. Combined with the combing method or the AMS however, the effect is amplified, as more than a tenth of the initial families survive the 1000-th generation, compared to one or two for lineages in the other three cases. It is clear then

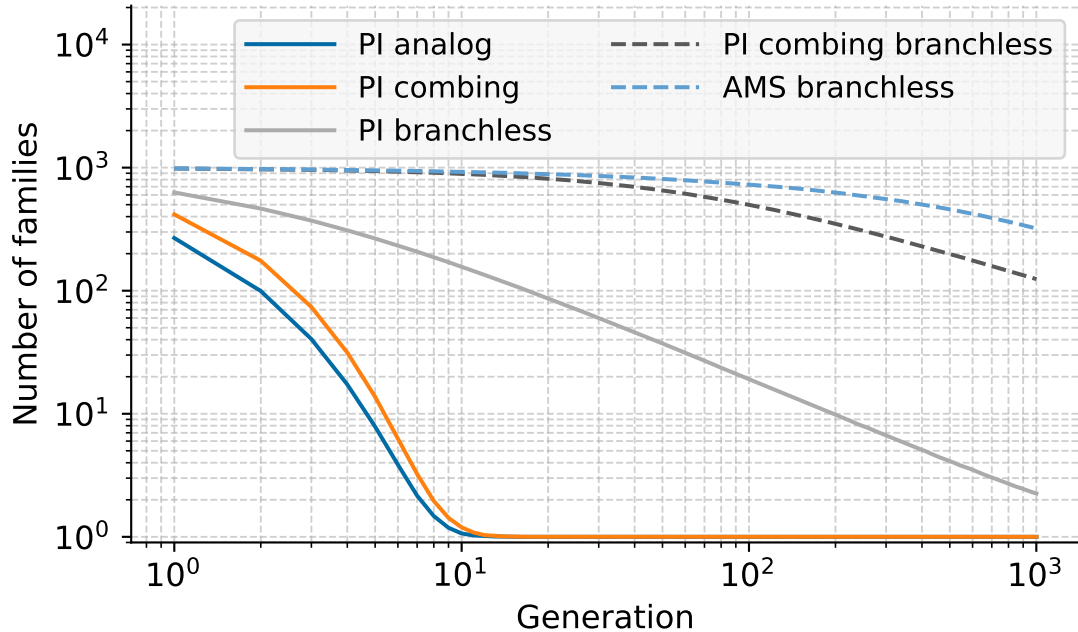


Figure 5.12: Mean number of families over generations.

that both how the physics is modeled, and how particles are sampled during population control matter when wanting to limit the increase of correlated pairs of neutrons, which corroborates recent work authored by Sutton [97]. To quantify the impact of these two parameters, the number of families ended by physical events and by population control are shown in Figures 5.13 and 5.14 respectively.

Regarding the death occurring due to physical events (i.e. during the transport step), their number is initially higher when the branchless collision is not used (about one to two orders of magnitude as seen in Figure 5.13) due to the fact that neutron lines can end by capture. It then drops relatively rapidly because the number of families remaining is low (the drop of the relative number of families killed occurs around generation 10, which corresponds to the point where the number of families reaches one in Figure 5.12). Whereas for the branchless cases, the relative number of losses remains constant and very low (around 1% according to Figure 5.13). Regarding population control, it is clear from Figure 5.14 that the combing method allows for a better conservation of independent particles, which seems to be enhanced when combined with the branchless collision. On the other hand, using branchless collision and a basic sampling with replacement is not very effective, because removing one neutron during population control is equivalent to removing one family from the batch, given that all families have only one neutron in this absence of splitting events (at least before the

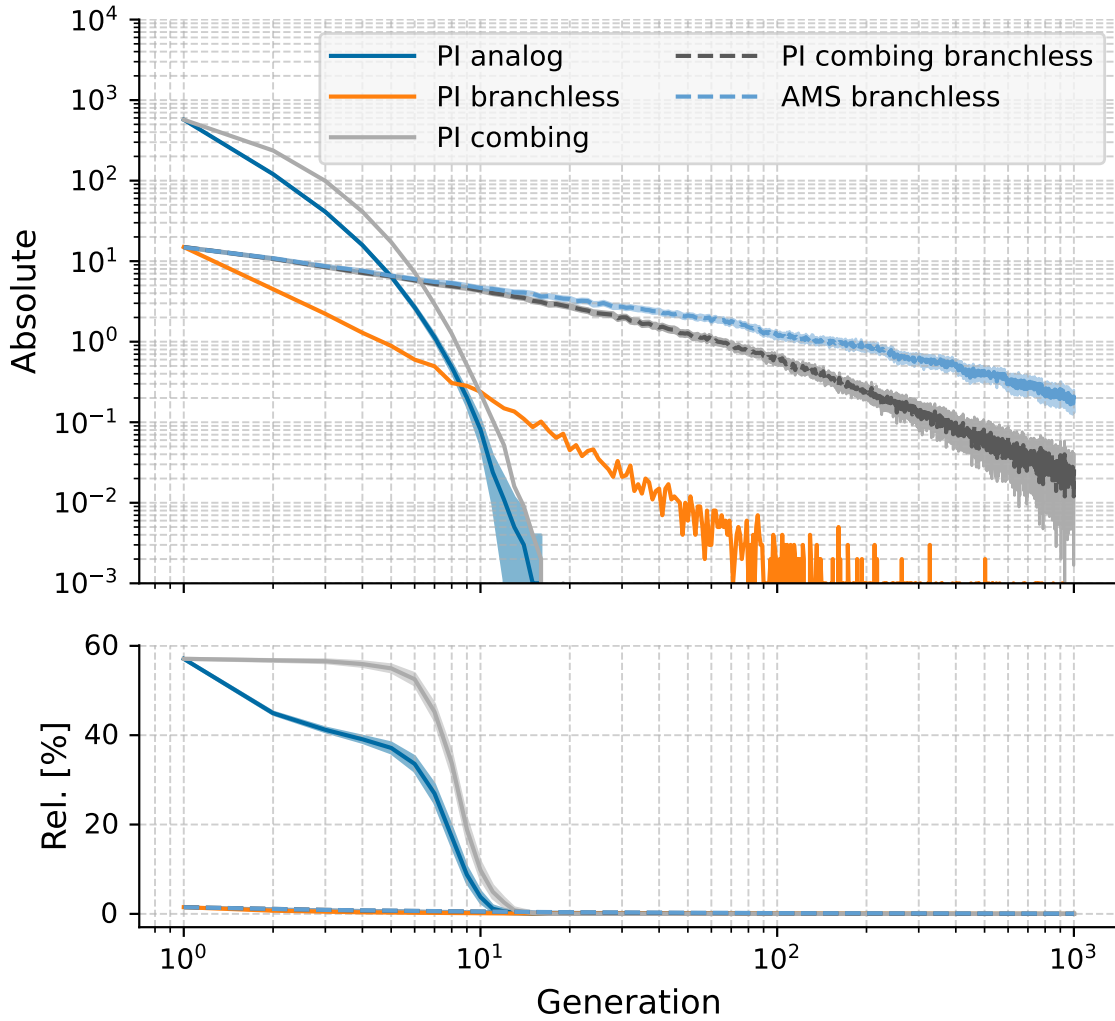


Figure 5.13: Absolute (top) and relative (bottom) number of independent neutron families killed by birth/death process

first population control step). In addition, since more families end up by capture in the analog collision cases, there are fewer of them at population control, and thus fewer terminated. As the AMS does not eliminate particles but only re-injects them into the simulation (see Section 5.1.3), the only source of disappearance is through the physical process.

With regard to the observations made above, using the AMS combined with the branchless collision method is the most effective way (amongst the ones presented in this chapter) to mitigate correlations and thus clustering. This combination is very closely followed by the combing plus branchless collision combination in terms of impact on the clustering phenomenon.

5.2.3.4 Variance estimation

Estimating the average of a score and its variance over successive cycles introduces correlations between observations and therefore a bias on the latter as stated in Section 3.1.3.2. Consequently, to quantify the bias on the flux estimates, generational corre-

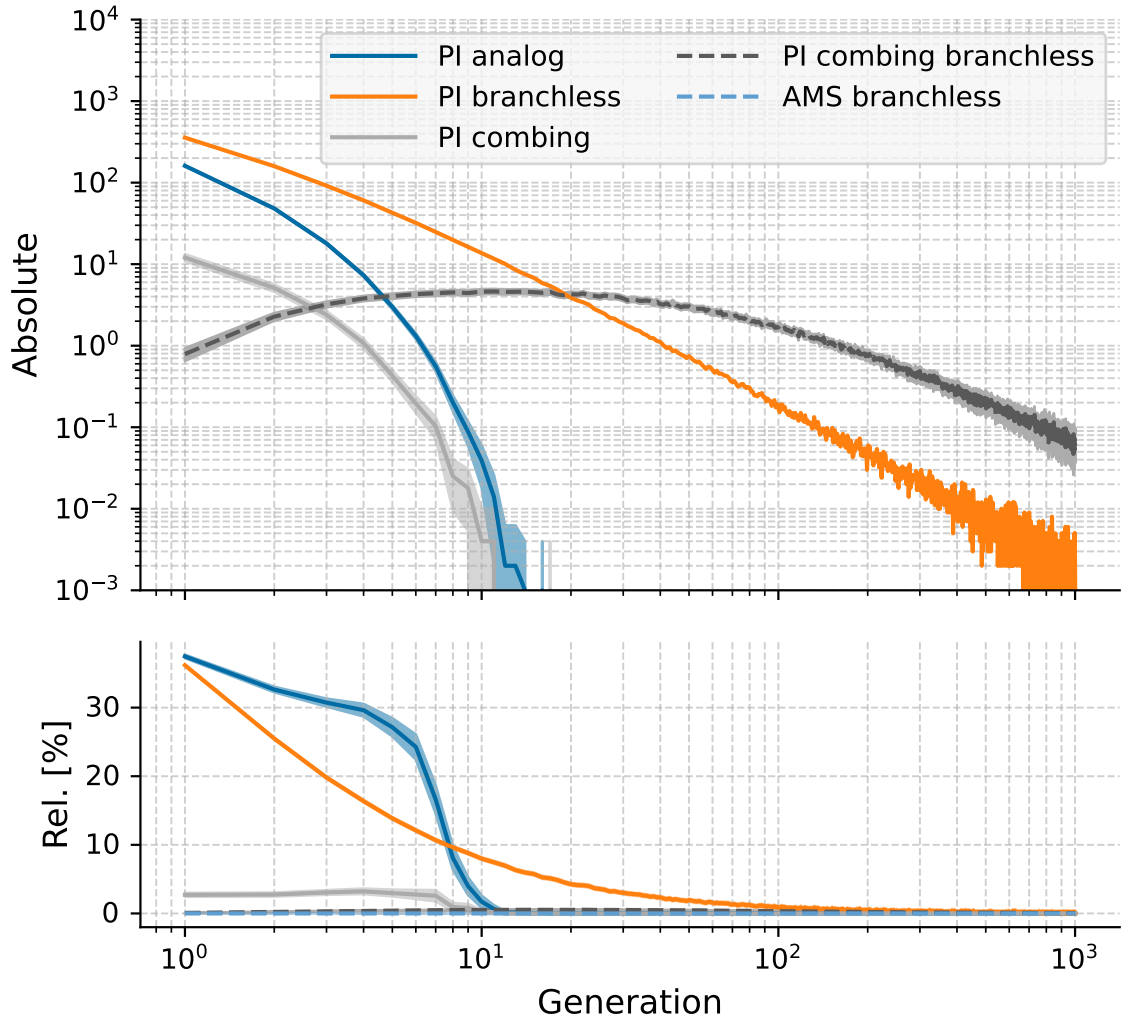


Figure 5.14: Absolute (top) and relative (bottom) number of independent neutron families killed by population control

lations were computed over active generations (i.e. once the spatial flux has allegedly converged regarding the entropy criterion) in all spatial bins.

The pattern of generational correlations for the PI analog case is displayed in Figure 5.15. If a cut were to be made in an arbitrary generation and the correlation factor displayed as a function of x , we would notice two maxima around -25 cm and 25 cm ($1/4$ and $3/4$ of the total length of the slab), and three minima in -50 , 0 and 50 cm (0 , $1/2$ and 1 of the total length). This behavior has been observed before [122], and is due to the oscillations of the higher modes of the eigenvector which leads to higher correlations on the antinodes of these modes inducing higher variances at these locations, and lower correlations hence lower variance on the nodes.

In regions of strong correlations (i.e., for $x = -25$ cm and $x = 25$ cm), the deviations between the methods are the largest as shown in Figure 5.16. It shows that the two cases for which clustering was a minor issue are also the two cases with almost nonexistent generational correlations, namely the PI combing branchless case and AMS branchless case. Back to the variance estimation, this implies that the gain in term of FoM might be more important in these areas. To compare the effectiveness of

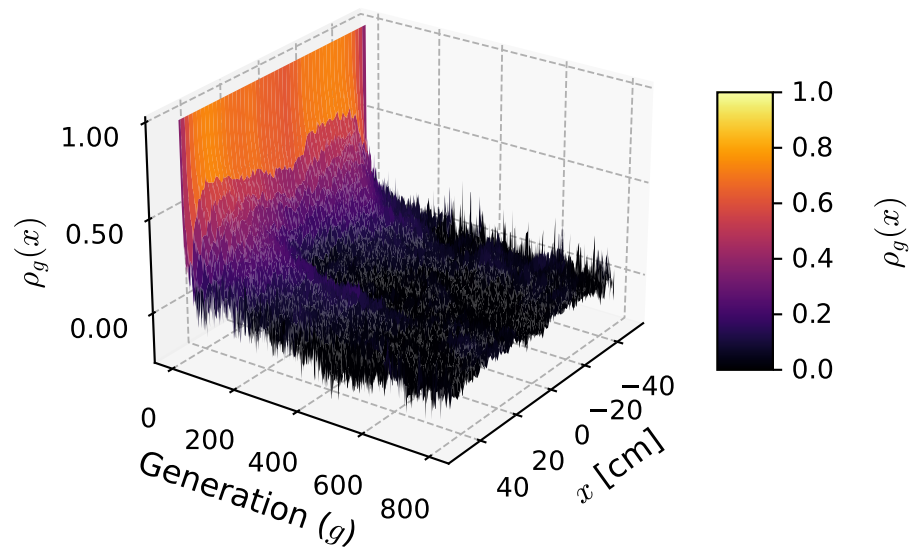


Figure 5.15: Generational correlations over space for the homogeneous 1D bare slab reactor (PI analog case).

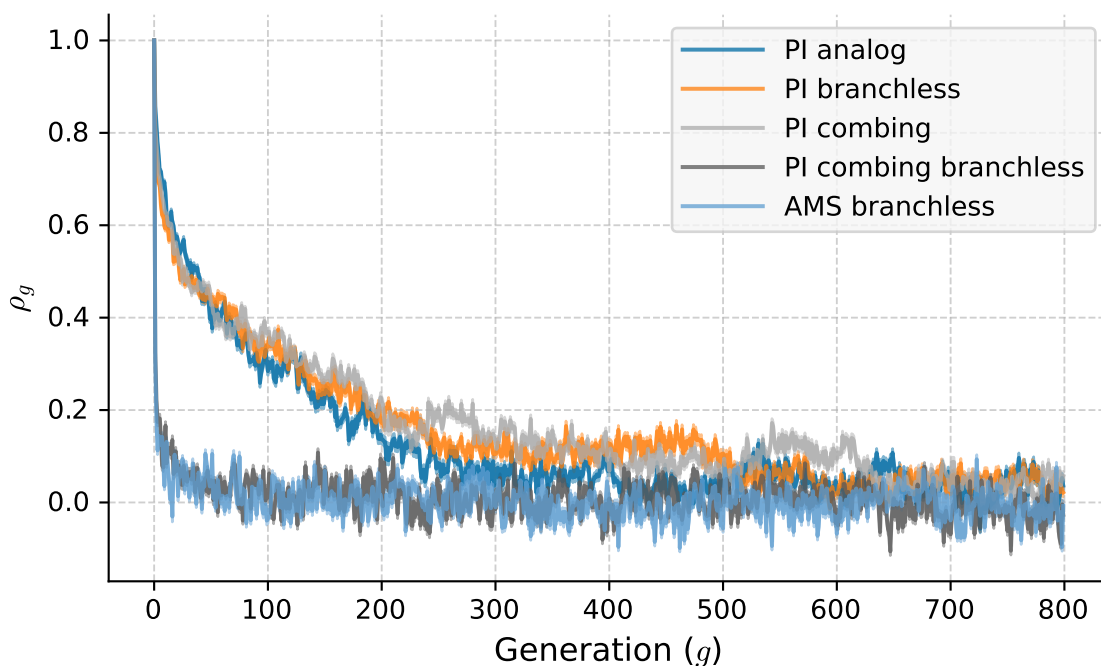


Figure 5.16: Generational correlations at $x = 23.5$ cm for the 1D bare slab reactor.

the methods, the FoM on the flux calculation was computed in each spatial bin, after correcting the variance by accounting for correlations using Equation 3.17. The results displayed in Figure 5.17 show an increase in the FoM up to two orders of magnitude when the branchless method is used in combination with the AMS or the combing

method, for any x .

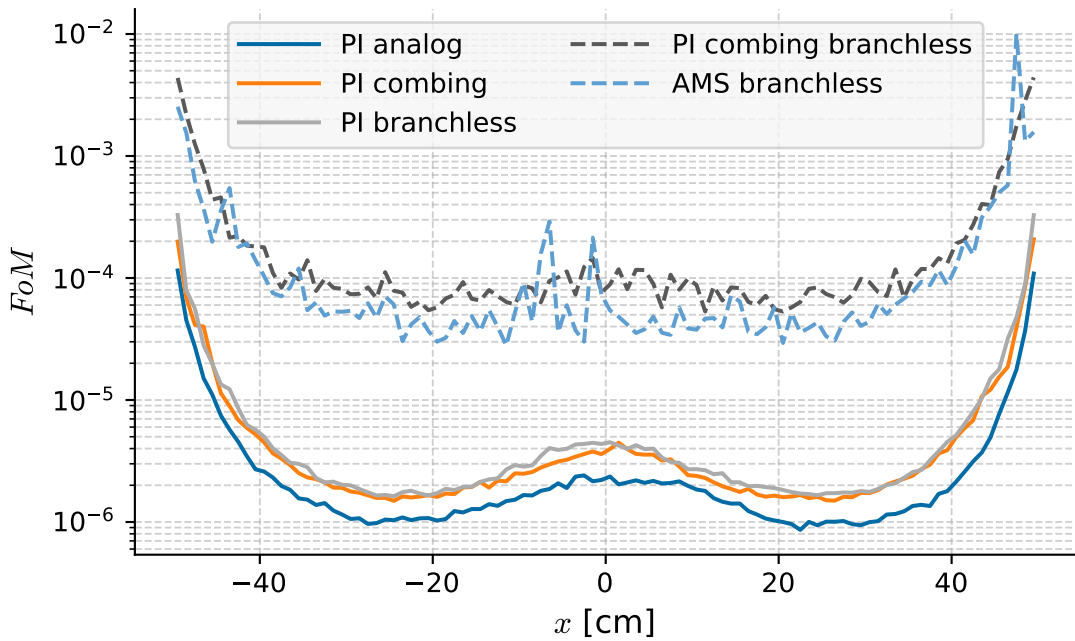


Figure 5.17: Figure of Merit for the flux estimation over x for the 1D bare slab reactor.

Table 5.4 also lists detailed computation times for all cases. It outlines that all methods show similar computation duration, with combing overall faster than the rest, and AMS slightly slower. The combing method allows to gain speed mainly because of the way source neutrons are sampled, which is more efficient than sampling with replacement which requires to browse across the particle buffer multiple times. Regarding the AMS, less time is spent in the transport algorithm since fewer collisions are sampled due to population fluctuations between generations (see Figure 5.7). However, the overall wall clock time is higher than for the other methods since it is necessary to sort tracks and add point into their structure (although the latter's impact is negligible). The ranking method used to find the K -th worst track (as regards to its importance level) has a complexity of $O(N)$ (for now the function `nth_element` from the C++ standard libraries is used) with N the number of tracks. But since the transport, sampling and scoring times are also roughly proportional to the number of particles, the ratio between the ranking time and the time spent in other functions should not differ much if the number of neutrons per cycle was to vary.

Table 5.4: Computation time [s] for each k -eigenvalue calculation on the homogeneous bare slab.

Case	Total	Transport	Sampling	Scoring	Sorting tracks	Adding points
PI analog	1.822×10^4	1.896×10^3 (10) %	7.954×10^3 (43) %	6.134×10^3 (33) %		
PI branchless	1.779×10^4	1.859×10^3 (10 %)	7.805×10^3 (43 %)	5.949×10^3 (33 %)		
PI combing	1.497×10^4	2.066×10^3 (13 %)	4.377×10^3 (29 %)	6.245×10^3 (41 %)		
PI combing branchless	1.392×10^4	1.896×10^3 (13 %)	3.990×10^3 (28 %)	5.885×10^3 (42 %)		
AMS branchless	2.094×10^4	1.660×10^3 (7 %)	4.166×10^1 (0 %)	1.199×10^4 (57 %)	3.638×10^1 (0 %)	3.867×10^3 (18 %)

To summarize, the AMS with branchless collisions allows to mitigate spatial and generational correlations compared to the population control methods generally used in power iteration based transport algorithms. By re-sampling new neutron histories (called *tracks* in the AMS framework), it pushes neutrons over generations without removing any particles (only physical events, like leakage or captures, or other methods, such as russian roulette, can remove particles from the system). In that sense, the AMS shows similarities with a Fleming-Viot particle system. A Fleming-Viot particle system can be used to study the distribution of a system conditioned to be alive at a specific time [123]. For neutron criticality calculations, it would consist in estimating the distribution of the neutron population at a given generation, assuming that the system is still alive (i.e., the number of neutrons is not zero). In practice, it requires to simulate the transport of neutrons as it would be in a free system (i.e., without population control), and re-sample a new particle each time one dies, by duplicating a particle that is alive at the exact time of death. Here, for a discrete time, this would coincide with an AMS simulation for the limit of $K = 1$.

5.3 Application to heterogeneous slab

It is time to challenge and further characterize the AMS method on a more complex study case. A level of complexity in space has thus been introduced by modeling a heterogeneous system with fissile and non-fissile media. Two differences arise compared to the homogeneous case:

- new particles are not re-sampled everywhere in space since only fission points are saved within the AMS in the case of k -eigenvalue calculations,
- the different material compositions will induce different branchless collisions coefficients, and thus a probable divergence regarding particle weights.

5.3.1 Description of the geometry, materials and simulation parameters

The geometry of the system, as well as the media macroscopic cross section are described in Figure 5.18, where Σ_t , Σ_s , Σ_c , Σ_f and ν are the total, scattering, radiative

capture, fission macroscopic cross sections and mean number of neutrons per fission respectively.

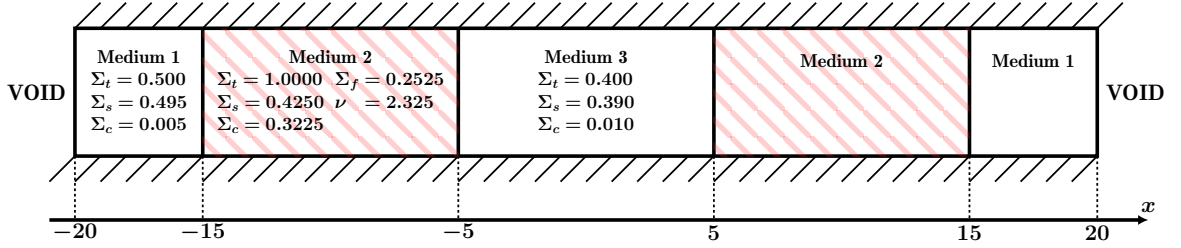


Figure 5.18: Heterogeneous one energy group slab geometry. The red dashed regions are fissile media.

This time, 10^4 neutrons per cycles (10^4 tracks for the AMS) were simulated to avoid clustering effects, and the number of independent calculations was decreased to 100 (the total number of generations is still 1000). The idea is to compare the methods performances regarding variance estimation when they all converge to the correct average value.

For the AMS method with branchless collisions, two calculations were done with two different importance functions in order to highlight improvements due to the importance function. For the one named "*AMS ϕ* ", the importance function was equal to

$$I_\phi = g + \phi_{D5} \quad (5.10)$$

where ϕ_{D5} is the spatial flux of the system computed with the deterministic code DRAGON5 using a S_n solver, and normalized to have values between 0 and 1. Since the problem is a one-speed problem, it is self adjoint in the time-independent diffusion theory [124]. ϕ_{D5} is therefore used as an approximation of the spatial adjoint flux for the current problem. The other one, named "*AMS $1 - \phi$* " uses the following importance

$$I_{1-\phi} = g + 1 - \phi_{D5}. \quad (5.11)$$

This time, the idea is to use a sub-optimal importance function, and to display the effects of the spatial shape of the importance function regarding the figure of merit by comparing the results obtained with I_ϕ and $I_{1-\phi}$.

5.3.2 Estimations of the fundamental mode

5.3.2.1 Estimation of k_{eff}

The k_{eff} distributions obtained with the different methods are plotted in Figure 5.19, and show that all methods lead to a main mode centered around the same value. While the AMS displays narrower distributions than the PI analog case around the main mode, these distributions have longer tails, meaning that more extreme values of the k_{eff} were sampled.

Surprisingly, the k_{eff} density obtained in the case AMS $1 - \phi$ also has two secondary modes. This phenomenon is due to strong oscillations of the k_{eff} estimates around the mean value as displayed in Figure 5.20. It emphasizes that a poor choice of importance function might induce higher variance in k_{eff} estimates, leading to a deterioration of the method performances.

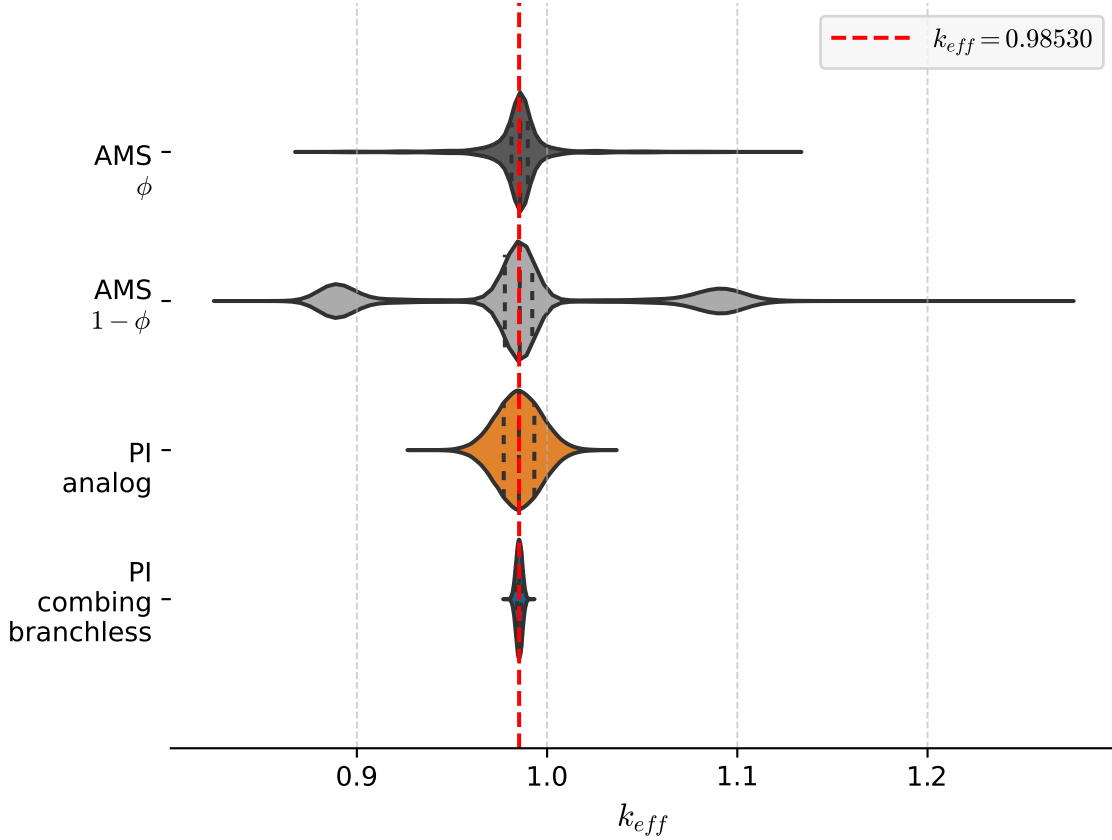


Figure 5.19: k_{eff} distribution for the heterogeneous one-speed slab problem.

5.3.2.2 Estimation of the fundamental flux distribution

As for the flux profile displayed in Figure 5.21, all methods converge to the exact same spatial distribution. Contrary to the performances observed previously, the AMS here strongly degrades the figure of merit in all spatial bins as seen in Figure 5.22, which features the spatial FoM for the flux estimates.

Since the system is heterogeneous, different multiplicative factors are defined depending on the collision position when applying the branchless collisions method. This can induce important discrepancies between particles weight, which can eventually lead to degradation of performances. The distribution of collision points and particle weights are thereby thoroughly analyzed in the following section.

5.3.3 Analyze of the particle weights over space and generations

Looking at collision point plotted in Figure 5.23, it is possible to discern the 5 slabs composing the system which are highlighted by the weights distribution over space. The spread of weights, as well as the overall window in which they are distributed, remain constant through generations because the population is re-normalized at each population control step (since the evolution of the total population weight is carried by the k_{eff} , each source neutron is sampled with a weight equal to one at each iteration).

On the other hand, the AMS does not stop neutrons at each generation, and they are followed by the algorithm until their death. This implies that weights vary not only

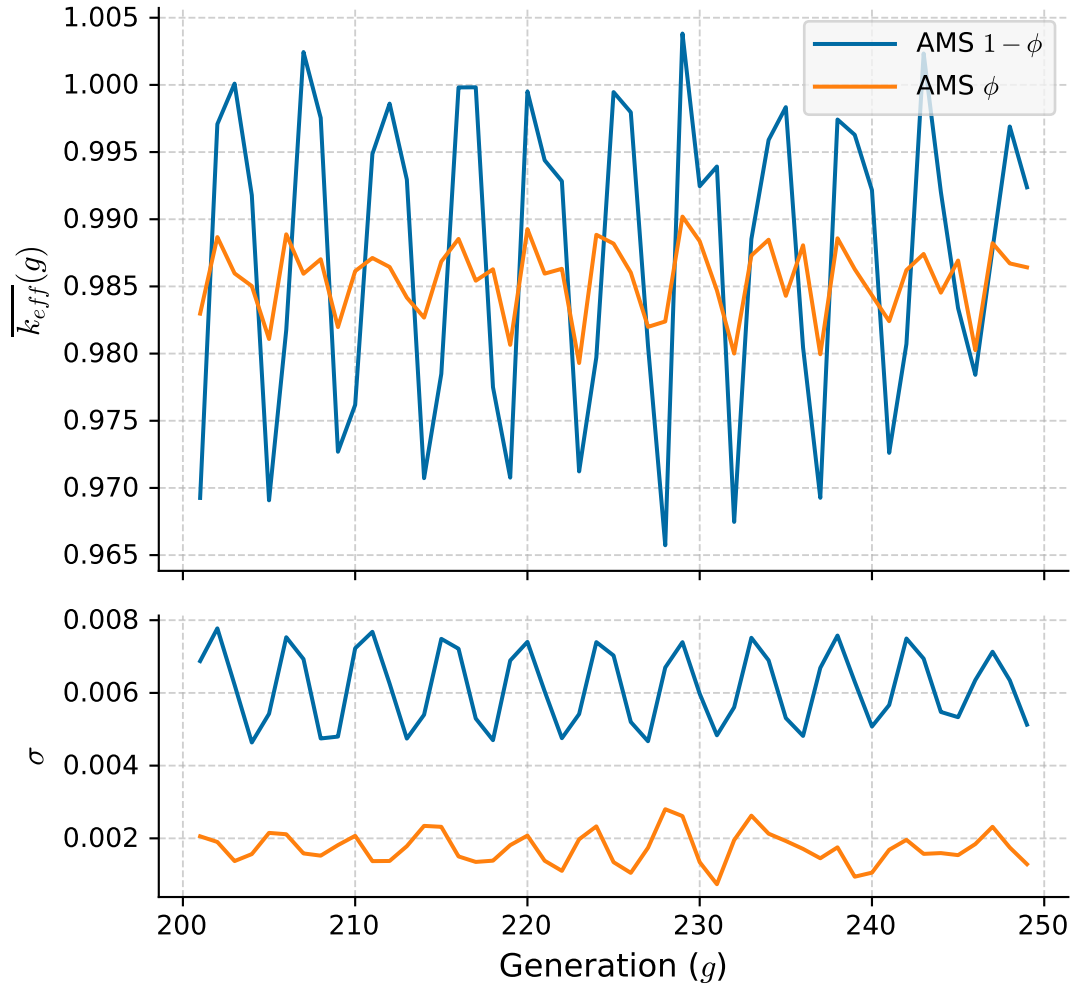


Figure 5.20: Average k_{eff} (computed over independent calculations) between generations 200 and 250 for the heterogeneous one-speed slab problem. Standard error of the mean is displayed on the bottom plot for readability issues.

on the scale of one generation, but over several successive generations. As a matter of fact, the spatial distribution of weights in the AMS cases⁴ is represented in Figure 5.24 and shows that not only weights increase along with the generation, but the spread of their distribution too.

By way of comparison, the same distributions were retrieved from the PI combing branchless and AMS branchless calculations of the homogeneous system presented above, displayed in Figures 5.25 and 5.26 respectively. In the homogeneous case, since the same multiplicative factor was applied to a particle weight, whether it collided in bin i or $j \neq i$, the weights are distributed uniformly in space, and do not spread much inside a generation, and this is also true for the AMS.

The evolution of the weight distribution is quantitatively illustrated in Figure 5.27 through four metrics highlighting its spread. The ratio $w_{\text{max}}/w_{\text{min}}$ plotted in the top left corner of the figure displays some kind of distance between extreme weight values in each generation, which increases over several orders of magnitude in the heterogeneous

⁴There is no noticeable differences between AMS ϕ and AMS $1 - \phi$ cases regarding the spatial distribution of weights.

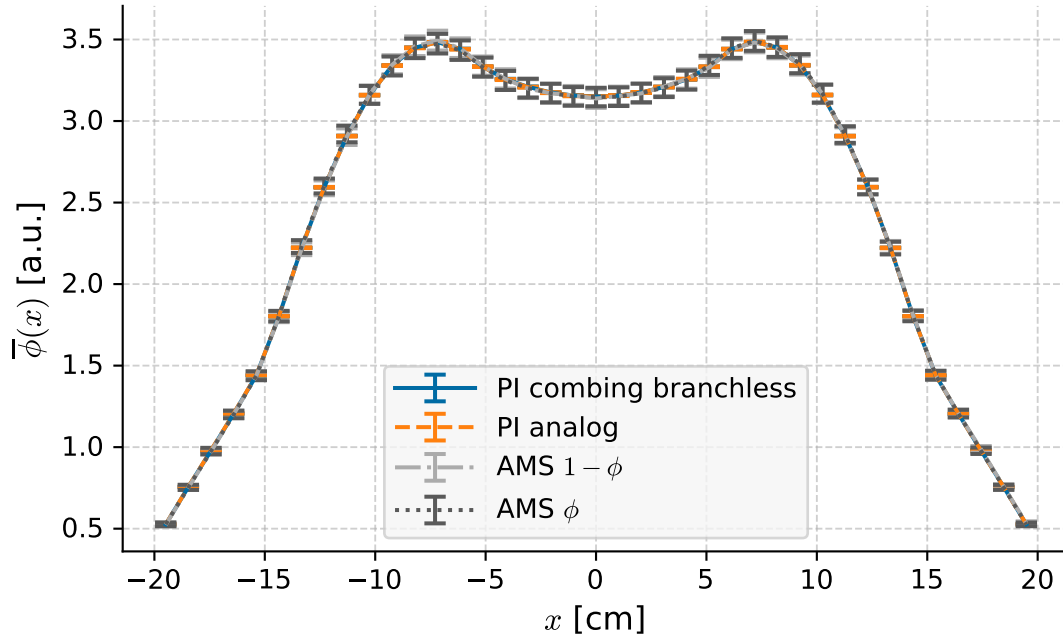


Figure 5.21: Flux distribution in the heterogeneous case.

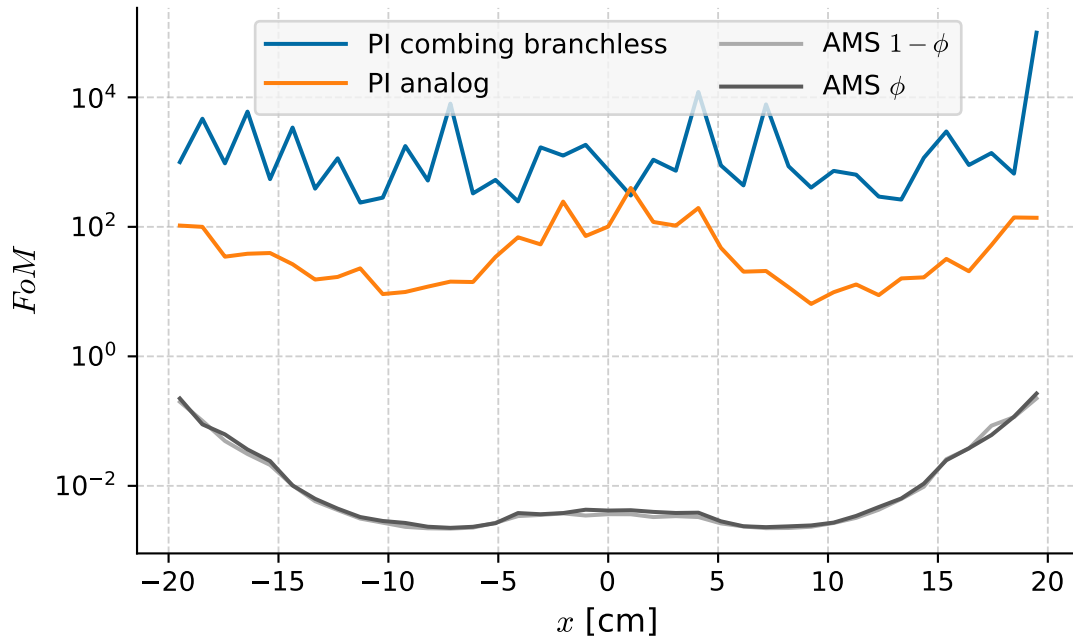


Figure 5.22: Flux FoM in the heterogeneous case.

AMS case while it remains much lower in the three other cases. Since this measure does not fairly represent the spread of the distribution by its own, the ratio of the mean value over the median of the distribution has been computed and plotted in the bottom left figure in Figure 5.27. It shows that the average particle weight diverges further from the median of the distribution as generations go by. Since the average is much more sensitive to extreme values than the median, this implies that a few particles with

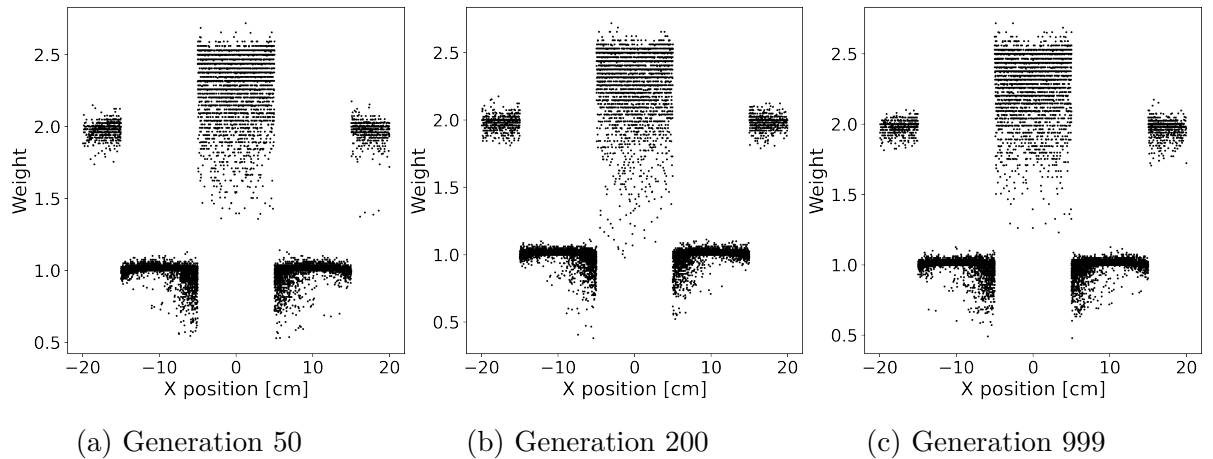


Figure 5.23: Weight VS position along x -axis for collision points in the case of the PI combing branchless heterogeneous case at generations 50 (a), 200 (b) and 999 (c).

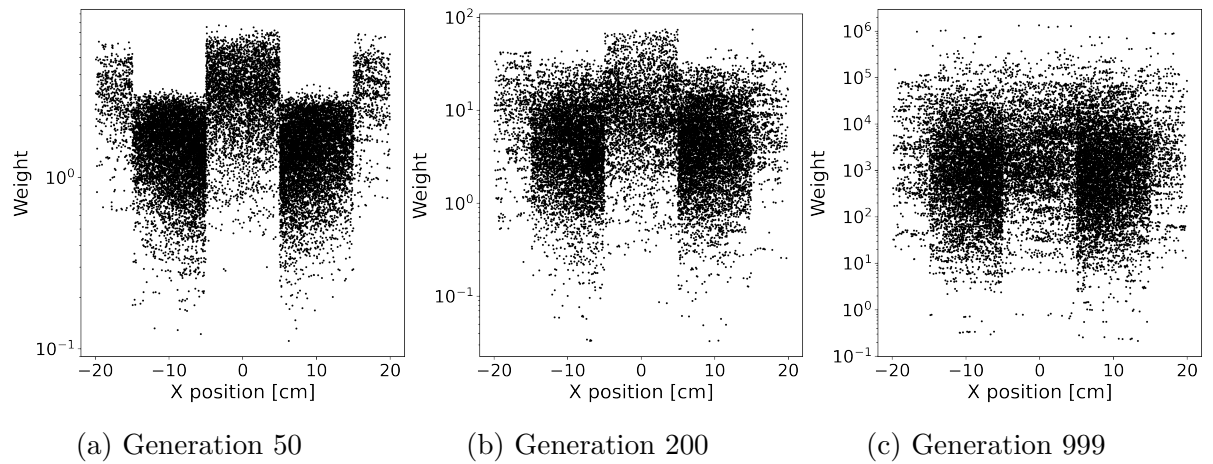


Figure 5.24: Weight VS position along x -axis for collision points in the case of the AMS branchless heterogeneous case at generations 50 (a), 200 (b) and 999 (c).

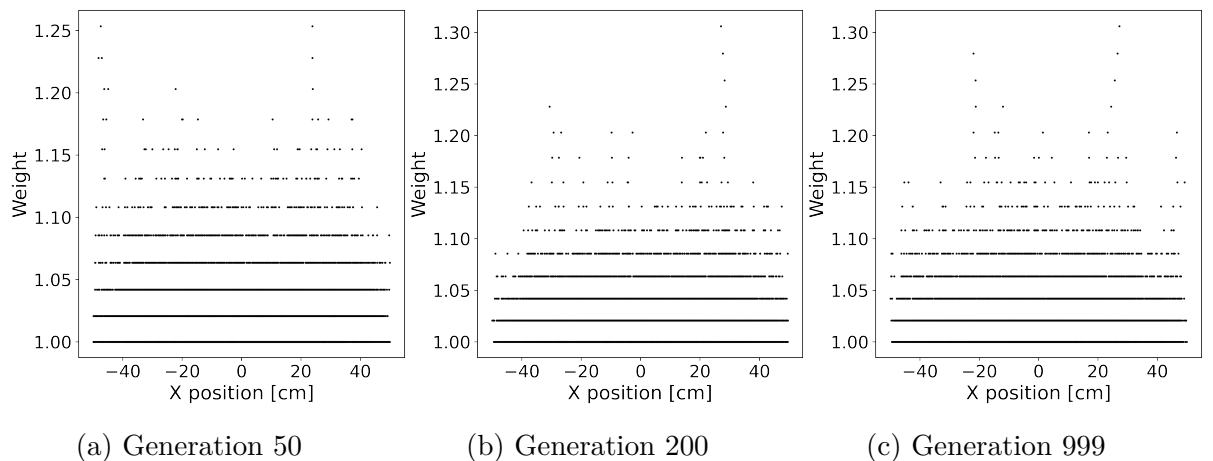


Figure 5.25: Weight VS position along x -axis for collision points in the case of the PI combing branchless homogeneous case at generations 50 (a), 200 (b) and 999 (c).

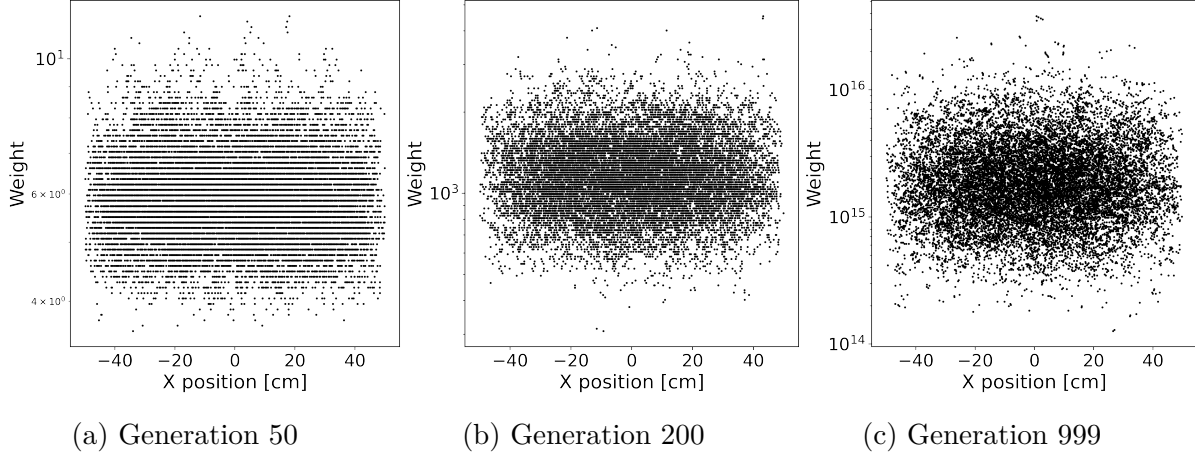


Figure 5.26: Weight VS position along x -axis for collision points in the case of the AMS branchless homogeneous case at generations 50 (a), 200 (b) and 999 (c).

high weight might induce variance jumps when they contribute to a tally.

The relative standard deviation as well as the relative IQR are also represented in the same figure in the top right and bottom right plots respectively, where the IQR is defined as

$$IQR = Q_{75} - Q_{25} \quad (5.12)$$

where Q_i is the i -th percentile of the distribution (which make Q_{25} , Q_{50} and Q_{75} the first quartile, the median and the third quartile respectively). These two plots illustrate that the particle weight distribution spreads from the mean and the median over generations in AMS cases, and that this spread is much larger in the heterogeneous case. It appears that weight distribution observed in power iteration cases remain constant regarding their extent, because the population control methods used here sample all new particles with the exact same weight so that the spreading does not propagate over generations.

From the above observations, it can be deduced that the poor figure of merit in the case AMS heterogeneous is merely due to a lot of low weight particles that are simulated but whose contribution to the score is negligible. It would be preferable to keep the weight of the particles within a window to limit the impact of this phenomenon, for example by using the Weight Window technique (cf. Section 2.2.4.1). In Korrigan current version, no mechanism has been implemented to constrain weights when using the branchless collisions. SERPENT2 code however systematically uses this technique after each branchless collisions. For this reason, further investigations regarding particle weight disparities were conducted using the SERPENT2 code. However, the AMS has only been implemented in SERPENT2 for kinetics calculations. For this specific reason, the results of this investigation are presented after the implementation of the AMS in kinetics, in Section 6.3.

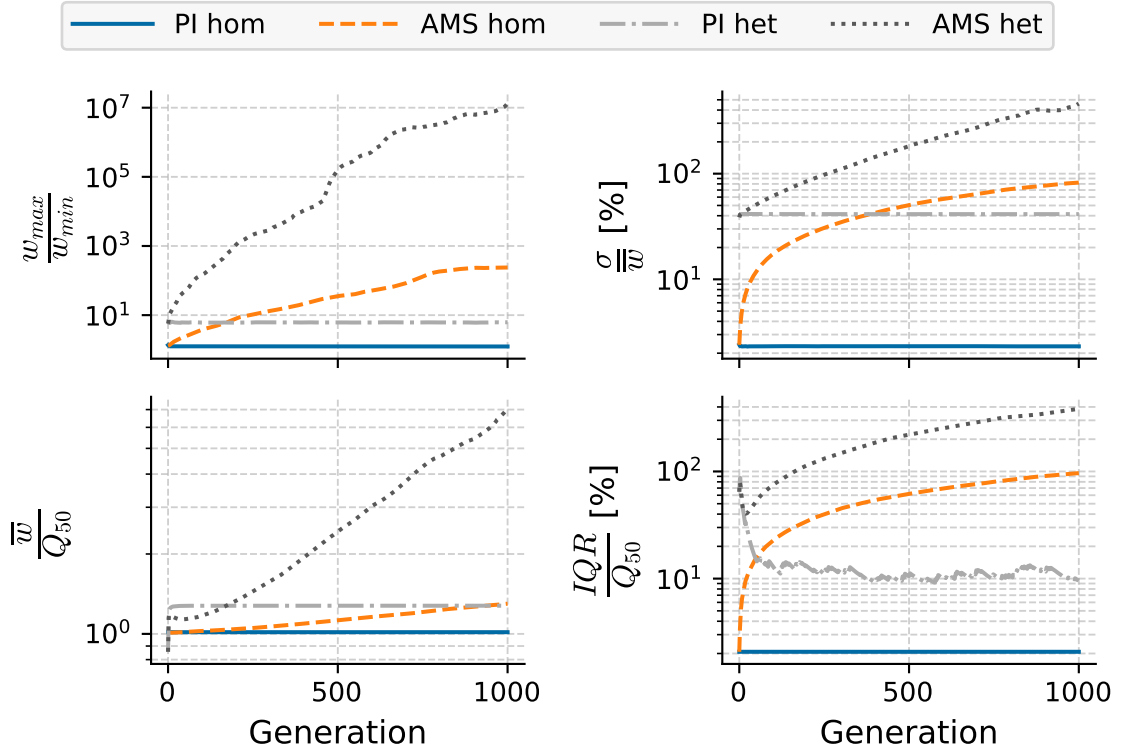


Figure 5.27: Statistical moments of the weights distribution over generations, smoothed using exponentially weighted moving average (original results are given in Figure A.1)

Conclusion

In conclusion, it has been shown that the Adaptive Multilevel Splitting (AMS) for particle transport could be extended to criticality calculations. The comparison between variations of the classical power iteration over a one-dimensional homogeneous reactor slab showed that the AMS combined with the branchless collision method induced increase in the FoM of the flux computation of a factor 100. This gain is of the same order of magnitude as the one obtained associating the power iteration with combing and branchless collision method. The computation times were overall similar, and the main gain has been achieved on the variance estimation, which is far less polluted by correlations in AMS branchless and PI combing branchless cases.

Regarding the estimation of the average value of the flux, the consideration of variance reduction was also aimed at reducing clustering phenomena. On that matter, both the AMS and the power iteration with combing (as long as the variance in fission chain lengths was reduced by the branchless collision method) showed great results with low or no clustering when few particles were simulated.

However, the simulation of heterogeneous slabs shed light on problems due to unconstrained particle weights, which resulted in a degraded figure of merit when the AMS was used. Keeping particle weights close to one another using a weight window technique might solve this problem. The AMS results might then show improved figures of merits. However, this method was not implemented in Korrigan due to time constraints. Regarding this issue, since the AMS was also implemented in SERPENT2 kinetics mode, and since the weight window technique is always combined to

the branchless collision in SERPENT2 kinetics calculations, further characterizations of the method were done on the one-speed heterogeneous slab case for an α -eigenvalue calculation in Section 6.3.

Besides reducing the spread of weight distribution, modeling more complex systems with a non-trivial adjoint solution should be necessary to further characterize this method behavior, especially for loosely coupled systems where neutrons would have difficulties reaching certain regions. The effects of the importance function would then be even more significant, and potentially sensitively better results could be achieved.

The idea of using the AMS for criticality was to study the state of a system conditioned to its survival to compute its asymptotic behavior (here characterized by the k_{eff} and the fundamental flux). While this approach requires an importance function to rank tracks and push neutron histories through time, it could be possible to get rid of this function by treating the system as a Fleming-Viot process, thus benefiting from the population control to regenerate particles without killing independent families.

As for reactor dynamics, the AMS has shown to be compatible with the computation of a spatial flux in a multiplicative system. By extending the importance function to the time domain, it could also be possible to perform a population control that better preserves independent neutron families, combined with variance reduction for a time-dependent detector (e.g., local tallies during a transient). In addition, branches in the AMS structure would also carry the type of the particle (neutron or delayed neutron precursor), hence making it possible to perform variance reduction with multiple importance functions depending on the particles nature. Nonetheless, particular attention must be paid to population growth in kinetics to limit the CPU and memory use of the AMS.

To sum up, it is possible to compute the fundamental mode of multiplicative system with the AMS, here for a supercritical configuration. Deriving scores such as the power distribution is thereby possible, whatever the k_{eff} of the system. In the next chapter, the method is extended to time-dependent problems, for which the score of interest may be defined over both space and time.

Chapter 6

Application of the AMS to kinetics calculations

Hic sunt dracones.

Contents

6.1	Kinetics calculations with SERPENT2	100
6.1.1	Time structure	101
6.1.2	Particles storage structure	102
6.1.3	Variance reduction and population control	104
6.1.3.1	Branchless collisions	104
6.1.3.2	Population control	105
6.1.3.3	Precursors creation in SERPENT2	107
6.2	Implementation of the AMS in SERPENT2	109
6.2.1	Handling delayed neutron precursors	109
6.2.2	Recasting the population control step	112
6.2.3	Distribution of particle weights over time	113
6.3	Computing the α fundamental mode of a one-speed 1D heteroge- neous slab reactor	118
6.3.1	Modeling of a dynamic α mode with SERPENT2 in one- speed theory	118
6.3.2	Test case	119
6.3.3	Results	121
	<i>Conclusion</i>	127

As in criticality, population control used in kinetics leads to the loss of independent lineages in the process. Besides, poor Figures of Merit regarding spatially localized

scores during a transient might benefit from a method to reduce variance in specific detectors. These two reasons seem to provide sufficient incentive to apply the AMS to kinetic calculations. Since kinetic calculations differ greatly from criticality calculations with respect to the treatment of particles over time (especially regarding precursors), the implementation of the AMS also increases in complexity compared to the version presented in the previous chapter. The aim of this chapter is to present kinetics calculations in SERPENT2 with and without using the Adaptive Multilevel Splitting. First, we present SERPENT2 specificities regarding kinetics calculations, and special features of interest regarding the AMS implementation are highlighted. The choices made for the AMS implementation are then presented and justified with simple case examples. Finally, the AMS algorithm is tested on two cases and the results are compared to reference calculations carried out with SERPENT2 official version 2.1.32.

6.1 Kinetics calculations with SERPENT2

SERPENT2 [118] is a Monte Carlo neutron transport code developed since 2004, initially intended for multigroup cross-sections generation in reactor physics [125, 126, 127]. It has been since used for a broad number of applications, including radiation protection calculation [128] and burn-up calculations [129]. SERPENT2 is among the first Monte Carlo codes that were extended to kinetics calculations in early 2010's [16] (along with TRIPOLI4 [117, 17, 100, 130] and MCNP [116, 130]).

Overall, kinetics calculation performed by the SERPENT2 code are based on methods presented earlier in Chapters 2 and 3. Unlike TRIPOLI4, whose initialization strategy relies on Faucher's procedure described in Section 3.2.2, all SERPENT2 independent kinetics calculation constituting a simulation are initialized from the same external source file. This source file is built from a k -eigenvalue calculation performed beforehand. When running a criticality calculation, live neutrons position, energy and direction can be recorded during active cycles to serve as initial source for a kinetics calculation. To obtain a neutron source as close as possible to a physical distribution, neutrons are prone to be saved at each collision according to a probability proportional to [131]

$$P \propto \frac{1}{\Sigma_{tot}(E)v(E)} \quad (6.1)$$

where $\Sigma_{tot}(E)$ is the total macroscopic cross section at energy E , and $v(E)$ is the neutron speed at energy E . Unlike the neutron source in a criticality calculation, which is comprised of fission neutrons only, the source in a kinetics calculation is comparable to a picture of the steady-state system at a given time, and includes neutrons at different stage of their life due to the slowing-down process. Once the critical neutron and precursor distributions have been saved as points in specific files (one for each type of particle), they are used to sample particles initially present in the subsequent kinetics simulation. Since only one set of points for each type of particle is used to sample source particles for multiple independent kinetics simulations, it is necessary to sample enough source points from the criticality calculation. If not enough source points are saved in these external files, supposed "independent" kinetics calculations might end up highly correlated due to similar initial conditions.

Once source files are computed, a transient can be simulated based on the following scheme:

1. source neutrons are sampled for the current time interval. At $t = 0$ it is based on the source files in which the critical source has been saved, whereas live neutrons are sampled from neutrons that survived the previous time interval and precursors are sampled to emit delayed neutrons for the rest of the transient,
2. source neutrons and their progeny arising in the current time interval are transported until death or until they reach the end of the interval,
3. some population control techniques may be applied to the neutron and precursor populations before the next time interval (i.e., between two transport steps),
4. if there are still time intervals before the end of the transient, a new time interval is simulated starting from point 1.

SERPENT2 specificities relevant to the implementation of the AMS in kinetics are described in the next subsections. There are three major aspects that deserve to be further detailed as the implementation of the AMS presented later required to modify part of already existing code. The time structures used to operate population control and score tallies, the storage structure which is used to differentiate particles of different types and in different states (e.g., to be transported or awaiting population control). Finally, SERPENT2 specificities regarding methods affecting particles populations such as population control, the branchless collisions technique and the precursors management will be presented.

6.1.1 Time structure

Unlike criticality calculations, particles are tracked continuously across time in kinetics calculations. A division of time into several intervals, also called time bins, is however necessary to apply some of the population control or variance reduction methods. Two types of time intervals are set. The first one is for the population control process, which will determine how the transport loop will be handled in time, whereas the second one is for the scoring of detectors tallies. While there can only be one time structure for transport steps, here called *simulation time intervals*, multiple binnings can be set for multiple detectors outputs. The simulation time intervals are a succession of time bins defined by a lower bound noted t_{BOI} (for time at the Beginning Of time Interval) and an upper bound denoted t_{EOI} (where *EOI* stands for End Of time Interval). An independent simulation of the transient is called a *batch*, and corresponds to the simulation of a neutron population for the entire transient. Random walks are performed for all particles in each time interval, one after another until all of them have been simulated. When particles are born during transport, they either undergo transport during the current time bin or are stored for the next time bin depending if they are emitted in this time interval or the next. Once all time intervals have been modeled, tallies are retrieved and the next independent batch is run. With regard to the parallelization of the simulation, it is important to note that the independent batches are run sequentially. Parallelization is indeed implemented at the level of a time step, during which the particles to be simulated are distributed between the different threads. Every time a particle crosses a time boundary (i.e., when a live particle reaches t_{EOI} without being terminated during the transport loop), it is stopped and stored to be used as a source particle in the next time bin. Depending on the particle state in the simulation (alive

in the current time bin, dead or pending for the next time interval), it will be stored in different lists composing the particles storage structure of SERPENT2.

6.1.2 Particles storage structure

Different storage structures are used in SERPENT2 to organize particles depending on their state. Dividing particles into these different lists allows to put in place tests and perform actions on a whole set of particles. For example, once the *queue* is empty, it means that all particles have been transported in the current time bin, and the next interval can be prepared. The following paragraphs present how particles go from one structure to another and which actions might be performed regarding particles stored in each one of them. A summary of their interactions is represented in Figure 6.1.

Stack

SERPENT2 is written in C and therefore allocates the memory needed for the calculation in a static way. Before the simulation of any particle, a certain amount of memory is allocated, say for N particles. The value of N must be greater than the initial number of particles per batch that the user wishes to simulate¹, which will be noted N_0 . In practice, the user defines a factor by which N_0 is multiplied to define the size of N . To allocate memory space for the N particles, a pool of numerical particles is created and will act as reservoir for the simulation. It is called the *stack*, and all particles which are not currently alive in the simulation are stored in it. During initialization, it is filled with N dummy particles for which memory has been allocated. If the algorithm tries to retrieve a particle in an empty stack, the calculation is immediately stopped, throwing an error, this is why it is important to allocate enough memory at the beginning. Each time the algorithm requires a new particle, it is retrieved from the stack and its state (nature, position, direction, energy, time) is set, before it is put in the relevant list.

Initially, all particles are stored in the *stack*, they will be redistributed throughout the calculation in the different lists (*queue*, *source*, *bank* or *store*) depending on their type and position in time.

Queue

The queue regroups particles that are to be followed in the current time bin. It is filled at the beginning of a time bin with neutrons whose time parameter is in $[t_{BOI}; t_{EOI}]$, but also with new neutrons born at a time lower than t_{EOI} (e.g., prompt neutrons born from fission, but also delayed neutrons emitted by forced decay during the transport loop). When particles die, they are passed to the stack, whereas they go to the bank if they reach the end of the current time bin. An empty queue means that the current time interval has been fully simulated.

¹Since time-dependent calculations allow the number of particles in the system to fluctuate, it is necessary to be able to inject more particles than the initial number should the system prove to be locally supercritical. Yet, the SERPENT2 run would abruptly stop without executing subsequent batches if it were to need a particle for which memory had not been reserved, hence the extra memory.

Source

There are two distinct sources, the neutron source and the precursor source. The neutron source is initially filled with live neutrons at the beginning of a time interval and emptied in the queue just before the transport loop. The precursors source on the other hand, is not emptied between time bins. Before each time interval, the population control over precursors is performed in this structure. Precursor weights are then adjusted to take into account the forced decay over the time interval. Between time bins, these objects then represent sources of particles for the next time bin.

Bank

When a neutron is recovered from the queue, it is simulated in the transport loop until it reaches the end of the current interval, or disappear (e.g., by absorption, leakage or Russian Roulette). If the neutron reaches the end of the interval without being terminated, it is put inside the bank to be used as a potential source point at the beginning of the next time step. The population control operated on neutrons, which will be described later, is done inside the bank and determines which ones will be kept as source for the next time interval.

Store

Finally, the last list that may contain particles is the store. The store is only used in dynamic calculations, when SERPENT2 is coupled to another physics code. Like the bank, the store is used to hold particles (neutrons and/or precursors), but this time reaching the end of the final interval regarding the neutronics part of a simulation. Particles stored in the store will be used to start the next neutronics stage, once the multi-physics coupling step has been done. This structure is not used in kinetics calculations (purely neutronics based), except if the user would like to save the final state of the neutron population in an external file.

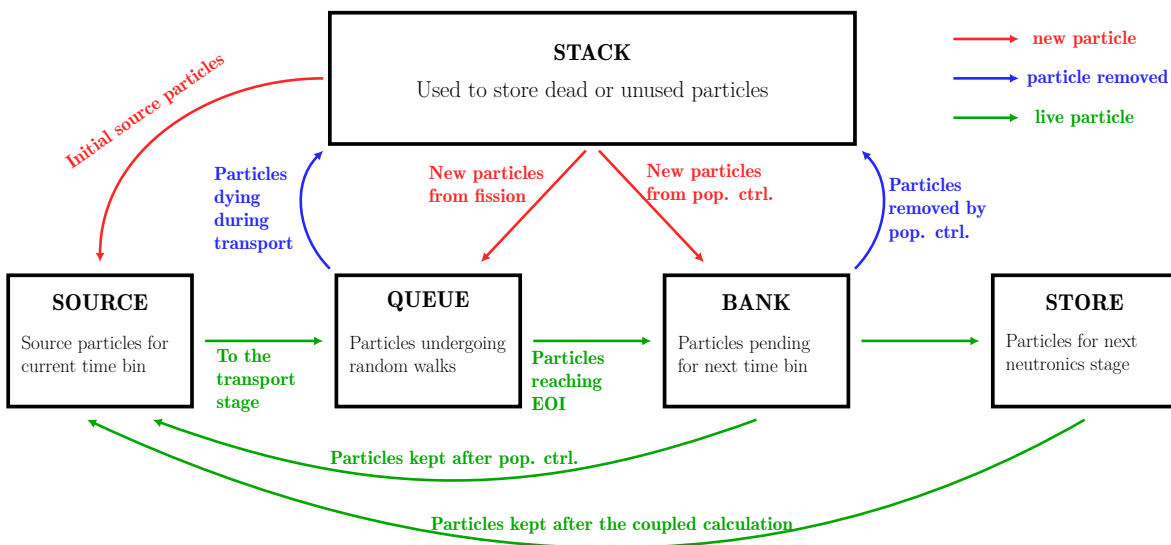


Figure 6.1: SERPENT2 particle storage structure

6.1.3 Variance reduction and population control

Conducting kinetics calculations can require the use of several techniques modifying the analog behavior of a real time-dependent system, as stated in Section 3.2. The SERPENT2 Monte Carlo code does not derogate from this rule and several population control and variance reduction methods can be used in its kinetics mode. Since the AMS might interact with already existing methods, as seen in criticality calculations presented in Chapter 5, it is important to depict a picture of SERPENT2 relevant methods for kinetics calculations. Albeit them having been introduced in Section 2.2.4, their specific implementation in SERPENT2 is described hereinafter.

As mentioned in Chapter 3, it is strongly advised to use the forced decay method to prevent scoring issues due to the different times scales between prompt and delayed neutrons. For that reason, the forced decay method (see Section 3.2.2.1) was implemented in SERPENT2 to handle precursors decay. It is even the only option available at the moment, since analog decay does not seem relevant for reactor physics applications in which real stochastic fluctuations regarding precursors decay are not of interest.

Neutron-nucleus collisions are by default analog in the sense that the outcome of a collision is randomly determined, and a capture kills the particle while branching reactions such as (n, xn) and fission create new particles according to probability distributions. It is however possible to turn on implicit reactions including capture, (n, xn) reactions, fission and even leakage. The branchless collisions method described in Section 2.2.4 is also available in the official code release version 2.1.32. In the case of AMS kinetics calculations presented later, only branchless collisions have been used.

Finally, SERPENT2 is able to perform population control at regular time intervals in time-dependent calculations. To do so, it is necessary to define a time-binning as explained in Section 6.1.1. However, would the user prefer to simulate a transient without population control, they would only need to define a unique time interval for the whole transient.

Overall principles for each of these methods were described in Chapter 2, but relevant specificities regarding their implementation in SERPENT2 are given hereinafter.

6.1.3.1 Branchless collisions

In SERPENT2, the branchless collisions globally follows the description presented in Section 2.2.4, and is always associated with the weight window to prevent the particle weights from scattering too much, which could lead to issues such as those presented for the heterogeneous slabs case in Section 5.3. The weight window method is applied after each collision. Its thresholds are based on the average neutron weight \bar{w} which is computed at the beginning of a time bin, with a lower bound defined as

$$w_{low} = WW_{min} \times \bar{w}, \quad (6.2)$$

and an upper bound equal to

$$w_{upp} = WW_{max} \times \bar{w}, \quad (6.3)$$

where WW_{min} and WW_{max} are set by the user. By default, $WW_{min} = 0.2$ and $WW_{max} = 10$. Changing the value of these two parameters should affect performances of kinetics calculations, but no further study has been conducted in the context of this

thesis. Since the \bar{w} is computed at the beginning of each time step as shown in Figure 6.2, this implies that the thresholds change and follow the evolution of $\bar{w}(t)$.

For information, it is also possible to configure the branchless collisions so that the number of neutrons born from collisions is superior to one. This option was not used in the present work but can be set by the user in the input data.

6.1.3.2 Population control

The population control step is done between two time bins, once all neutrons have been simulated in a time-bin, and reached t_{EOI} or have been terminated during the random walk. The whole process can be decomposed as three main steps, the precursor population control, the live neutrons (i.e., coming from previous time interval) population control and the emission of delayed neutrons. The scheme displayed in Figure 6.2 describes the whole strategy, which will be explained in the following section.

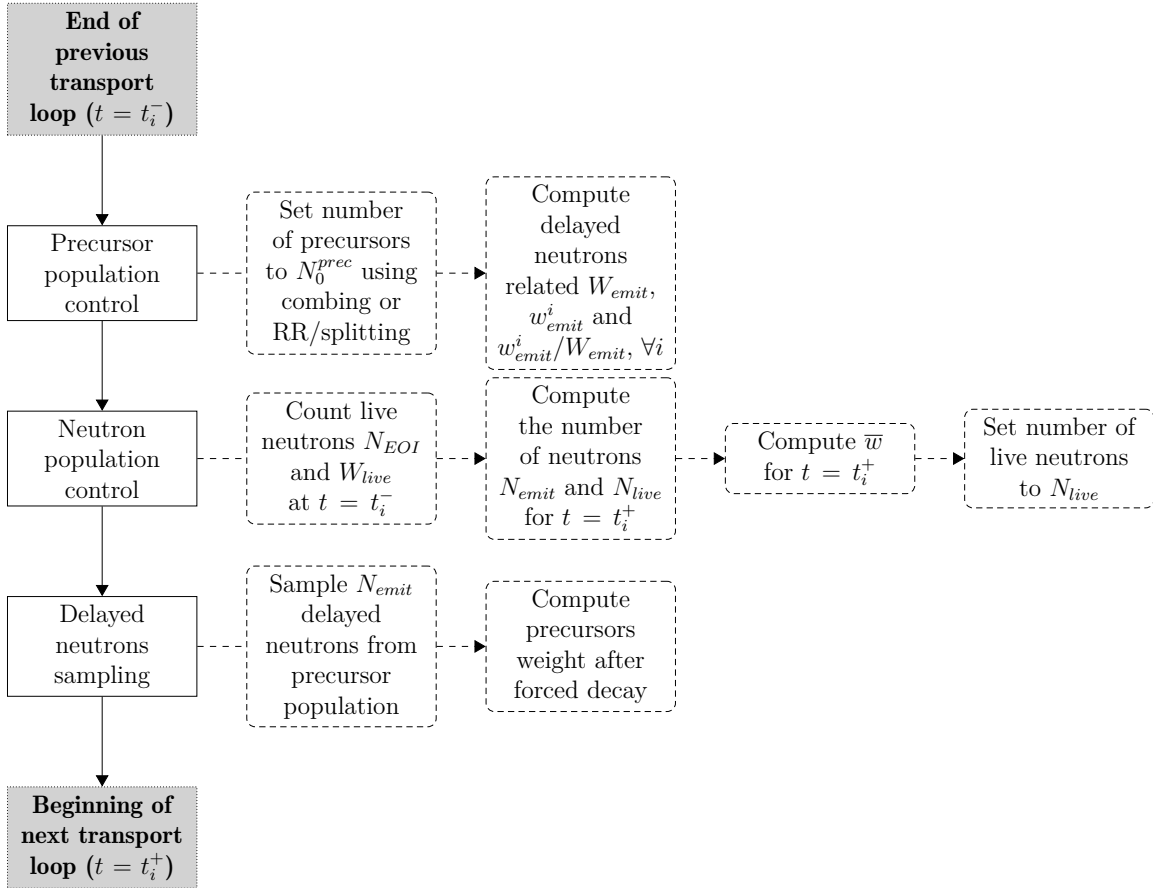


Figure 6.2: Population control sequence in SERPENT2

First, the precursor population is either combed or undergoes Russian Roulette (RR) / splitting so the number of precursors remains equal to its initial number here noted N_{prec}^0 . During this step, the expected weight of delayed neutrons to be emitted by forced decay in the next time bin is computed. The weight which should be emitted by delayed neutron precursor i is noted w_{emit}^i and is computed following Equation 3.20 which equivalent is reminded here

$$w_{emit}^i(t) = w_p \Delta t \lambda_i e^{-\lambda_i(\Delta t)} \quad (6.4)$$

where the expected delayed neutron weight is $w_{emit}^i(t)$, w_p is the weight of the precursor at the beginning of the time step of length Δt and with decay constant λ_i . The total weight to be emitted is then computed as such

$$W_{emit} = \sum_{i=1}^{N_{prec}^0} w_{emit}^i. \quad (6.5)$$

In the meantime, the contribution of each precursor particle to the total weight to be emitted, w_{emit}^i/W_{emit} , is computed. This value is used later to sample which precursors will be selected to emit a delayed neutron as detailed in the remainder of this section.

Then, the live neutron population coming from the previous time interval undergoes population control too. Once all remaining neutrons have been gathered in the bank at the end of the previous time interval, the total number of neutrons still alive at the end of interval, N_{EOI} , along with the total weight of these neutrons, W_{live} , are evaluated. From this point, the neutron population is brought from N_{EOI} to N_{live} , either by combing or RR/splitting, so that

$$N_{live} = \left\lfloor N_0 \frac{W_{live}}{W_{live} + W_{emit}} + \frac{1}{2} \right\rfloor \quad (6.6)$$

where N_0 is the number of neutrons per batch set by the user (which corresponds to the initial number of neutrons sampled in a time bin before running the transport loop), and $\lfloor x + 1/2 \rfloor$ means the rounding-off of x to the closest integer.

The number of delayed neutrons emitted during the following interval is then computed. Since the forced decay method can introduce issues regarding neutron weights, as mentioned in Section 3.2.2.1, it is necessary to find a way so that delayed neutrons carry weights close to live neutron weights. The strategy adopted in SERPENT2 relies on fewer delayed neutrons than precursors to increase the weight of each delayed neutron. The number of delayed neutrons emitted during the following interval is noted N_{emit} , and is computed as such

$$N_{emit} = \left\lfloor N_0 \frac{W_{emit}}{W_{live} + W_{emit}} + \frac{1}{2} \right\rfloor. \quad (6.7)$$

This way, the number of neutrons (prompt and delayed) sampled in the next time bin is not necessarily equal to N_0 due to rounding-off values, but close to it

$$N_0 \approx N_{live} + N_{emit}. \quad (6.8)$$

During the same step, the average neutron weight is computed for the next interval to update weight window bounds defined by Equations 6.2 and 6.3

$$\bar{w} = \frac{W_{emit} + W_{live}}{N_0}. \quad (6.9)$$

This way, all neutrons (live and delayed) have the same weight at the beginning of the following time interval. During this step, the update of \bar{w} is always done, but is only relevant to the weight window when the branchless collisions method is used.

Finally, the N_{emit} delayed neutrons are emitted from N_{emit} precursors randomly sampled. Since there are more precursors that are stored than delayed neutrons to emit, only part of them are to be selected to produce a delayed neutron. The probability

to sample precursor i is equal to its relative contribution to W_{emit} , which is equal to w_{emit}^i/W_{emit} . Every delayed neutron resulting from this sampling process is created with a weight strictly equal to

$$w_{del} = \frac{W_{emit}}{N_{emit}}. \quad (6.10)$$

In the end, the initial number of neutrons in the following time interval is equal to $N_{live} + N_{emit}$, with N_{live} neutrons coming from the previous time interval (i.e., their time is equal to $t = t_{BOI}$), and N_{emit} delayed neutrons appearing between $t = t_{BOI}$ and $t = t_{EOI}$.

6.1.3.3 Precursors creation in SERPENT2

In a kinetics calculation, delayed neutron precursors either come from the initial steady-state source, or appear due to fissions occurring during the transient. Depending on the transient, the latter is more or less important. For example, in prompt supercritical transients, delayed neutron precursors created during the transient may not impact much the power peak compared to prompt neutrons, but mainly affect the slow decay after the peak. Whereas their effect is much more significant in slower transients in which kinetics is mainly driven by their decay. In SERPENT2, delayed neutron precursors born during the transient are created implicitly. It means that exactly one precursor is created at each collision in fissile media to allow to have more source points, and thus, increased statistics. To ensure unbiased results, the weight of such a precursor is modified according to the expected number of precursors born by fission. The weight of the newly created precursor is thus equal to

$$w_{prec}(t_{col}) = w_{col}\nu_d \frac{\Sigma_f}{\Sigma_t} \quad (6.11)$$

where w_{col} is the weight of the colliding neutron, ν_d is the average number of delayed neutrons emitted by fission, Σ_f and Σ_t are the fission and total macroscopic cross sections of the medium in which the collision happened. This weight is defined at the time of occurrence of the collision, $t = t_{col}$. Because of the forced decay (see Section 3.2.2.1), a portion of this precursor takes the form of a delayed neutron emitted in the current time interval, between t_{col} and t_{EOI} . The weight and time of the delayed neutron emission are computed following forced decay rules, which make the weight of the delayed neutron equal to

$$w_{del} = w_{col}\nu_d \frac{\Sigma_f}{\Sigma_t} (1 - e^{-\lambda\Delta t_1}) \quad (6.12)$$

where $\Delta t_1 = t_{EOI} - t_{col}$ is the time interval between the collision and the end of the current time bin. The delayed neutron immediately undergoes Russian Roulette with a threshold equal to the incoming neutron weight. The remaining part of the precursor particle is to be stored to be part of the delayed neutron source for the following time bins.

Creating and storing one precursor at each collision (in a fissile medium) might end up in many precursor particles created during a time bin. Besides, SERPENT2 stores weighted precursors due to multiple reasons such as population control and variance reduction methods. Yet, storing too many precursors with very different weights would be inefficient regarding the figure of merit of a calculation. For this reason, Russian

Roulette is systematically applied to precursors created during transport, which are discarded with a probability $1 - P_{store}$. The adjustment of P_{store} is not based on the weights of the precursors themselves, but on the weight of the delayed neutrons emitted by forced decay. In a nutshell, the lower the weight of an emitted delayed neutron compared to those emitted by other precursors, the less likely the precursor is to be retained. In the current implementation of this process, the comparison is done for a *fictive delayed neutron emitted in the current time interval*. P_{store} is thus equal to

$$P_{store} = \frac{\left[w_{col} \nu_d \frac{\Sigma_f}{\Sigma_t} e^{-\lambda \Delta t_1} \right] \left(1 - e^{\lambda \Delta t_2} \right)}{w_{thresh} \frac{W_{emit}}{N_{prec}^0}} \quad (6.13)$$

where

- λ is the precursor decay constant,
- $\Delta t_2 = t_{EOI} - t_{BOI}$ is the length of the current time interval,
- W_{emit} is the total delayed neutron weight emitted at the beginning of the interval,
- N_{prec}^0 is the number of precursors at the beginning of the current time interval,
- w_{thresh} is a multiplicative constant set by the user.

Let us detail physical implications of Equation 6.13. The remaining weight of the precursor particle once a delayed neutron has been emitted following Equation 6.12 is equal to

$$w_{prec}(t_{EOI}) = w_{col} \nu_d \frac{\Sigma_f}{\Sigma_t} e^{-\lambda \Delta t_1}. \quad (6.14)$$

$w_{prec}(t_{EOI}) \times (1 - e^{\lambda \Delta t_2})$ is then the weight of the fictive delayed neutron mentioned above. Hence, the numerator of the r.h.s. in Equation 6.13 is equal to $w_{prec}(t_{EOI}) \times (1 - e^{\lambda \Delta t_2})$, which is the expected weight a fictive delayed neutron coming from forced decay of that precursor between t_{BOI} and t_{EOI} . It is compared to W_{emit}/N_{prec}^0 , which is the average delayed neutron weight emitted by a precursor initially present at the beginning of the current interval. If the remaining part of the precursor would lead to a delayed neutron with a small weight (compared to W_{emit}/N_{prec}^0 for the current time interval), the new precursor is not likely to be stored. The user can adjust this probability through w_{thresh} to store more or less precursors from collisions (by default it is equal to 1). In the case where $P_{store} > 1$, the precursor is stored with probability 1 and its weight is increased. This process is described by Figure 6.3. If stored, the precursors weight is set equal to

$$w_{prec}^{store} = \frac{w_{col} \nu_d \frac{\Sigma_f}{\Sigma_t} e^{-\lambda \Delta t_1}}{P_{store}} \quad (6.15)$$

so that the precursor balance is kept unbiased.

Compared to criticality calculations, kinetics calculations have a more complex structure because of the more numerous requirements induced by different particle types and because of population fluctuations over time. Implementing the AMS algorithm in the already existing structure is not straightforward as multiple strategies may be possible. In the next section, the choices of implementation made, as well as the observations that led to these choices, are presented.

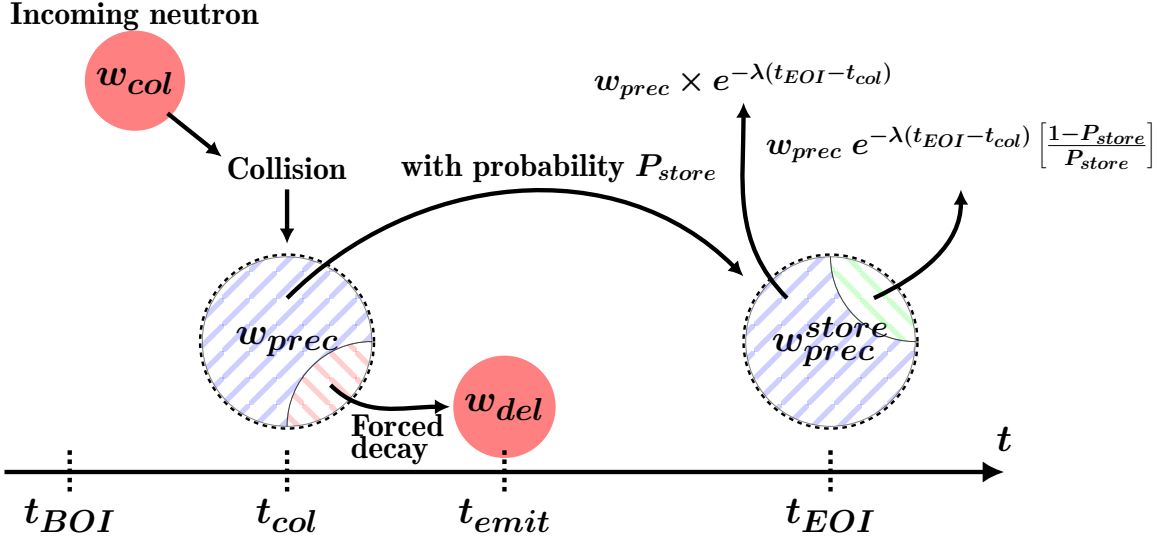


Figure 6.3: Creation of a delayed neutron precursor in SERPENT2. A precursor is implicitly created at each collision and immediately undergoes forced decay over the remaining of the current time interval. Russian roulette is then applied to the weight that has not decayed.

6.2 Implementation of the AMS in SERPENT2

Ideally, implementing the AMS algorithm should be as non-intrusive as possible to avoid too many modifications of SERPENT2's source code. However, due to the interactions between biasing methods in kinetics calculations and limited knowledge of the code, plugging the AMS iterative algorithm into SERPENT2's transport loop without modifying it turned out to be anything but trivial. The following choices may not be optimal at this time, but they allowed to establish a proof of concept regarding the use of the AMS in kinetics. Three major points drew our particular focus

- managing delayed neutron precursors,
- recasting the population control over time intervals,
- preventing particle weight distribution to spread too much inside a time bin.

The choices of implementation regarding these three issues are presented below.

6.2.1 Handling delayed neutron precursors

Since there are two types of particles in kinetics calculation, it would be beneficial to be able to differentiate precursors from neutrons regarding the importance function to further optimize the variance reduction. Yet, the AMS includes the possibility to handle multiple types of particles such as precursors and neutrons with different importance functions. The nature of the particle can also be considered as an additional parameter of the same importance function. It is expected that taking an importance function with more detailed physics, thus getting closer to the real adjoint function (considered to be the best option), will improve performances of the variance reduction method. As mentioned earlier in the conclusion of Chapter 5, delayed neutron precursors could be considered as branches in the AMS framework. Ideally, this should be the case so

that an importance function specific to these particles could be used to discriminate between live neutrons and precursors.

There are two possible options, integrating precursors into AMS tracks as branches, or not taking them into account in the AMS framework. In the second scenario, delayed neutrons would still be emitted following kinetics calculations settings, but no point corresponding to a delayed neutron precursor would be added into an AMS batch. This second option would hinder using precursors importance in the AMS, which could prevent it from reaching its full potential, but is the simplest one to set up.

Pursue the first scenario would imply having a high number of tracks. Indeed, since tracks are in the first place defined from the population present at the beginning of the simulation, this would mean creating as many tracks as there are live neutrons and precursors initially in the system. In view of this observation, it is possible to evaluate the proportion of precursors and neutrons at $t = 0$. For the sake of simplicity, and since the majority of kinetic calculations are rather intended for the modeling of accidental transients, the reactor is considered initially critical. Referring to Equation 1.11 describing the evolution of the concentration of the precursors over time, and considering the system at equilibrium, the concentration of precursor in family i is equal to

$$\lambda_i C_i = \beta_i \Sigma_f \phi \quad (6.16)$$

$$C_i = \frac{\beta_i n}{\lambda_i \Lambda} \quad (6.17)$$

where n is the neutron density and $\Lambda = \frac{1}{\nu \Sigma_f}$ is the mean generation time. The total concentration of precursors is therefore equal to

$$C = \frac{n}{\Lambda} \sum_i \frac{\beta_i}{\lambda_i}. \quad (6.18)$$

Considering the eight precursors families whose constants β_i and λ_i appear in Table 6.1, in a system with $\Lambda = 7 \times 10^{-5}$ s the ratio between precursors and live neutrons is equal to

$$\frac{C}{n} \approx 10^3. \quad (6.19)$$

From the user's perspective, this ratio implies that for N_0 neutrons requested at $t = 0$, the number of tracks is equal to $1000N_0$. In SERPENT2 however, the number of precursors that are stored in memory is not equal to the physical number of precursors that should be present. The initial number of precursor particles stored in memory is equal to

$$N_{\text{prec}}^0 = \text{precsrcf} \times N_0 \quad (6.20)$$

where *precsrcf* is set by the user (default is 10). Setting a low value for *precsrcf* could result in an undersampling of precursors in some regions of the geometry. Thus, for N_0 neutrons at the beginning of the simulation, the following number of AMS tracks would be created

$$N_{\text{tracks}} = N_{\text{prec}}^0 + N_0^{\text{live}} = \text{precsrcf} \times N_0 + N_0^{\text{live}} \quad (6.21)$$

where N_0^{live} is the number of initial neutrons sampled from the steady-state distribution, equal to

$$N_0^{\text{live}} = N_0 - N_0^{\text{del}} \quad (6.22)$$

Table 6.1: Precursor group constants for ^{235}U ($\beta = 0.00585$) for a 1 eV incident neutron (JEFF3.1.1 data library [132]).

Family i	λ_i [s $^{-1}$]	β_i/β
1	0.0127	0.0340
2	0.0283	0.1501
3	0.0425	0.0992
4	0.1330	0.2001
5	0.2925	0.3122
6	0.6665	0.0932
7	1.6348	0.0872
8	3.5546	0.0240

where N_0^{del} is the number of delayed neutrons emitted in the first time bin at initialization (for more detail on how N_0^{live} and N_0^{emit} are computed, please refer to Section 6.1.3.2). Thus, the order of magnitude of N_{tracks} is about $\text{precsrcf} \times N_0$ (by default 10 times N_0). In the current AMS implementation in SERPENT2, the user does not define K directly, but the ratio $r = \frac{\text{number of re-sampled tracks}}{\text{total number of tracks}}$ (equivalent to K/N in AMS notations presented in Chapter 4). The stopping criterion regarding the minimum number of tracks reaching the detector² would then be equal to

$$N_{\text{tracks}} - K + 1 = N_{\text{tracks}}(1 - r) + 1 = (\text{precsrcf} \times N_0 + N_0^{\text{live}}) \times (1 - r) + 1. \quad (6.23)$$

Compared to the case with neutrons only, (i.e. $N_{\text{tracks}} - K + 1 = N_0(1 - r) + 1$), the ratio of tracks in the detector is equal to

$$\frac{(\text{precsrcf} \times N_0 + N_0^{\text{live}}) \times (1 - r) + 1}{N_0(1 - r) + 1} \approx \frac{(\text{precsrcf} \times N_0 + N_0^{\text{live}})(1 - r)}{N_0(1 - r)} \approx \text{precsrcf}. \quad (6.24)$$

For $\text{precsrcf} = 10$, 10 times more particles would be pushed into the detector. For a neutron detector (i.e., only neutrons may reach the maximum importance in the detector), $10N_0$ neutrons would reach the detector when N_0 initial neutrons would have been required by the user. Asking for $N_0/10$ initial neutrons in return would probably result in too few particles at the beginning of the transient (fewer neutrons also implies fewer precursors) and thus a poor variance (if not low particle number related problems such as clustering) in early stages of the transient. Of course, having too many particles is not a problem regarding Monte Carlo estimates, but it can lead to long calculation times and memory footprint issues. So as not to run the risk of having to deal with problems of computing power or unexpected excessive numerical fluctuations (e.g., such as clustering) in the first test calculations presented later, this option has been ruled out for the rest of this thesis. It would certainly be worthwhile to look at this issue in more detail in the future.

The second option is to consider only neutrons in the AMS framework. In this situation, the number of tracks would be equal to the initial of neutrons sampled at the beginning of the transient, i.e., the number of neutrons sampled from the steady-state distribution (N_0^{live}) plus the number of delayed neutron emitted at initialization

²As mentioned in Chapter 4, re-sampling K tracks until the K -th worst track (regarding the importance) have reached the maximum importance is equivalent to having at least $N - K + 1$ tracks inside the detector.

(N_0^{del}), that is N_0 . Since not all precursors emit a delayed neutron, delayed neutrons emitted in the following time bins from an initial precursor that has not decayed in the first time interval do not belong to an already existing track. A new track must therefore be created and added to the AMS batch. This raises the following issue: new tracks would appear throughout the simulation, yet the number of AMS tracks must be constant over time. To avoid such problems, modifications were made so that delayed neutrons are emitted over the whole transient through precursors forced decay at the beginning of an iteration, to ensure that no new track would appear during following time bins. Compared to a classic SERPENT2 kinetics simulation, one unique time bin is defined for decay instead of several. This way, the contribution of all initial precursors are taken into account for the whole transient, and no new track appears over the course of the simulation. Since delayed neutrons that appear from fission occurring during the simulation come from an already existing track, there is no issue regarding them, a new branch is just appended to the track. This is the option that was adopted for the calculations presented later in this chapter.

6.2.2 Recasting the population control step

As seen earlier, the AMS is capable of partial³ population control in case of subcritical systems. It is also possible to model any system whatever its reactivity as a numerically subcritical system⁴ by using methods such as the branchless collision, where neutrons are still lost upon leakage, making the system numerically subcritical. In that context, the AMS was also used as some sort of population control thanks to its particle resampling mechanism, in place of current methods such as RR/splitting and combing.

When applying AMS to kinetics simulation, thanks to its inherent partial population control, the SERPENT2 native population control methods described in figure 6 could be eventually left out. In-between time intervals stage described in Figure 6.2 could have been skipped, making the definition of multiple unnecessary time intervals (except for scoring, but that has no impact on the simulation conduct whatsoever since scoring bins are defined over a different time mesh). Yet, particle weights average value is the chief ingredient in the Weight Window technique which is applied after each branchless collision, and this value is updated between each time step in a classical SERPENT2 run. In order to preserve means of updating the Weight Window over time, the binning structure was modified in regards to Figure 6.2 to define multiple time intervals without applying population control during the simulation. Between each time interval, the average particle weight is computed from particles alive before the next random walk, which includes live neutrons from the previous interval, and delayed neutrons which were sampled at the beginning of the next interval. The weights have also to be updated for each AMS iteration. To improve statistics, this average value at iteration i is computed by taking tracks which were sampled in previous AMS iterations into account, such that

$$\bar{w}_i(t_{BOI}) = \frac{\sum_{j=1}^i [W_{live}^j + W_{emit}^j]}{\sum_{j=1}^i [N_{live}^j + N_{emit}^j]}. \quad (6.25)$$

³By partial we mean here that it cannot strictly control the population since it is only capable of injecting around K new particles.

⁴Where *numerically subcritical* means that the number of numerical particles (the particles actually modeled in the simulation, not to be mistaken with the physical particles they represent) decreases with time.

The different terms composing this equation are

- $\bar{w}_i(t_{BOI})$ is the average particle weight for time interval $[t_{BOI}; t_{EOI}]$ during AMS iteration i ,
- N_{live}^j and N_{emit}^j are the number of live neutrons and delayed neutrons respectively present in interval $[t_{BOI}; t_{EOI}]$ at iteration j , before the transport loop,
- W_{live}^j and W_{emit}^j are the total live and delayed neutrons weights computed for interval $[t_{BOI}; t_{EOI}]$ at iteration j .

As AMS iterations go by, the total number of particles sampled in bin $[t_{BOI}; t_{EOI}]$ from the beginning of the simulation increases (as the AMS re-samples extra particles), leading to a better estimation of $\bar{w}_i(t_{BOI})$. If the average weight were to be computed only from N_{live}^i and N_{emit}^i (i.e., the particles simulated only in the current AMS iteration), poor estimates of $\bar{w}_i(t_{BOI})$ should be expected due to the relatively low number of particles re-sampled in one AMS iteration.

Consequently, the multiple time-bin structure was retained to allow the neutron average weight to be updated in each time-step, thus allowing to follow its variations over time. It has however been modified to switch off the population control of SERPENT2. The average weight is exclusively used by the Weight Window method to constrain particle weight in a range, and following the average weight variations over time hence allows for a finer use of the Weight Window technique as described in the following section.

6.2.3 Distribution of particle weights over time

In the previous chapter (cf. Section 5.3), light was shed on performance problems due to high variance between particle weights in the heterogeneous slab system simulated as a criticality problem. Although the performed calculation was encompassed in the one-speed theory, this issue is also met in continuous energy problems, even in spatially homogeneous systems. Indeed, the branchless collision coefficient (see Equation 2.19) depends on materials cross sections which in turn depend on the neutron energy.

As a reminder, it was observed that for a heterogeneous geometry, the successive multiplication of branchless collision coefficients would lead to a weight distribution spreading over time, resulting in poor performances as regards to the Figure of Merit. To fix this issue, it was assumed that using some sort of weight control method would allow for the weights variance not to skyrocket over time. To test the current AMS framework described in Sections 6.2.1 and 6.2.2, and illustrate the importance of the Weight Window technique, a homogeneous test case will be presented hereinafter.

Three different strategies were therefore tested

- no Weight Window (WW) was applied after branchless collisions, and duplicated tracks are uniformly sampled during the re-sampling step,
- the Weight Window technique is applied after each collision, and duplicated AMS tracks are sampled uniformly
- the Weight Window is applied plus new AMS tracks are sampled based on track weights as presented at the end of Section 4.1.2.

Table 6.2: SERPENT2 AMS simulation in the case of a homogeneous box for different Weight Window parameters.

Case	weight window	Sampling	Number of batches ⁵
AMS uniform sampling	$[10^{-10}; 10^{10}]^6$	uniform	7850
AMS uniform sampling + WW	$[0.2\bar{w}; 10\bar{w}]$	uniform	9600
AMS weighted sampling + WW	$[0.2\bar{w}; 10\bar{w}]$	weighted	9800

Table 6.3: Materials composition for the homogeneous PWR like box.

Isotope	Atomic density fraction ⁷	
	Near criticality	During transient
1H	6.3333×10^{-1}	6.3333×10^{-1}
^{16}O	3.1667×10^{-1}	3.1667×10^{-1}
^{10}B	1.1440×10^{-4}	1.1515×10^{-4}
^{235}U	1.5000×10^{-3}	1.5000×10^{-3}
^{238}U	4.8500×10^{-2}	4.8500×10^{-2}

Actually, it is not possible to turn off the Weight Window when the branchless collision is used in SERPENT2. To circumvent this issue for the first strategy just presented, a single time interval was set with a broad Weight Window so that as few particles as possible would undergo splitting or RR.

The AMS re-sampling kernel is currently either based on uniform sampling or weighted sampling (as in the combing method). It is possible to choose between these two modes in the input data file. To compare the impact on the score variance over time, three simulations were performed with these three sets of options over one single time bin and are summarized in Table 6.2)

The system modeled is a simple homogeneous 2-dimensional square (3-dimensional cuboid with reflections along z -axis) filled with PWR-like material described in Table 6.3. Boundary conditions along x and y -axes are leakage (absorbing) conditions. The material density is equal to 2.923 g.cm^{-3} . The ^{235}U enrichment is equal to 3%, and a subcritical transient is initiated by increasing the ^{10}B concentration at $t = 0$.

Even if the system is homogeneous, weights are expected to diverge over time. Since the branchless collision weight coefficient appearing in Equation 2.19 depends on the medium cross sections which in turn depend on the neutron energy, it can be superior to one in some regions of the energy domain and lower than one in some others.

The spatially integrated power over time is plotted in Figure 6.4. As expected, the case where no Weight Window was performed displays strong fluctuations with

⁵Due to errors encountered on some computational nodes, some simulations were not completed. Therefore, the number of independent batches is displayed for the reader's information. It should not impact the FoM however since this indicator is not sensitive to the number of histories simulated.

⁶To limit the number of neutrons that would undergo splitting or RR, a wide window was defined with only one time interval so that the average weight considered by the method would remain equal to 1.

⁷The fraction is renormalized before running the calculation so that the sum of all isotopes fraction is equal to unity.

variance jumps (as seen in Figure 6.5) while the two other cases remain more stable over time. Such important fluctuations can not be entirely explained by the lower number of independent calculations that were done for this case (7850 against 9600 and 9800 for cases with Weight Window), and, thus the variance between particle weights during the simulation should be the cause of these poor results.

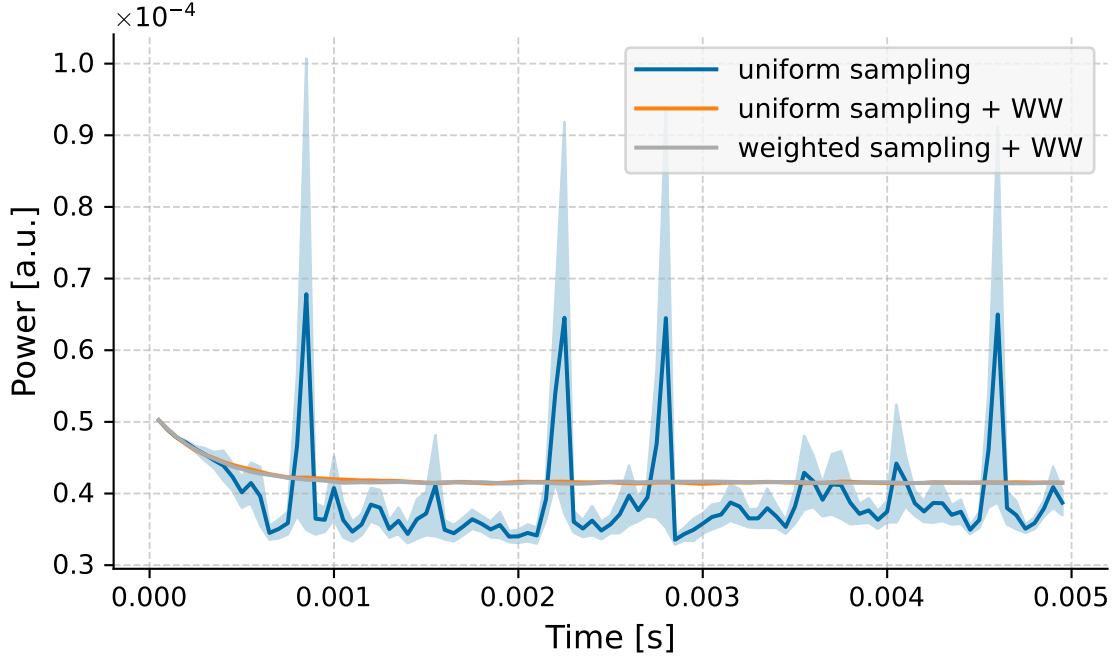


Figure 6.4: Spatially integrated power over 5 ms with 1σ confidence intervals for a subcritical (-0.23%) axially infinite cuboid (absorption boundary conditions along x and y axes).

To assess the effect of the AMS re-sampling kernel on performances, the two cases with Weight Window, which only differ by how tracks are sampled for duplication, were further compared using the figure of merit for the spatially integrated power over time and the distribution of the spatial figure of merit for different times, see Figure 6.6. The two methods show quite similar performances in regard to the spatially integrated power over time estimation as seen in Figures 6.5 and 6.6. However, weighted sampling seems to perform better at the end of the transient as the variance decreases faster with time which results in a better FoM at the end.

To analyze possible spatial effects, the FoM was evaluated on a fine regular spatial mesh, and the obtained distribution of the FoMs for each method are presented in Figure 6.7. The fission power was tallied in each time bin over 20 slabs equally distributed along the x -axis, and the FoM was computed in each slab over time. Regarding the spatial distribution of the FoM, both methods seem to have the same behavior with slightly better results for the uniform sampling as displayed in Figure 6.7. For both methods, the FoM increases very rapidly and then remains of the same order of magnitude over the majority of the transient, despite some fluctuations. A slight increase in the FoM with time is however apparent, which seems to be consistent with spatially integrated results presented in Figure 6.6. Only a few values fall above the 3rd quartile which indicates that only a few bins have a variance significantly lower than in the rest of the system.

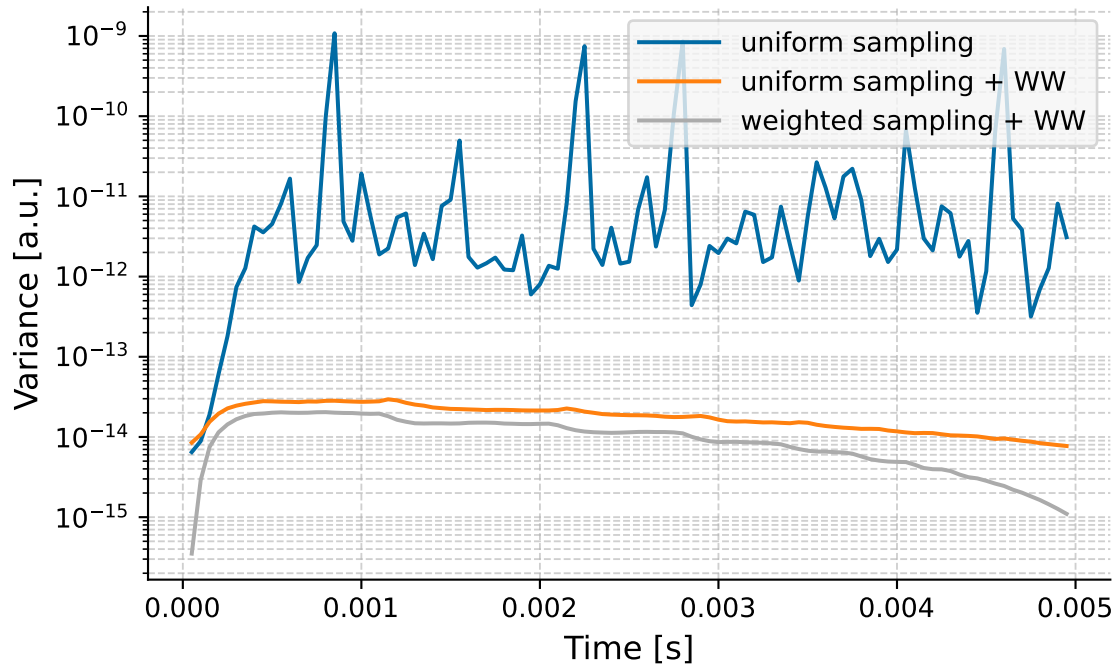


Figure 6.5: Variance of spatially integrated power over 5 ms for a subcritical (-0.23%) axially infinite cuboid for different AMS re-sampling kernels.

Finally, we can conclude that using the Weight Window technique should be mandatory as long as the branchless collision method is used, while the sampling strategies in the AMS framework may vary. Compared to the already used uniform sampling, weighted sampling not only gives unbiased results but it can also perform better in some cases (e.g., for the spatially integrated power at the end of the transient shown above).

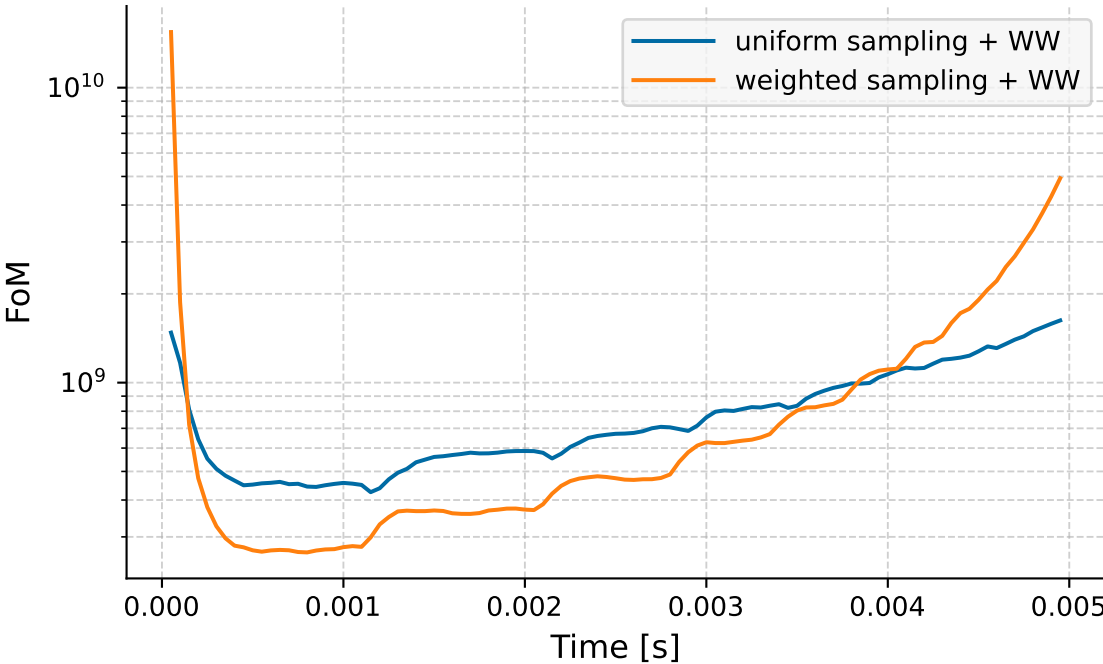


Figure 6.6: Figure of Merit for the spatially integrated power over 5 ms for a subcritical (-0.23%) axially infinite cuboid for different AMS re-sampling kernels.

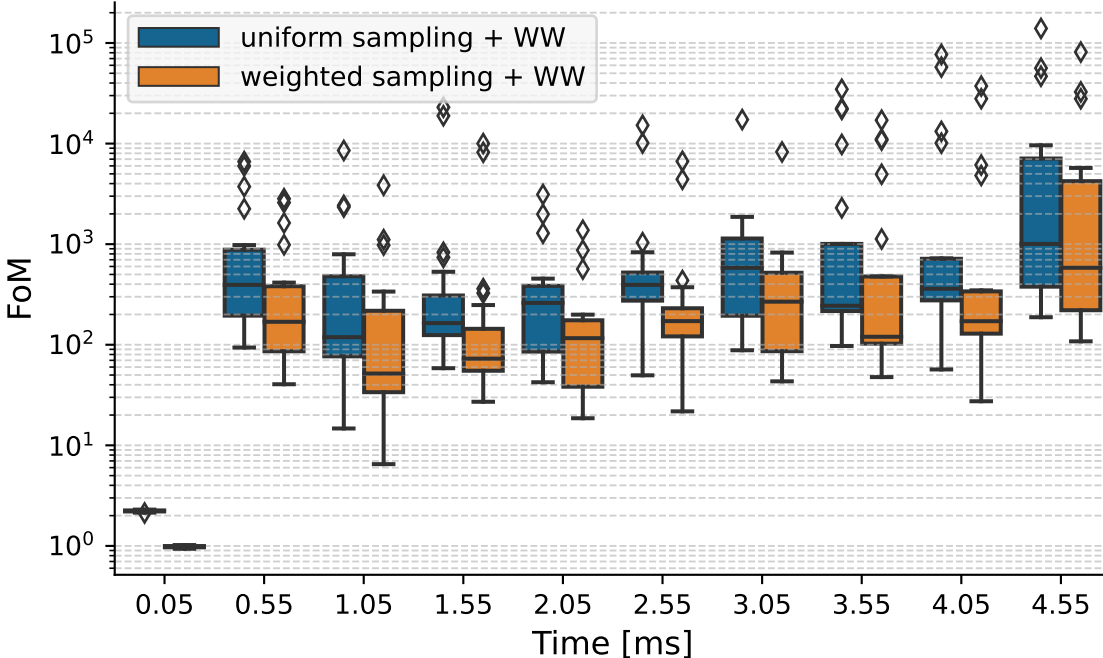


Figure 6.7: Box plots of the spatial FoM distribution for the local power evaluation with time for a subcritical (-0.23%) axially infinite cuboid for different AMS re-sampling kernels. White diamond markers indicate values either below the first quartile or above the third quartile.

To sum it up, the main features of the AMS for kinetics calculations in its current state are listed below.

- Precursors are not taken into account in the AMS structure (but are still created at collisions as in classical SERPENT2 simulations). This could be an avenue for future improvement.
- Time bins are still defined but no population control is performed between them. Instead, only the average neutron weight is updated so that the Weight Window changes over time, taking into account all previous AMS iterations when computing it for improved estimates.
- Finally, the Weight Window technique is used in combination with the branchless collision method to prevent the weight distribution from spreading too much. Regarding that point, there is no difference with SERPENT2 current kinetics calculations.

Up to this point, the content of this chapter was intended to present the particularities of the implementation of AMS in SERPENT2 that may be necessary for the characterization of the method. Before presenting a more complex case in Chapter 7, the following section presents a brief parenthesis to validate the hypothesis made at the end of Chapter 1. Indeed, it was foreseen that using the Weight Window method coupled with the branchless collision could improve the results of the AMS in the case of criticality calculations over heterogeneous geometries.

6.3 Computing the α fundamental mode of a one-speed 1D heterogeneous slab reactor

In the previous chapter, results presented on the heterogeneous slab test case led to the discussion of the need to use a weight control method to improve the FoM of the AMS in heterogeneous criticality calculations. In the previous section, it was shown that using the weight window technique available in SERPENT2 prevented large weight disparities induced by the branchless collision method for continuous energy calculations. The idea is now to verify our hypothesis on a heterogeneous geometry of a one-speed problem in SERPENT2. However, the AMS was not implemented into SERPENT2 criticality mode and SERPENT2 is not able to perform one-speed calculations in its current state, which prevent us from modelling the exact same case as the one presented in Section 5.3. To overcome these issues, we based our analysis on another type of eigenvalue problem describing the asymptotic behavior of a fissile system, the α -eigenvalue equation previously mentioned in Section 3.2. This section compiles modifications operated on SERPENT2 kinetics module to perform a one-speed α -eigenvalue calculation.

6.3.1 Modeling of a dynamic α mode with SERPENT2 in one-speed theory

SERPENT2 is not able to perform α -eigenvalue equations, neither is the AMS available in another mode than kinetics. But since the fundamental α mode describes the

asymptotic neutron population in time, computing the behavior of the neutron population over a long period of time in a kinetics calculation should give the same results [75]. Since the AMS is only available in SERPENT2 kinetics mode, we aimed at reproducing some sort of criticality calculation using kinetics calculations. A few tweaks described below were done so that the physics modeled would be as close as possible to the one in our one-speed criticality calculation.

SERPENT2 is not intended to perform one-speed calculations. To simulate mono-energetic neutrons, ACE files in which nuclear data read by SERPENT2 are stored were modified so that each cross section was set constant over the whole energy domain. Nevertheless, ACE files also contain the transition probabilities used to sample the change in neutrons energy due to collisions. Since we did not modify these probabilities in our modified ACE files, neutrons would still be able to loose energy. As a workaround, we also adjusted SERPENT2 so that the energy and speed of neutrons were also set constant to model the one-speed problem.

Furthermore, delayed neutron precursors have to be turned off. Because delayed and prompts neutrons have different time scales, considering delayed neutrons precursors in this simulation would delay the convergence to any asymptotic behavior the system could display. Since the system is mono-energetic and the time of emission does not play a role in criticality calculations, turning off precursors should not invalidate our model. In a similar fashion as how one-speed neutrons were added into SERPENT2, precursor tracking was turned off in kinetics calculation so that only prompt neutrons were simulated. All these modifications can be turned on and off at will using a flag read during compilation.

Notwithstanding the modifications explained above, performing a kinetics or α -eigenvalue calculation over an heterogeneous system should give different results from the ones obtained from a k -eigenvalue calculation, if the system k_{eff} were to be different from 1. Indeed, as explained by Cullen et al. in Ref. [75], operating population control between generations over fission neutrons exclusively introduces a bias regarding the real system. Kinetics calculations should be closer to real system from this point of view. As a result, the neutron population control in criticality calculations impacts both the energy spectrum and the spatial distribution of the flux. To illustrate the fundamental difference between the equations solved by the two types of eigenvalue equations (k and α), the spatial flux shape obtained from a k -eigenvalue criticality calculation was also computed for the test case presented below.

6.3.2 Test case

The objective was first to reproduce issues observed previously in Section 5.3 by modeling a system as close as the one that was simulated at that time, to then look at the impact of the weight window on the AMS FoM. The following test case is a set of heterogeneous one-dimensional slabs and is represented in Figure 6.8. Compared to the geometry previously modeled in our toy-model in criticality mode, this one was simplified so that only two different media remained. The k_{eff} of this configuration was computed with SERPENT2 in criticality mode and is equal to 1.00674 ± 2 pcm, which makes the system slightly supercritical. To estimate the fundamental α mode, kinetics calculations were performed according to the following modus operandi:

- starting from an arbitrary initial neutron distribution, a kinetics calculation is performed for a long time interval (about 10000 times the average generation

time constant of the system),

- tallies are computed when the distribution seems to have reached an asymptotic behavior, i.e., the first part of the transient is discarded as are inactive cycles in a k -eigenvalue calculation,
- flux tallies are saved over time intervals about the length of a hundred generations. Each time interval defines a scoring batch. This way, we intend to temper correlations between successive scoring batches when computing the average of their tallies,
- estimates of the space distribution of the flux are then averaged over successive scoring batches.

The average weight modifier due to the branchless collision method was computed in this substitute α -eigenvalue calculation once the spatial distribution had converged and is about 1.00285.

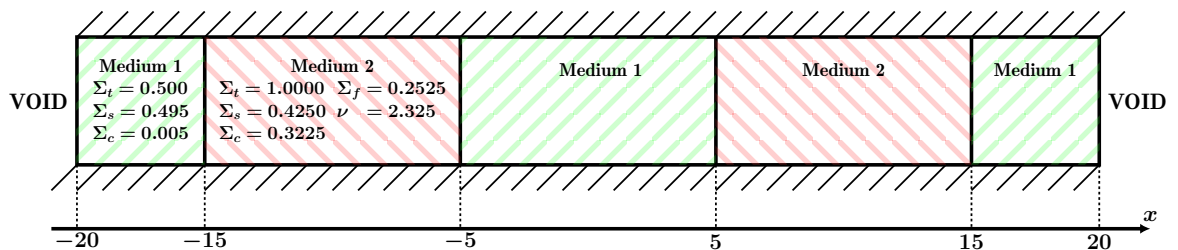


Figure 6.8: Heterogeneous slab geometry in SERPENT2 for the α -like kinetics calculation. The red dashed regions are fissile media, while the green ones are only diffusive/absorbing.

As previously mentioned, the solution of a k -eigenvalue calculation was also computed to illustrate the fundamental difference between kinetics or α -eigenvalue calculations and k -eigenvalue calculations. The k -eigenvalue calculation was performed using 10^4 particles per cycle to match the number of particles set in Section 5.3, over 10^4 successive generations. Cycles were regrouped in batches of 100 successive generations to tally the neutron flux. Scores were then computed by averaging over these batches, which should mitigate generational correlations on the variance estimation [82]. The first 100 cycles were discarded to wait for the source convergence. To get as close as the generational tracking as possible, the length of time steps had to be comparable to the average time between two neutron generations. If this is done, the population control in kinetics is performed with approximately the same frequency as in the equivalent k -eigenvalue calculation. The fission cross section in fissile material is equal to 0.25 cm^{-1} and the neutron speed was set to $1.75 \times 10^6 \text{ cm.s}^{-1}$. The average generation time in fissile material was therefore about 10^{-5} s . Kinetics calculations were therefore performed as such:

- 10^4 neutrons per time step,
- 10^4 time steps uniformly distributed from $t = 0 \text{ s}$ to $t = 0.1 \text{ s}$ (which makes the length of a time step equal to 10^{-5} s) for both scoring and population control,

6.3. COMPUTING THE α FUNDAMENTAL MODE OF A ONE-SPEED 1D HETEROGENEOUS SLAB REACTOR

Table 6.4: SERPENT2 simulation cases for the one-speed heterogeneous slab geometry in kinetics.

Case	Mode	AMS (on / off)	branchless (on / off)	weight window	Sampling (AMS only)
Criticality	criticality	off	off	-	-
Ref branchless	kinetics	off	on	$[0.2\bar{w}; 10\bar{w}]$	-
Ref analog	kinetics	off	off	-	-
AMS uniform sampling	kinetics	on	on	$[10^{-10}; 10^{10}]$	uniform
AMS uniform sampling + WW	kinetics	on	on	$[0.2\bar{w}; 10\bar{w}]$	uniform
AMS weighted sampling + WW	kinetics	on	on	$[0.2\bar{w}; 10\bar{w}]$	weighted

- the first 10^{-3} s were discarded to account for the asymptotic shape convergence before scoring (given that the average generation time in fissile material is about 10^{-5} s, this account for around 100 generations, which matches the number of inactive cycles discarded in the criticality calculation described above),
- spatial tallies were normalized to the flux integrated over the whole surface in each time bin before being regrouped in batches of 100 bins to compute scores.

Five kinetics calculations apart from the k -eigenvalue calculation were run to compare performances of the different sets of options. Three AMS calculations with different sampling and weight window parameters were compared to two SERPENT2 kinetics calculations without AMS (namely cases *Ref branchless* and *Ref analog*) but with time-dependent population control (whose frequency is given above). Table 6.4 gathers other simulation parameters of interest for all six cases performed.

6.3.3 Results

The flux spatial shape averaged over scoring batches (over generations for the k -eigenvalue calculations, or over time for kinetics calculations) was plotted in Figure 6.9 for all six simulations. For each line, a shaded area of the same color stands for the 3σ confidence interval. Apart from the orange line (case *AMS uniform sampling*), all confidence intervals are too narrow to be seen. To look closer at the differences between kinetics calculation, the deviations from the reference analog case was plotted in Figure 6.10. The ref. analog case was taken as a reference for these plots. An asymmetric pattern arises from this Figure, suggesting that the ref. analog case results do not display a symmetric shape. This might be due to light clustering effects. Apart from that point, it appears that all kinetics calculations remain coherent as their absolute differences with the ref. analog case remain between 3σ . They do, however, differ from the criticality calculation solution by more than their 3σ uncertainties as seen in Figure 6.9, which was expected as explained above.

Despite these differences between criticality and kinetics results, it was possible to reproduce the problem previously observed in Section 5.3, this time in kinetics, as shown by Figures 6.11 and 6.12. A pattern can be identified in Figures 6.11a, 6.11b, 6.11c and 6.12a. This pattern is due to the fact that the particles all have the same weight at the beginning of the time interval. Particles initially have the same weight because either this is the first time step (Figures 6.11a and 6.12a) or the population control in the reference calculation has assigned the same weight to the particles it

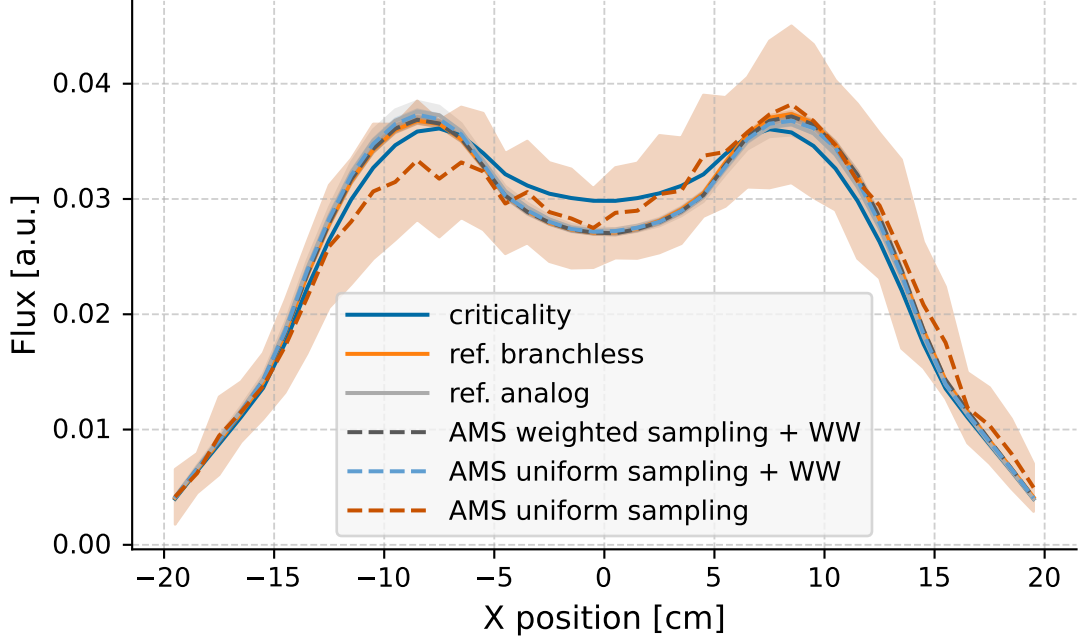


Figure 6.9: Flux over space for the heterogeneous slab case in kinetics with 3σ confidence intervals displayed as shaded areas.

sampled (Figures 6.11b and 6.11c). The pattern disappears as particle weights spread over multiple orders of magnitude when they are not constrained nor reset to unique value at the beginning of a time interval (Figures 6.12b and 6.12c).

From a qualitative point of view, these graphs show that particle weights remain really close to one another over time when a classical population control is performed (Figure 6.11). Whereas, their spread increases with time when the AMS is used without any technique to constrain the weight (Figure 6.12). As illustrated by Figure 6.12, the ratio between lower and higher weights goes from around 1 at $t = 10^{-5}$ s (Figure 6.12a) to about 12 orders of magnitudes at $t = 5 \times 10^{-2}$ s (Figure 6.12c).

Introducing the weight window technique into the calculation drastically reduces the weight difference between particles in the same time bin as seen in Figure 6.13. There is now only about two orders of magnitude between high and low weights (although most of weights are of the same order of magnitude and only a few ones are really low compared to others, as seen in Figure 6.13c). Replacing uniform sampling by weighted sampling in the AMS re-sampling stage does not seem to bring in much change regarding the spread, which means that the weight window is already efficient enough regarding that point. However, it allows particle weights not to soar as much as when uniform sampling is used. Indeed, when looking at the weight distribution at $t = 5 \times 10^{-2}$ s, weights range approximately from 10^{24} to 10^{26} for the *AMS uniform sampling + WW* case (Figure 6.13c) and from 4×10^{21} to 4×10^{23} for the *AMS weighted sampling + WW* case (Figure 6.14c).

These observations are quantitatively summarized in Figure 6.15 through the same metrics used in the previous chapter to assess the spread of the weight distribution. Metrics include the ratio between the maximum and minimum weights, the relative standard deviation, the ratio between the mean and the median, and the relative interquartile range (IQR). This time however, they are monitored over time bins instead

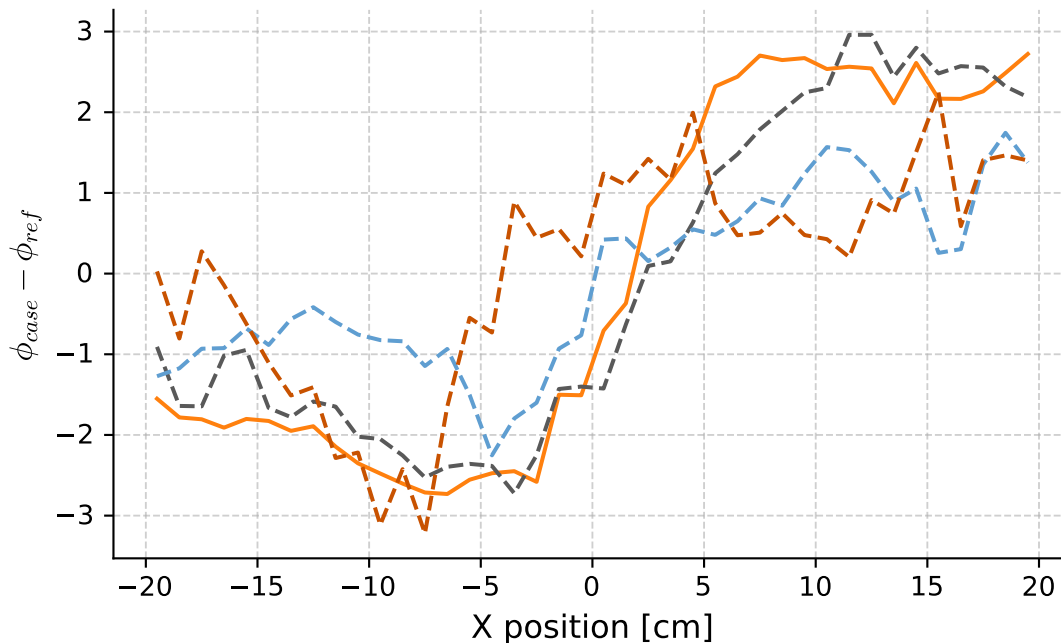


Figure 6.10: Deviations from a reference solution (ref. analog case) for the heterogeneous slab in kinetics. Differences are displayed in terms of the combined standard error of the mean.

of generations. The reference branchless case exhibits stationary behavior regarding these indicators because particles sampled during the population control process are assigned the average weight, which concentrates the distribution around the mean value at the beginning of each time bin. These results also confirm that, as long as the weight window method is used, sampling new tracks based on a uniform sampling or a weighted one is quantitatively quite similar compared to the case where no weight window was applied.

Once again, the resulting FoM relative to spatial flux estimates was then computed and is plotted in Figure 6.16. Again, a difference of several orders of magnitude was found between the initial AMS situation represented by case *AMS uniform sampling* and the reference calculation, which is case *ref branchless*. As a reminder, the case *AMS uniform sampling* is supposed to mimic the AMS criticality calculation performed in Section 5.3. Among the three improved AMS cases presented above, the one displaying the best FoM is the one in which WW was used in combination with weighted sampling during the AMS re-sampling step. Still, its figure of merit remains lower than that of the reference calculation.

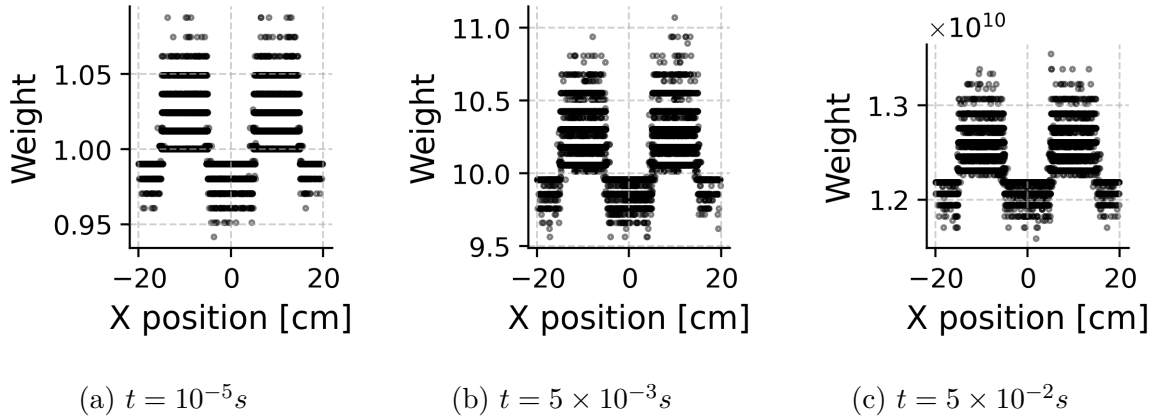


Figure 6.11: Weight VS position along x -axis for collision points in the kinetics heterogeneous slab geometry (case *ref branchless*) for different times.

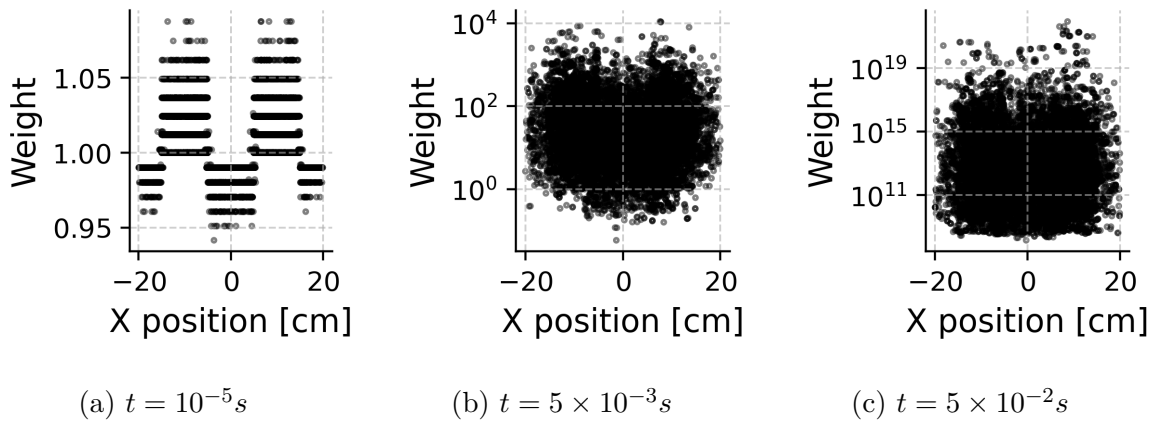


Figure 6.12: Weight VS position along x -axis for collision points in the kinetics heterogeneous slab geometry when no weight window is used in the AMS (case *AMS uniform sampling*) for different times.

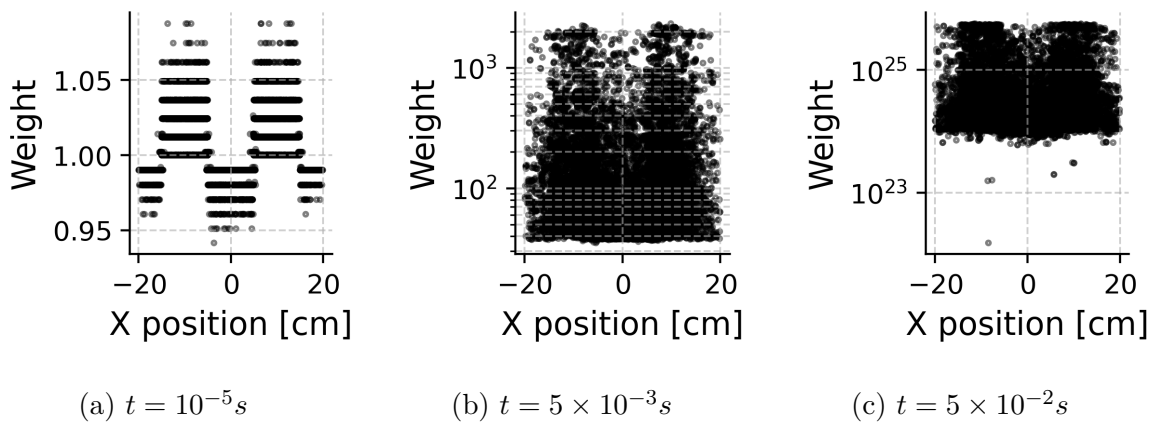


Figure 6.13: Weight VS position along x -axis for collision points in the kinetics heterogeneous slab geometry when the weight window technique is used in the AMS (case *AMS uniform sampling + WW*) for different times.

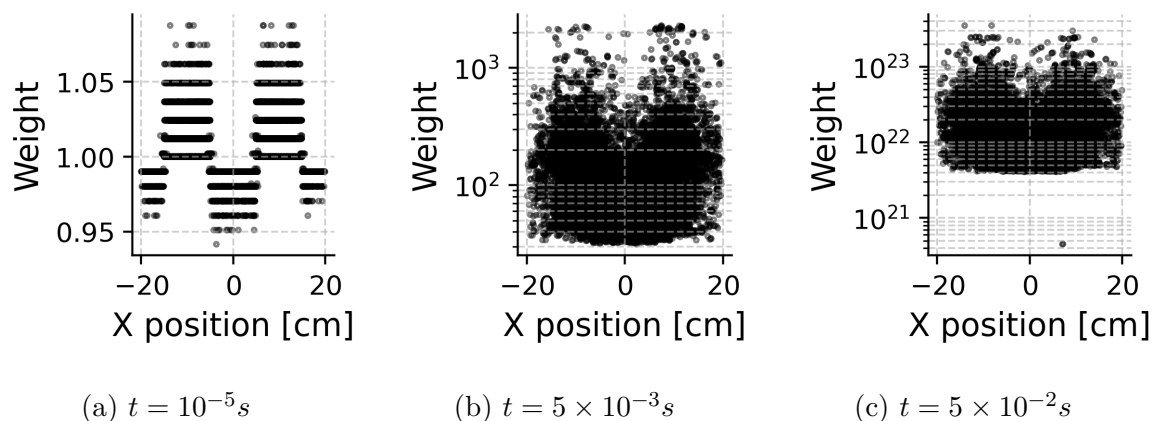


Figure 6.14: Weight VS position along x -axis for collision points in the kinetics heterogeneous slab geometry when the weight window technique is used in the AMS and weighted sampling is used (case *AMS weighted sampling + WW*) for different times

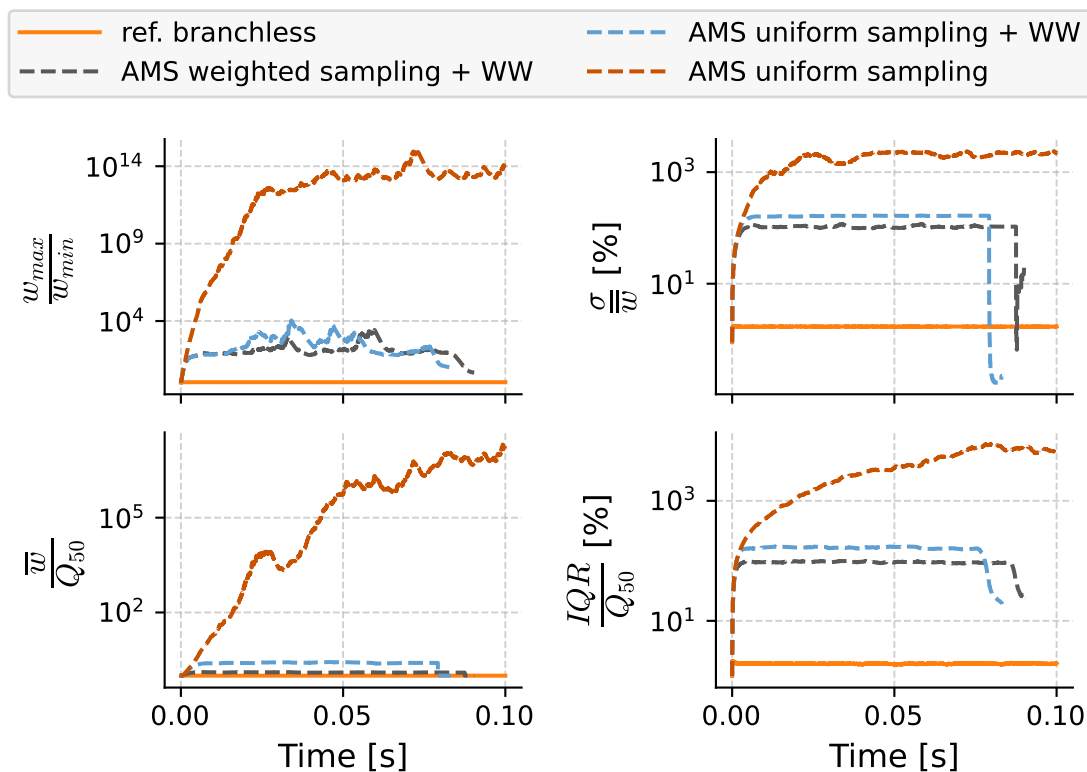


Figure 6.15: Statistical moments of the weights distribution over time, smoothed using exponentially weighted moving average (original results are given in Figure A.2)

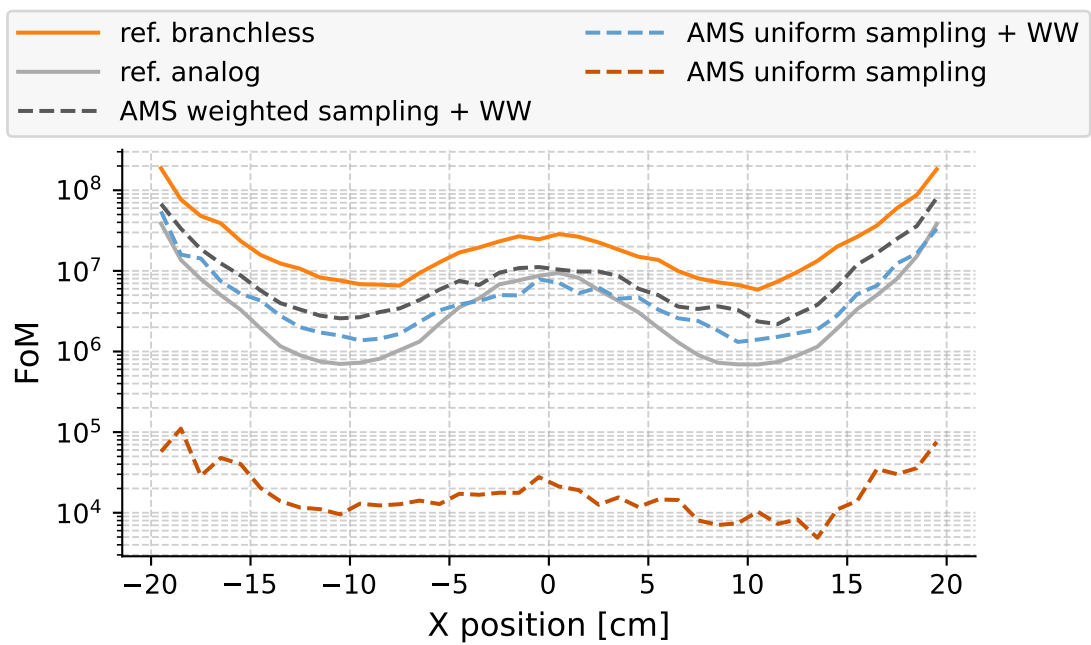


Figure 6.16: Figure of Merit over space for the heterogeneous slab case in kinetics.

In definitive, the unanswered problem highlighted in Section 5.3 was reproduced. Using SERPENT2 kinetics mode, the asymptotic behavior of a time-dependent supercritical system was modeled, hence computing the spatial flux distribution similar to the fundamental α mode of the system. By modifying SERPENT2 kinetics calculations, it was possible to simulate mono-energetic neutrons to be as close as possible to the original problem. Results have shown that using the weight window method allowed to improve the performances of AMS calculations with the branchless collision method in heterogeneous configurations, thus fixing the issue highlighted in criticality calculation in our toy-model. Regarding the FoM of the time averaged spatial flux estimate, it was increased over multiple orders of magnitude, bringing it close to that of a baseline SERPENT2 branchless calculation including population control.

Conclusion

In this chapter, we presented how the Adaptive Multilevel Splitting was implemented in SERPENT2 Monte Carlo code, for kinetics calculations only. As a reminder, the objective was to outline how the AMS algorithm was integrated to a recent version of SERPENT2, by detailing the reasons that led to the current implementation of the method. The results presented here were not destined to characterize the global efficiency of the method compared to state-of-the-art kinetics simulations, this specific topic is reserved for the next chapter.

Population control techniques available in SERPENT2 were turned off as the AMS was used to re-inject neutrons over time (as in criticality calculations presented in previous chapter). The time bins structure has, however, been kept in order to be able to stop all neutrons at regular time boundaries so as to update the average weight of the population. Indeed, this average weight is always used by the weight window method during each branchless collision, that need was demonstrated in this chapter. For now, delayed neutron precursors are not followed within the AMS framework (but are still simulated during the transport step). The method was then successfully applied to neutron kinetics calculations. Test cases included a homogeneous system with continuous energy cross sections, and a one-dimensional heterogeneous slab geometry with mono-energetic neutrons. In both cases, the weight window technique was found to be essential to compensate for the increasing dispersion of particle weights over time due to the diversity of branchless collision weight modifiers. Thus, controlling particle weights in this way allowed to fix the issue illustrated earlier in Section 5.3 for heterogeneous cases.

Nonetheless, the current implementation of the Adaptive Multilevel Splitting in SERPENT2 remains somewhat rigid, as it does not allow for delayed neutron precursors to be resampled by its iterative algorithm. Future perspectives include investigating modifications of the code which would allow handling delayed neutron precursors in the AMS framework.

Until this point, the importance fed to the Adaptive Multilevel Splitting algorithm remained quite simple. Indeed, we took advantage of the AMS robustness to use the time, or the neutron generation as the main driver of importance. In the next chapter, the method is used to model a transient in a fuel assembly cluster and compared to a SERPENT2 baseline reference calculation. Its performances are analyzed regarding

spatially integrated and local tallies over the whole transient for different importance maps.

Chapter 7

Transient simulation with the Adaptive Multilevel Splitting

A wizard is never late. Nor is he early; he arrives precisely when he means to.

- Gandalf

Contents

7.1	The adjoint flux as neutron importance	130
7.1.1	Relation between neutron importance and adjoint flux . . .	130
7.1.2	Setting the "initial" condition in the backward approach . .	133
7.1.3	Formulation of the adjoint point kinetics	134
7.2	Control rods drop on a 3x3 UOX assembly cluster	137
7.2.1	Case description	137
7.2.2	Adjoint flux calculation for the rod drop transient	139
7.3	Numerical results	142
7.3.1	Impact of the AMS on the power over time estimate	143
7.3.2	Impact of the multi-parameter importance map on the FoM degradation	144
7.3.3	FoM gains regarding the power distribution in the last time bin	146
	<i>Conclusion</i>	149

Now that the AMS was implemented in SERPENT2 and can be used to simulate transients without any major hindrance, it is time to take a closer look at the importance function used to rank tracks. The work presented hereafter in this chapter aims at presenting the Adaptive Multilevel Splitting capabilities to compute local tallies during a transient simulation. In contrast to Chapters 5 and 6, non-trivial importance functions will be used here with the objective of comparing the method effects regarding variance reduction of local and spatially integrated tallies.

After a brief review on the link between the concept of neutron importance and the solution of the adjoint transport equations, a subcritical transient initiated by dropping control rods in a 3x3 fuel assemblies cluster is presented along with the strategy used to retrieve an importance over time and space. Finally, comparisons between current SERPENT2 performances and those of the Adaptive Multilevel Splitting are presented for different importance maps.

7.1 The adjoint flux as neutron importance

Variance reduction methods in neutron physics often rely on the concept of particle *importance*. The term *importance sampling* (cf. Section 2.2.4.2) for example refers to using a modified sampling kernel to favor particles "of interest" over particles that have a low probability to contribute, thus improving the variance estimation of a score. Multiple authors introduced the adjoint equations for time-independent neutron transport problems [133, 134, 135, 136] and interpreted their solution as the *importance* of neutrons. Later, Lewins derived the time-dependent importance (or adjoint) equations in neutron transport and diffusion theories for both neutrons and delayed neutron precursors [137, 138].

In addition, zero-variance schemes have been developed based on this definition of the importance for both static and time-dependent problems [57, 54, 55]. If zero-variance schemes are already several decades old for stationary calculations, the use of a *true time-dependent*¹ adjoint flux is very recent [57] and has not been used in practical kinetics applications to our knowledge yet. The solution of these adjoint equations, or the *adjoint flux* in case of neutrons, is thus often considered as the best option for importance based variance reduction methods for that reason.

It is important to note that, unlike importance sampling methods (such as the exponential biasing method) which are a key ingredient of zero-variance schemes [54], the AMS is only able to achieve a minimum, non-zero, variance [113], and thus cannot lead to zero-variance schemes. Its advantage lies in its robustness, which allows worthwhile variance reduction to be obtained only with an approximation (even a relatively coarse one) of the optimal importance function [7] (i.e., the committor in applied mathematics). But in practice, the exact importance of the system is beyond calculation (it would require to know the solution to the problem one seeks to find in advance, and with an infinite accuracy), and practical computational schemes are therefore based on more or less approximate importance estimates.

In this section, a brief presentation of the concept of importance in reactor physics is done, as it may be used by the Adaptive Multilevel Splitting algorithm [7]. The concept of importance and the derived equations presented in this section mostly follow Lewins' formalism in Ref. [124].

7.1.1 Relation between neutron importance and adjoint flux

To derive the importance equations, one can start from the physical interpretation of the importance, defined as such

Definition 1 *The importance of one neutron or precursor at t is equal to its probable contribution to a detection process at a time $t' \geq t$.*

¹By *true time-dependent*, we mean here not computed from quasi-static approximations.

to the following differential equations for the neutron and precursor importances

$$\begin{aligned}
 -\frac{1}{v} \frac{\partial \phi^*}{\partial t}(\mathbf{r}, E, \boldsymbol{\Omega}, t) - \boldsymbol{\Omega} \cdot \nabla \phi^*(\mathbf{r}, E, \boldsymbol{\Omega}, t) + \Sigma_t(\mathbf{r}, E) \phi^*(\mathbf{r}, E, \boldsymbol{\Omega}, t) = \\
 \int_0^\infty \int_{4\pi} \left[\Sigma_s(\mathbf{r}, E) f_s(E, \boldsymbol{\Omega}, \rightarrow E', \boldsymbol{\Omega}') + (1 - \beta) \nu \Sigma_f(\mathbf{r}, E) \frac{\chi_{f,p}(E')}{4\pi} \right] \phi^*(\mathbf{r}, E', \boldsymbol{\Omega}', t) d^2\Omega' dE' \\
 + \sum_k \beta_k \nu \Sigma_f(\mathbf{r}, E) C_k^*(\mathbf{r}, t) + \eta_\phi(\mathbf{r}, v, \boldsymbol{\Omega}, t) \quad (7.2)
 \end{aligned}$$

$$-\frac{\partial C_k^*}{\partial t}(\mathbf{r}, t) = \lambda_k \iiint \frac{\chi_{f,d}^k(E')}{4\pi} \phi^*(\mathbf{r}, E', \boldsymbol{\Omega}', t) d^2\Omega' dE' - \lambda_k C_k^*(\mathbf{r}, t). \quad (7.3)$$

where

- $\phi^* = vN^*$ is the adjoint flux or neutron importance²,
- Σ_s and Σ_f are the macroscopic cross section for scattering and fission,
- $f_s(E, \boldsymbol{\Omega}, \rightarrow E', \boldsymbol{\Omega}')$ is the transition kernel for scattering,
- ν is the mean number of neutrons produced by fission,
- $\chi_{f,p}$ is the fission spectrum for prompt neutrons,
- β is the total delayed neutrons fraction, and β_k is the delayed neutrons fraction for precursor family k ,
- $\chi_{f,d}^k$ is the fission spectrum for delayed neutrons emitted by precursors of family k ,
- λ_k is the decay constant for precursor family k ,
- C_k^* is the adjoint precursor concentration in family k , it is also the precursor importance,
- η_ϕ is the detector response function for the neutron flux.

Let us highlight the difference with the source term in the direct system of equations (cf. Equations 1.10 and 1.11). Although the general form is quite similar, the detail of the terms may be slightly different. In the adjoint problem, a particle loses importance as it is distributed among its progeny (cf. Axiom 1). Equations 7.2 and 7.3 thus describe how the importance is transmitted to the progeny rather than where the importance comes from when placed at point $(\mathbf{r}, E, \boldsymbol{\Omega}, t)$. This is why the integration over energy and direction is done on the kernels (for scattering, fission and decay) outputs rather than their inputs (as it is done in the direct system of equation). For the same reason, the roles of Σ_f and $\chi_{f,p}$ (or $\chi_{f,d}^k$ in Equation 7.3) are flipped regarding the integration over energy.

²Solving the adjoint equations in terms of adjoint neutron density or adjoint flux is equivalent. For more details, see Appendix B.

In time-dependent diffusion theory (i.e., space-time kinetics), the importance equations are written

$$\begin{aligned}
 -\frac{1}{v}\frac{\partial\phi^*}{\partial t}(\mathbf{r}, E, t) - \nabla \cdot D(\mathbf{r})\nabla\phi^*(\mathbf{r}, E, t) + \Sigma_t(\mathbf{r}, E)\phi^*(\mathbf{r}, E, t) = \\
 \int [\Sigma_s(\mathbf{r}, E)f_s(E \rightarrow E') + (1 - \beta)\nu\Sigma_f(\mathbf{r}, E)\chi_{f,p}(E')] \phi^*(\mathbf{r}, E', t)dE' \\
 + \sum_k \beta_k\nu\Sigma_f(\mathbf{r}, E)C_k^*(\mathbf{r}, t) + \eta_\phi(\mathbf{r}, E, t) \quad (7.4)
 \end{aligned}$$

$$-\frac{\partial C_k^*}{\partial t}(\mathbf{r}, t) = \lambda_k \int \chi_{f,d}^k(E') \phi^*(\mathbf{r}, E', t)dE' - \lambda_k C_k^*(\mathbf{r}, t) \quad (7.5)$$

To define a unique solution, it is necessary to set boundary conditions, in particular regarding time.

7.1.2 Setting the "initial" condition in the backward approach

The negative sign in front of the term $\frac{1}{v}\frac{\partial\phi^*}{\partial t}$ suggests some sort of backward approach with regard to time to compute the importance (as opposed to the direct or forward approach for the direct neutron flux). Hence, while it is necessary to define an initial state of the system to solve the direct equations, solving the importance equations requires final conditions.

For the direct approach, an initial neutron population has to be imposed. When the system is initially at equilibrium, the initial precursor concentrations can be deduced using the equations characterizing precursor concentrations at steady-state :

$$\frac{\partial C_k}{\partial t}(\mathbf{r}, t = 0) = 0 = \beta_k\nu\Sigma_f\phi(\mathbf{r}, t = 0) - \lambda_k C_k(\mathbf{r}, t = 0) \quad (7.6)$$

$$C_k(\mathbf{r}, t = 0) = \frac{\beta_k\nu\Sigma_f}{\lambda_k}\phi(\mathbf{r}, t = 0). \quad (7.7)$$

For the importance equations however, Axiom 1 implies that a final condition must be set instead of an initial one. An importance is defined for a given detector, and changing the detector would change the definition of the importance. Just as a direct transport problem depends on the initial source, the adjoint problem depends on the definition of the detector in the final state. In addition, there is no reason to consider some sort of equilibrium at t_f . $\phi^*(\mathbf{r}, E, t_f)$ (or $C_k^*(\mathbf{r}, t_f)$) should thus be equal to the probability for a neutron (or a delayed neutron precursor) alive at t_f to contribute to the score at the exact same time, thereby

$$\phi^*(\mathbf{r}, E, t_f) = \eta_\phi(\mathbf{r}, E, t_f) \quad (7.8)$$

$$C_k^*(\mathbf{r}, t_f) = \eta_{C_k}(\mathbf{r}, t_f) \quad (7.9)$$

where η_ϕ is the detector response function for the neutron flux, and η_{C_k} is the detector response function for the precursor concentration C_k . These two equations mean that for a given system, there are several importances depending on the considered detector, in the same way that there are several solutions to a time dependent transport problem for different initial sources. For a neutron-sensitive only detector, the reaction rate in

the detector is defined as $R = \int \int \int \int \phi(\mathbf{r}, E, \boldsymbol{\Omega}, t) \eta_\phi(\mathbf{r}, E, \boldsymbol{\Omega}, t) d^3r dE d^2\Omega dt$, and thus we have $\eta_{C_k} = 0$. Note that the same reaction rate could also be written as a function of the importance

$$R = \int \int \int \int \phi^*(\mathbf{r}, E, \boldsymbol{\Omega}, t) Q(\mathbf{r}, E, \boldsymbol{\Omega}, t) d^3r dE d^2\Omega dt \quad (7.10)$$

where $Q(\mathbf{r}, E, \boldsymbol{\Omega}, t)$ is the neutron source at point $(\mathbf{r}, E, \boldsymbol{\Omega}, t)$.

In the direct formulation of the neutron kinetics equations, the time integration scheme develops $\phi(\mathbf{r}, E, t_n)$ and $C_k(\mathbf{r}, t_n)$ as functions of $\phi(\mathbf{r}, E, t_{n-1})$ and $C_k(\mathbf{r}, t_{n-1})$ (we go forward in time). For the backward approach, the idea is to formulate $\phi^*(\mathbf{r}, E, t_{n-1})$ and $C_k^*(\mathbf{r}, t_{n-1})$ as functions of $\phi^*(\mathbf{r}, E, t_n)$ and $C_k^*(\mathbf{r}, t_n)$ (starting from a final condition we try to rewind to $t = 0$). Numerically solving this equation legitimately raises questions. For deterministic solvers, it seems sufficient to change the direction of the temporal integration scheme to go backward in time, which should not be a major problem. Indeed, the matrix system, although different, seems to have the same properties as the direct system. The same numerical methods should therefore be applicable. For Monte Carlo methods however, the backward approach is quite different from the direct one regarding transport and collision kernels, and may cause difficulties from a rigorous backward computation of the time-dependent adjoint transport equation. It is nevertheless possible to compute it using a brut force forward approach, but this technique would be extremely expensive (and thus counterproductive if the objective is then to model the exact same system using the adjoint solution to reduce the variance...).

7.1.3 Formulation of the adjoint point kinetics

Ideally, the more accurate the importance, the more efficient the variance reduction. However, having access to the ideal adjoint flux implies knowing the solution of the adjoint problem which is at least as difficult to solve as the direct problem. Since solving the problem a first time to improve the speed of the second resolution of the same problem at the same level of accuracy is pointless, the goal is to use less accurate but faster solvers to obtain an estimate of the adjoint flux. Besides, the AMS method has shown to be quite robust regarding the choice of importance for time-independent attenuation problems [7], which is rather encouraging to investigate the possibility of using an approximation. For steady-state configurations, deterministic methods are often used to compute an adjoint flux estimate that is then used by variance reduction methods [105, 106]. But to date, there is no space-time kinetics solvers to the author's knowledge that can handle time-dependent adjoint problems. An attempt has been made during this thesis to extend DONJON5 kinetics solver capabilities to compute time-dependent adjoint flux. However, the method is not yet mature and it is not possible to use it.

In order to study the behavior of the time-dependent AMS regarding the importance function, a simplified time-dependent importance was used. Based on the same base principle as the point kinetics approximation, the importance $\phi^*(\mathbf{x}, t)$ (where \mathbf{x} refers to all variable of the phase-space except for time, and t is the time) is decomposed as the product of two functions with separated variables $N^*(t)$ and $\psi^*(\mathbf{x})$. Here, $N^*(t)$ represents the variations of the amplitude of the total importance over time, while $\psi^*(\mathbf{x})$ is the shape of the importance over the phase-space.

Two approaches could be used to derive equations for $N^*(t)$. The most simple one is to consider the direct point kinetics equations and develop the adjoint equations

based on the mathematical definition of an adjoint operator. For an operator \mathcal{L} , the adjoint operator \mathcal{L}^* is defined from the dot product by [124]

$$\int \psi^* \mathcal{L} \psi d\mathbf{x} = \int \psi \mathcal{L}^* \psi^* d\mathbf{x}. \quad (7.11)$$

The other method would be to consider Equations 7.2 and 7.3 and apply the point reactor approximation (i.e., the separation of the time variable with other phase-space variables). The most simple one, which was adopted for the following derivation, is to consider the mathematical adjoint to the direct point kinetics equations.

Considering the direct point kinetics equations, the evolution of the neutron population amplitude and precursor concentrations over time are written

$$\frac{dN}{dt}(t) = \frac{\rho(t) - \beta}{\Lambda} N(t) + \sum_k \lambda_k C_k(t) \quad (7.12)$$

$$\frac{dC_k}{dt}(t) = \frac{\beta_k}{\Lambda} N(t) - \lambda_k C_k(t) \quad (7.13)$$

where N is the neutron population amplitude, C_k is the precursor concentration in group k , ρ is the reactivity of the system, β is the total effective delayed neutron fraction (β_k being the effective delayed neutron fraction for precursor group k), Λ is the effective generation time and λ_k is the precursor decay constant for group k . Transforming Equations 7.12 and 7.13 into a matrix system equation leads to

$$\frac{d}{dt} [\varphi(t)] = [R(t)] [\varphi(t)] \quad (7.14)$$

with matrix $[R(t)]$ and vector $[\varphi(t)]$ defined as

$$[R(t)] = \begin{bmatrix} \frac{\rho(t) - \beta}{\Lambda} & \lambda_1 & \dots & \lambda_K \\ \frac{\beta_1}{\Lambda} & -\lambda_1 & & \\ \vdots & & & \\ \frac{\beta_K}{\Lambda} & & & -\lambda_K \end{bmatrix} \quad (7.15)$$

$$[\varphi(t)] = \begin{bmatrix} N(t) \\ C_1(t) \\ \vdots \\ C_K(t) \end{bmatrix}. \quad (7.16)$$

Following the method presented in Ref. [139], the mathematical adjoint to Equation 7.14 is built by adding a minus sign in front of the time derivative and transposing the matrix $[R(t)]$. Which, once developed in a differential form, gives

$$-\frac{dN^*}{dt}(t) = \frac{\rho(t) - \beta}{\Lambda} N^*(t) + \sum_k \frac{\beta_k}{\Lambda} C_k^*(t) \quad (7.17)$$

$$-\frac{dC_k^*}{dt}(t) = \lambda_k N^*(t) - \lambda_k C_k^*(t). \quad (7.18)$$

It is important to mention that References [137, 138, 124] highlight that the concept of time-dependent importance could lead either to the time-dependent adjoint transport equation or the time-dependent adjoint diffusion equation. None of the references cited

above show that the derivation of the physical concept of importance would be equal to the adjoint point kinetics equations in the case of the point reactor approximation. In that regard, deriving the adjoint operators from the direct point kinetics equations (i.e., what has been done above and led to Equations 7.17 and 7.18) only allows to compute an estimation of the importance for neutrons and precursors. The derivation of the point reactor approximation starting from Equations 7.2 and 7.3 could also be used to compute an estimation of the importance. There is no indication that these two approaches should lead to the same system of equations. Similarly, there is no conclusion as to which approach should give the best approximation.

Since neutronics codes are implemented to compute only direct kinetics effective parameters for now, the approach that was selected in the framework of this thesis is the one presented above leading to Equations 7.17 and 7.18. We can afford this kind of approximation without high risk because the importance function is only used to select histories of interest³, and does not bias estimators used to compute the average value of the result.

Choosing the condition at t_f

The final conditions for adjoint quantities will differ depending on the choice of the observable of interest. For example, if the total number of neutrons at t_f is of interest, the adjoint amplitude will be

$$N^*(t_f) = 1 \quad (7.19)$$

since the contribution of a neutron to the total neutron population at t_f is 1. On the other hand, a precursor that is present at t_f has not decayed and thus did not contribute to the neutron population through a delayed neutron at t_f . Hence, the importance of precursors at t_f is defined as

$$C_k^*(t_f) = 0. \quad (7.20)$$

The choice of the detector (represented by η_ϕ) is quite important as it defines the aim of the variance reduction method. For the time-dependent importance equations, the choice of t_f is also important as it determines when one seeks to define the detector. For example, in a transient simulation, the goal can either be to reduce the variance at the peak power or later during the transient.

Regarding the shape function $\psi^*(x)$, plenty of codes are able to compute a static adjoint flux [140, 141, 142, 143]. For the calculations presented hereinafter, the code ADVANTG [144] was used (cf. Section 7.2.2). The resulting importance function used in this thesis is therefore of the form

$$I(\mathbf{r}, E, t) = N^*(t)\psi_{ADVANTG}^*(\mathbf{r}, E) \quad (7.21)$$

where $N^*(t)$ was computed by solving Equations 7.17 and 7.18 with a homemade python code. The time discretization is based on the Crank-Nicolson method and is presented in Appendix B.2.

³And in the case of AMS, a high precision on its absolute value is not useful since it is only used to order tracks.

7.2 Control rods drop on a 3x3 UOX assembly cluster

Kinetics calculations were performed on a 3x3 UOX assembly cluster with SERPENT2 official version 2.1.32 and with our in-house version⁴ including the AMS method. The aim was to assess potential improvements regarding the figure of merit for time-dependent and spatially defined detectors. The case that was simulated is a control rod drop inducing a subcritical transient in the system. This choice was made for three reasons:

- a 3x3 assembly cluster is heterogeneous to test different spatial importance maps, but simple enough to avoid any unexpected issue regarding the physics (and its size is close to benchmarks previously modeled in kinetics, such as SPERT-III [145] or TMI-Minicore [24]),
- supercritical transients are more difficult to optimize with respect to simulation parameters (number of time steps, Weight Window parameters, ...), which makes them more appropriate for any further validation of the method,
- a subcritical transient allows to emphasize the population control mechanism of the AMS, which is valuable as a first step towards full characterization of the method.

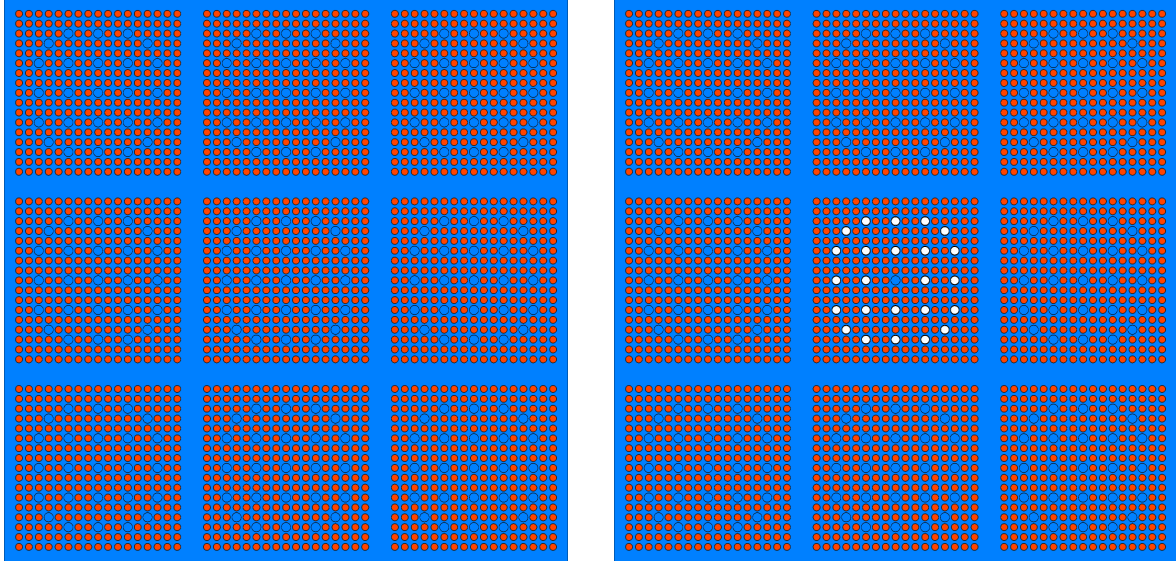
The purpose of this study case was to present a proof of concept for the AMS for a time-dependent simulation. The following section presents the characteristics of the problem and the calculation parameters used in the different simulations.

7.2.1 Case description

The modeled geometry represents a 2-dimensional cluster of 9 initially identical assemblies with absorption boundary conditions applied on the external boundaries. All physical characteristics of the system are listed in the SERPENT2 input data file in Appendix C. At $t < 0$, the system is critical and the configuration of the cluster is displayed in Figure 7.1a. Compared to more realistic assembly clusters, water gaps between assemblies are much wider here, around two pincells large. The width of these water gaps was arbitrarily set to that of a fuel pincell (which simplified the input data file), and should not significantly impact the behavior of the different methods. At $t = 0$, AIC control rods are instantly inserted in 24 of the 25 guide tubes of the central assembly as depicted in Figure 7.1b, thus inducing a subcritical transient of -6819 pcm of reactivity (which corresponds to approximately -9.2%)⁵. By taking into account heterogeneity, continuous energy and time dependence, the idea was to make the problem on which the AMS would be applied more complex. This way, the result of the interactions between the different methods (e.g., branchless collision, weight window and AMS) used in a simulation could be analyzed on a plausible system. As for the size of the cluster, it fulfills two purposes. First, we wanted to keep the leakage boundary conditions to be sure that some neutrons would still disappear despite the branchless collision method so that the AMS could iterate (we could not be sure of the

⁴As a reminder, the base version from which the changes were made to implement the AMS is also 2.1.32.

⁵This reactivity change was computed from a steady-state k -eigenvalue calculation



(a) Before the rod drop : $t < 0$. Each assembly has 25 empty guide tube and 17x17 fuels rods.

(b) After the rod drop : $t \geq 0$. Outer assemblies are similar to the initial configurations, whereas the inner assembly has 24 AIC control rods and 1 central empty guide tube.

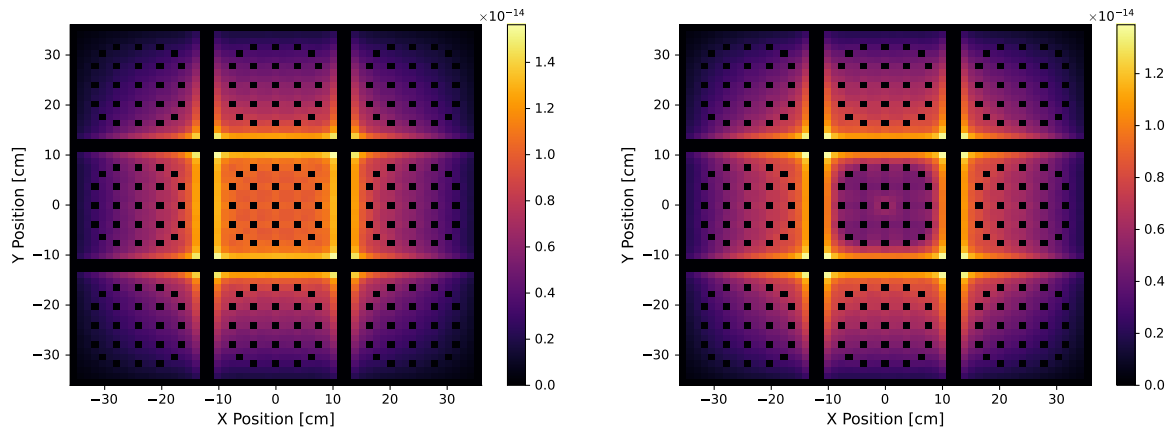
Figure 7.1: Cluster geometry for the rod drop transient case. Colors: blue: borated water, red: fuel, white: AIC control rods, black: control rod cladding, grey: zircaloy fuel and guide tubes cladding.

effect of the Russian Roulette before running the calculation). And second, modeling a single assembly with a leakage condition would have led to a strongly subcritical configuration, which would not have been relevant for a time-dependent analysis.

To grasp an idea of the power distribution that is expected, the power distributions before and after the rod drop were computed using SERPENT2 criticality mode and are displayed in Figure 7.2. As expected, highly absorbing control rods distort the distribution which was initially peaked at the center of the cluster, albeit in a symmetric way.

The power and flux distributions were monitored over 10 milliseconds for all simulations described in Table 7.1. Three AMS cases were performed to evaluate the impact of the importance map regarding the power distribution over time. Case *AMS alpha* uses an importance equal to the time, as in previous chapter, while *AMS central* and *AMS surround* maps are based on the adjoint point kinetics. These two cases only differ regarding the detector defined when computing the space and energy dependent adjoint flux $\psi(\mathbf{r}, E)$. For the former, namely *AMS central*, more importance is given to particles leading to tallies regarding the power in the central assembly. Whereas, the latter favors particles contributing to the power in the peripheral assemblies. More details about the simulation parameters for the computation of these importance functions are presented in the following subsection.

For all these simulations, 10^3 independent batches were performed, with $N_0 = 10^5$ initial particles per batch to avoid large empty areas. SERPENT2 default parameters regarding precursors storage were used, i.e., $10N_0$ precursors were initially sampled and the coefficient w_{thresh} in Equation 6.13 was left equal to 1. Apart from what is listed in Table 7.1, default SERPENT2 parameters were used. For the branchless


 (a) Before the rod drop : $t < 0$.

 (b) After the rod drop : $t > 0$.

Figure 7.2: Power distribution for the assembly cluster case retrieved from a k -eigenvalue calculation with SERPENT2 (baseline, v2.1.32) before and after the rod drop. Power is displayed with arbitrary units, the default normalization was used in both cases (unit total loss rate).

Table 7.1: Simulation parameters for the cluster rod drop simulation.

Case	AMS	Importance	Branchless collision	Population control
Analog	no	-	no	no
Branchless	no	-	yes	every 1 ms
AMS alpha	yes	t	yes	-
AMS central	yes	$N^*(t)\psi_{central}^*(\mathbf{r}, E)$	yes	-
AMS surround	yes	$N^*(t)\psi_{surround}^*(\mathbf{r}, E)$	yes	-

collision method, it means that the weight windows are equal to $[0.2\bar{w}; 10\bar{w}]$. Running the transient over 10 ms allowed to obtain relatively small statistical uncertainties with this number of batches and particles per batch, thus limiting the need for high performances computational resources.

7.2.2 Adjoint flux calculation for the rod drop transient

ADVANTG : An Automated Variance Reduction Parameter Generator

To compute the importance shape function $\psi^*(\mathbf{r}, E)$, the code ADVANTG [144] was used. It is developed and maintained by Oak Ridge National Laboratory to serve as a generator for variance reduction parameters based on the neutron and photon importance for MCNP [116]. More specifically, it is the package Denovo [42] that was used to compute the adjoint flux for our problem. Denovo is a deterministic solver using the discrete ordinates (S_N) method [35] to solve the steady-state adjoint Boltzmann equation for neutrons. In our case, there are three major arguments in favor of using ADVANTG. First, it is a deterministic solver which simplifies the generation of importance based weight window parameters. At the completion of the calculation, ADVANTG directly provides what is commonly used in MCNP as the lower bound of the weight window in each spatial and energy mesh, which is proportional to the inverse

of the adjoint flux (i.e., $\propto 1/\psi^*(\mathbf{r},E)$ in our specific case). Secondly, ADVANTG can extract information from a MCNP input dataset to build the geometry and materials, and since it is possible to convert a SERPENT2 geometry into a MCNP quite easily, it simplifies the problem definition. Finally, ADVANTG writes WWINP files which are directly readable by MCNP, but also by the SERPENT2 code. Hence, without much modifications of the SERPENT2 source code, it was possible to reroute the information read by SERPENT2 for its weight window into the AMS method to be used as a spatial and energetic importance shape function.

Computation of $\psi^*(\mathbf{r},E)$ with ADVANTG/Denovo

As mentioned earlier, two different importance maps were used, $N^*(t)\psi_{central}^*(\mathbf{r},E)$ and $N^*(t)\psi_{surround}^*(\mathbf{r},E)$. The static parts of the maps were generated over a 572×572 spatial mesh covering the entire cluster in the configuration after the rod drop (see Figure 7.1b)⁶, over 47 energy groups. The multigroup energy structure is the BPLUS library displayed in Table D.1. The two maps were compared to observe effects due to the definition of the detector used to compute the importance on the spatial variance reduction during the transient.

The first static importance map, namely $\psi_{central}^*(\mathbf{r},E)$, was computed for a detector defined as the pinwise fission rate integrated over all fuel pin cells of the central assembly. To be more specific, it is the product of the neutron flux in the central assembly UOX material multiplied by the material fission cross section.

The second static adjoint flux, noted $\psi_{surround}^*(\mathbf{r},E)$, was computed using the same methodology, but integrated over all assemblies excepting the central one.

Regarding the spatial distribution, $\psi_{central}^*(\mathbf{r},E)$ and $\psi_{surround}^*(\mathbf{r},E)$ are displayed for energy groups 7 and 32 in Figures 7.3 and 7.4. It is possible to distinguish cells for low-energy neutrons while this is not the case for high-energy neutrons (this is due to the mean free path being higher in the high-energy groups than in the low-energy groups), but other than that, the spatial distribution of importance does not seem to change substantially with energy. In the case of $\psi_{central}^*(\mathbf{r},E)$, the importance distribution is more spread out than in the case of $\psi_{surround}^*(\mathbf{r},E)$ because the distance between the cells farthest from the detector is greater. Thus, for a similar gradient, the distance between the cell of lowest and highest importance will be larger.

This is verified in Figure 7.5, which displays the average value in space of $\psi_{central}^*$ and $\psi_{surround}^*$ for each of the 47 energy groups along with the minimum and maximum values. Both functions tend to behave approximately the same in energy, albeit the spatial importance distribution is more spread in the case of $\psi_{central}^*$. As the energy increases, the average value of $\psi_{surround}^*(\mathbf{r},E)$ in space remains quite stable and tends to homogenize due to a higher average neutron free path than in the slow domain. Consequently, the energy parameter does not seem to play a major role here.

We chose a reactor like case (i.e., monitoring the power distribution over a large area), but many different detectors could have been modeled. For example, a different importance gradient could have been obtained by defining a more localised detector such as a reaction rate within a single pincell (e.g., like a mobile fission chamber such as those used in the nuclear industry). More heterogeneous assemblies (e.g., different enrichments, MOX, ...) could have produced more heterogeneous maps too.

⁶Since our SERPENT2 geometry was defined in two dimensions and Denovo expects a three-dimensional geometry from the MCNP input file, one mesh was defined along z -axis.

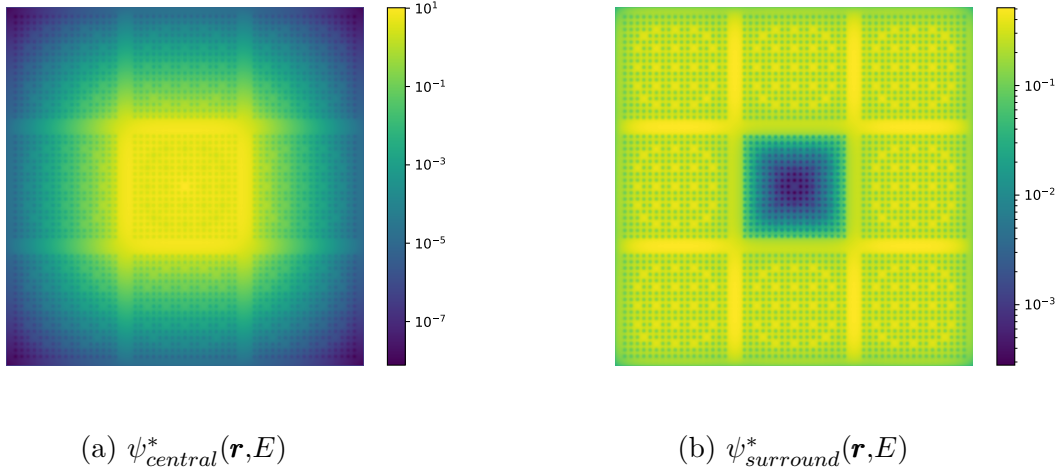


Figure 7.3: Spatial importance distribution in energy group 7 ($37.3\text{eV} < E < 101\text{eV}$) for the assembly cluster with AIC control rods.

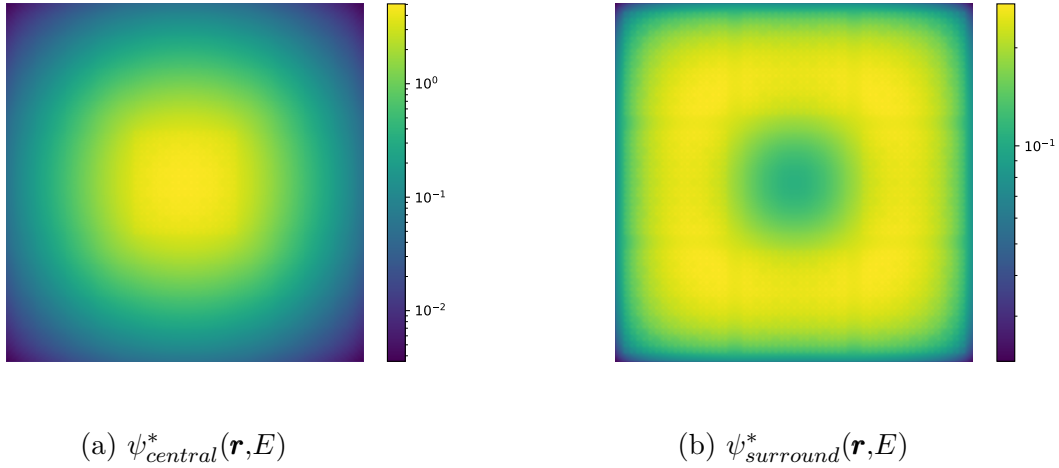


Figure 7.4: Spatial importance distribution in energy group 32 ($1.92\text{MeV} < E < 2.23\text{MeV}$) for the assembly cluster with AIC control rods.

Computation of the adjoint point kinetics solution

As explained before, in this particular case, the time-dependent behavior of the importance is entirely driven by the solution of adjoint point kinetics equations, denoted $N^*(t)$. For both cases (AMS central and AMS surround), the same function was used to represent the temporal evolution of the importance. The amplitude of the adjoint flux over time, retrieved from adjoint point kinetics equations, is displayed in Figure 7.6 for two cases: one with delayed neutron precursors, and one without precursors. The case without precursors was also computed because precursors are not tracked by the AMS yet. Indeed, since it is not possible to rely on precursors importance to re-sample these particles, only the neutron importance is taken into account. As one can see in Figure 7.6, the neutron importance is constant over time for a few milliseconds if precursors are taken into account when solving the adjoint equations, and only contribute at the end of the transient. This is due to the short lifetime of fission chains. Delayed

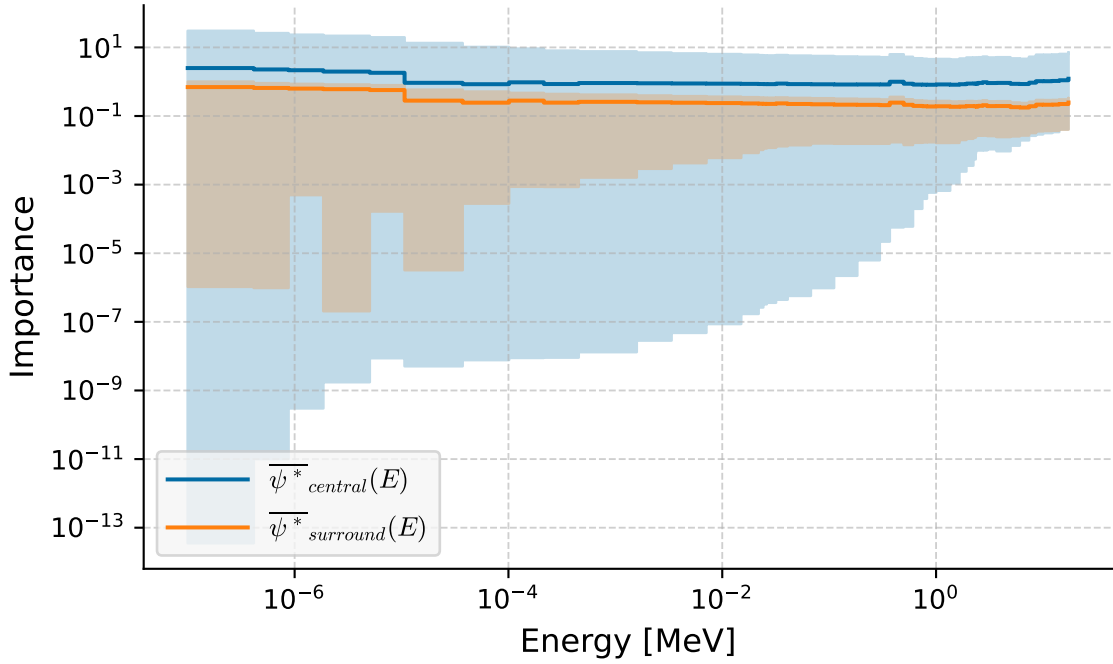


Figure 7.5: Spatially averaged importance in each energy group for the assembly cluster with AIC control rods. Solid line represents the average while shades delimit the min and max of the importance in each energy group.

neutrons precursors, which live much longer, are then more important for a long time before reaching the detector as they act as a neutron reservoir over time. The constant importance value over the first part of the transient implies that only $\psi(\mathbf{r}, E)^*$ would be used to discriminate neutrons for most of the transient if precursors were taken into account when solving the adjoint point kinetics equations (i.e., $\beta \neq 0$). Thus, to avoid a stagnation in time of the neutron pool (since they would only be re-sampled according to $\psi(\mathbf{r}, E)^*$ and not particularly pushed over time) and increase the effect of the time-dependent part of the importance, precursors were ignored to compute $N^*(t)$.

In the next section, results obtained for the different SERPENT2 calculations are presented. The effects of calculation parameters presented in Table 7.1 are compared over space and time to assess the effect of the AMS method on the variance estimate, and the impact of different importance maps described above.

7.3 Numerical results

As previously said, the power during the transient was monitored over 10 milliseconds. Two types of results were computed:

- the spatially integrated power over time, which was compared to the direct point kinetics solution for which effective parameters and reactivity were retrieved from a static k -eigenvalue calculation,
- spatial distribution of the power, and the neutron flux, over time.

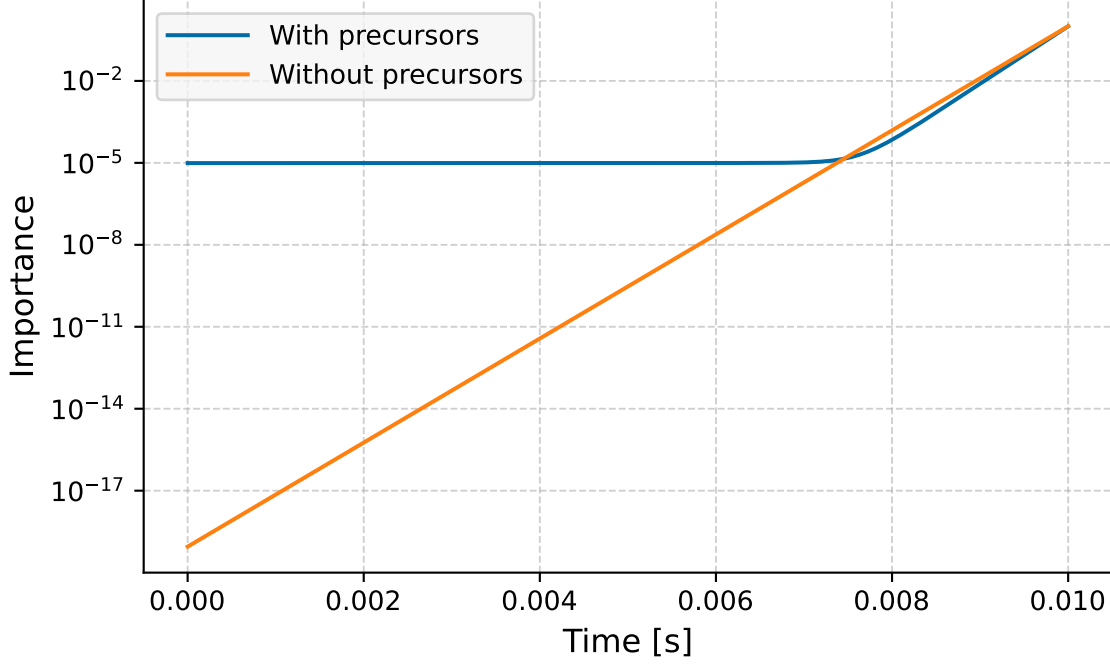


Figure 7.6: Solution of the adjoint point kinetics neutron equation for the control rods drop subcritical transient with and without precursors.

7.3.1 Impact of the AMS on the power over time estimate

The spatially integrated power results are shown in Figure 7.7, and regroup the power over time, its relative uncertainty and the Figure of Merit (FoM). The right column displays the same results but for $t > 1$ ms with a linearly scaled time axis. All Monte Carlo results seem coherent with point kinetics results which are included in all the 3σ confidence intervals for the whole transient. A prompt drop of the power occurs between $t = 0$ ms and $t = 1$ ms, during which the relative variance increases until it stabilizes once the power seems to have reached a steady state. Actually, the power is not really constant after 10 milliseconds since the reactivity is still negative, but because of the delayed neutrons, it appears as such because of the time scale. The variance seems to stabilize after the prompt drop.

Regarding SERPENT2 reference calculations (namely *analog* and *branchless* cases), the analog case presents a higher FoM. Between $t = 10^{-4}$ s and $t = 10^{-3}$ s, both calculations behave almost the same in terms of statistical uncertainty. Indeed, the branchless collisions method has low impact for subcritical systems [15], and the population control has not started yet. As a reminder, the weight window parameters were left to their default value, and therefore not optimized for this specific case. Since the population control injects extra particles every millisecond in the branchless case (see calculation parameters in Table 7.1), the overall calculation time is expected to be higher. As a reminder, only the final computation time is used to compute the FoM. This results in a lower FoM for the first part of the transient (between 10^{-4} and 10^{-3} s) in the branchless case, compared to the analog case. As for the teeth that can be observed every millisecond, they are caused by a decrease of the variance due to population control, as in the subcritical system the population control increases the number of particles, and more particles induce less variance. But since it takes about 1 ms for the prompt

drop to fully occur, and the largest variations of the variance occur over this period, population control does not allow for significant improvements regarding the FoM. For this reason, the analog case was kept as a reference to compare later results.

As for AMS calculations, the error on the average results is higher (and the FoM lower) than other SERPENT2 calculations until the end of the transient, where the uncertainty drops. For the last millisecond of the transient, the FoM becomes greater than the analog FoM up to a factor 10. This result is globally positive, as the FoM was improved where the detector was defined from a temporal point of view. As for the FoM along the transient, one would have to define a more time extended detector and see what the impact would be. To take the analysis further, it would also be possible to simulate exactly the same transient, but longer, and see how the FoM would behave if we always defined the detector at the end of the transient. It could then be seen if the FoM would continue to increase after 10 ms, or if it would be lower than in the other simulations except in the last milliseconds of the new transient.

It appears that using a "physical" importance map (in our case using ADVANTG maps described earlier as spatial and energetic shape of the importance to mimic an approximate time-dependent adjoint flux) does not allow a better FoM and lower error compared to the time-only importance map. This is represented by the case AMS alpha, which performs better than cases AMS central and AMS surround over the whole time interval. This could be due to the fact that the importance maps that were chosen were not ideal adjoint fluxes and may be less efficient than just pushing neutrons through time.

7.3.2 Impact of the multi-parameter importance map on the FoM degradation

The FoM is computed from the variance of the score and the computation time necessary to compute this score. To understand how using spatially and energy-defined shapes led to poorer performances regarding the FoM, details on calculation times and sampled collisions were retrieved and displayed in Table 7.2. The five columns represent the total CPU time of the simulation, the time spent sampling particle flights, the number of flights sampled, the fraction of these flights leading to real collisions (a fraction of the total number of flights may end in a virtual collision due to delta tracking, to a leakage or a boundary crossing), and finally the fraction of collisions that are scored. For SERPENT2 baseline calculations (i.e., branchless and analog cases), 100% of real collisions contribute to the score estimate. It is expected since there is no reason not to tally a collision (a value lower than 100% should raise questions regarding the implementation of the scoring). However, not all collisions may lead to increased statistics in the different AMS cases due to the on-the-fly scoring procedure described in Section 4.2.4.

It appears that, as mentioned earlier, the branchless case has a much higher calculation time than all other cases due to extra particles that are re-sampled due to population control. But the most interesting finding, is that only 46% of collisions that occurred during the simulation were used to compute tallies when $\psi^*(\mathbf{r}, E)$ was used, compared to 96% for the AMS alpha case, and 100% in classical SERPENT2 kinetics calculations. It means that in AMS with a physical importance map more than 50% of collisions required calculation time, but were not tallied, which resulted in a poorer FoM. This phenomenon is due to the AMS *on-the-fly* scoring procedure

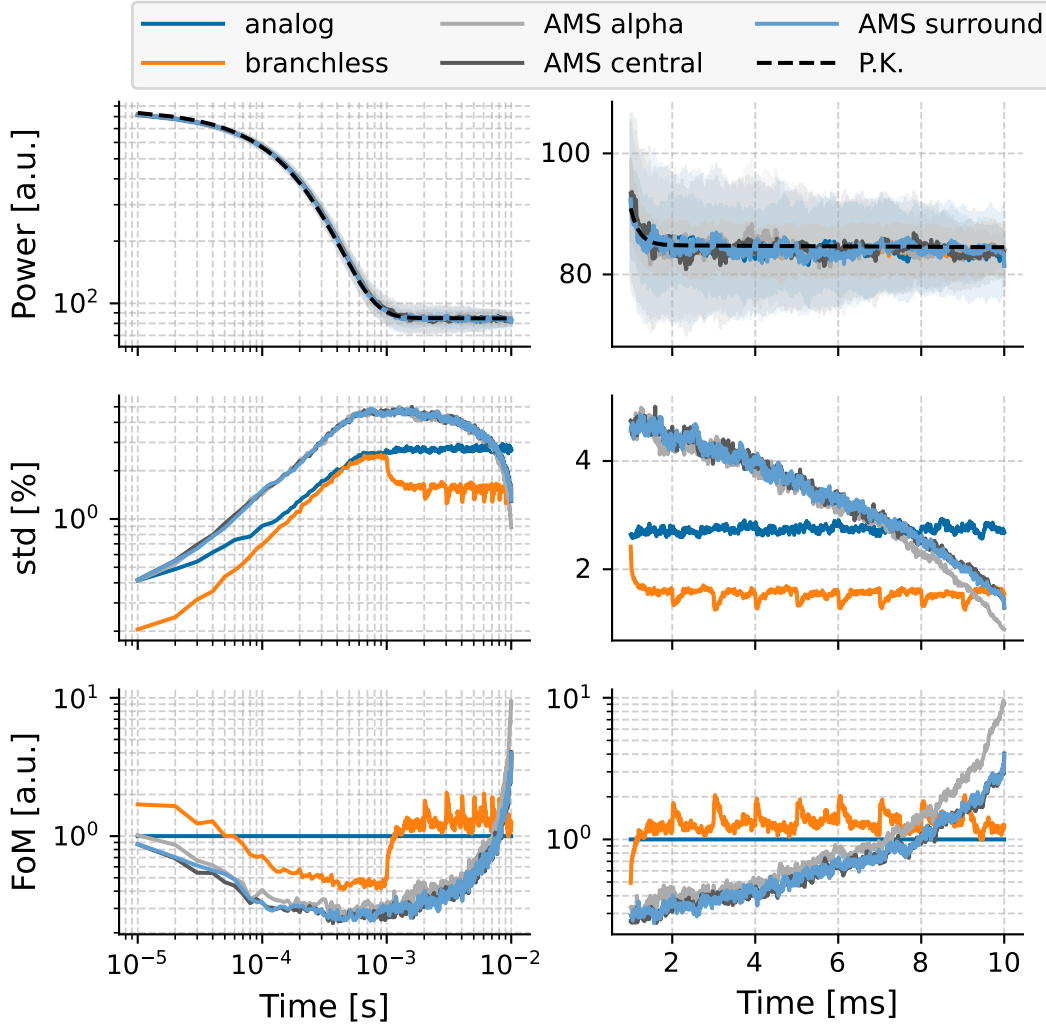


Figure 7.7: Power over time for the UOX cluster rod drop. On the right column, the power with 3σ confidence intervals, the relative standard error of the mean and the normalized FoM ($FoM(analog) = 1$) are displayed over a 10 ms time interval (log-scale). On the left column, the same measures are presented between 1 and 10 ms (linear scale)

briefly described in Section 4.2.4, and was mentioned earlier in the context of criticality calculations (see Section 5.1.4, more particularly Figure 5.3). Since the importance map has a spatial dependence, as well as a time dependence, some particles may be re-sampled in other regions of the geometry, but in the past. But since scoring volumes are defined as entire time bins (i.e., the entire space and energy phase-space in a particular time-bin defines a unique scoring volume), newly re-sampled neutrons might not collide in active scoring volumes for a few collisions (they might even disappear before contributing to any scoring). These collisions are lost for scoring, which results in values lower than 100% (50% for *AMS central* and *AMS surround* cases) in the last column of Table 7.2.

This phenomenon is very seldom for the *AMS alpha* case since particles may not be re-sampled in the past (in this specific case, the importance is strictly equal to time). Whereas, it is much more prominent in the *AMS central* and *AMS surround* cases since

the spatial shape of the importance may lead to re-sampling particles at earlier time than their time of death. To prevent this issue from happening, it could be beneficial to define a finer scoring volumes mesh, including a spatial discretization on top of the time bins. Note that this solution has not been tested yet, and might be part of future improvements. This problem did not occur in the simulations presented in Chapter 5 because the space-dependent part of the importance was kept between 0 and 1, as the generational part (equivalent to a discrete time dimension) was incremented by 1 at each generation (i.e., it was not possible to re-sample a particle in the past compared to the moment of its death).

Table 7.2: Computation times and fraction of sampled flights leading to real collisions and scored collisions over the whole transient for the cluster rod drop simulation. Values in the *collisions* column represent the percentage of sampled flights leading to the sampling of real collisions. Values in the *scored* column represent the percentage of real collisions that are effectively scored⁷.

Case	Total time [CPU.min]	Transport time [CPU.min]	Number of flights	Collisions [%]	Scored [%]
Analog	1019	1014	8.71×10^6	39.4	100
Branchless	2436	2431	3.39×10^7	35.9	100
AMS alpha	1013	1009	2.16×10^7	35.5	96.9
AMS central	1152	1146	2.62×10^7	36.2	46.4
AMS surround	1143	1137	2.60×10^7	36.2	46.7

To make the parallel with a case of spatial attenuation with AMS, it would be like the importance map would be multi-parameter (energy and space), and that re-sampling neutrons to a more interesting energy would lead to them being moved spatially away from the detector. If scoring-volumes were closed along the path between the location of the newly sampled particle and the detector, the particle would have to travel in regions without contributing to the score in them. This phenomenon does not affect the last time bin, because it is closed only during the final iteration of the AMS, i.e., there cannot be any non scored collision in the target volume, which is here the last time bin.

7.3.3 FoM gains regarding the power distribution in the last time bin

Nonetheless, our initial goal was to improve local tallies at the end of the transient⁸. With that in mind, the FoM was computed in each spatial bin of each one of the 1000 time intervals. The distribution of the ratio $FoM(case)/FoM(analog)$ was plotted in Figure 7.8 for the very last time interval only for AMS cases. It appears that all AMS cases allowed to increase the FoM in all spatial bins at the end of the transient (all distributions start for a FoM ratio higher than 1). Once again, the AMS alpha

⁷The AMS on-the-fly scoring procedure can lead to collisions that are not taking into account for the scoring, to avoid duplication of contributions to a score (cf. Sections 4.2.4 and 5.1.4 for more detail).

⁸The detector was defined for the final time of the transient, which makes it the time of interest regarding FoM comparisons. It would be interesting to test different definitions of the detector to look at a time range during the transient for example.

produced better results, with a FoM up to 40 times better than the reference calculation (it even goes up to 70 for very few spatial bins).

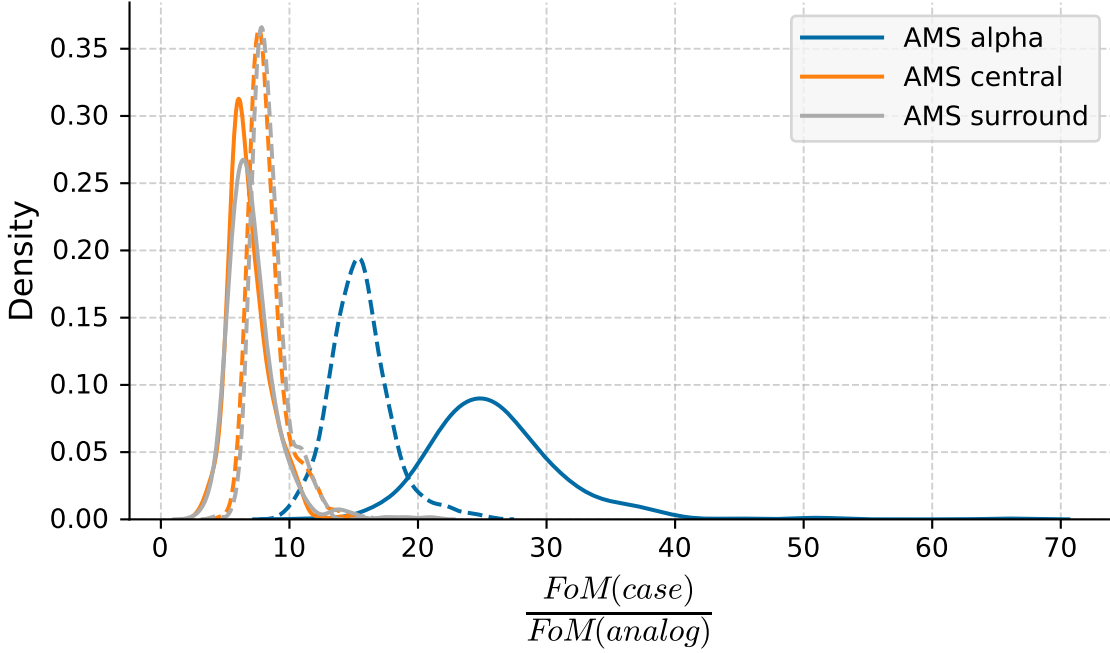


Figure 7.8: Distribution of the relative increase of the spatial FoM at $t = 10^{-2}$ s for the cluster rod drop simulation for the 3 AMS calculations. Solid lines represent the FoM increase for the power distribution while dashed lines represent the FoM increase for the flux distribution.

However, there is a worth mentioning difference between the behavior of the cases with and without multi-parameter importance with regard to the difference between the FoM on the flux and the FoM on the power. For the *AMS central* and *AMS surround* simulations, the FoM gain is higher for the local flux than for the local power. This means that using $\psi_{central}^*$ and $\psi_{surround}^*$ led to a higher increase of the FoM for the fast neutron flux than for the thermal neutron flux. Indeed, fissions are mainly induced by thermal neutrons in the modeled system. It is a bit surprising, and at the same time, Figure 7.5 indicates that the minimum importance increases with energy. This implies that there are more regions of low importance in low energy compared to high energy groups, and therefore new tracks have a lower probability to be re-sampled as thermal neutrons than fast neutrons. This could explain this rather unexpected result (as a reminder, detectors that were defined in the ADVANTG simulations were both fission rates). The opposite behavior is observed for the *AMS alpha* case. Thus, just pushing the neutrons in time seems to improve the estimation of the thermal flux more than the fast flux in this specific case.

Finally, Figure 7.9 shows a comparison between the FoMs of the AMS central and AMS surround cases over space in the last time bin. No proper pattern seems to stand out as results remain noisy. From results presented here, changing from a purely time-dependent importance map to a more complex adjoint flux seems to worsen the performances of the AMS. However, it is important to put these results in perspective of the case studied and the maturity of the AMS implementation in SERPENT2. The fact that scoring volumes are still defined as whole time bins did not allow cases *AMS*

central and *AMS surround* to reach their full potential. Besides, the adjoint fluxes computed with ADVANTG did not present strong gradients for more than a few mean free path, and were not strictly equal to the space/energy distribution of a true time-dependent adjoint flux. Defining much more specific detectors rather than the fission rate in an entire assembly, or modeling more heterogeneous systems, may highlight differences in performance between different importance maps.

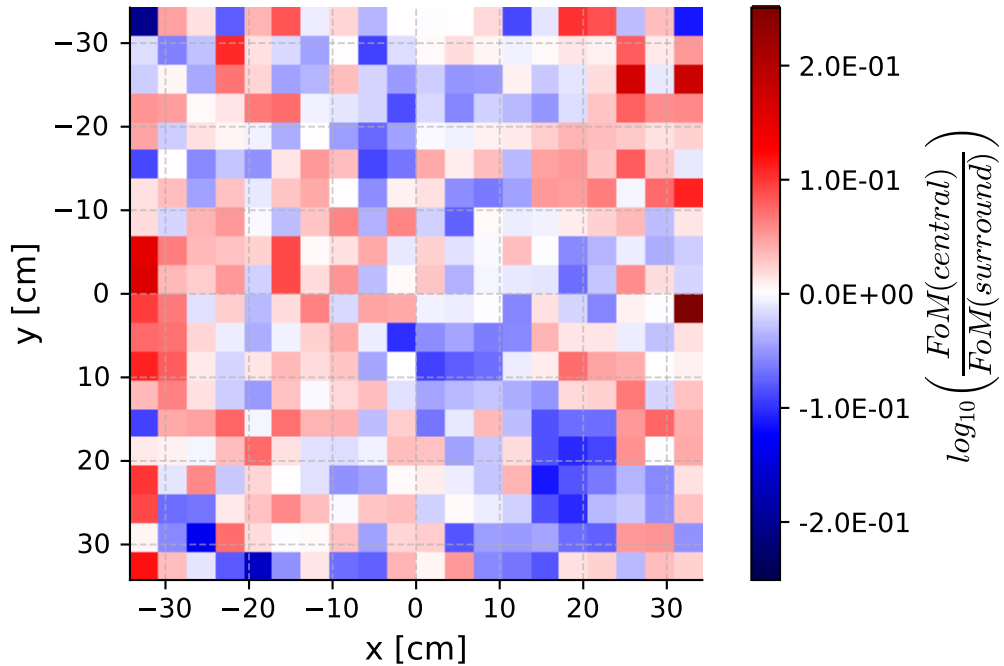


Figure 7.9: Ratio of the FoM obtained with 2 different spatial importance maps for the cluster rod drop simulation (detector is either central assembly fission rate or surroundings assemblies fission rate). The log of the ratio is displayed and a value of zero means that both FoM are equivalent. Red bins mean that the case *AMS central* has a better FoM at these locations that the case *AMS surround* (blue bins mean the opposite).

Overall, it was shown that it is possible to compute the power distribution over a sub-critical transient for a heterogeneous assembly cluster using the AMS method. This method allowed to increase the FoM for scores tallied at the end of the transient, over the whole space. Regarding that specific point, it might be interesting to use the AMS as a way to efficiently decrease the variance just before operating a coupled calculation with different physics codes requiring tallies from the Monte Carlo calculations. Finally, improvements of the on-the-fly the scoring should be implemented so further investigations could be conducted, especially regarding the importance map that is used.

Conclusion

In this chapter, the AMS has been successfully applied to a fuel assembly cluster in the context of a subcritical transient initiated by control rods drop. The aim of these calculations was to demonstrate the feasibility of transient calculations on more realistic systems than homogeneous geometries or mono-energetic configurations. As far as subcritical transients are concerned, it was shown that the AMS allowed for substantial improvements of the FoM (around a factor 25) compared to the analog regarding the spatial distribution of the power at the end of the transient, i.e., *where* our time detector was defined.

However, another objective was to characterize the contribution of the importance map to these improvements by using different importance functions. To do so, some AMS calculations relied on multi-parameter importance functions by including energy and space in addition to time in the importance function. In the first part of this chapter, the link between the concept of neutron importance and the solution of the adjoint problem was presented. The idea was to motivate the use of external solvers, for example relying on deterministic methods, to compute adjoint flux estimates allowing intuiting the optimal importance function. Since deterministic codes are not able to compute time-dependent adjoint flux for the moment, an estimation based on the variable separation of the point reactor approximation was computed. To retrieve the space and energy dependent part of the importance, the solver Denovo from the code ADVANTG was used. Two different importance maps, differing only by the defined detector, were computed. The obtained results did not show significant improvements, and these calculations even performed worse than a simple AMS calculation with time as the importance, as used in the previous chapter. Multiple reasons might explain these results. Firstly, regarding results along the transient, scoring volumes on which the *on-the-fly* scoring procedure used by the AMS to compute unbiased scores outside the detector are still space-independent. As a matter of fact, each volume represents an entire time bin, without distinction between different parts of the geometry. Due to this feature, a lot of collisions are simulated by the Monte Carlo algorithm but are not used to estimate tallies, which resulted in a poor FoM. Then, importance maps retrieved from ADVANTG seemed different enough in terms of adjoint flux distribution over space, but might not be sufficient to lead to significant differences regarding the spatial FoM.

Following these observations, several perspectives arise. With regard to the two issues that were just mentioned, it could be of interest to implement more complex scoring volumes into the AMS framework to take into account both space and time and reduce the number of wasted collisions. To focus on the importance map itself, re-engaging with more localized variance reduction would probably be easier than to tackle global variance reduction. Going from one or several fuel assemblies to a more localized detector might induce more differences regarding the spatial FoM. To do so, it would not be necessary to modify the AMS, but only the detector which is defined to compute the adjoint flux. In the end, it would be profitable to work on a more accurate time-dependent adjoint flux, free from the approximations we made, by extending deterministic solvers capabilities to adjoint kinetics. Regarding the test case presented here, it is necessary to further characterize the method by modeling longer transients, as well as supercritical systems. Due to strong interactions between biasing methods in kinetics, studying the impact of simulation parameters (such as the number

of resampled tracks K , or the width of the weight window) is also recommended to fully identify the capabilities and application scope of the AMS.

Chapter 8

Prospects regarding consequences of neutron transport's stochasticity on computational schemes

We live on an island surrounded by a sea of ignorance. As our island of knowledge grows, so does the shore of our ignorance.

- John Archibald Wheeler

Contents

8.1	Application of the AMS to neutron noise	152
8.1.1	Zero-power neutron noise	152
8.1.2	AMS and correlations	154
8.2	Stochastic fluctuations in time-dependent neutronics	154
8.2.1	Distribution of the number of particles at time t and power law	154
8.2.2	Detecting a power law in time-dependent neutronics calculations	155
8.3	Thoughts on the relevance of coupled calculations in the case of dynamics Monte Carlo calculations	156

Monte Carlo simulations aim at estimating the average response for a given detector. Due to random samplings performed by Monte Carlo methods, these average estimates are always associated with a measure of statistical uncertainty. The purpose of variance reduction techniques is to reduce the variance of Monte Carlo estimators (i.e., the uncertainty), without modifying the first statistical moment of those estimators, i.e., the average value. While these methods allow to overcome some difficulties arising from numerical aspects of the simulation (e.g., the particle clustering phenomenon addressed in Chapter 5), they distort the real problem one seeks to model. If one is interested in

the average value of a score, variance reduction should not be a problem given that unbiased variance reduction schemes should conserve this average value compared to the real system. For some applications, however, being able to compute real fluctuations is of paramount importance, as in zero-power neutron noise simulations, for example. In that case, a pure analog simulation of processes involved in neutron transport is required. For most Monte Carlo simulations, pure analog simulations are not currently performed outside the scope of noise calculations since methods such as population control are often used to make numerical simulations possible.

In this chapter, a short discussion on stochastic fluctuations inherent to the very nature of branching processes is carried out with the objective of presenting some ideas for Monte Carlo neutron noise and dynamics calculations. This part is merely intended to outline perspectives regarding the use of the AMS method in other contexts than those already mentioned in this thesis, as well as presenting thoughts on dynamics Monte Carlo calculations.

8.1 Application of the AMS to neutron noise

It has been observed in Chapter 5 that a combination of an appropriate population control technique and the branchless collision method allowed to greatly reduce spatial correlations in the case of criticality calculations, and thus reduce the neutron clustering phenomenon (cf. Section 5.2.3.3). In Chapter 7, the variance of the integrated power estimator was also decreased using the combination of the AMS and the branchless collision method. As mentioned above, changing correlations and variance meant changing statistical moments (except for the mathematical expectation) of the Markov chains that represent neutron histories. Thus, stochastic fluctuations of the real system were modified. Conserving the natural random behavior of the system yet allows to gain information on its dynamic properties through noise analysis.

While the branchless collision method was designed to reduce the variance due to fission chains (i.e., inherent to the stochastic nature of neutron transport), the AMS was used in this thesis to tame correlations due to neutron population control (i.e., numerically induced). In previous work, the AMS was successfully used to compute non-Boltzmann scores in the context of gamma-ray-spectrometry using the fixed-source method [7]. For this application, it was necessary to conserve natural correlations between secondary particles arising from splitting events (in this specific context, photons, electrons and positrons) so as to ensure a correct calculation of the energy deposit in the detector.

Unlike the work presented here until this point (i.e., the objective was to compute an average behavior of the reactor), this section aims at giving some thoughts on the perspective of using the AMS to further study the stochastic behavior of multiplicative systems through zero-power neutron noise calculations.

8.1.1 Zero-power neutron noise

The average power output of a stationary fissile system is constant over time. While the actual power output of a deterministic system follows the same behavior, real systems see their power fluctuate over time as illustrated in Figure 8.1.

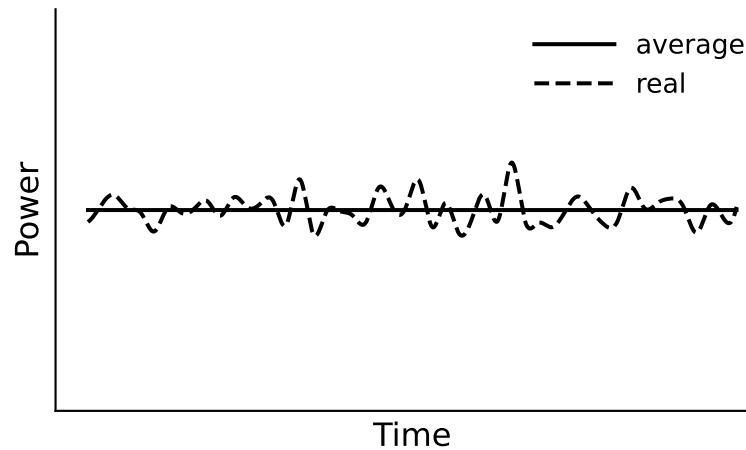


Figure 8.1: Illustration of neutron noise in a critical reactor.

For zero-power systems¹, the system's noise is solely driven by the stochasticity of branching processes (e.g., fission chains). For power reactors, on the other hand, noise may arise from temperature oscillations or mechanical vibrations inducing fluctuating feedback effects on the neutron transport. The following part only focus on the so-called *zero-power* neutron noise.

The analysis of neutron noise at zero power can be motivated by different problems, from the characterization of unknown nuclear waste [146] to reactivity measurement [147]. In the first case, one will try to characterize the contents of the cask without being able to open it, based on the intrinsic noise of the system or under the effect of an external neutron flux. The second case mentioned above makes use of the modification of the noise generated by a change of the system's reactivity.

Zero-power noise analysis may be conducted using multiple observables, such as the Feynmann variance to mean whose estimator is defined as $Var(N)/\langle N \rangle$, where $N(t)$ is the number of neutrons detected and $\langle N \rangle$ is the average value of that number over a time period. While this specific estimator requires a conservation of the first and second order statistical moments of the particle distribution in the system, the ideal setting would be to conserve the natural distribution itself.

In practice, the numerical study of zero-power neutron noise is quite similar to a fixed-source calculation, except that all collisions must be analog and no variance reduction scheme should be applied. Since local detectors are often considered to compare results to real measurements, tallies of interest are localized. Thus, to get enough statistics inside the detector, it may be necessary to simulate a lot of particles.

Just as in some radiation protection applications, for which the fraction of the total number of particles reaching the detector is quite low, zero-power neutron noise simulation could benefit from using simulation methods to increase the number of detected particles, provided that they do not alter the natural fluctuations one seeks to measure.

¹A system may be labeled as *zero-power* if neglecting temperature feedback effects when modeling it does not lead to significant discrepancies with the real system.

8.1.2 AMS and correlations

As mentioned earlier, the AMS primal mechanism acts as a Fleming-Viot particle system in which the system is conditioned to its survival in order to simulate a rare event with enough statistics. Thus, while the AMS is used to decrease the variance associated to the estimate of the mean score in the case of a rare event, the probability distribution of the physical process is preserved, given that the system is alive.

Indeed, the AMS as presented in Chapter 4 does not introduce particle weights in the case of a uniform sampling of the new tracks. Due to that particularity, natural correlations between particles in the same iteration are preserved. As a matter of fact, the correction performed using the iteration weight defined by Equation 4.12 applies a global weight to the whole population of N tracks. As a reminder, this weight at iteration q is given by

$$\alpha_{\text{AMS}}^q = \prod_{i=1}^q \left(1 - \frac{K_i}{N}\right) \quad (8.1)$$

where N is the total number of tracks (i.e., particle histories) and K_i is the number of tracks that are resampled by the algorithm at iteration i .

The AMS method could then be used for noise simulation in the case of attenuation problems, typically in the case of localized ex-core detectors.

Besides direct application of fluctuations measurements, the stochastic nature of neutron transport also raises interest to grasp differences between simulations and reality, and quantify them if necessary, as illustrated in the subsequent sections.

8.2 Stochastic fluctuations in time-dependent neutronics

In the work presented in this thesis, the random variables representing the number of neutrons in the system at time t , noted X_i , were supposed to be independent and identically distributed, and have a finite variance. But the study of the critical catastrophe (cf. Section 1.3) shows that while the average number of particles is preserved, the population almost surely dies. Thus, it indicates that at large times, the probability to have n neutrons at time t noted $P(n,t)$ is rather pathological with regard to the physics. Indeed, for the classical Central-Limit Theorem (CLT) to hold, and to properly compute an average with Monte Carlo methods, the variance has to be finite, i.e.,

$$\sum_{n=0}^{\infty} n^2 P(n,t) < \infty. \quad (8.2)$$

However, in the case σ^2 would be infinite, the classical CLT could not be applied, and a generalized version of this theorem would be of need.

8.2.1 Distribution of the number of particles at time t and power law

Considering a probability density function f whose tail follows a power law of parameter α , i.e.,

$$f(x) \propto \frac{1}{x^\alpha}, \quad \forall x > A \quad (8.3)$$

where A is a threshold above which the tail may be approximated by the power law of parameter α . For the sake of simplicity, only the case $f(x < 0) = 0$ is considered here. The variance associated with such probability density function is equal to

$$\sigma^2 = \int_0^\infty x^2 f(x). \quad (8.4)$$

For $x > A$, one has

$$\int_A^\infty x^2 f(x) \propto \int_A^\infty \frac{x^2}{x^\alpha} = \int_A^\infty x^{2-\alpha}. \quad (8.5)$$

For $\alpha > 3$, this integral converges. For $\alpha \leq 3$, however, the variance σ^2 is infinite. In that case, the classical CLT does not hold anymore, and the sum of random variables independently distributed according to f does not converge towards a random variable distributed according to a Gaussian distribution.

While this behavior has not been observed in the work presented in this thesis, questions arise in the case of time-dependent population fluctuations. Indeed, it was shown that the total number of particles in a fission chain may be described by a power law of parameter $\alpha = 3/2$ [34]

$$P(n) \propto n^{-3/2}. \quad (8.6)$$

In that case, even the average (i.e., the first statistical moment) is not finite. While here the number of particles is integrated over generations, this observation questions the validity of the CLT when studying a fluctuating population of neutrons over time.

As a reminder, the critical catastrophe presented in Section 1.3 illustrates a practical case in which only one simulation has survived (in the sense that there are still neutrons). While it is possible to compute empirical average and variance of the population, they make no sense with regard to the CLT. In practice, population control methods are used to prevent this issue. These methods do not bias the average estimation, but clearly change the dynamic of population fluctuations compared to the real system. Thus, they may hide CLT hypotheses issues.

8.2.2 Detecting a power law in time-dependent neutronics calculations

Deriving a formal expression for $P(n,t)$ for different regimes (subcritical, critical or supercritical) is a cumbersome task. Another approach would be to highlight potential issues by empirically assessing the shape of $P(n,t)$ for high values of n in an analog simulation. Apart from the difficulties linked to analog calculations in dynamics, detecting a power law distribution from empirical data is a non-trivial task [148]. Indeed, despite the fact that such a law is in theory easily recognizable via a log-log plot of its probability density function, a naive fit of the data may not be enough to validate the law as a power law [149] (e.g., some log-normal laws have been mistaken for power laws in the past [150]). Empirically evaluating the tail of $P(n,t)$ is thus complex because as time goes by, the number of independent fission chains decreases due to the critical catastrophe (cf. Section 1.3), leaving few statistics to estimate the tail's shape.

Since observing high values of n is a rare event, investigating the use the AMS to empirically evaluate the tail of the distribution could eventually be worthwhile. As an example, the AMS has already been used in the past to estimate tail probabilities of stochastic differential equation's solutions [113].

Only the impact of fluctuations due to branching processes in kinetics has been addressed here. It is, however, appropriate to place these questions about the validity of current computational schemes in the context of coupled calculations. The following section is based on recent work to raise the question of the potential limits of numerical modeling in dynamics.

8.3 Thoughts on the relevance of coupled calculations in the case of dynamics Monte Carlo calculations

Dynamics calculations aim at solving a strongly coupled set of non-linear equations which describe the behavior of a stochastic multiplicative system in which feedback effects modify the properties of the medium. Indeed, as power increases, local feedback effects increase in intensity due to temperature changes leading to thermal and mechanical effects. The stabilizing role that intrinsic sources may fill for highly sub-critical systems is thus replaced by a local "*population control*" phenomenon due to those feedback effects.

In practice, whether using deterministic or Monte Carlo methods, the simulation of the neutron transport and other equations describing feedback effects are numerically decoupled. First the power due to fission is computed during the neutronics calculation, followed by the deterministic calculation of the changes in temperature and mechanical properties of the system. It is possible to iterate until convergence of a solution, but it does not strictly consist in simulating the parallel flights of neutrons which interact with nuclei of the medium, which in return affect the flight of other neutrons. While this difference may be insignificant with regard to the solution of the coupled transport problem, recent literature shed light on potential discrepancies between the solution of a formal Brownian process with feedback effects, and the numerical eigenvector stemming from a coupled criticality calculation [151]. According to these results, it is possible to show that in the case of neutron diffusion, the diffusion operator defined as

$$\frac{\partial}{\partial t} N(\mathbf{x}, t) = D\Delta N(\mathbf{x}, t) + \rho N(\mathbf{x}, t) + \sigma N_{\text{ref}} N(\mathbf{x}, t) - \sigma N^2(\mathbf{x}, t) \quad (8.7)$$

(where $\sigma N_{\text{ref}} N(\mathbf{x}, t) - \sigma N^2(\mathbf{x}, t)$ model the Doppler effect) shifts towards a super-diffusion operator written

$$\frac{\partial}{\partial t} N(\mathbf{x}, t) = -D(-\Delta)^{z/2} N(\mathbf{x}, t) + \rho N(\mathbf{x}, t) + \sigma N_{\text{ref}} N(\mathbf{x}, t) - \sigma N^2(\mathbf{x}, t) \quad (8.8)$$

with $z = 1.901$ in the one-dimensional case. Hence, the tightly coupling of non-linear effects within the stochastic derivation of neutron diffusion distort the well-known diffusion equation compared to the case where feedback effects are taken into account afterwards, once the average equation has been derived. In that context, current numerical approaches, limited to a macroscopic coupling of neutronics and other multiphysics effects², do not allow to answer the question of the validity of these numerical schemes. To further investigate this issue, a formal theoretical framework might be necessary.

²As opposition to microscopic coupling, in which multiphysics phenomena are taken into account when deriving the neutronics equation itself.

With these simple examples, one sees that Monte Carlo simulations may fail to give the correct model of the real system due to population control in simulations, or the macroscopic coupling between multiphysics codes. While these issues were not witnessed in the rest of the work presented in this document, and may be negligible in most (if not all) current applications, it seems valuable to further investigate discrepancies between numerical simulations and realistic neutron transport theory they aim to model. Indeed, it seems important to assess the representativeness of "*high fidelity*" results produced by state of the art computational methods when applied to safety related inquiries.

Conclusion

Summary of the work presented in this thesis

The goal of this Ph.D. work was to extend the use of the Adaptive Multilevel Splitting variance reduction method to reactor physics calculations. As a first step, the method was extended from fixed-source calculations to steady state k -eigenvalue calculations. The objective in the particular case of k -eigenvalue calculation was to reduce the effects of spatial and temporal correlations that can for example lead to the neutron clustering phenomenon. In a second step, the method was extended to transient calculations in the SERPENT2 code. The work presented in each chapter is detailed below.

Chapter 3 introduced various ongoing issues in k -eigenvalue and kinetics Monte Carlo simulations, motivating the use of variance reduction methods in reactor physics calculations.

While steady-state calculations (i.e., k -eigenvalue calculations) have been studied for decades and do not present a major hindrance in most configurations, the study of loosely coupled systems in the past decade has highlighted potential statistical issues affecting the Monte Carlo simulation, amongst which the neutron clustering phenomenon. Although particle clustering is a natural effect rooted in the asymmetry between neutrons being captured everywhere but being born only close to their parents, its occurrence in power reactors operated in the power regime is a priori impossible, since its magnitude is inversely proportional to the number of neutrons in the system. Monte Carlo simulations, on the other hand, do not allow to simulate a number of particles of the same order of magnitude as in real system. The relatively low number of neutrons, coupled with population control methods used to keep the neutron population stable may in fact increase already existing correlations, thus leading to stronger spatial correlations. Recently, it has been observed that the disappearance of independent lineages of neutrons over generations may be one of the root causes of neutron clustering, especially in Monte Carlo simulations. This observation motivated the use of a variance reduction method to tackle the attenuation of independent families of neutrons over the course of the simulation.

Monte Carlo kinetics calculations are much more recent than steady-state simulations, at least in practice. For years, the computing power of calculation units was not sufficient to perform such calculations with enough statistics to obtain accurate scores (or even finish the simulation). Due to the increasing power of computers, this subject has undergone a renewed interest in the 2010s. Several Monte Carlo codes are now able to run transient simulations taking into account the different time scales of prompt and delayed neutrons. Despite major advances in terms of variance reduction, these

calculations remain very costly in terms of computing time compared to other simulations involved in multi-physics coupled schemes. To go further, it seems necessary to propose new proper variance reduction schemes for these time-dependent calculations.

Chapter 4 was dedicated to the presentation of the Adaptive Multilevel Splitting (AMS) method as it was previously implemented for neutral particle transport in the context of fixed-source calculation. The AMS method was initially intended to increase the number of events leading to a measure of interest in the case of rare events. For particle transport, it was implemented in the TRIPOLI-4® Monte Carlo code for fixed-source calculations (mostly used in the context of spatial attenuation problems such as shielding), to increase the number of particles reaching a remote detector.

The algorithm iteratively re-samples new neutron histories by duplicating existing histories according to an importance criterion. The importance function is user-defined, and should in principle give more importance to the particles that are most likely to contribute to the score. However, it is only used as a ranking criterion by the AMS. It was therefore observed that even with a rough approximation of the ideal importance function, the AMS algorithm was robust enough to improve the estimation of the variance in the detector.

The algorithm has shown valuable results regarding the estimation of rare events in a remote detector, and ease of use / robustness since it exhibits a good efficiency even in the case of a poor knowledge of the optimum importance function. The method, as implemented for particle transport, also allows accounting for branching processes and is able to compute unbiased estimates not only in the detector, but also along the path of particles in intermediate volumes thanks to an *on-the-fly* scoring procedure. These features motivated its extension to steady-state and kinetic reactor physics calculations.

Once the motivations and the method were introduced, AMS's specificities in the context of k -eigenvalue calculations were outlined in Chapter 5. A particular attention has been paid on how to define a generational attenuation problem out of any configuration (even supercritical). To do so, the branchless collision method was used in combination with the AMS so that the number of neutrons cannot be modified by fission or capture and is only affected by leakage (or by artificial removal from the system, such as Russian roulette).

An ad hoc Monte Carlo code, named Korrigan, capable of modeling one-speed heterogeneous (Cartesian geometries) neutron transport problems was developed during this Ph.D., and used to test the performances of the AMS for steady-state reactor physics. It allowed to compare the AMS algorithm to the power iteration based iterative scheme that has been used for decades by criticality codes, and also to estimate its efficiency regarding the reduction of the particle clustering phenomenon.

Both the k -eigenvalue of a homogeneous supercritical bare slab reactor, as well as its associated eigenvector (i.e., the average flux distribution) could be calculated with the AMS which moreover displayed advantageous results regarding neutron clustering. Indeed, its underlying ranking scheme has the capacity to reduce the loss of independent neutron families over time compared to existing population control method. A similar

mechanism might however be in principle obtained with a simpler particle resampling algorithm based on the Fleming-Viot particle system.

The branchless collision method also proved to be a key element for the improvement of performances with regard to both clustering and variance reduction for the k_{eff} and the flux's estimations. This advantage brought by branchless collisions has been confirmed when used in combination with the AMS and other methods.

A heterogeneous one-dimensional fissile slab system was also simulated with the simulation parameters adapted so as to get rid of spatial correlations, as the objective was to assess the performance of the "branchless" AMS in terms of figure of merit, relative to the standard "branchless" power iteration. Due to significant and uncontrolled weight disparities induced by the branchless collision method, results were not compelling. A comprehensive analysis of underlying mechanisms has, however, been presented, and led to the conclusion that a weight control technique should be used to mitigate the degradation of the results. This technique was therefore used in the forthcoming studies.

The next two chapters presented the application of AMS to kinetics calculations, which was implemented in the coupled neutron-precursor mode of the SERPENT2 code. The implementation is extensively described in Chapter 6. As kinetic calculations are generally more complex than fixed source calculations from an algorithmic point of view, due to the different prompt and delayed neutrons time scales and population control, a thorough analysis of the kinetics of SERPENT2 has been performed and is presented in the first part of this chapter.

The choices resulting from the observations made were then illustrated on a simple homogeneous case. The following features were retained:

- delayed neutron precursors are not tracked within the AMS framework yet (but are still simulated during the transport step),
- the AMS has been systematically used in combination with the branchless collision method,
- the branchless collision method was always used in combination with the weight window technique to keep particle weights under control (as already implemented in SERPENT2),
- no population control, other than the resampling of particles by the AMS, is performed,
- a time bins structure is, however, used to track the average particle weight over time, since it is required by the weight window method in SERPENT2.

The last part of this chapter was devoted to the study of the effect of the weight window technique regarding the poor performances observed on the heterogeneous case in Chapter 5. As a reminder simulation parameters were chosen to avoid particle clustering. The time asymptotic state of a mono-energetic heterogeneous system was computed using SERPENT2 kinetics capabilities. Several modifications, which are detailed in this chapter, were made to adapt SERPENT2 kinetics simulations to reproduce an eigenvalue problem close to the one solved in Korrigan. To model one-speed neutrons,

changes have been made compared to a classical SERPENT2 simulation. First, nuclear data files have been modified to account for one energy group cross sections. The SERPENT2 code has also been modified to force the neutron speed to be constant at every step of the transport stage. In order to reproduce the weight problems discussed at the end of Chapter 5, the spatial flux shape was averaged over time, hence allowing to retrieve the α -eigenproblem solution. Results showed that, using weight windows allowed to fix the particle weight divergence, thus improving the FoM of the spatial flux. Nonetheless, for a basic importance function, no significant improvement, nor degradation, of the FoM was noted for the time-averaged flux shape compared to other state-of-the-art population control methods which were also combined with branchless collisions. This test case confirmed the choices implemented in the case of kinetics calculations in SERPENT2. Further analysis of real time-dependent calculations were then presented in the next chapter.

A proof of concept of the application of AMS on a transient is presented in Chapter 7. In this final part of the thesis work, more complex importance functions are used, based on the physical concept of neutron importance.

After an introduction on the relation between the concept of neutron importance and the adjoint time-dependent transport equations. Since so far no numerical code is capable of solving the time-dependent adjoint transport problem, an approximation of the importance equations was used by deriving the adjoint counterpart of the point kinetics equations. An importance function depending on space, energy and time has therefore been calculated assuming the separation of time and space/energy variables. The static part, i.e., $\psi^*(\mathbf{r}, E)$, has been computed using the Denovo S_n solver from the ADVANTG code, whereas the time-dependent amplitude, noted $N^*(t)$, has been evaluated by solving the adjoint point kinetics equations with a Crank-Nicolson time integration scheme.

Two static maps were estimated by defining two different detector of interest, and tested in a transient calculation using the AMS. The modeled system was a 3x3 fuel assembly cluster, in which AIC control rods were instantly dropped to initiate a subcritical transient of about -9.2% . The results were compared to another AMS simulation, this time only considering a time-dependent importance function used to push neutrons over time to the final 10^{-5} s of the transient. Two reference calculations were also performed using SERPENT2 baseline version 2.1.32.

Results have shown that all three AMS calculations performed worse for the first part of the transient, regarding the usual FoM criterion, while a better FoM of the AMS is observed for the last part of the transient. This matches the time definition of the detector used for the importance computation, which was set to the final time. The obtained FoM gains were up to a factor 10 for the integrated power, and around 5, 15 and 25 depending on the observable (flux or power) and the importance map for the spatial distributions. Regarding the two different estimations of the static adjoint flux, no particular spatial pattern has been observed with regard to the FoM of the spatial distribution of the power. It is possible that the two detectors that were defined in the ADVANTG calculation were too close to a global detector to induce significant spatial differences with regard to the FoM. Improvements of the on-the-fly scoring procedure in time and space-dependent problems must also be carried out.

Finally, Chapter 8 outlines some thoughts on the modeling of realistic fluctuations of the neutron population. Two aspects were addressed, zero-power neutron noise calculations and the impact of numerical models' limitations regarding real population fluctuations. Regarding the former, the potential usefulness of the AMS to reduce calculation costs of such simulations was highlighted. As for the latter, open questions about the representativeness of numerical schemes to model population dynamics problems was raised with respect to recent and past results.

General conclusion of this thesis

The Adaptive Multilevel Splitting has been extended to steady state k -eigenvalue and neutron kinetics Monte Carlo calculations. This thesis presents a progressively unfolded proof of concept, which showed that it was possible to use this iterative algorithm to estimate power in reactor physics.

Regarding the coupling between a Monte Carlo neutronics code and another physics solver (e.g., a CFD code), such adaptive algorithm could be beneficial to reduce statistical fluctuations just before the meeting point of the two codes in particular. By all means, the work presented in this document does not claim to be a thorough validation of the method, and obviously requires further analysis of the different methods involved in the dynamics calculations.

The results obtained for k -eigenvalue calculations have shown a potential propensity for a combination of the AMS and the branchless collision method to reduce spatial and generational correlations compared to some population control methods that may be used in iterated fission source calculations. This reduction of correlations thus induces lower probabilities for neutron clusters, which induce a bias in score estimates, to appear in loosely coupled systems. As regards to the AMS performances out of a clustering context, the figure of merit for the averaged fundamental flux was not so different from that of a power iteration based calculation combined with branchless collisions. Both figures of merit were however higher than that of a classical power iteration based calculation due to the branchless collision method. Put in perspective with the results observed in kinetics, the AMS does not seem to increase the figure of merit for each estimation of the flux (i.e., in each successive generation) but only near the detector (i.e., around the last generation), which result in an overall lukewarm improvement of the performances in the context of eigenvalue calculations.

In definitive, no clear advantage in terms of performance of the AMS over power iteration based algorithms was highlighted in eigenvalue calculations, excepting for the reduction of correlations, which makes the AMS a significant asset regarding the mitigation of the clustering phenomenon.

Prospects for future work

Several prospects arise from the work conducted in this thesis to head towards Monte Carlo dynamics simulations with the AMS. Concerning the implementation of the AMS in kinetics calculations, several improvements are to be made.

First of all, it would be desirable to add the monitoring of precursors within the AMS

method. The major impediment comes from the large number of precursors compared to the number of neutrons in a kinetics calculation. From the user's point of view, the presence of precursors should not change the number of neutron histories expected in the detector. Thus, when a user sets a number of initial neutrons equal to N , and a number of resampled tracks of K , there should be about $N - K + 1$ histories having reached the detector. A possible solution would be to introduce hidden parameters N' and K' such that the stopping criterion would not change from the user's perspective (i.e., at least $N - K + 1$ tracks were brought to the detector). The ratio of precursors and neutrons over time would then be mainly driven by the importance functions provided by the user (as the precursors become less important, resampled tracks would mainly contain neutrons, and vice versa). In that case, it would then be essential to analyze the behavior of the method regarding how tracks are resampled over time.

The second major point concerns the on-the-fly scoring procedure of the AMS. In spatial attenuation problems, quantities of interest are mostly spatially defined reaction rates. In the current implementation of the AMS in SERPENT2, on-the-fly scoring *volumes*³ are only defined over time. However, tallies of interest in time-dependent reactor physics problems (e.g., the power distribution over time) are both time and space-dependent. It has been noted that a significant number of collisions were not scored in some AMS calculations, due to the fact that the definition of scoring volumes does not take space and time into account at the moment. Extending the definition of these scoring volumes to both dimensions would be valuable regarding the method performances for the estimation of both local and integrated detector scores.

The work presented here aimed at setting up a proof of principle and exploring the possibilities of the AMS method. It brought promising results regarding steady-state and kinetics applications, but performances may still be increased by optimizing the choices for K the weight window parameters.

Regarding the uses of the AMS, only steady-state eigenvalue and kinetics calculations were studied here. As mentioned in Chapter 8, extending the application scope of an adaptive re-sampling of neutrons to stochastic fluctuations problem, such as zero-power noise studies could provide an original and valuable way of dealing with such problems.

On a larger scale, several questions were raised on kinetics calculations but were not answered here, such as the role and impact of population control compared to real stochastic processes, for example. The simulation of dynamics transients with Monte Carlo calculations is still a relatively emerging field compared to steady-state simulations, and would probably benefit from further theoretical investigation on the very nature of what is modeled.

³As a reminder, the word *volume* does not describe a spatial volume here, but is used in a broader sense.

Appendix A

Data analysis tools

A.1 Variance for correlated random variables

Let X be a random variable, with n realizations $(X_i)_{i \in \llbracket 1;n \rrbracket}$, identically distributed according to a given distribution with mean μ and variance σ^2 . $\bar{X}(n)$ is an unbiased estimator of μ defined as

$$\bar{X}(n) = \frac{1}{n} \sum_{i=1}^n X_i, \text{ with } \mathbb{E}[\bar{X}] = \mu. \quad (\text{A.1})$$

The corresponding variance of the mean is defined as

$$\text{Var}[\bar{X}(n)] = \frac{1}{n^2} \text{Var}\left[\sum_{i=1}^n X_i\right]. \quad (\text{A.2})$$

If the X_i are independent, $\text{Var}[\sum_{i=1}^n X_i] = \sum_{i=1}^n \text{Var}[X_i]$ and since the X_i are identically distributed,

$$\text{Var}[\bar{X}(n)] = \frac{\sigma^2}{n}, \quad (\text{A.3})$$

$\frac{\sigma}{\sqrt{n}}$ being referred to as the *standard error of the mean*.

In practice, σ^2 is estimated from the sample of size n , using the unbiased estimator s^2 defined as

$$s^2(n) = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}(n))^2, \text{ with } \mathbb{E}[s^2] = \sigma^2. \quad (\text{A.4})$$

Therefore, given the central-limit theorem

$$\bar{X} \sim \mathcal{N}(\bar{X}(n), s^2(n)) \quad (\text{A.5})$$

For correlated variables however, the variance of the sum differ from the sum of variances

$$\text{Var}\left[\sum_{i=1}^n X_i\right] = \sum_{i=1}^n \text{Var}[X_i] + \sum_{\substack{i,j=1 \\ i \neq j}}^n \text{Cov}(X_i, X_j) \text{ (Bienaymé's identity)}. \quad (\text{A.6})$$

Therefore, the variance of the mean becomes

$$\text{Var}[\bar{X}(n)] = \frac{1}{n^2} \left[\sum_{i=1}^n \text{Var}[X_i] + \sum_{\substack{i,j=1 \\ i \neq j}}^n \text{Cov}(X_i, X_j) \right] \quad (\text{A.7})$$

$$\text{Var} [\bar{X}(n)] = \frac{1}{n^2} \left[\sum_{i=1}^n \text{Var} [X_i] + 2 \sum_{i=1}^{n-1} \text{Cov}(X_i, X_j) \right]. \quad (\text{A.8})$$

The covariance can be expressed using the correlation factor

$$\rho_{ij} = \frac{\text{Cov}(X_i, X_j)}{\sigma(X_i)\sigma(X_j)} = \frac{\text{Cov}(X_i, X_j)}{\sigma^2} \quad (\text{A.9})$$

leading to

$$\text{Var} [\bar{X}(n)] = \frac{\sigma^2}{n^2} \left[n + 2 \sum_{i=1}^{n-1} \rho_{ij} \right]. \quad (\text{A.10})$$

A.2 Exponentially Weighted Moving Average

The Exponentially Weighted Moving Average (EWMA) method was used as a smoothing technique to highlight tendencies when results were too noisy. The chief idea is to transform noisy data by doing rolling average. As an example, here are original data used in Sections 5.3 and 6.3 to compute results plotted in Figures 5.27 and 6.15.

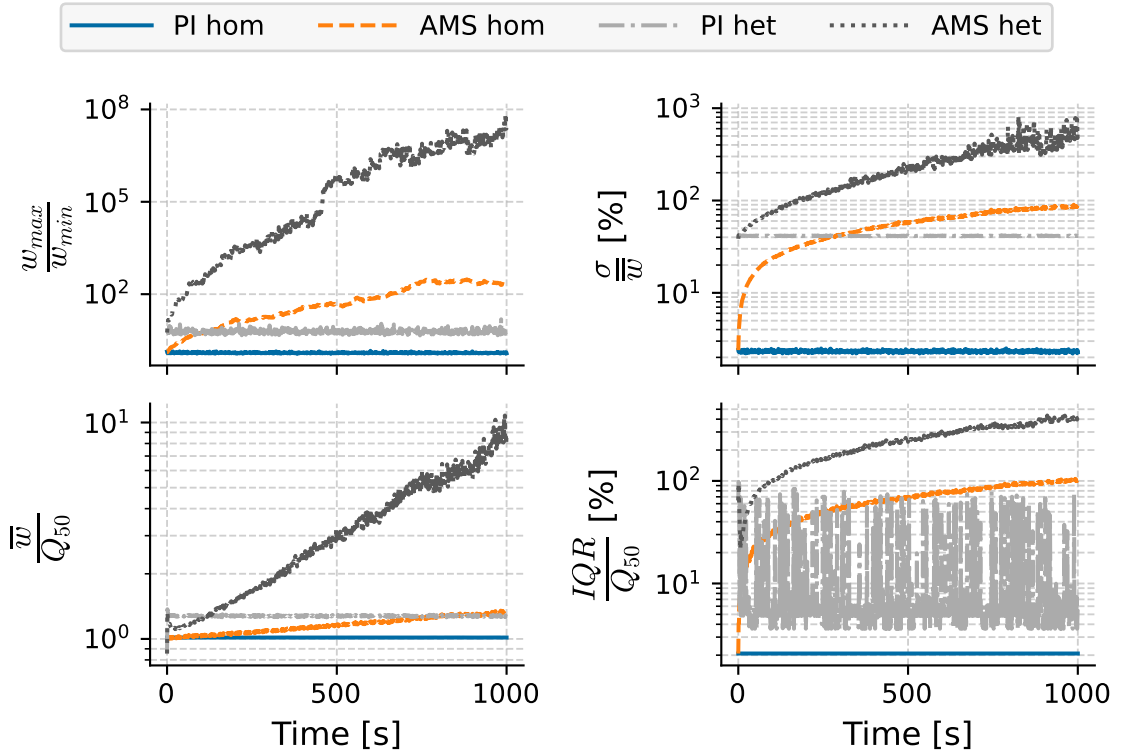


Figure A.1: Statistical moments of the weights distribution over generations.

Unlike usual moving average, data are weighted as such

$$EWMA(t) = \alpha r(t) + (1 - \alpha) \times EWMA(t - 1) \quad (\text{A.11})$$

where $EWMA(t)$ is the averaged data at time index t , $r(t)$ is the raw data at time t and α is a user defined weight used for smoothing. Formulated in terms of raw data

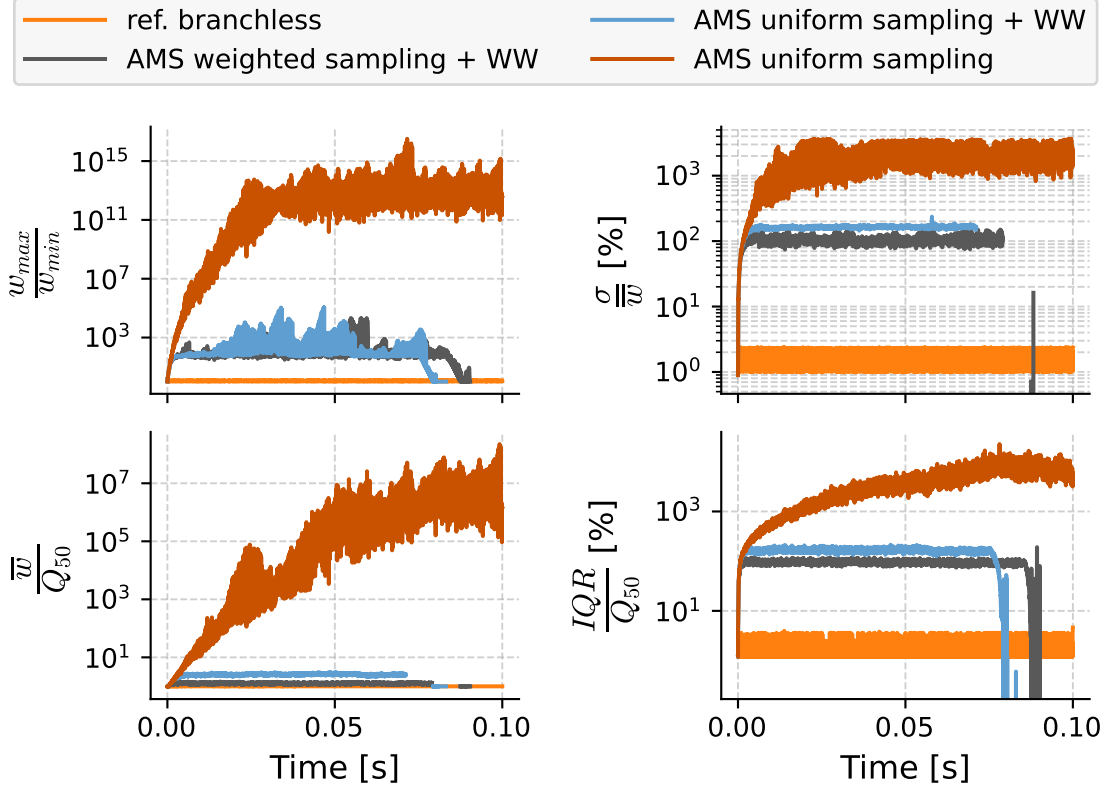


Figure A.2: Statistical moments of the weights distribution over generations.

only this gives

$$EWMA(t) = \sum_{i=0}^t \alpha(1 - \alpha)^{t-i} r(t). \quad (\text{A.12})$$

From this equation, it is manifest that the further the data from the current time index, the lower its contribution to the resulting average.

Appendix B

Derivation of the adjoint equations

B.1 Adjoint transport equation

B.1.1 Derivation of the importance equation

The balance equation for the neutron importance in transport theory can be written [124]

$$N^*(\mathbf{r}, v, \mathbf{\Omega}, t) = [1 - v\Sigma_t\delta t] N^*(\mathbf{r} + \delta\mathbf{r}, v, \mathbf{\Omega}, t + \delta t) + v\Sigma_t\delta t + v\eta_N(\mathbf{r}, v, \mathbf{\Omega}, t)\delta t. \quad (\text{B.1})$$

When δt approaches 0, one obtains

$$\lim_{\delta t \rightarrow 0} \frac{N^*(\mathbf{r} + \delta\mathbf{r}, v, \mathbf{\Omega}, t + \delta t) - N^*(\mathbf{r}, v, \mathbf{\Omega}, t)}{\delta t} = v\Sigma_t N^* - v\Sigma_t \chi^* + P_{score}(\mathbf{r}, v, \mathbf{\Omega}, t) \quad (\text{B.2})$$

$$\frac{\partial N^*}{\partial t} + v\mathbf{\Omega} \cdot \nabla N^* = v\Sigma_t N^* - v\Sigma_t \chi^* + P_{score}(\mathbf{r}, v, \mathbf{\Omega}, t) \quad (\text{B.3})$$

It is possible to write the direct transport equation using the neutron flux $\phi = vN$ which leads to

$$\frac{1}{v} \frac{\partial \phi}{\partial t} + \mathbf{\Omega} \cdot \nabla \phi + \Sigma_t \phi = Q \quad (\text{B.4})$$

where Q is the source term (neutrons from fission, scattering, external, etc...). The adjoint to this equation is

$$-\frac{1}{v} \frac{\partial \phi^*}{\partial t} - \mathbf{\Omega} \cdot \nabla \phi^* + \Sigma_t \phi^* = Q^* \quad (\text{B.5})$$

$$\Leftrightarrow \frac{1}{v} \frac{\partial \phi^*}{\partial t} + \mathbf{\Omega} \cdot \nabla \phi^* = \Sigma_t \phi^* - Q^* \quad (\text{B.6})$$

$$\Leftrightarrow \frac{\partial \phi^*}{\partial t} + v\mathbf{\Omega} \cdot \nabla \phi^* = v\Sigma_t \phi^* - vQ^* \quad (\text{B.7})$$

with $Q^* = \Sigma_t \chi^* - P_{score}$. One can see that equation B.7 is equivalent to equation B.3. Therefore, the adjoint flux ϕ^* solution to equation B.5 is also the neutron importance.

Let us now detail the term χ^* using ϕ^* instead of N^* , which is the mean importance generated from one collision. Lewins give the following expression [137]

$$\chi^*(\mathbf{r}, v, \boldsymbol{\Omega}, t) = \iiint \underbrace{c(\mathbf{r}, v, \boldsymbol{\Omega}, t)}_{\substack{\text{yield of} \\ \text{neutrons} \\ \text{per neutron} \\ \text{removed}}} \underbrace{f(\mathbf{r}, v, \boldsymbol{\Omega} \rightarrow v', \boldsymbol{\Omega}', t)}_{\substack{\text{normalized distribution} \\ \text{density of velocity} \\ \text{after a collision}}} \times \phi^*(\mathbf{r}, v', \boldsymbol{\Omega}', t) dV' d^2\Omega' \quad (\text{B.8})$$

This term does not take delayed neutron precursors into account. Using energy E instead of the neutron speed v and taking importance from precursors into account leads to

$$\begin{aligned} \chi^*(\mathbf{r}, E, \boldsymbol{\Omega}, t) = & \iiint \underbrace{c_p(\mathbf{r}, E, \boldsymbol{\Omega}, t)}_{\substack{\text{yield of} \\ \text{prompt neutrons} \\ \text{per neutron} \\ \text{removed}}} \underbrace{f_p(\mathbf{r}, E, \boldsymbol{\Omega} \rightarrow E', \boldsymbol{\Omega}', t)}_{\substack{\text{normalized distribution} \\ \text{density of velocity} \\ \text{for prompt neutrons} \\ \text{after a collision}}} \times \phi^*(\mathbf{r}, E', \boldsymbol{\Omega}', t) dE' d^2\Omega' \\ & + \sum_k \underbrace{c_d^k(\mathbf{r}, E, \boldsymbol{\Omega}, t)}_{\substack{\text{yield of delayed} \\ \text{neutrons precursors} \\ \text{per neutron removed}}} \times C_k^*(\mathbf{r}, t) \end{aligned} \quad (\text{B.9})$$

where $C_k^*(\mathbf{r}, t)$ is the importance of precursors from family k present at \mathbf{r} at time t . It is also necessary to formulate an equation for the precursors importance, one that includes the velocity distribution for delayed neutrons. After δt , a fraction $\lambda_k \delta t$ of precursors have decayed into delayed neutrons to leave a fraction $(1 - \lambda_k \delta t)$ of surviving precursors with importance $C_k^*(\mathbf{r}, t + \delta t)$, where λ_k is the decay constant from precursors family k . Therefore, the delayed neutrons emitted have a total importance equal to $\lambda_k \delta t \phi^*$. The balance equation fulfilling Axiom 1 is then

$$C_k^*(\mathbf{r}, t) = (1 - \lambda_k \delta t) C_k^*(\mathbf{r}, t + \delta t) + \lambda_k \delta t \iiint f_d^k(\mathbf{r}, E', \boldsymbol{\Omega}, t) \phi^*(\mathbf{r}, E', \boldsymbol{\Omega}', t) dE' d^2\Omega'. \quad (\text{B.10})$$

And with δt approaching 0 it yields

$$\frac{\partial C_k^*}{\partial t}(\mathbf{r}, t) = \lambda_k C_k^*(\mathbf{r}, t) - \lambda_k \iiint f_d^k(\mathbf{r}, E', \boldsymbol{\Omega}, t) \phi^*(\mathbf{r}, E', \boldsymbol{\Omega}', t) dE' d^2\Omega' \quad (\text{B.11})$$

$$\frac{\partial C_k^*}{\partial t}(\mathbf{r}, t) = \lambda_k C_k^*(\mathbf{r}, t) - \lambda_k \iiint \underbrace{f_d^k(\mathbf{r}, E', \boldsymbol{\Omega}, t)}_{\substack{\text{normalized distribution} \\ \text{density of velocity} \\ \text{for delayed neutrons}}} \phi^*(\mathbf{r}, E', \boldsymbol{\Omega}', t) dE' d^2\Omega' \quad (\text{B.12})$$

We now have Equations B.3 and B.9 describing the neutron importance, and Equation B.12 for the precursors importance. The functions c_p , f_p , c_d^k and f_d^k can be formulated using the problem physical properties as follow

$$\begin{aligned}
 c_p(\mathbf{r}, E, \mathbf{\Omega}, t) f_p(\mathbf{r}, E, \mathbf{\Omega} \rightarrow E', \mathbf{\Omega}', t) &= \underbrace{\frac{\Sigma_s(\mathbf{r}, E)}{\Sigma_t(\mathbf{r}, E)} f_s(\mathbf{\Omega}, E \rightarrow \mathbf{\Omega}', E')}_{\text{scattering}} \\
 &+ \underbrace{\frac{(1 - \beta)\nu\Sigma_f(\mathbf{r}, E)}{\Sigma_t(\mathbf{r}, E)} \frac{\chi_{f,p}(E')}{4\pi}}_{\text{prompt fission}}
 \end{aligned} \tag{B.13}$$

$$c_d^k(\mathbf{r}, E, \mathbf{\Omega}, t) = \frac{\beta_k \nu \Sigma_f(\mathbf{r}, E)}{\Sigma_t(\mathbf{r}, E)} \tag{B.14}$$

$$f_d^k(\mathbf{r}, E', \mathbf{\Omega}, t) = \frac{\chi_{f,d}^k(E')}{4\pi} \tag{B.15}$$

where

- Σ_s is the macroscopic scattering cross section,
- f_s is the energy and angle distribution for scattering,
- β is the total delayed neutron fraction,
- ν is the mean number of neutron emitted from fission (including prompt and delayed),
- Σ_f is the macroscopic fission cross section,
- $\chi_{f,p}$ is the fission emission spectrum for prompt neutrons,
- β_k is the delayed neutrons fraction for k -th precursors family,
- $\chi_{f,d}^k$ is the delayed neutron emission spectrum for k -th family.

Replacing Equations B.13, B.14, B.15 and B.9 into Equations B.7 and B.12 finally leads to

$$\begin{aligned}
 -\frac{1}{v} \frac{\partial \phi^*}{\partial t}(\mathbf{r}, E, \mathbf{\Omega}, t) - \underbrace{\mathbf{\Omega} \cdot \nabla \phi^*(\mathbf{r}, E, \mathbf{\Omega}, t)}_{\text{leakage out of } dV} + \Sigma_t(\mathbf{r}, E) \phi^*(\mathbf{r}, E, \mathbf{\Omega}, t) = \\
 \iiint \left[\Sigma_s(\mathbf{r}, E) f_s(E, \mathbf{\Omega}, \rightarrow E', \mathbf{\Omega}') + (1 - \beta)\nu\Sigma_f(\mathbf{r}, E) \frac{\chi_{f,p}(E')}{4\pi} \right] \phi^*(\mathbf{r}, E', \mathbf{\Omega}', t) d^2\Omega' dE' \\
 + \sum_k \beta_k \nu \Sigma_f(\mathbf{r}, E) C_k^*(\mathbf{r}, t) + S^*(\mathbf{r}, v, \mathbf{\Omega}, t) \tag{B.16}
 \end{aligned}$$

$$-\frac{\partial C_k^*}{\partial t}(\mathbf{r}, t) = \lambda_k \iiint \frac{\chi_{f,d}^k(E')}{4\pi} \phi^*(\mathbf{r}, E', \mathbf{\Omega}', t) d^2\Omega' dE' - \lambda_k C_k^*(\mathbf{r}, t) \tag{B.17}$$

where $S^*(\mathbf{r}, v, \mathbf{\Omega}, t)$ is related to the probability for a neutron present at \mathbf{r} with velocity v and direction $\mathbf{\Omega}$ at t to score in the chosen detector. Following this definition, S^* is here the detector response function for the neutron flux η_ϕ .

Following the same derivation for the adjoint diffusion equation gives [124]

$$\begin{aligned}
 -\frac{1}{v} \frac{\partial \phi^*}{\partial t}(\mathbf{r}, E, t) - \underbrace{\nabla \cdot D(\mathbf{r}) \nabla \phi^*}_{\text{leakage out of } dV}(\mathbf{r}, E, t) + \Sigma_t(\mathbf{r}, E) \phi^*(\mathbf{r}, E, t) = \\
 \int [\Sigma_s(\mathbf{r}, E) f_s(E \rightarrow E') + (1 - \beta) \nu \Sigma_f(\mathbf{r}, E) \chi_{f,p}(E')] \phi^*(\mathbf{r}, E', t) dE' \\
 + \sum_k \beta_k \nu \Sigma_f(\mathbf{r}, E) C_k^*(\mathbf{r}, t) + \eta_\phi(\mathbf{r}, E, t) \quad (\text{B.18})
 \end{aligned}$$

$$-\frac{\partial C_k^*}{\partial t}(\mathbf{r}, t) = \lambda_k \int \chi_{f,d}^k(E') \phi^*(\mathbf{r}, E', t) dE' - \lambda_k C_k^*(\mathbf{r}, t). \quad (\text{B.19})$$

B.1.2 Multigroup adjoint diffusion equation

$$\begin{aligned}
 -\frac{1}{v_g} \frac{\partial \phi_g^*}{\partial t}(\mathbf{r}, t) - \nabla \cdot D(\mathbf{r}) \nabla \phi_g^*(\mathbf{r}, t) + \Sigma_{rg}(\mathbf{r}) \phi_g^*(\mathbf{r}, t) = \\
 \sum_{h \neq g} [\Sigma_{s,h \leftarrow g}(\mathbf{r}) + (1 - \beta) \nu \Sigma_{f,g}(\mathbf{r}) \chi_{f,p,h}] \phi_h^*(\mathbf{r}, t) \\
 + \sum_k \beta_k \nu \Sigma_{f,g}(\mathbf{r}) C_k^*(\mathbf{r}, t) + \eta_{\phi,g}(\mathbf{r}, t) \quad (\text{B.20})
 \end{aligned}$$

$$-\frac{\partial C_k^*}{\partial t}(\mathbf{r}, t) = \lambda_k \sum_h \chi_{f,d,h}^k \phi_h^*(\mathbf{r}, t) - \lambda_k C_k^*(\mathbf{r}, t) \quad (\text{B.21})$$

with

$$\Sigma_{rg} = \Sigma_{t,g} - \Sigma_{s,g \leftarrow g}. \quad (\text{B.22})$$

B.2 Time discretization of the adjoint point kinetics equations

Considering the system described by Equations

$$\frac{d}{dt} [\Phi^*(t)] = [R^*(t)] [\Phi^*(t)] \quad (\text{B.23})$$

where

$$[\Phi^*(t)] = \begin{bmatrix} N^*(t) \\ C_1^*(t) \\ \vdots \\ C_K^*(t) \end{bmatrix} \quad (\text{B.24})$$

and

$$[R^*(t)] = \begin{bmatrix} \frac{\beta - \rho(t)}{\Lambda} & \frac{-\beta_1}{\Lambda} & \frac{-\beta_2}{\Lambda} & \dots & \frac{-\beta_K}{\Lambda} \\ -\lambda_1 & \lambda_1 & & & \\ -\lambda_2 & & \lambda_2 & & \\ \vdots & & & \ddots & \\ -\lambda_K & & & & \lambda_K \end{bmatrix}. \quad (\text{B.25})$$

Using the Θ method the system is discretized following

$$\frac{d\Phi^*}{dt} = \frac{\Phi^{*,(n+1)} - \Phi^{*,(n)}}{\Delta t} = (1 - \Theta)R^*\Phi^{*,(n)} + \Theta R^*\Phi^{*,(n+1)}. \quad (\text{B.26})$$

$$\Phi^{*,(n)} = (I + (1 - \Theta)\Delta t R^*)^{-1}(I - \Theta\Delta t R^*)\Phi^{*,(n+1)} \quad (\text{B.27})$$

The specific case of the Crank-Nicolson method corresponds to $\Theta = 1/2$. The Crank-Nicolson method is unconditionally numerically stable, and therefore a safe choice to solve this time-dependent system of equation.

Appendix C

SERPENT2 input data for the geometry and materials of the assembly cluster transient

```
/*  
* Material definitions *  
*/  
  
% — 6.5-% enriched UO2  
%   Temperature is set to 900 K  
  
% — Fuel material (6.5 %wt enriched uranium dioxide),  
   density 10.3 g/cm3  
  
mat uox65    sum    tmp 900 rgb 255 68 0  
92235.09c   1.5122e-3  
92238.09c   2.1477e-2  
8016.09c    4.5945e-2  
  
% — Cladding material for fuel rod  
%   Zr-Nat  
  
mat cladZr  -6.53 tmp 600 rgb 96 96 96  
40090.06c  -0.5145  
40091.06c  -0.1122  
40092.06c  -0.1715  
40094.06c  -0.1738  
40096.06c  -0.0280  
  
% — Steel cladding for AIC rods  
  
mat steel sum rgb 0 0 0  
tmp 600 % Kelvin
```

APPENDIX C. SERPENT2 INPUT DATA FOR THE GEOMETRY AND MATERIALS OF THE ASSEMBLY CLUSTER TRANSIENT

26054.06c 3.23849590E-03
26056.06c 5.12129068E-02
26057.06c 1.22839503E-03
26058.06c 1.56341179E-04
28058.06c 5.13793342E-03
28060.06c 1.96426059E-03
28061.06c 8.50427023E-05
28062.06c 2.70179909E-04
28064.06c 6.84857150E-05
24050.06c 6.65122119E-04
24052.06c 1.28116282E-02
24053.06c 1.45256566E-03
24054.06c 3.60847887E-04
25055.06c 8.03962292E-04
14028.06c 5.82413049E-04
14029.06c 2.95564168E-05
14030.06c 1.95148132E-05

% — AIC control rod
mat aic sum rgb 255 255 255
tmp 600 % Kelvin

47107.06c 2.32816897E-02
47109.06c 2.16376428E-02
48106.06c 3.28119022E-05
48108.06c 2.33620740E-05
48110.06c 3.28381517E-04
48111.06c 3.36256373E-04
48112.06c 6.33400981E-04
48113.06c 3.20769148E-04
48114.06c 7.53886299E-04
48116.06c 1.96083929E-04
49115.06c 7.57471379E-03

% — Coolant is water with 650 ppm soluble boric acid added
% The temperature of water is 600 K
% Density is calculated based on a pressure of 15.5 MPa
and a Temperature of 315 C
% Hydrogen is flagged as a bound scatterer with the "moder
"-card

mat water -0.66 tmp 600 moder lwtr 1001 rgb 0 128 255
O-16.06c 3.330861e-01
H-1.06c 6.663259e-01
5010.03c 0.000293053904

% — Define thermal scattering libraries associated with
hydrogen in light water
% As there are no readymade thermal scattering libraries

```

    for 583 K
%    We will tell Serpent to interpolate using two bounding
    libraries:
%    -lwj3.11t (H-1 in light water at 574 K)
%    -lwj3.13t (H-1 in light water at 624 K)
%    See also: http://montecarlo.vtt.fi/download/SSS\_THERMAL.pdf

therm lwtr 600 h-h2o.15t h-h2o.16t

/*****
 * Geometry definitions *
 *****/

% — Fuel rod

pin FF
uox65 0.412
cladZr 0.476
water

% — AIC control rod

pin cr
aic 0.50
steel 0.54
water 0.57
cladZr 0.61
water

% — Empty instrumentation thimble

pin ii
water 0.57
cladZr 0.61
water

% — Empty control rod channel

pin cc
water 0.57
cladZr 0.61
water

% — Empty lattice position (just water)

pin ww

```

APPENDIX C. SERPENT2 INPUT DATA FOR THE GEOMETRY AND MATERIALS OF THE ASSEMBLY CLUSTER TRANSIENT

water

```
% — Pin lattice definition , name of the lattice "assUOX"
%   Lattice type 1 (square lattice)
%   Lattice centered at 0.0 0.0
%   19 x 19 lattice elements (17x17 fuel rods + 1 layer of
%   water)
%   Lattice pitch 1.265 cm
```

```
lat assUOX 1 0.0 0.0 19 19 1.265
```

```
ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF cc FF FF cc FF FF cc FF FF FF FF FF FF ww
ww FF FF FF cc FF FF FF FF FF FF FF FF FF FF FF cc FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF cc FF FF cc FF FF cc FF FF cc FF FF cc FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF cc FF FF cc FF FF cc FF FF cc FF FF cc FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
```

```
lat assAIC 1 0.0 0.0 19 19 1.265
```

```
ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF cr FF FF cr FF FF cr FF FF FF FF FF FF ww
ww FF FF FF cr FF FF FF FF FF FF FF FF FF FF FF cr FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF cr FF FF cr FF FF cr FF FF cr FF FF cr FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF cr FF FF cr FF FF cr FF FF cr FF FF cr FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww FF FF FF cr FF FF FF FF FF FF FF FF FF FF FF cr FF FF FF ww
ww FF FF FF FF FF cr FF FF cr FF FF cr FF FF cr FF FF FF ww
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
```

```
ww FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF ww
ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww ww
```

```
% — assembly cluster, name of the cluster "clusterUOX"
%   Lattice type 1 (square lattice)
%   Lattice centered at 0.0 0.0
%   3 x 3 lattice elements
%   Lattice pitch 24.035 cm
lat clusterUOX 1 0.0 0.0 3 3 24.035
assUOX assUOX assUOX
assUOX assAIC assUOX
assUOX assUOX assUOX
```

```
% — A square surface around the fuel assembly / cluster
surf s1 sqc 0.0 0.0 36.0525
```

```
% — Cell c1 belongs to the base universe 0, is filled with
   the lattice lat1
%   and covers everything inside surface s1
cell c1 0 fill clusterUOX -s1
```

```
% — Cell c2 belongs to the base universe 0, is defined as an
   "outside" cell
%   and covers everything outside surface s1
cell c2 0 outside s1
```

```
% — Boundary condition (1 = black, 2 = reflective, 3 =
   periodic)
set bc 1
```


Appendix D

BPLUS energy mesh

Table D.1: BPLUS library groups

Group	Upper energy boundary (MeV)	Group	Upper energy boundary (MeV)
0	1.7332E+01	24	2.9721E-01
1	1.4194E+01	25	1.8316E-01
2	1.2214E+01	26	1.1109E-01
3	1.0000E+01	27	6.7379E-02
4	8.6071E+00	28	4.0868E-02
5	7.4085E+00	29	3.1828E-02
6	6.0654E+00	30	2.6058E-02
7	4.9659E+00	31	2.4176E-02
8	3.6788E+00	32	2.1875E-02
9	3.0119E+00	33	1.5034E-02
10	2.7253E+00	34	7.1017E-03
11	2.4660E+00	35	3.3546E-03
12	2.3653E+00	36	1.5846E-03
13	2.3457E+00	37	4.5400E-04
14	2.2313E+00	38	2.1445E-04
15	1.9205E+00	39	1.0130E-04
16	1.6530E+00	40	3.7266E-05
17	1.3534E+00	41	1.0677E-05
18	1.0026E+00	42	5.0435E-06
19	8.2085E-01	43	1.8554E-06
20	7.4274E-01	44	8.7643E-07
21	6.0810E-01	45	4.1399E-07
22	4.9787E-01	46	1.0000E-07
23	3.6883E-01		1.0000E-11

Résumé en français

Introduction

Le fonctionnement des réacteurs nucléaires est basé sur un phénomène de réaction en chaîne impliquant un très grand nombre de fissions de noyaux induites par des neutrons. Mais la conception des réacteurs, la prédiction du comportement des systèmes fissiles, l'évaluation des retombées des accidents nucléaires ou encore la conception d'expériences et l'approfondissement de la compréhension des phénomènes physiques, sont des activités qui ne peuvent s'appuyer uniquement sur des expériences, et ce pour des raisons économiques et de sûreté. L'industrie nucléaire s'appuie donc fortement sur les outils de simulation pour de nombreuses applications.

Les méthodes Monte Carlo sont largement utilisées afin de produire des simulations dites de référence pour des applications comme la sûreté-criticité ou la radioprotection, par exemple. Certaines applications de la physique des réacteurs ne sont néanmoins pas concernées par ce point. Les calculs dits de dynamique notamment, ont été pendant longtemps effectués à l'aide de méthodes déterministes exclusivement. Ces calculs visent à modéliser le comportement d'un système fissile au cours du temps suite à une perturbation, comme lors d'un accident de réactivité par exemple. Jusqu'au début des années 2010, aucun code de production n'était en mesure d'effectuer de tels calculs du fait des coûts de calcul prohibitifs associés aux méthodes Monte Carlo dans ce contexte.

Pour réduire les coûts des simulations Monte Carlo, il est possible d'utiliser les propriétés mathématiques de l'échantillonnage aléatoire pour mettre en œuvre des méthodes appelées méthodes de *réduction de la variance*. Ces méthodes consistent à modifier les règles naturelles des simulations de Monte Carlo afin de réduire l'incertitude statistique associée aux résultats de ces simulations, pour un coût de calcul égal (ou bien accélérer l'exécution du calcul sans changer l'incertitude sur le résultat). Elles ont été largement utilisées dans les calculs de transport de particules, notamment pour les problèmes d'atténuation en radioprotection.

Les méthodes de réduction de variance sont généralement moins utilisées dans le cas de la physique des réacteurs, pour la simple raison qu'elles sont le plus souvent développées pour réduire la variance (c'est-à-dire l'incertitude) pour des détecteurs fortement localisés. Or, c'est tout l'espace des phases (ou presque) qui est d'intérêt dans le cas où l'on chercherait à étudier le fonctionnement global d'un réacteur. D'autres problèmes peuvent néanmoins profiter de schémas de réduction de la variance.

Les calculs de neutronique en régime permanent, par exemple, consistent à caractériser la *criticité*, c'est-à-dire l'évolution de la population de particules à long terme, d'un système. Ils sont également couramment utilisés pour évaluer la distribution des particules dans le système (et donc la puissance) en régime permanent. Bien que ces simulations soient utilisées depuis les débuts de l'industrie nucléaire civile et bénéficient de décennies de perfectionnement, des améliorations sont encore possibles. Il a en effet été démontré qu'en raison de la nature aléatoire du mécanisme de branchement (fission), couplé au transport des particules, de fortes fluctuations peuvent se développer dans des systèmes faiblement couplés, conduisant à la formation de paquets disparates de neutrons menant à des erreurs d'estimation des résultats.

Les simulations de la dynamique des systèmes fissiles quant à elles sont encore très récentes et trop coûteuses pour en faire une application industrielle. Afin de les rendre plus abordables, il est nécessaire de développer des schémas performants notamment à l'aide de réduction de la variance.

Ce travail de thèse s'inscrit donc dans ce cadre. L'objectif de ce manuscrit est de présenter l'application d'un schéma de réduction de variance innovant en physique des réacteurs en se basant sur l'utilisation d'une méthode appelée *Adaptive Multilevel Splitting (AMS)*. La méthode AMS est une méthode de réduction de la variance développée pour augmenter les performances des simulations d'événements rares, et précédemment mise en œuvre dans le code TRIPOLI-4® pour le transport de particules neutres en radioprotection. Cette méthode se base sur la multiplication des particules qui ont plus de chances d'atteindre un détecteur donné, augmentant ainsi le nombre de particules contribuant à la mesure. Elle s'est avérée assez robuste dans les configurations de forte atténuation spatiale dans lesquelles seules quelques particules (voire aucune) atteignent le détecteur.

Dans le cadre de cette thèse, l'AMS a d'abord été implémentée dans un code Monte Carlo simplifié entièrement développé pendant la thèse, afin d'être étendue aux calculs de neutronique en régime permanent afin de réduire les corrélations spatiales menant à la formation de paquets de neutrons. Pour cela, le transport de neutrons sur des générations successives a été interprété comme un problème d'atténuation générationnel, et le contrôle de population généralement opéré par un algorithme de type itération sur la puissance a été remplacé par l'AMS. Dans un second temps, l'AMS a été implémenté dans le code de transport SERPENT2 pour les calculs de cinétique des réacteurs, c'est-à-dire avec dépendance en temps et prise en compte de la double échelle de temps due aux neutrons prompts et aux neutrons retardés. Une analyse détaillée du code a dû être effectuée afin de caractériser au mieux les interactions entre les méthodes préexistantes dans SERPENT2 et l'AMS.

Problème de *clustering* en physique statique des réacteurs

La théorie du transport neutronique est utilisée pour décrire les systèmes neutroniques de diverses manières, chacune d'entre elles reposant sur une variation différente de l'équation de transport linéaire. La majorité des calculs neutroniques en physique des réacteurs sont effectués sur des systèmes statiques (ou du moins numériquement statiques). Pendant des décennies, ces calculs, dits de *criticité* ont été utilisés pour

caractériser l'aspect multiplicatif d'un système, en résolvant l'équation aux valeurs propres k . Cette équation interprète le système sous la forme d'un problème pour lequel les sources neutroniques externes (c'est-à-dire qui ne dépendent pas du flux neutronique actuel, comme les fissions spontanées) sont négligées. Pour s'assurer que le système est statique, le terme de production de fission dans l'équation de transport est modifié, révélant ainsi une équation aux valeurs propres. Cette description fait consensus et est actuellement largement utilisée non seulement en sûreté-criticité, mais aussi en physique des réacteurs nucléaires.

La méthode d'itération sur la puissance est utilisée, tant par les codes déterministes que Monte Carlo, afin d'estimer la solution de cette équation. Dans le cas des méthodes Monte Carlo, elle prend la forme d'une renormalisation de la population de neutrons à chaque génération, de sorte à la maintenir constante tout au long de la simulation. Cette étape de contrôle de la population est néanmoins susceptible d'entraîner une amplification des fluctuations aléatoires inhérentes au transport de neutron dans des milieux fissiles. Dans les cas les plus pathologiques, ces fluctuations peuvent mener à la formation de paquets de particules souvent appelés *clusters*, provoquant alors des erreurs dans les estimations des scores.

L'observation de ce phénomène, dit de *clustering*, a été reliée à la mesure de fortes corrélations spatiales et temporelles caractéristiques dans les systèmes faiblement couplés, c'est à dire pour lesquels l'information d'une perturbation locale est difficilement transmise au reste du système. Plus récemment, l'apparition de ces corrélations a été expliquée par la disparition de lignées indépendantes de neutrons (c'est-à-dire des neutrons ne partageant pas d'ancêtres communs dans la simulation). C'est dans le but d'agir sur cette disparition des familles indépendantes de neutrons que l'AMS a été mis en place pour les calculs de transport de neutrons stationnaires.

Application de l'AMS dans les calculs de criticité

Pour que l'AMS puisse être utilisé dans les calculs de criticité, une refonte du problème est nécessaire pour le rendre adapté à la réduction de la variance. L'AMS pour le transport de particules a été typiquement conçu pour estimer les événements rares dans le contexte des problèmes de blindage. En criticité, si $k_{\text{eff}} < 1$, le système tend à s'éteindre au fil des générations. Par conséquent, il est clair que plus le k_{eff} est faible, moins la descendance d'un neutron a de chances de survivre au fil des générations, et atteindre une génération éloignée est finalement un événement rare. Dans ce contexte, l'AMS peut être utilisée pour ré-échantillonner les histoires et les pousser à travers les générations, comme une alternative aux techniques plus classiques de contrôle de la population déjà utilisées dans l'itération de puissance. Néanmoins, même pour les systèmes critiques (et surcritiques), la probabilité de survie d'une chaîne de fission diminue avec le temps. Par conséquent, il est possible d'appliquer l'AMS pour ré-échantillonner les historiques de neutrons même dans ces contextes. La divergence probable du nombre de branches à l'intérieur d'une histoire peut cependant nécessiter une refonte du problème pour imposer une plus forte atténuation du nombre de particules au cours du temps.

S'il est improbable que la population de particules diminue drastiquement dans les systèmes critiques et supercritiques, il est possible de simuler un système numériquement sous-critique dans lequel la population numérique (c'est-à-dire le nombre de particules dans la simulation) diminue tout en maintenant la population physique (le

nombre de particules physiques représentées par les particules numériques) constante ou en l'augmentant. Ces deux nombres peuvent différer dans un calcul non analog, puisqu'une particule numérique peut représenter plusieurs particules physiques par son poids statistique. Ainsi, en modifiant le nombre de particules échantillonnées en sortie de collision, il est possible d'inhiber les événements branchant, et donc le nombre de particules dans la simulation. Ceci entraîne alors une diminution du nombre de particules numériques même dans les systèmes surcritiques. La méthode des collisions non branchantes a été utilisée à cette fin afin de modéliser des systèmes numériquement sous-critiques quel que soit leur k_{eff} . De cette façon, il a été possible de généraliser la description d'un problème d'atténuation à des systèmes critiques et surcritiques, à condition qu'il existe un moyen de tuer les particules numériques (par exemple, par fuite ou par roulette russe). Il existe un autre avantage à utiliser la méthode des collisions non branchantes, même pour les systèmes sous-critiques. Si l'on considère la façon dont les histoires branchantes sont échantillonnées par l'AMS, on remarque que plus une histoire contient de branches, plus le nombre de particules ré-échantillonnées est élevé (rappelons qu'une branche représente une particule). Cela implique que pour les systèmes branchant, le nombre de particules ré-échantillonnées pourrait augmenter à chaque itération de l'AMS, entraînant un ralentissement des itérations. Par conséquent, limiter le nombre de branches à l'intérieur des histoires permet également de garder le temps de calcul sous contrôle.

Dans l'algorithme Monte Carlo basé sur l'itération de puissance, la descendance des neutrons est simulée de la génération 0 (celle de notre source initiale) à la génération G . On passe d'une génération à la suivante par fission et le contrôle de la population permet de maintenir le nombre de particules constant au fil des générations (en l'empêchant de s'atténuer ou de diverger). Dans un calcul de criticité reformulé comme un problème d'atténuation générationnelle, on ne s'intéresse pas seulement à un détecteur spécifique localisé dans l'espace des phases, mais à une génération de neutrons dans laquelle il y a suffisamment de particules (c'est-à-dire la génération G). Ici encore, le mécanisme qui amène un neutron d'une génération à l'autre est toujours la fission, mais aucun contrôle de population autre que l'AMS n'est utilisé, ce qui conduit à une diminution du nombre de particules au fil des générations.

Une fois implémentée dans un code Monte Carlo simplifié, la méthode a été testée sur un milieu fissile homogène en une dimension d'espace avec des neutrons monocinétiques. La taille du système et les sections efficaces macroscopiques décrivant le matériau ont été fixées de sorte à simuler un système faiblement couplé. De plus, le nombre de neutrons modélisés a été choisi de sorte à faire volontairement apparaître un phénomène de clustering dans ce système simplifié.

Les résultats obtenus montrent une réduction des corrélations spatiales et générationnelles par l'utilisation de l'AMS, en agissant sur le mécanisme de contrôle de la population. De plus, l'utilisation de la méthode des collisions non branchantes permet une réduction significative de l'incertitude sur le score (k_{eff} et flux), mais aussi des corrélations spatiales dans le système. De fait, le phénomène de clustering a pu être réduit en comparaison aux cas pour lesquels la méthode de contrôle de la population était grossière, ou lorsque les collisions étaient traitées de manière analog. Le biais sur l'estimation de la variance a lui aussi été réduit du fait de corrélations inter générations plus faibles.

Calculs de cinétique par les méthodes Monte Carlo

Dans les calculs de dynamique, le but est de modéliser un transitoire en couplant une simulation cinétique (c'est-à-dire le transport de neutrons en fonction du temps) avec différents solveurs de physique pour tenir compte des phénomènes de rétroaction induits par la température. Dans les calculs cinétiques, tout l'intervalle de temps est peuplé de neutrons afin d'obtenir un résultat à presque n'importe quel moment du transitoire. Dans les calculs de cinétiques Monte Carlo actuels, des marches aléatoires dépendantes du temps sont simulées sur des intervalles de temps successifs, et la moyenne de la contribution de chaque marche dans chaque intervalle de temps est calculée. En raison de la nature aléatoire de l'apparition et de la disparition des neutrons, la population numérique de neutrons est sujette à d'importantes variations dans le temps, ce qui peut entraîner des problèmes de temps de calcul et d'empreinte mémoire. Pour éviter ces problèmes, un contrôle de la population est utilisé à intervalle régulier pour contenir les fluctuations du nombre de particules simulées.

Contrairement aux calculs de criticité, les simulations cinétiques ne débutent pas d'une distribution arbitraire des particules. La modélisation d'un transitoire se fait généralement au départ d'un état initial stable et donc avec une distribution particulière des neutrons. De plus, puisque les différentes échelles de temps des neutrons prompts et retardés sont prises en compte par le suivi des précurseurs de neutrons retardés en plus des neutrons, il est également nécessaire d'initialiser la distribution des précurseurs. Pour ce faire, un calcul en régime stationnaire peut être effectué au préalable afin de converger vers la distribution à l'équilibre du système.

Le principal facteur limitant des calculs de dynamique Monte Carlo a trait aux ressources de calcul requises, notamment en temps de calcul. Ce dernier est principalement dû à la simulation de la neutronique (c'est-à-dire la cinétique) avec les méthodes de Monte Carlo. En effet, des calculs couplés récents ont montré un temps de calcul négligeable pour l'étape de couplage par rapport à l'étape de transport Monte Carlo dépendant du temps, elle-même très coûteuse. Pour surmonter ce problème, des travaux sur de nouveaux schémas de réduction de la variance appliqués aux problèmes de cinétique sont d'actualité.

Implémentation de l'AMS dans SERPENT2 et premiers tests

Dans le but de produire une preuve de principe sur l'utilisation de l'AMS en cinétique, la méthode a été implémentée dans le code Monte Carlo SERPENT2. Une version récente du code (v2.1.32) a servi de base aux développements, et l'exécution d'un calcul de cinétique a été étudiée en détail afin d'envisager les possibilités d'implémentation de la méthode AMS. En effet, les calculs de cinétique font intervenir plusieurs méthodes pour limiter les coûts associés à ces simulations. Il a donc été nécessaire d'analyser les interactions entre ces méthodes et l'AMS, tant du point de vue de l'implémentation que des performances des simulations une fois la méthode en place. Dans son fonctionnement actuel, l'AMS est utilisé en combinaison avec la méthode des collisions non branchantes afin de limiter le nombre de branches au sein de chaque histoire de neutrons (ce qui, comme dans le cas des calculs stationnaires, est susceptible de provoquer des coûts de calculs importants). Pour limiter la dispersion des poids au sein d'une

population de particules, une technique de contrôle du poids (la technique *weight window*) est associée aux collisions non branchantes dans SERPENT2. Tout comme les calculs stationnaires susmentionnés, cela permet également de faire émerger un problème d'atténuation de la population, cette fois au cours du temps, de sorte à remplacer les méthodes classiques de contrôle de population par l'AMS.

La méthode a été ensuite testée sur un transitoire sous-critique initié par la chute de barres de contrôles dans un motif de 3x3 assemblages de combustible fissile. Le calcul de la puissance intégrée au cours du temps a montré que l'AMS permettait d'améliorer la figure de mérite dans les derniers instants du transitoire modélisé par rapport à un calcul de référence SERPENT2. Concernant la distribution spatiale de la puissance dans le système, les différentes cartes d'importance utilisées n'ont pas semblé entraîner de différence majeure dans la figure de mérite spatiale.

Réflexions sur la modélisation des comportements temporels de milieux fissiles avec des méthodes Monte Carlo

Les simulations de Monte Carlo visent généralement à estimer la réponse moyenne d'un détecteur donné. En raison des échantillonnages aléatoires effectués par les méthodes Monte Carlo, l'estimation d'une moyenne est toujours associée à une incertitude statistique. L'objectif des techniques de réduction de la variance est de réduire la variance (c'est-à-dire l'incertitude) des estimateurs Monte Carlo, sans modifier le premier moment statistique de ces estimateurs, c'est-à-dire la valeur moyenne. Si ces méthodes permettent de surmonter certaines difficultés découlant des aspects numériques de la simulation, elles dénaturent le problème réel que l'on cherche à modéliser.

L'étude du bruit neutronique associé aux fluctuations naturelles des processus branchant (appelé bruit neutronique à puissance nulle), par exemple, nécessite la modélisation fidèle des distributions de particules dans le système, et ne peut donc pas profiter des schémas de réduction de la variance classiques. Or, si l'AMS permet de diminuer la variance associée à l'estimateur de la probabilité d'un événement rare, elle ne modifie pas les corrélations naturelles dues au processus de fission dans sa forme générale. Cette méthode permettrait donc éventuellement de réduire la variance associée aux estimateurs des observables du bruit neutronique.

Enfin, des travaux récents ont soulevé d'éventuelles limites des schémas numériques actuels dans le cadre de calculs neutroniques avec prise en compte des contre-réactions. En effet, des différences de comportement ont été observées entre le comportement moyen simulé par les codes de calcul et la forme analytique développée d'après une formulation stochastique théorique d'un mouvement brownien (c'est-à-dire de diffusion) de neutrons, dès lors qu'un phénomène de contre-réaction de type élargissement Doppler était pris en compte dans le modèle. Cela soulève aussi des questions encore ouvertes sur la nature du problème modélisé par les outils de calculs actuels (déterministes et Monte Carlo) en dynamique. Pour approfondir le sujet, une formulation théorique du transport stochastique des neutrons, comparée aux calculs actuels, serait nécessaire.

Conclusion

L'utilisation de la méthode Adaptive Multilevel Splitting a été étendue aux calculs de valeurs propres k et aux calculs de cinétique des réacteurs. Cette thèse présente une preuve de concept montrant qu'il est possible d'utiliser cet algorithme itératif pour estimer la puissance en physique des réacteurs.

En ce qui concerne le couplage entre un code Monte Carlo de neutronique et le solveur d'une physique différente (par exemple, un code de mécanique des fluides), un tel algorithme adaptatif pourrait être bénéfique pour réduire les fluctuations statistiques au point de rencontre des deux codes. Une analyse plus approfondie des différentes méthodes impliquées dans les calculs de dynamique est toutefois nécessaire.

Les résultats obtenus pour les calculs de valeurs propres k ont montré une tendance à réduire les corrélations spatiales et générationnelles lorsque l'AMS est utilisée en combinaison avec la méthode des collisions sans branches, par rapport à certaines méthodes de contrôle de la population utilisées dans ces calculs. Cette réduction des corrélations a diminué les risques de formation de paquets de neutrons menant au phénomène de clustering dans les systèmes faiblement couplés, réduisant ainsi les biais dans les estimations des scores. En ce qui concerne les performances de l'AMS en dehors d'un contexte de clustering, les performances du calcul du flux dans le mode fondamental étaient similaires à celles de méthodes modernes de contrôle de la population.

En définitive, l'AMS mais aussi la méthode des collisions non branchantes montrent des caractéristiques intéressantes pour la réduction du risque de clustering dans les calculs de physique des réacteurs. Les calculs de cinétiques ont quant à eux montré qu'il était possible d'obtenir une réduction de la variance localement en temps.

Bibliography

- [1] PRIS - Home.
- [2] France monthly energy production data | RTE.
- [3] Guide de l'ASN n°28, 9 2021.
- [4] Frédéric Cérou and Arnaud Guyader. Adaptive Multilevel Splitting for Rare Event Analysis. <https://doi.org/10.1080/07362990601139628>, 25(2):417–443, 3 2007.
- [5] Charles Edouard Bréhier, Maxime Gazeau, Ludovic Goudenège, Tony Lelièvre, and Mathias Rousset. Unbiasedness of some generalized adaptive multilevel splitting algorithms. <https://doi.org/10.1214/16-AAP1185>, 26(6):3559–3601, 12 2016.
- [6] Henri Louvin, Eric Dumonteil, Tony Lelièvre, Mathias Rousset, and Cheikh M. DIop. Adaptive Multilevel Splitting for Monte Carlo particle transport. *EPJ Web of Conferences*, 153:06006, 9 2017.
- [7] Henri Louvin. *Development of an adaptive variance reduction technique for Monte Carlo particle transport*. PhD thesis, Université Paris-Saclay, 2017.
- [8] Eric Dumonteil, Fausto Malvagi, Andrea Zoia, Alain Mazzolo, Davide Artusio, Cyril Dieudonné, and Clélia De Mulatier. Particle clustering in Monte Carlo criticality simulations. *Annals of Nuclear Energy*, 63:612–618, 2014.
- [9] A. Zoia, E. Dumonteil, A. Mazzolo, C. De Mulatier, and A. Rosso. Clustering of branching Brownian motions in confined geometries. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 90(4), 10 2014.
- [10] Eric Dumonteil, Giovanni Bruna, Fausto Malvagi, Anthony Onillon, and Yann Richet. Clustering and traveling waves in the Monte Carlo criticality simulation of decoupled and confined media. *Nuclear Engineering and Technology*, 49(6):1157–1164, 9 2017.
- [11] M W Waddell, Jr and H L Dodds. A method for transient, three-dimensional neutron transport calculations. Technical report, Oak Ridge Y-12 Plant, 12 1992.
- [12] C Bentley, R DeMeglio, M Dunn, S Goluoglu, K Norton, R Pevey, I Suslov, and H Dodds. Development of a hybrid stochastic/deterministic method for transient, three dimensional neutron transport. *Progress in Nuclear Energy*, 105:236–246, 6 1997.

- [13] U. Rohde, S. Baier, S. Duerigen, E. Fridman, S. Kliem, and B. Merk. Development and verification of the coupled 3D neutron kinetics/thermal-hydraulics code DYN3D-HTR for the simulation of transients in block-type HTGR. *Nuclear Engineering and Design*, 251:412–422, 10 2012.
- [14] Bart L. Sjenitzer and J. Eduard Hoogenboom. Dynamic Monte Carlo Method for Nuclear Reactor Kinetics Calculations. *Nuclear Science and Engineering*, 175(1):94–107, 2013.
- [15] BL Sjenitzer. *The dynamic Monte Carlo method for transient analysis of nuclear reactors*. PhD thesis, Delft university of technology, 2013.
- [16] Jaakko Leppänen. Development of a dynamic simulation mode in SERPENT 2 Monte Carlo code. In *M&C 2013*, pages 117–127, Sun Valley, Idaho, 5 2013. American Nuclear Society.
- [17] Bart L. Sjenitzer and J. Eduard Hoogenboom. Implementation of the dynamic Monte Carlo method for transient analysis in the general purpose code Tripoli. In *M&C 2011: International conference on mathematics and computational methods applied to nuclear science and engineering*, Rio de Janeiro, Brazil, 2011.
- [18] Balazs Molnar, Gabor Tolnai, and David Legrady. A GPU-based direct Monte Carlo simulation of time dependence in nuclear reactors. *Annals of Nuclear Energy*, 132:46–63, 10 2019.
- [19] Liam Russell, Adriaan Buijs, and Guy Jonkmans. G4-STORK: A Geant4-based Monte Carlo reactor kinetics simulation code. *Annals of Nuclear Energy*, 71:451–461, 9 2014.
- [20] Antonios G. Mylonakis, M. Varvayanni, D. G.E. Grigoriadis, and N. Catsaros. Developing and investigating a pure Monte-Carlo module for transient neutron transport analysis. *Annals of Nuclear Energy*, 104:103–112, 6 2017.
- [21] Margaux Faucher. *Coupling between Monte Carlo neutron transport and thermal-hydraulics for the simulation of transients due to reactivity insertions*. PhD thesis, Université Paris-Saclay, 2019.
- [22] Diego Ferraro, Ville Valtavirta, Manuel García, Uwe Imke, Riku Tuominen, Jaakko Leppänen, and Victor Sanchez-Espinoza. OECD/NRC PWR MOX/UO₂ core transient benchmark pin-by-pin solutions using Serpent/SUBCHANFLOW. *Annals of Nuclear Energy*, 147:107745, 11 2020.
- [23] Diego Ferraro, Manuel García, Ville Valtavirta, Uwe Imke, Riku Tuominen, Jaakko Leppänen, and Victor Sanchez-Espinoza. Serpent/SUBCHANFLOW pin-by-pin coupled transient calculations for a PWR minicore. *Annals of Nuclear Energy*, 137:107090, 3 2020.
- [24] Margaux Faucher, Davide Mancusi, and Andrea Zoia. Multi-physics transient simulations with TRIPOLI-4®. *EPJ Web of Conferences*, 247:07019, 2021.
- [25] John R Lamarsh. *Introduction to nuclear reactor theory*. Addison-Wesley, 1966.

-
- [26] James Terrell. Distributions of Fission Neutron Numbers. *Physical Review*, 108(3):783, 11 1957.
- [27] Karl O. Ott and R.J. Neuhold. *Introductory Nuclear Reactor Dynamics*. American Nuclear Society, 1st edition, 1981.
- [28] Paul Reuss. *Neutron physics*. EDP sciences, 2012.
- [29] Eric Dumonteil, Rian Bahran, Theresa Cutler, Benjamin Dechenaux, Travis Grove, Jesson Hutchinson, George McKenzie, Alexander McSpaden, Wilfried Monange, Mark Nelson, Nicholas Thompson, and Andrea Zoia. Patchy nuclear chain reactions. *Communications Physics* 2021 4:1, 4(1):1–10, 7 2021.
- [30] Clélia De Mulatier, Eric Dumonteil, Alberto Rosso, and Andrea Zoia. The critical catastrophe revisited. *Journal of Statistical Mechanics: Theory and Experiment*, 2015(8):P08021, 8 2015.
- [31] Crispin W. Gardiner. *Handbook of stochastic methods - for physics, chemistry and the natural sciences*. Springer Series in Synergetics. Springer Berlin Heidelberg, Berlin, Heidelberg, second edition edition, 1985.
- [32] M. M. R. Williams. *Random Processes in Nuclear Reactors*. Pergamon press, 1st edition edition, 1974.
- [33] Clélia de Mulatier. *A random walk approach to stochastic neutron transport*. PhD thesis, Université Paris-Saclay, 2015.
- [34] Theodore Edward. Harris. *The theory of branching processes*. Springer-Verlag, 1963.
- [35] Alain Hébert. *Applied reactor physics*. Presses internationales Polytechnique, Montréal, 3rd edition, 11 2020.
- [36] G. I. Bell and S. Glasstone. *Nuclear Reactor Theory*. Van Nostrand Reinhold Company, 10 1970.
- [37] Alain Hébert. DRAGON5 and DONJON5, the contribution of École Polytechnique de Montréal to the SALOME platform. *Annals of Nuclear Energy*, 87:12–20, 1 2016.
- [38] Joel Rhodes, Kord Smith, and Deokjung Lee. CASMO-5 Development and Applications. In *PHYSOR 2006*, Vancouver, 2004. Canadian Nuclear Society.
- [39] Richard Sanchez, Igor Zmijarevic, M. Coste-Delclaux, Emiliano Masiello, Simone Santandrea, Emanuele Martinolli, Laurence Villate, Nadine Schwartz, and Nathalie Guler. APOLLO2 YEAR 2010. *Nuclear Engineering and Technology*, 42(5):474–499, 2010.
- [40] Florence Dolci, D Schneider, F Gabriel, J.-M Palau, M Guillo, B Pothet, P Archier, K Ammar, F Auffret, R Baron, A.-M Baudron, P Bellier, L Bourhrara, L Buiron, M Coste-Delclaux, C De, J.-M Do, B Espinosa, E Jamelot, V Jouault, J.-J Lautard, R Lenain, J.-C Le Pallec, L Lei Mao, E Masiello, S Mengelle, F Moreau, P Mosca, M Muniglia, N Odry, V Pascal, S Pastoris, B Roque,

- A Targa, C Patricot, S Santandrea, D Sciannandrone, A Tsilanizara, J.-F Vidal, and I Zmijarevic. APOLLO3[®] : CEA/DEN deterministic multi-purpose code for reactor physics analysis. In *PHYSOR 2016 - Unifying Theory and Experiments in the 21st Century*, Sun Valley, 2016.
- [41] B. A. Lindley, J. G. Hosking, P. J. Smith, D. J. Powney, B. S. Tollit, T. D. Newton, R. Perry, T. C. Ware, and P. N. Smith. Current status of the reactor physics code WIMS and recent developments. *Annals of Nuclear Energy*, 102:148–157, 4 2017.
- [42] Thomas M. Evans, Alissa S. Stafford, Rachel N. Slaybaugh, and Kevin T. Clarno. Denovo: A New Three-Dimensional Parallel Discrete Ordinates Code in SCALE. <http://dx.doi.org/10.13182/NT171-171>, 171(2):171–200, 2017.
- [43] Thomas Haigh, Mark Priestley, and Crispin Rope. Los Alamos bets on ENIAC: Nuclear Monte Carlo simulations, 1947-1948. *IEEE Annals of the History of Computing*, 36(3):42–63, 2014.
- [44] William L. Dunn and J. Kenneth Shultis. *Exploring Monte Carlo Methods*. Elsevier, 2012.
- [45] Joel A Kulesza, Terry R Adams, Jerawan C Armstrong, Simon R Bolding, Forrest B Brown, Jeffrey S Bull, Timothy P Burke, Alexander R Clark, Robert Arthur Forster III, Jesse F Giron, Avery S Grieve, Colin J Josey, Roger L Martz, Gregg W McKinney, Eric J Pearson, Michael E Rising, Clell J Solomon Jr., Sriram Swaminarayan, Travis J Trahan, Stephen C Wilson, and Anthony J Zukaitis. MCNP[®] Code Version 6.3.0 Theory & User Manual. Technical Report LA-UR-22-30006, Rev. 1, Los Alamos National Laboratory, Los Alamos, NM, USA, 9 2022.
- [46] Thomas E Booth. A weight (charge) conserving importance-weighted comb for Monte Carlo. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 1996.
- [47] David Legrady, Mate Halasz, Jozsef Kophazi, Balazs Molnar, and Gabor Tolnai. Population-based variance reduction for dynamic Monte Carlo. *Annals of Nuclear Energy*, 149:107752, 12 2020.
- [48] H Kahn. Modification of the Monte Carlo Method. Technical report, Rand Corporation, 11 1948.
- [49] Gerald Goertzel and Malvin H Kalos. Monte Carlo methods in transport problems. *Progress in nuclear energy*, 2:315–369, 1958.
- [50] M H Kalos, F R Nakache, and J Celnik. *Monte Carlo methods in reactor computations*. Gordon and Breach Publishers, New York, 1968.
- [51] J. E. Hoogenboom. Optimum Biasing of Integral Equations in Monte Carlo Calculations. <http://dx.doi.org/10.13182/NSE79-A19656>, 70(2):210–212, 1979.
- [52] S. R. Dwivedi. Zero Variance Biasing Schemes for Monte Carlo Calculations of Neutron and Radiation Transport Problems. <http://dx.doi.org/10.13182/NSE82-A21413>, 80(1):172–178, 1982.

- [53] H. C. Gupta. A Class of Zero-Variance Biasing Schemes for Monte Carlo Reaction Rate Estimators. <http://dx.doi.org/10.13182/NSE83-A18212>, 83(2):187–197, 1983.
- [54] Iván Lux and László Koblinger. *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations*. CRC Press, 5 1991.
- [55] J. Eduard Hoogenboom. Zero-Variance Monte Carlo Schemes Revisited. *Nuclear Science and Engineering*, 160(1):1–22, 2008.
- [56] Stavros Christoforou and J. Eduard Hoogenboom. A Zero-Variance-Based Scheme for Monte Carlo Criticality Calculations. <http://dx.doi.org/10.13182/NSE09-107>, 167(1):91–104, 2011.
- [57] Davide Mancusi and Andrea Zoia. Zero-variance schemes for kinetic Monte Carlo simulations. *The European Physical Journal Plus* 2020 135:5, 135(5):1–33, 5 2020.
- [58] A. J.M. Plompen, O. Cabellos, C. De Saint Jean, M. Fleming, A. Algora, M. Angelone, P. Archier, E. Bauge, O. Bersillon, A. Blokhin, F. Cantargi, A. Chebboubi, C. Diez, H. Duarte, E. Dupont, J. Dyrda, B. Erasmus, L. Fiorito, U. Fischer, D. Flammini, D. Foligno, M. R. Gilbert, J. R. Granada, W. Haeck, F. J. Hamsch, P. Helgesson, S. Hilaire, I. Hill, M. Hursin, R. Ichou, R. Jacqmin, B. Jansky, C. Jouanne, M. A. Kellett, D. H. Kim, H. I. Kim, I. Kodeli, A. J. Koning, A. Yu Konobeyev, S. Kopecky, B. Kos, A. Krása, L. C. Leal, N. Leclaire, P. Leconte, Y. O. Lee, H. Leeb, O. Litaize, M. Majerle, J. I. Márquez Damián, F. Michel-Sendis, R. W. Mills, B. Morillon, G. Noguère, M. Pecchia, S. Pelloni, P. Pereslavytsev, R. J. Perry, D. Rochman, A. Röhrmoser, P. Romain, P. Romojaro, D. Roubtsov, P. Sauvan, P. Schillebeeckx, K. H. Schmidt, O. Serot, S. Simakov, I. Sirakov, H. Sjöstrand, A. Stankovskiy, J. C. Sublet, P. Tamagno, A. Trkov, S. van der Marck, F. Álvarez-Velarde, R. Villari, T. C. Ware, K. Yokoyama, and G. Žerovnik. The joint evaluated fission and fusion nuclear data library, JEFF-3.3. *The European Physical Journal A* 2020 56:7, 56(7):1–108, 7 2020.
- [59] D A Brown, M B Chadwick, R Capote, A C Kahler, A Trkov, M W Herman, A A Sonzogni, Y Danon, A D Carlson, M Dunn, D L Smith, G M Hale, G Arbanas, R Arcilla, C R Bates, B Beck, B Becker, F Brown, R J Casperson, J Conlin, D E Cullen, M.-A Descalle, R Firestone, T Gaines, K H Guber, A I Hawari, J Holmes, T D Johnson, T Kawano, B C Kiedrowski, A J Koning, S Kopecky, L Leal, J P Lestone, C Lubitz, J I Márquez Damián, C M Mattoon, E A Mccutchan, S Mughabghab, P Navratil, D Neudecker, G P A Nobre, G Noguere, M Paris, M T Pigni, A J Plompen, B Pritychenko, V G Pronyaev, D Roubtsov, D Rochman, P Romano, P Schillebeeckx, S Simakov, M Sin, I Sirakov, B Sleaford, V Sobes, E S Soukhovitskii, I Stetcu, P Talou, I Thompson, S Van Der Marck, L Welsersherrill, D Wiarda, M White, J L Wormald, R Q Wright, M Zerkle, G Žerovnik, and Y Zhu. ENDF/B-VIII.0: The 8 th Major Release of the Nuclear Reaction Data Library with CIELO-project Cross Sections, New Standards and Thermal Scattering Data. *Nuclear Data Sheets*, 148:1–142, 2018.

- [60] K. Shibata, O. Iwamoto, T. Nakagawa, N. Iwamoto, A. Ichihara, S. Kunieda, S. Chiba, J. Katakura, and N. Otuka. JENDL-4.0: A New Library for Innovative Nuclear Energy Systems. *Journal of the Korean Physical Society*, 59(2):1046–1051, 8 2011.
- [61] A. J. Koning, D. Rochman, J. Ch Sublet, N. Dzysiuk, M. Fleming, and S. van der Marck. TENDL: Complete Nuclear Data Library for Innovative Nuclear Science and Technology. *Nuclear Data Sheets*, 155:1–55, 1 2019.
- [62] Laura Clouvel. *Uncertainty quantification of the fast flux calculation for a PWR vessel*. PhD thesis, Université Paris-Saclay, 2019.
- [63] Vivian Salino. *Incertitudes et ajustements de données nucléaires au moyen de méthodes déterministes, probabilistes et de mesures effectuées sur des réacteurs à eau sous pression*. PhD thesis, Polytechnique Montréal, 2022.
- [64] Cyrille de Saint Jean, Pierre Tamagno, Pascal Archier, Gilles Noguere, and Cyrille De Saint Jean. CONRAD – a code for nuclear data modeling and evaluation. *EPJ N - Nuclear Sciences & Technologies*, 7:10, 2021.
- [65] A. J. Koning and D. Rochman. Modern Nuclear Data Evaluation with the TALYS Code System. *Nuclear Data Sheets*, 113(12):2841–2934, 12 2012.
- [66] Nancy M Larson. Updated User’s Guide for Sammy: Multilevel R-Matrix Fits to Neutron Data Using Bayes’ Equations. Technical report, Oak Ridge National Laboratory (ORNL), Oak Ridge, TN, 10 2008.
- [67] Robert Macfarlane, Douglas W. Muir, R. M. Boicourt, Albert Comstock Kahler, III, and Jeremy Lloyd Conlin. The NJOY Nuclear Data Processing System, Version 2016. Technical report, Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), 1 2017.
- [68] Kenichi Tada, Yasunobu Nagaya, Satoshi Kunieda, Kenya Suyama, and Tokio Fukahori. Development and verification of a new nuclear data processing system FRENDY. <http://dx.doi.org/10.1080/00223131.2017.1309306>, 54(7):806–817, 7 2017.
- [69] Dorothea Wiarda, Mark L Williams, Cihangir Celik, and Michael E Dunn. AMPX: A Modern Cross Section Processing System for Generating Nuclear Data Libraries. In *ICNC 2015*, Charlotte, NC, USA, 1 2015. American Nuclear Society.
- [70] D. Dickinson and G. E. Whitesides. The Monte Carlo Method for Array Criticality Calculations. <http://dx.doi.org/10.13182/NT76-A31614>, 30(2):166–189, 2017.
- [71] W. Goad and R. Johnston. A Monte Carlo Method for Criticality Problems1. <http://dx.doi.org/10.13182/NSE59-1>, 5(6):371–375, 6 2017.
- [72] M. R. Mendelson. Monte Carlo Criticality Calculations for Thermal Reactors. <http://dx.doi.org/10.13182/NSE68-A20214>, 32(3):319–331, 6 2017.

- [73] JOHN G. MOORE. The solution of criticality problems by Monte Carlo methods. *Advances in Nuclear Science and Technology*, pages 73–98, 1 1976.
- [74] H. Rief and H. Kschwendt. Reactor Analysis by Monte Carlo. <http://dx.doi.org/10.13182/NSE67-A18401>, 30(3):395–418, 12 2017.
- [75] D E Cullen, C J Clouse, R Procassini, and R C Little. Static and Dynamic Criticality: Are They Different? Technical report, Lawrence Livermore National Laboratory (LLNL), Livermore, CA (United States), 12 2003.
- [76] Forrest B Brown. Fundamentals of Monte Carlo particle transport. *Los Alamos National Laboratory, LA-UR-05-4983*, 2005.
- [77] Taro Ueki, Takamasa Mori, and Masayuki Nakagawa. Error estimations and their biases in Monte Carlo eigenvalue calculations. *Nuclear Science and Engineering*, 125(1):1–11, 1996.
- [78] Taro Ueki, Forrest B. Brown, D. Kent Parsons, and Drew E. Kornreich. Autocorrelation and Dominance Ratio in Monte Carlo Criticality Calculations. <http://dx.doi.org/10.13182/NSE03-04>, 145(3):279–290, 2003.
- [79] J Lieberoth. Monte carlo technique to solve the static eigenvalue problem of the boltzmann transport equation. *Nukleonik*, 11:213– 219, 9 1968.
- [80] RC Gast. Monte Carlo eigenfunction iteration strategies that are and are not fair games (LWBR Development Program). Technical report, Bettis Atomic Power Lab., Pittsburgh, PA (USA), 9 1969.
- [81] D. B. MacMillan. Monte Carlo Confidence Limits for Iterated-Source Calculations. <http://dx.doi.org/10.13182/NSE73-A22590>, 50(1):73–75, 1 2017.
- [82] E. M. Gelbard and R. Prael. Computation of standard deviations in Eigenvalue calculations. *Progress in Nuclear Energy*, 24(1-3):237–241, 1990.
- [83] R. J. Brissenden and A. R. Garlick. Biases in the estimation of Keff and its error by Monte Carlo methods. *Annals of Nuclear Energy*, 13(2):63–83, 1986.
- [84] E. Dumonteil and T. Courau. Dominance ratio assessment and Monte Carlo criticality simulations: Dealing with high dominance ratio systems. *Nuclear Technology*, 172(2):120–131, 2010.
- [85] Nicholas Horelik, Bryan Herman, Matthew Ellis, Shikhar Kumar, Jingang Liang, Benoit Forget, and Kord Smith. Benchmark for evaluation and validation of reactor simulations (BEAVRS), v1. 0.1. crpg.mit.edu, 2018.
- [86] Bryan R. Herman. *Monte Carlo and thermal hydraulic coupling using low-order nonlinear diffusion acceleration*. PhD thesis, Massachusetts Institute of Technology, 2014.
- [87] C. E. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(3):379–423, 1948.

- [88] Taro Ueki and Forrest B. Brown. Stationarity modeling and informatics-based diagnostics in Monte Carlo criticality calculations. *Nuclear Science and Engineering*, 149(1):38–50, 2005.
- [89] Forrest B Brown. On the use of Shannon entropy of the fission distribution for assessing convergence of Monte Carlo criticality calculations. In Canadian Nuclear Society, editor, *ANS topical meeting on reactor physics (PHYSOR 2006)*., Canada, 2006.
- [90] Eric Dumonteil and Fausto Malvagi. Variance Estimation In Monte Carlo Criticality Simulations Application To The Axial Study Of A Pwr Cell With Tripoli-4. In *PHYTRA2 - The Second International Conference on Physics and Technology of Reactors and Applications.*, Fez, Morocco, 9 2011.
- [91] J. Theodore Cox and David Griffeath. Occupation Times for Critical Branching Brownian Motions. <https://doi.org/10.1214/aop/1176992799>, 13(4):1108–1132, 11 1985.
- [92] B. Houchmandzadeh. Clustering of diffusing organisms. *Physical Review E*, 66(5):052902, 11 2002.
- [93] B. Houchmandzadeh. Neutral clustering in a simple experimental ecological community. *Physical Review Letters*, 101(7), 8 2008.
- [94] Bahram Houchmandzadeh. Theory of neutral clustering for growing populations. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 80(5), 11 2009.
- [95] W. R. Young, A. J. Roberts, and G. Stuhne. Reproductive pair correlations and the clustering of organisms. *Nature*, 412(6844):328–331, 7 2001.
- [96] Thomas M. Sutton and Anudha Mittal. Neutron clustering in Monte Carlo iterated-source calculations. *Nuclear Engineering and Technology*, 49(6):1211–1218, 9 2017.
- [97] Thomas M. Sutton. Toward a More Realistic Analysis of Neutron Clustering. *Nuclear Science and Engineering*, pages 1–12, 5 2022.
- [98] D. Brockway, P. Soran, and P. Whalen. Monte-Carlo eigenvalue calculation. *Monte-Carlo Methods and Applications in Neutronics, Photonics and Statistical Physics*, pages 378–387, 7 1985.
- [99] David Legrady and J. Eduard Hoogenboom. Scouting the feasibility of Monte Carlo reactor dynamics simulations. In *International Conference on the Physics of Reactors "Nuclear Power: A Sustainable Resource"*, Interlaken, Switzerland, 2008. Paul Scherrer Institut PSI.
- [100] Bart L Sjenitzer and J Eduard Hoogenboom. General purpose dynamic Monte Carlo with continuous energy for transient analysis. In *PHYSOR 2012 Advances in Reactor Physics Linking Research, Industry, and Education*, Knoxville, Tennessee, USA, 2012. American Nuclear Society.

-
- [101] Bart L. Sjenitzer, J. Eduard Hoogenboom, Javier Jiménez Escalante, and Victor Sanchez Espinoza. Coupling of dynamic Monte Carlo with thermal-hydraulic feedback. *Annals of Nuclear Energy*, 76:27–39, 2 2015.
- [102] Margaux Faucher, Davide Mancusi, and Andrea Zoia. New kinetic simulation capabilities for Tripoli-4®: Methods and applications. *Annals of Nuclear Energy*, 120:74–88, 10 2018.
- [103] Manuele Aufiero, Axel Laureau, Ville Valtavirta, Carlo Fiorina, and Pablo Rubiolo. SERPEN-OpenFOAM coupling in transient mode: simulation of a Godiva prompt critical burst. In *Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC)*, Nashville, Tennessee, 2015. American Nuclear Society.
- [104] Bart L Sjenitzer and J Eduard Hoogenboom. A Monte Carlo Method for Calculation of the Dynamic Behaviour of Nuclear Reactors. *Progress in NUCLEAR SCIENCE and TECHNOLOGY*, 2:716–721, 2011.
- [105] John C. Wagner and Alireza Haghighat. Automated Variance Reduction of Monte Carlo Shielding Calculations Using the Discrete Ordinates Adjoint Function. <http://dx.doi.org/10.13182/NSE98-2>, 128(2):186–208, 1998.
- [106] John C Wagner, Edward D Blakeman, and Douglas E Peplow. Forward-weighted CADIS method for variance reduction of Monte Carlo calculations of distributions and multiple localized quantities. In *International Conference on Mathematics, Computational Methods & Reactor Physics (M&C 2009)*. American Nuclear Society, 2009.
- [107] Théophile Bonnet, Davide Mancusi, and Andrea Zoia. Space and time correlations for diffusion models with prompt and delayed birth-and-death events. *Physical Review E*, 105(6):064105, 6 2022.
- [108] Paul Glasserman, Philip Heidelberger, Perwez Shahabuddin, and Tim Zajic. Multilevel Splitting for Estimating Rare Event Probabilities. <https://doi.org/10.1287/opre.47.4.585>, 47(4):585–600, 8 1999.
- [109] Frédéric Cérou, Arnaud Guyader, and Mathias Rousset. Adaptive multilevel splitting: Historical perspective and recent results. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 29(4):043108, 4 2019.
- [110] Herman Kahn and Theodore E. Harris. Estimation of particle transmission by random sampling. *National Bureau of Standards applied mathematics series*, 12:27–30, 1951.
- [111] Pierre L’Ecuyer, Valérie Demers, and Bruno Tuffin. Splitting for rare-event simulation. *Proceedings - Winter Simulation Conference*, pages 137–148, 2006.
- [112] Pierre Del Moral. *Feynman-Kac Formulae*. Probability and its Applications. Springer New York, New York, NY, 2004.
- [113] Charles Edouard Bréhier and Tony Lelièvre. On a new class of score functions to estimate tail probabilities of some stochastic processes with adaptive multilevel splitting. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 29(3):033126, 3 2019.

- [114] Kévin Fröhlicher, Eric Dumonteil, Loïc Thulliez, Julien Taforeau, and Mariya Brovchenko. Improving the variance in monte carlo criticality calculations with adaptive multilevel splitting. In *International Conference on Physics of Reactors (PHYSOR 2022)*, pages 2512–2521, Pittsburgh, PA, 5 2022. American Nuclear Society.
- [115] K. Fröhlicher, E. Dumonteil, L. Thulliez, J. Taforeau, and M. Brovchenko. Generational variance reduction in monte carlo criticality simulations as a way of mitigating unwanted correlations. *Nuclear Science and Engineering*, (under reviewing).
- [116] F B Brown, R F Barrett, T E Booth, J S Bull, L J Cox, R A Forster, T J Goorley, R D Mosteller, S E Post, R E Prael, E C Selcow, A Sood, J Sweezy, Forrest B Brown, and J T Goorley. MCNP version 5. *Trans. Am. Nucl. Soc.*, 2002.
- [117] E. Brun, F. Damian, C. M. Diop, E. Dumonteil, F. X. Hugot, C. Jouanne, Y. K. Lee, F. Malvagi, A. Mazzolo, O. Petit, J. C. Trama, T. Visonneau, and A. Zoia. TRIPOLI-4®, CEA, EDF and AREVA reference Monte Carlo code. *Annals of Nuclear Energy*, 82:151–160, 8 2015.
- [118] Jaakko Leppänen, Maria Pusa, Tuomas Viitanen, Ville Valtavirta, and Toni Kaltiaisenaho. The Serpent Monte Carlo code: Status, development and applications in 2013. *Annals of Nuclear Energy*, 82:142–150, 8 2015.
- [119] B. Cochet, A. Jinaphanh, L. Heulers, and O. Jacquet. Capabilities overview of the MORET 5 Monte Carlo code. In *SNA + MC 2013 - Joint International Conference on Supercomputing in Nuclear Applications + Monte Carlo*, page 06014, Paris, 2014. EDP Sciences.
- [120] P.F. Zweifel. *Reactor physics*. Virginia Polytechnic Inst. and State Univ., Blacksburg (USA), 1977.
- [121] Michel Nowak, Jilang Miao, Eric Dumonteil, Benoit Forget, Anthony Onillon, Kord S. Smith, and Andrea Zoia. Monte Carlo power iteration: entropy and spatial correlations. *Ann. Nucl. Energy*, 94:856–868, 3 2016.
- [122] Eric Dumonteil and Fausto Malvagi. Automatic treatment of the variance estimation bias in TRIPOLI-4 criticality calculations (Conference) | OSTI.GOV. In *Proceedings of the 2012 International Congress on Advances in Nuclear Power Plants - ICAPP '12*, page 2799, Chicago, 2012. American Nuclear Society, 555 North Kensington Avenue, La Grange Park, IL 60526 (United States).
- [123] Frédéric Cérou, Bernard Delyon, Arnaud Guyader, and Mathias Rousset. A central limit theorem for Fleming–Viot particle systems. <https://doi.org/10.1214/19-AIHP976>, 56(1):637–666, 2 2020.
- [124] Jeffery Lewins. *Importance: the adjoint function*. Pergamon Press, 1965.
- [125] Jaakko Leppanen. *Development of a new Monte Carlo reactor physics code*. PhD thesis, VTT Technical Research Centre of Finland, 2007.

- [126] E. Fridman and J. Leppänen. On the use of the Serpent Monte Carlo code for few-group cross section generation. *Annals of Nuclear Energy*, 38(6):1399–1405, 6 2011.
- [127] Emil Fridman and Jaakko Leppänen. Revised Methods for Few-Group Cross Section Generation in the Serpent Monte Carlo Code R&D on fast systems View project McSAFE: Monte Carlo Based Multiphysics methods for transient analysis of reactor systems View project REVISED METHODS FOR FEW-GROUP CROSS SECTIONS GENERATION IN THE SERPENT MONTE CARLO CODE. In *PHYSOR 2012 – Advances in Reactor Physics – Linking Research, Industry, and Education*. American Nuclear Society, 2011.
- [128] Jaakko Leppänen. Response Matrix Method–Based Importance Solver and Variance Reduction Scheme in the Serpent 2 Monte Carlo Code. <https://doi.org/10.1080/00295450.2019.1603710>, 205(11):1416–1432, 11 2019.
- [129] M. Aufiero, A. Cammi, C. Fiorina, J. Leppänen, L. Luzzi, and M. E. Ricotti. An extended version of the SERPENT-2 code to investigate fuel burn-up and core material evolution of the Molten Salt Fast Reactor. *Journal of Nuclear Materials*, 441(1-3):473–486, 10 2013.
- [130] J. Eduard Hoogenboom and Bart L. Sjenitzer. Extensions of the MCNP5 and TRIPOLI4 Monte Carlo Codes for Transient Reactor Analysis. In *SNA + MC 2013 - Joint International Conference on Supercomputing in Nuclear Applications + Monte Carlo*. EDP Sciences, 2014.
- [131] Ville Valtavirta, Muhammad Hessian, and Jaakko Leppänen. Delayed neutron emission model for time dependent simulations with the Serpent 2 Monte Carlo code - First results. In *PHYSOR 2016 : Unifying Theory and Experiments in the 21st Century*, pages 1568–1582, Sun Valley, 2016. American Nuclear Society.
- [132] A Santamarina, D Bernard, P Blaise, M Coste, A Courcelle, T D Huynh, C Jouanne, P Leconte, O Litaize, S Mengelle, G Noguère, J-M Ruggiéri, O Sérot, J Tommasi, C Vaglio, J-F Vidal, and Y Rugama. The JEFF-3.1.1 Nuclear Data Library Validation Results from JEF-2.2 to JEFF-3.1.1. Technical report, Nuclear Energy Agency, Organisation for Economic Co-operation and Development, 2009.
- [133] E. P. Wigner. Effect of Small Perturbations on Pile Period. *Nuclear Energy*, pages 540–552, 1948.
- [134] H Jr Hurwitz. Note on the theory of danger coefficients. Technical report, Knolls Atomic Power Lab., Schenectady, N.Y., New York, 9 1948.
- [135] Harry Soodak. *The science and engineering of nuclear power*. Addison-Wesley, New York, 1948.
- [136] L. N. Ussachoff. Equation for the importance of neutrons, reactor kinetics and the theory of perturbations. In *Proc. Int. Conf. on the Peaceful Uses of Atomic Energy, Geneva, Switzerland, Aug. 8-21, 1955*, volume 5, pages 503–510, 1956.

- [137] Jeffery Lewins. A derivation of the time-dependent adjoint equations for neutron importance in the transport, continuous slowing-down and diffusion models. *Journal of Nuclear Energy. Part A. Reactor Science*, 13(1-2):1–5, 10 1960.
- [138] Jeffery Lewins. The Time-Dependent Importance of Neutrons and Precursors*. <http://dx.doi.org/10.13182/NSE60-A25713>, 7(3):268–274, 1960.
- [139] J D Lewins. Construction of adjoint equations and their adjoint operators—Terminology? *Annals of Nuclear Energy*, 11(6):311, 1984.
- [140] T. Courau and Guy Marleau. Adjoint and Generalized Adjoint Flux Calculations Using the Collision Probability Technique. <http://dx.doi.org/10.13182/NSE02-A2265>, 141(1):46–54, 2017.
- [141] Nicholas Terranova, Guillaume Truchet, Igor Zmijarevic, and Andrea Zoia. Adjoint neutron flux calculations with Tripoli-4®: Verification and comparison to deterministic codes. *Annals of Nuclear Energy*, 114:136–148, 4 2018.
- [142] Qu Wu, Xingjie Peng, Xiao Tang, Yingrui Yu, Qing Li, and Kan Wang. Whole-core forward-adjoint neutron transport solutions with coupled 2-D MOC and 1-D SN and kinetics parameter calculation. *Progress in Nuclear Energy*, 108:310–318, 9 2018.
- [143] Francesco Tantillo and Simon Richards. Adjoint neutron flux estimator implementation and verification in the continuous energy Monte Carlo code MONK. *Nuclear Engineering and Design*, 356:110368, 1 2020.
- [144] Scott W. Mosher, Seth R. Johnson, Aaron M. Bevill, Ahmad M. Ibrahim, Charles R. Daily, Thomas M. Evans, John C. Wagner, Jeffrey O. Johnson, and Robert E. Grove. ADVANTG An Automated Variance Reduction Parameter Generator, Rev. 1. Technical report, Oak Ridge National Laboratory (ORNL), Oak Ridge, TN (United States), 8 2015.
- [145] Davide Mancusi, Margaux Faucher, and Andrea Zoia. Monte Carlo simulations of the SPERT III E-core transient experiments. *The European Physical Journal Plus*, 137(1):127, 1 2022.
- [146] S. Croft, A. Favalli, D. K. Hauck, D. Henzlova, and P. A. Santi. Feynman variance-to-mean in the context of passive neutron coincidence counting. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 686:136–144, 9 2012.
- [147] Carl Berglöf. *On measurement and monitoring of reactivity in subcritical reactor systems*. PhD thesis, Kungliga Tekniska högskolan (KTH), Stockholm, 2010.
- [148] Martin Hilbert. Scale-free power-laws as interaction between progress and diffusion. *Complexity*, 19(4):56–65, 3 2014.
- [149] Aaron Clauset, Cosma Rohilla Shalizi, and M. E.J. Newman. Power-Law Distributions in Empirical Data. <https://doi.org/10.1137/070710111>, 51(4):661–703, 11 2009.

- [150] Michael Mitzenmacher. A Brief History of Generative Models for Power Law and Lognormal Distributions. <http://dx.doi.org/10.1080/15427951.2004.10129088>, 1(2):226–251, 2011.
- [151] Benjamin Dechenaux, Thomas Delcambre, and Eric Dumonteil. Percolation properties of the neutron population in nuclear reactors. *Physical Review E*, 106(6):064126, 12 2022.