



HAL
open science

Development of numerical methods for neutronics in continuous media

Hunter Belanger

► **To cite this version:**

Hunter Belanger. Development of numerical methods for neutronics in continuous media. Génie civil nucléaire. Université Paris-Saclay, 2022. English. NNT : 2022UPASP111 . tel-04213597

HAL Id: tel-04213597

<https://theses.hal.science/tel-04213597>

Submitted on 21 Sep 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.

Development of numerical methods for neutronics in continuous media

*Développement de méthodes numériques
pour la neutronique en milieu continu*

Thèse de doctorat de l'université Paris-Saclay

École doctorale n°576, particules, hadrons, énergie et noyau :
instrumentation, imagerie, cosmos et simulation (PHENIICS)
Spécialité de doctorat: Energie Nucléaire
Graduate School : Physique, Référent : Faculté des sciences d'Orsay

Thèse préparée dans l'unité de recherche
Service d'Études des Réacteurs et de Mathématiques Appliquées
(Université Paris-Saclay, CEA),
sous la direction d'Andrea ZOIA (Ingénieur-Chercheur, HDR)
et l'encadrement de Davide MANCUSI (Ingénieur-Chercheur)

Thèse soutenue à Paris-Saclay, le 14 octobre 2022, par

Hunter Christophe BELANGER

Composition du jury

Cheikh Diop Directeur de Recherche, CEA/DES	Président
Forrest Brown Professeur, University of New Mexico (Etats-Unis)	Rapporteur
Jaakko Leppänen Professeur, VTT (Finlande)	Rapporteur
Sandra Dulla Professeure, Politecnico di Torino (Italie)	Examinatrice
Davide Mancusi Ingénieur-Chercheur, CEA/DES	Examineur
Elsa Merle Professeure (HDR), Grenoble INP/LPSC	Examinatrice
Gaël Poëtte Ingénieur-Chercheur (HDR), CEA/DAM	Examineur
Andrea Zoia Ingénieur-Chercheur (HDR), CEA/DES	Directeur de thèse

Titre: Développement de méthodes numériques pour la neutronique en milieu continu

Mots clés: Monte-Carlo, transport de particules, annulation de poids, bruit neutronique

Résumé: La simulation Monte Carlo est la méthode “étalon” pour les problèmes de transport de particules en radioprotection, physique des réacteurs, sûreté-criticité et instrumentation nucléaire. Au prix d’un coût de calcul élevé, des solutions de référence sont obtenues, pouvant ensuite être utilisées pour la validation numérique de solveurs déterministes, plus rapides mais approchés. Le développement de nouvelles stratégies de réduction de variance et la disponibilité accrue de ressources HPC ont énormément élargi le champ d’application des simulations Monte Carlo, couvrant des régimes non stationnaires couplés aux multi-physiques, et des observables auparavant inaccessibles. Dans cette thèse, nous illustrerons comment les algorithmes de suivi de particules, au cœur des méthodes de Monte Carlo, doivent évoluer pour faire face à ces nouveaux défis. Nous présenterons d’abord les stratégies de suivi existantes, basées sur la description standard “con-

stante par morceaux” des milieux, et examinerons leurs avantages et leurs inconvénients. Ensuite, nous étendrons ces méthodes au cas des milieux aléatoires et des milieux continus, chacun impliquant des défis spécifiques. Pour le cas des milieux continus, l’objet principal de cette thèse, nous montrerons que la méthode negative-weighted delta tracking est très efficace, puisqu’elle ne nécessite pas une section efficace majorante stricte (contrairement à la méthode delta tracking classique). Afin de surmonter les possibles instabilités dues à la présence de particules de poids négatifs dans cet algorithme, nous développerons et appliquerons avec succès des techniques d’annulation de poids, exactes ou approchées. Enfin, en nous appuyant sur ces résultats, nous montrerons que l’annulation de poids est également essentielle pour assurer la convergence et réduire considérablement l’incertitude statistique dans les problèmes de bruit neutronique.

Title: Development of numerical methods for neutronics in continuous media

Keywords: Monte Carlo, particle transport, weight cancellation, neutron noise

Abstract: Monte Carlo simulation is the “gold standard” for particle transport problems emerging in radiation shielding, reactor physics, criticality-safety and nuclear instrumentation. Reference solutions with almost no approximations can be obtained, at the expense of high computational cost, which can then be used for the numerical validation of faster but approximate deterministic solvers. The development of novel variance reduction strategies and the increased availability of HPC resources has enormously widened the field of application for Monte Carlo simulations, covering non-stationary regimes with multi-physics feedback, and previously inaccessible observables. In this PhD thesis we will illustrate how particle-tracking algorithms, which lie at the heart of Monte Carlo methods, should evolve in order to cope with these new challenges. We will first review the existing particle-tracking strategies, based

on the standard piece-wise-constant medium description, and examine their merits and drawbacks. Then, we will extend these methods to the case of random media and continuously-varying media, each involving specific challenges. For the case of continuously-varying media, which is the main focus of this thesis, we will show that negative-weighted delta tracking is highly effective, in that it does not require a strict majorant cross section (contrary to regular delta tracking). In order to overcome the possible instabilities due to the presence of negative-weighted particles in this algorithm, we will develop and successfully test exact and approximate weight cancellation techniques. Finally, by building on these findings, we will show that weight cancellation can also be key to achieving convergence and greatly reducing the statistical uncertainty in neutron noise problems.

Contents

1	Introduction	9
1.1	New Challenges for Monte Carlo Particle Simulations in Reactor Physics	10
1.2	The Voyage from Continuous Media to Neutron Noise	10
1.3	Chenille	11
1.3.1	Geometry	12
1.3.2	Particle Tracking	12
1.3.3	Physics	12
1.3.4	Miscellaneous	13
1.4	Structure of Thesis	14
	References	15
I	Particle Tracking Methods	18
2	Particle Tracking in Continuous Media	19
2.1	Overview of Existing Tracking Methods	19
2.1.1	Surface Tracking	20
2.1.2	Delta Tracking	21
2.1.3	Negative-Weighted Delta Tracking	23
2.1.4	Regionalization	24
2.2	Transport in Stochastic Geometries	25
2.2.1	Properties of Poisson Tessellations	26
2.2.2	Particle Tracking in Random Media	27
2.3	Transport in Continuous Media	30
2.3.1	Sampling Flight Distances with Spatially-Continuous Cross Sections	32
	References	34
3	Optimization of Particle Tracking Methods for Stochastic Media	38
3.1	Introduction	39
3.2	Markov Media	40
3.3	Particle Tracking Methods	41
3.3.1	Surface Tracking	41
3.3.2	Delta Tracking	42
3.3.3	Neighbor Maps	42
3.3.4	Volume Search Mesh	43
3.3.5	Geometry Kernel Memory	43
3.4	Benchmark Specifications	43
3.5	Simulation Results	44
3.6	Conclusions	45

4	Review of Monte Carlo Methods for Particle Transport in Continuously-Varying Media	49
4.1	Introduction	50
4.2	Particle Tracking in Continuous Media	53
4.2.1	Ray tracing in Piecewise Homogeneous Media	53
4.2.2	Delta (Woodcock) Tracking	54
4.2.3	Direct Sampling	55
4.2.4	New (and Old) Delta Tracking Variants	55
4.3	Benchmarking in a 1D System	57
4.3.1	Examined Transport Methods	57
4.3.2	Evaluation of Methods	59
4.4	Results and Discussion	59
4.4.1	Integral Collision Rate and Leakage	59
4.4.2	Collision Density	60
4.4.3	Regional Splitting Effects on FOM	62
4.4.4	Collision Estimators	64
4.4.5	Discussion	65
4.5	Conclusions	65
5	Conclusions for Particle Tracking Methods	67
5.1	Acceleration Techniques for Stochastic Geometries	67
5.2	Transport in Spatially Continuous Media	67
II	Weight Cancellation	69
6	Negative-Weighted Delta Tracking with Weight Cancellation	70
6.1	Failure of Negative-Weighted Delta Tracking with Power Iteration	70
6.1.1	Accommodating Negative Weights in Power Iteration	70
6.1.2	Coupled Transport Equations	72
6.1.3	Necessity of Weight Cancellation	75
6.2	Approximate Regional Weight Cancellation	76
6.3	Exact Regional Weight Cancellation	77
6.3.1	Conceptual Introduction	78
6.3.2	Conditions for the Unbiasedness of Regional Cancellation	79
6.3.3	Optimization of Exact Regional Cancellation	82
	References	83
7	Solving Eigenvalue Transport Problems with Negative Weights and Regional Cancellation	86
7.1	Introduction	87
7.2	Transport with Negative Particles	89
7.3	Exact Regional Cancellation: 1D Benchmark	90
7.4	Approximate Regional Cancellation: C5G7	92
7.5	Conclusions	94

8	Exact Weight Cancellation in Monte Carlo Eigenvalue Transport Problems	97
8.1	Introduction	98
8.2	Power Iteration Debacle	99
8.3	Necessity of Weight Cancellation	102
8.3.1	Failure of Power Iteration	102
8.3.2	Modeling Weight Cancellation	104
8.3.3	Deterministic Proof of Concept	105
8.4	An Exact Regional Cancellation Scheme for 1D Problems	106
8.4.1	Results of the 1D Methodology	108
8.5	Exact 3D Multigroup Regional Cancellation Scheme	109
8.6	Implementation and Results	111
8.7	Conclusion	114
8.8	Appendix: Derivation of Coupled Boltzmann Transport Equations	115
9	Unbiasedness and Optimization of Regional Weight Cancellation	119
9.1	Introduction	120
9.2	Regional Weight Cancellation	121
9.3	Unbiasedness of Cancellation	122
9.3.1	Integral Formulation of the Transport Equation	123
9.3.2	Averaging over all Scattering Events	124
9.3.3	Expected-Value Estimators for Cancellation	126
9.3.4	Intermediate Collision Points	127
9.3.5	Non-Uniform Cancellation	129
9.3.6	Fission Emission Density Function with Delta-Tracking Schemes	129
9.4	Optimization of Cancellation Efficiency	130
9.4.1	Replacing ζ with $\langle \zeta \rangle$	131
9.4.2	Optimization of $\langle \Gamma \rangle$	131
9.4.3	Small Region Limit	131
9.5	Monte Carlo Implementation	132
9.5.1	Cancellation with Distributed Memory Simulations	132
9.5.2	Calculation of $\langle \zeta \rangle$ and $\langle 1/\zeta \rangle$	132
9.5.3	Heterogeneous Cancellation Regions	132
9.5.4	Monte Carlo Implementation in the Open-Source Code MGMC	133
9.6	Simulation Results	133
9.6.1	Comparison of Optimization Strategies	133
9.6.2	Strategies for Evaluating the Average Fission Emission Densities	134
9.6.3	Demonstration of Heterogeneous Cancellation Regions on the C5G7 Benchmark	135
9.7	Conclusion	135
9.8	Appendix A: Optimization of Γ^*	136
9.9	Appendix B: Optimization of $\langle \Gamma \rangle$	136

10	Conclusions for Weight Cancellation	139
10.1	Negative-Weighted delta tracking with Power Iteration	139
10.2	Weight Cancellation	139
III	Neutron Noise	141
11	Application of Weight Cancellation to Neutron Noise	142
11.1	Derivation of the Canonical Noise Equations	143
11.2	Sampling of the Noise Source	145
11.2.1	Cross Section Oscillations	145
11.2.2	Mechanical Vibrations	146
11.2.3	Frequency Dependence of the Noise Equation	149
11.3	Transport of Noise Particles	149
11.4	Branchless Sampling for the Noise Source	151
11.5	Weight Cancellation for Variance Reduction in Noise Problems	153
11.5.1	How to Apply Weight Cancellation to Fixed-Source Problems	153
11.5.2	Observed Performance Improvements	155
	References	156
12	Variance Reduction Techniques for Monte Carlo Neutron Noise Simulations	159
12.1	Introduction	160
12.2	Monte Carlo Simulations of Neutron Noise	161
12.3	Branchless Noise Source Sampling	162
12.4	Weight Cancellation	164
12.5	Simulation Results	164
12.5.1	Effects of Branchless Noise Source Sampling on the FOM	166
12.5.2	Effects of Weight Cancellation on the FOM	166
12.6	Conclusions	167
13	Variance Reduction and Noise Source Sampling Techniques for Monte Carlo Simulations of Neutron Noise Induced by Mechanical Vibrations	170
13.1	Introduction	171
13.2	The Neutron Noise Equations	172
13.3	Sampling the Neutron Noise Source	174
13.3.1	Cross Section Oscillations	174
13.3.2	Mechanical Vibrations	175
13.4	Transport of Noise Particles	178
13.5	Weight Cancellation for Variance Reduction	179
13.5.1	When to Perform Cancellation	179
13.5.2	Approximate Cancellation	180
13.5.3	Exact Cancellation	180
13.6	Simulation Results	182

13.6.1	Sampling of the Noise Source	183
13.6.2	Application of Weight Cancellation	183
13.6.3	The Effects of Vibration Frequency and Amplitude	188
13.7	Conclusions	188
14	Conclusions for Neutron Noise	192
15	Conclusions and Future Work	193
	References	195
A	Résumé Détaillé en Français	197
A.1	Introduction	197
A.1.1	Nouveaux Défis pour les Simulations Monte Carlo en Physique des Réacteurs	198
A.1.2	Des Milieux Continus au Bruit Neutronique	199
A.1.3	Chenille	200
A.1.4	Structure de la Thèse	201
A.2	Conclusions et Travaux Futurs	202
	References	205

Acknowledgements

The emotions that one feels as they near the end of a Ph.D. are rather perplexing; amongst the sentiments of excitement and relief which accompany the impending precipice of achieving the zenith of many years of work and research, a somberness inevitably seeps in. To me, these feelings of sorrow are an indicator of having cultivated relationships of significant importance during the past three years. Completing a doctorate can be a difficult endeavor, and having the right people by your side makes it all the easier, and more enjoyable. There are therefore several people to whom I would like to express my gratitude.

I would first like to thank the members of my committee, Sandra Dulla, Elsa Merle, Gaël Poëtte, and Cheikh Diop, for having taken the time out of their busy schedules to prepare for and participate in my defense. This is doubly true for Forrest Brown and Jaakko Leppänen, who additionally took the time to review my manuscript and provide detailed evaluations and comments on my work.

To my thesis director, Andrea Zoia, and my supervisor, Davide Mancusi, the words I write here could never properly express my gratitude for your mentorship and guidance. Both of you have been instrumental in my formation as a researcher and Monte Carlo developer. I have truly cherished our many hours of discussions, and laughter.

My family has always been my bedrock. I will never be able to thank all of you enough for your support and encouragement, not just during my Ph.D., but throughout my entire life.

Before beginning my doctoral studies in France, I spent five years completing a Bachelors and Masters degree at Rensselaer Polytechnic Institute (RPI). While there are many people from my time at RPI to whom I owe a great deal of thanks, none more so than Jenn Collins and James Flamino. You somehow made the many hundreds of hours spent on home work and studying enjoyable.

J'aimerais remercier deux autres doctorants du SERMA : Théophile Bonnet et Emeline Rosier. Sans vous, la thèse aurait été beaucoup moins désopilante, et plus ennuyeuse.

Finalement, je souhaite remercier Héloïse Muller, pour ton soutien et tes encouragements. Je n'ignore point le fait que j'ai tellement de chance de t'avoir dans ma vie. Tu m'as montré que Le Petit Prince avait bien raison, parce que je ne verrais plus rien sans toi à mes côtés.

1 - Introduction

The distribution of neutrons and photons in phase space within the systems of interest in radiation shielding and nuclear reactor physics applications can be described by the Boltzmann transport equation. Two families of numerical methods exist to solve the Boltzmann equation for neutrons. Deterministic methods are the best known category, and encompass a variety of techniques which rely on the discretization of phase space. Virtually all deterministic solvers utilize the multi-group approximation, where nuclide cross sections are approximated as being constant within a given energy interval [1]. The direction component of the solutions are often discretized along several discrete ordinates (which leads to the S_n methods), or can be approximated as a finite sum of spherical harmonics (which leads to the P_n methods) [2]. Spatial coordinates are typically discretized by means of a geometric mesh, although certain techniques such as the Method of Characteristics do not require a spatial mesh but assume that the neutron source is a piece-wise constant or piece-wise linear function of the phase space coordinates [2]. Deterministic codes such as APOLLO3[®], developed at CEA, are able to solve the transport equation relatively fast,¹ and are therefore the most commonly used solvers in the nuclear industry. In order to finely discretize the six dimensions of phase space (or seven, for non-stationary problems where time must be added to the phase space variables), the number of degrees of freedom for a full reactor problem could reach 10^{21} , which is beyond reach of the current generation of computers [3]. To alleviate this problem, reduced-order models and approximations are introduced in the deterministic solvers, which adds a modeling bias, in addition to the unavoidable discretization errors, in the final results. This is particularly delicate for the treatment of the energy variable; the use of multi-group cross sections, which requires highly sophisticated self-shielding models, might result in errors whose magnitude is very difficult to predict [2].

The Monte Carlo method was initially developed during the 1940's, to solve the Boltzmann equation by a probabilistic approach [4]. In this technique, a population of neutrons is followed through a sequence of free flights and collisions with the traversed materials. The evolution of each particle is inherently stochastic and must be sampled according to the physical laws (cross sections, energy-angle distributions, and yields) provided in the nuclear data libraries. Random numbers are used in conjunction with known probabilities for different events to decide the fate of the particle, from birth all the way to death due to capture or leakage. Each time an event of interest is sampled, the corresponding particle contributes to the estimated value of the observable associated with the event, known as a score or tally. After the entire population has been processed², the ensemble average over all the particle contributions provides an unbiased estimate of the sought observable.

Knowledge of the flight and collision kernels of the Boltzmann equation (in addition to the source) is sufficient to sample the particle trajectories: a key advantage of the Monte Carlo method is that it does not require the discretization of the Boltzmann equation (although the phase space must be decomposed into regions to record the sampled events for each portion of interest). Different quantities such as flux, current, and reaction rates can be estimated by averaging the contributions of the particles to the corresponding scores within the given portion of phase space. By directly simulating the physical process for individual particles, it is possible to obtain an unbiased solution with no approximations, and to estimate the associated statistical uncertainty (i.e. the level of confidence that we can have in the obtained result).³ The absence

¹It is generally accepted that deterministic solvers are much faster with respect to Monte Carlo solvers. This is to be nuanced, since the calculation time for deterministic codes typically does not include the time spent performing self-shielding calculations and homogenization to obtain multi-group cross sections.

²The use of random numbers with different seeds to sample stochastic processes ensures that the results of each simulation are different and independent. A deterministic code will always yield the same result for a given problem.

³The Monte Carlo solution is exact for the given nuclear data, geometry, and material compositions. If there are uncertainties in the geometric configuration and material composition, or if the provided nuclear data is inadequate, the results from a Monte Carlo simulation may not align with experimental results.

of bias makes Monte Carlo methods the gold standard for the verification of deterministic codes. The main drawback of Monte Carlo simulations is that they are slower than deterministic solvers, notably because the statistical uncertainty decreases as $1/\sqrt{N}$ as a function of the number of simulated particles, N . For this reason, traditionally the use of Monte Carlo codes in the nuclear industry is limited compared to deterministic solvers, and is typically restricted to stationary problems.

1.1 . New Challenges for Monte Carlo Particle Simulations in Reactor Physics

In the past few decades, the computational resources which are available to most users has increased drastically. Monte Carlo simulations which used to take hours or days on once state-of-the-art super-computers can now be run on a personal laptop in a matter of minutes. While Monte Carlo codes were once almost exclusively used for the study of research reactors or full cores in stationary conditions, several projects in Europe (HPMC⁴ and McSAFE⁵) and in the USA (CESAR⁶ and CASL⁷) have stimulated new simulation methods and variance reduction techniques enabling unprecedented achievements. Monte Carlo codes are now used to perform full core analysis for fuel depletion calculations with coupled thermal-hydraulics and thermo-mechanics solvers to take into account multi-physics feedback [5–8]. Recently, dynamic Monte Carlo simulations of full core transients including the time scale of the delayed neutron precursors, with coupled multi-physics have been performed [7, 9, 10]. With the continual improvements being made to high performance computing (HPC) infrastructures, the number of possibilities for Monte Carlo simulations in reactor physics is still expanding. New types of simulations which have never been previously considered are now being examined. Some research has focused on solving for the higher harmonics of the transport equation, to improve source convergence and calculate the dominance ratio, which is key to reactor safety [11, 12]. Another new topic of interest, developed in the framework of the EU H2020 CORTEX project (2017-2021), has been solving the neutron noise equation in the frequency domain, in order to understand the effects of vibrating fuel pins and assemblies in a reactor core [13, 14].

Both the advancements in HPC, and the investigation of new types of simulations, force Monte Carlo code developers to completely reconsider how the transport routines are implemented. The first such example is perhaps the development of techniques to perform on the fly Doppler broadening of cross sections [15, 16]. Another prime example is the rewrite of codes from a history-based to an event-based algorithm for better utilization of GPUs [17]. As GPUs were developed to be most efficient when performing the same operation on multiple pieces of data, efficiently harnessing their computational potential requires sorting particles which are undergoing the same type of reaction. It is clear from these examples that the advancement of Monte Carlo analysis for nuclear reactor physics will require code developers to reconsider traditional methods, inventing new techniques that better match the constraints of modern computing architectures.

1.2 . The Voyage from Continuous Media to Neutron Noise

The primary goal of this thesis is to research a possible approach to improve the fidelity of the modeling of material properties in relation to particle transport by Monte Carlo simulations. In most current production-level Monte Carlo codes, the material properties are assumed to be piece-wise constant functions of spatial position. Each elementary volume (or cell) of the geometric model typically has an associated temperature, density, and isotopic composition: within this spatial region, these physical quantities are assumed to be spatially homogeneous. Except in a few cases, this hypothesis is typically not met in reality, as all of these properties may be continuous functions of the spatial position. This is particularly relevant in reactor physics applications, when we consider neutron transport simulations which are coupled with thermal-hydraulic, thermo-mechanic, and depletion fields: in order to take into account multi-physics feedback, it is necessary

⁴High performance Monte Carlo reactor core analysis. <https://www.fp7-hPMC.eu>

⁵High-Performance Monte Carlo Methods for SAFETY Demonstration. <https://cordis.europa.eu/project/id/755097>

⁶Center for Exascale Simulation of Advanced Reactors.

⁷Consortium for Advanced Simulation of Light Water Reactors. <https://casl.gov>

to look-up the temperature, density, and isotopic concentrations at any position in the reactor, which makes the material properties seen by the neutrons along their displacements also continuously dependent on the particle position. Utilizing this information would enable a more accurate representation of real world systems. Libraries are already being developed to allow Monte Carlo neutronics codes to query temperatures and densities calculated by the multi-physics solvers in memory [18]. Spatially continuous cross sections could also be applied to depletion problems, where the isotopic concentrations can vary spatially. In current codes, fuel pins must be discretized into many rings (≈ 10), in order to account for skin effects. Spatially continuous isotopic concentrations could potentially make this ring discretization unnecessary.

The temperature, density, and isotopic concentrations are all used to determine the macroscopic cross section at a given position and energy. If these material properties are continuous functions of position, then so is the macroscopic cross section. Sampling how far a particle will travel before undergoing a collision is an integral component to the Monte Carlo algorithm, as will be outlined in Sec. 2.3. This task becomes quite difficult, however, once the cross sections are no longer piece-wise constant. Our goal was therefore to first critically review the existing Monte Carlo particle tracking methods, and to explore how they can be adapted to sample flight distances from continuously-varying cross sections.

In Part I, we consider several different particle tracking algorithms, and examine their performance when used in fixed-source problems with spatially-continuous cross sections. Of the considered algorithms, our analysis shows that a method that we refer to as negative-weighted delta tracking [19, 20] is the most promising. Negative-weighted delta tracking has the peculiarity that particles can have either a positive or negative statistical weight. After this initial investigation in Chapter 4, we attempt to examine the performance of negative-weighted delta tracking in k -eigenvalue power iteration simulations. Although the algorithm performs very well in fixed-source problems, all of the power iteration simulations fail to complete, much to our surprise. In Part II we analyze why power iteration with negative-weighted delta tracking fails, and propose a strategy based on weight cancellation as a solution. Loosely speaking, weight cancellation is an operation which takes a positively-weighted particle and a negatively-weighted particle and “combines” them in such a way that their weights may cancel with one another. Accomplishing this in an unbiased manner in a Monte Carlo simulation is highly nontrivial, as particles are each located at different phase-space coordinates, and it is therefore impossible for a positive and negative particle to end up at the exact same coordinates. We develop two different weight cancellation methods in Part II, and apply them to power iteration with negative-weighted delta tracking: we show that this approach indeed allows these simulations to complete normally.

While the development of weight cancellation techniques initially appears unrelated to the problem of spatially-continuous media, it is absolutely mandatory when using negative-weighted delta tracking. Other problems in the field of nuclear reactor physics exist which could also potentially benefit from the application of weight cancellation. One such case is solving the neutron noise equation in the frequency domain. A previous PhD at CEA focused on the development of Monte Carlo methods to solve the neutron noise equation [21], and we decided that the weight cancellation techniques developed in Part II ought to be tested in this domain. Part III of this thesis applies the newly developed weight cancellation techniques to the neutron noise problem as a variance reduction technique.

1.3 . Chenille

TRIPOLI-4[®] is the production-level, reference Monte Carlo code for particle transport which has been under development at CEA since the mid 90’s [22]. As a production-level code used in industry, TRIPOLI-4[®] has a plethora of features (including the possibility of propagating neutrons, photons and electrons/positrons in the electromagnetic shower) for applications ranging from reactor physics to radiation shielding and nuclear instrumentation: currently the code base sits at approximately 400 kSLOC (Source Lines Of Code). Attempting to make even small changes in such a large program can be challenging, especially for students who have yet to become acquainted with the code and its peculiarities. The work presented in this thesis required implementing novel particle transport algorithms, restructuring how power iteration and fixed-

source transport problems are decomposed, and adding weight cancellation meshes. All these tasks are highly invasive with respect to the architecture of a Monte Carlo code: doing the work which was required by the scientific goals set for the course of this thesis would have been hardly feasible in TRIPOLI-4[®], under the time constraints of a PhD program. As such, although the implementations existing in TRIPOLI-4[®] were often taken as a starting point, all of the new algorithms which were conceived and tested were implemented in *Chenille*, a new Monte Carlo mini-app developed from scratch over the course of this thesis.

Chenille is a Monte Carlo mini-app which is able to simulate the transport of neutrons in three-dimensional geometries, using either multi-group or continuous-energy nuclear data. Being only ≈ 20 kSLOC, and written in a very modular manner, *Chenille*'s size and code style facilitate the fast implementation and testing of new algorithms, even those requiring extensive rewriting of the existing architecture. Due to *Chenille*'s importance to this thesis, we will briefly outline some of its characteristics and capabilities. We mention that some of the one-dimensional single-speed simulations, which are found at various locations in the manuscript for benchmark calculations, were run in a different toy program, also developed during this thesis [23].

1.3.1 . Geometry

Chenille is able to perform transport in general three-dimensional geometries. Surfaces such as planes, cylinders, and spheres can be defined by the user, each with a unique identifier. The configuration space can be bisected by the equation of a surface. All points in space are either on the positive side, or the negative side of the surface. This combination of a side and surface is sometimes referred to as a half-space [24]. Half-spaces are combined using Boolean operators to define a volume, which is associated with a material to define a cell. Unlike in other codes, a cell can only be filled by a material in *Chenille*. Cells can be grouped together into universes, which can in turn be placed inside of a lattice. A lattice is a special generalization of a universe in *Chenille*, which is used to replicate the same universe in a structured and repeated manner. Their main use case is the construction of fuel assemblies, and then the placement of fuel assemblies within the core. There are two available lattices in *Chenille*, regular rectilinear lattices, and regular hexagonal lattices. Not being able to nest universes inside of cells is admittedly a limitation, although for most reactor physics applications this has not been an issue. A built-in plotting functionality allows the user to plot a two-dimensional slice of the geometry, which helps ensure that the input file is correct.

1.3.2 . Particle Tracking

As one of the main investigations of this thesis was the examination of different particle tracking methods, *Chenille* has several options. Standard surface tracking is the default tracking method. Delta tracking is also available; currently it is not possible to use a hybrid combination of surface tracking and delta tracking, as is done in *Serpent* [25]. The majorant cross section is calculated by unionizing the energy grid of all nuclides present in the problem, and then evaluating the total macroscopic cross section of all materials at all energy points. A variant of negative-weighted delta tracking, as proposed by Carter et al., has also been implemented in *Chenille* [19]. When in multi-group mode, either the majorant cross section, or a user-provided sampling cross section is used, as is the case in Chapter 8; currently, only the majorant cross section is used in continuous-energy mode, making it equivalent to delta tracking.

1.3.3 . Physics

Three different simulation modes are available: fixed-source, k -eigenvalue, and neutron noise. All three modes are available in both multi-group and continuous-energy particle transport. Macroscopic multi-group cross sections are provided for each material in the input file. It is possible to simulate any arbitrary number of energy groups. The user may also provide up to five Legendre moments for anisotropic scattering. Fission is always isotropic, and the fission spectrum is assumed to be independent of the incident energy. For continuous-energy physics, *Chenille* uses the Papillon Nuclear Data Library (PapillonNDL), which is developed and maintained by myself [26]. PapillonNDL is a C++20 library which reads ACE files for continuous-energy neutron data, and is released as free and open-source software under the GPLv3 license. It provides the classes which are responsible for evaluating microscopic cross sections, and sampling scattering

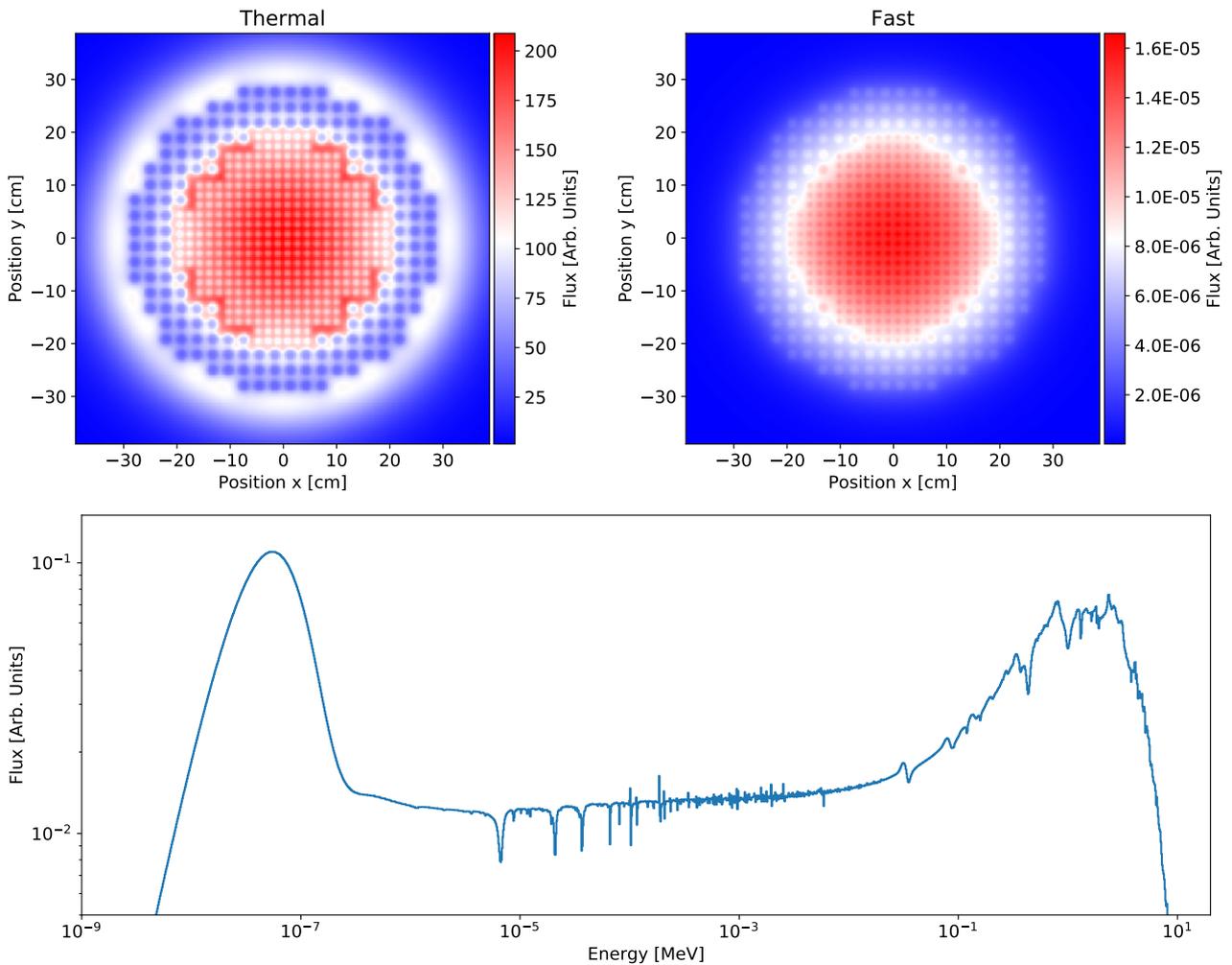


Figure 1.1: Flux profile for the CROCUS research reactor at EPFL, simulated using Chenille.

and fission distributions. Thermal scattering laws are also supported, and can be used with PapillonNDL and Chenille. Currently, Chenille does not utilize unresolved resonance region probability tables. Basic verification has been done using a suite of analytic multi-group problems [27], and comparison of results against OpenMC for a set of criticality benchmark problems in continuous-energy. Figure 1.1 depicts the static flux for the CROCUS research reactor benchmark [28], calculated using Chenille. A previous study conducted with TRIPOLI-4[®] using ENDF/B-VII.0 estimated a multiplication factor of $k_{\text{eff}} = 1.00249 \pm 0.00002$ [29]. Using ACE files produced from the ENDF/B-VII.0 evaluation [30] and processed using version 2 of FRENDDY [31], a multiplication factor of $k_{\text{eff}} = 1.00249 \pm 0.00002$ was obtained by Chenille. Despite not using probability tables for the unresolved resonance region, this estimate is in excellent agreement with the results from TRIPOLI-4[®] and MCNP [29].

1.3.4 . Miscellaneous

Chenille has a basic tally system which can score the flux or reaction rates over a regular rectilinear mesh. Both collision estimators and track-length estimators are available, although the track-length estimators for reaction rates are only available in surface tracking. At the end of each simulation, all the tallies are written to binary NumPy files, allowing the data to be read quickly in Python, for data analysis and visualization [32].

Chenille has been written with both shared memory and distributed memory parallelism, using the OpenMP and MPI standards respectively. Input files are written in YAML, making them very easy to read and write. A multi-group-only version of Chenille, referred to as MGMC, has been made publicly available as free software under the CeCILL-v2.1 license [33]. MGMC has all of the main features of Chenille, but

lacks the necessary classes and functions to interact with the PapillonNDL API.

1.4 . Structure of Thesis

This thesis is organized into three main parts, each focusing on a specific theme:

1. Part I is devoted to assessing the performance of a variety of different particle tracking methods. Two different use cases are considered: stochastic media and spatially continuous cross sections.
2. Part II considers the application of negative-weighted delta tracking to k -eigenvalue power iteration problems, showing that weight cancellation algorithms are mandatory for these simulations to converge.
3. Part III applies the newly developed weight cancellation methods to neutron noise simulations, as a variance reduction technique.

Each part of the manuscript begins with an introductory chapter, which is followed by several research publications (each in a separate chapter), and then a final concluding chapter. The introductory chapter is intended to provide the reader with the necessary prerequisite knowledge and tools that will then be used in the problems described in the subsequent publications. Some of the main methods used in the papers are outlined, and the important results are summarized. All of the detailed technical analysis and results of numerical simulations are provided in these papers. They are presented in an order which should allow for a smooth transition between the different parts of the manuscript and make the train of thought clear. One investigation, while answering some questions, leads to others in a slightly different area; eventually, after enough questions have been asked and then answered, it is quite possible to find oneself very far from where one started (which is the definition of a random walk, as simulated by Monte Carlo!). Finally, each thematic part is terminated with a concluding chapter, which summarizes the major results, and mentions the open questions that should be addressed by future research. The list of chapters which have either been previously published, or been submitted for publication, is provided below. Along with each chapter, one will find the appropriate reference for the associated publication.

Chapter 3: H. Belanger, C. Larmier, D. Mancusi, and A. Zoia, "Optimization of Particle Tracking Methods for Stochastic Media," In *Proceedings of the International Conference on Physics of Reactors 2022 (PHYSOR 2022)*, May 2022, Pittsburgh, PA, p. 294-303.

Chapter 4: H. Belanger, D. Mancusi, and A. Zoia, "Review of Monte Carlo methods for particle transport in continuously-varying media," *European Physical Journal Plus*, vol. 135, no. 11, p. 877, 2020, doi: 10.1140/epjp/s13360-020-00731-y.

Chapter 7: H. Belanger, D. Mancusi, and A. Zoia, "Solving Eigenvalue Transport Problems with Negative Weights and Regional Cancellation," In *Proceedings of the The International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering 2021 (M&C 2021)*, October 2021, p. 46-55.

Chapter 8: H. Belanger, D. Mancusi, and A. Zoia, "Exact weight cancellation in Monte Carlo eigenvalue transport problems," *Physical Review E*, vol. 104, no. 1, p. 015306, 2021, doi: 10.1103/physreve.104.015306.

Chapter 9: H. Belanger, D. Mancusi, and A. Zoia, "Unbiasedness and optimization of regional weight cancellation," *Physical Review E*, vol. 106, no. 2, p. 025302, 2022, doi: 10.1103/physreve.106.025302.

Chapter 12: H. Belanger, D. Mancusi, and A. Zoia, "Variance Reduction Techniques for Monte Carlo Neutron Noise Simulations," In *Proceedings of the International Conference on Physics of Reactors 2022 (PHYSOR 2022)*, May 2022, Pittsburgh, PA, p. 544-553.

Chapter 13: H. Belanger, D. Mancusi, A. Rouchon, and A. Zoia, "Variance Reduction and Noise Source Sampling Techniques for Monte Carlo Simulations of Neutron Noise Induced by Mechanical Vibrations," *Nuclear Science and Engineering*, accepted for publication.

References

- [1] G. Bell and S. Glasstone, *Nuclear Reactor Theory*. Van Nostrand Reinhold Company, 1970.
- [2] A. Hébert, *Applied Reactor Physics*, 3rd ed. Presses Internationales Polytechnique, 2020.
- [3] M. Coste-Delclaux, C. Diop, A. Nicolas, and B. Bonin, "La neutronique," 2013.
- [4] N. Metropolis and S. Ulam, "The Monte Carlo Method," *Journal of the American Statistical Association*, vol. 44, no. 247, pp. 335–341, 1949.
- [5] D. J. Kelly, A. E. Kelly, B. N. Aviles, A. T. Godfrey, R. K. Salko, and B. S. Collins, "MC21/CTF and VERA multiphysics solutions to VERA core physics benchmark progression problems 6 and 7," *Nuclear Engineering and Technology*, vol. 49, no. 6, p. 1326–1338, 2017.
- [6] J. Yu, H. Lee, M. Lemaire, H. Kim, P. Zhang, and D. Lee, "MCS based neutronics/thermal-hydraulics/fuel-performance coupling with CTF and FRAPCON," *Computer Physics Communications*, vol. 238, p. 1–18, 2019.
- [7] A. Levinsky, V. Valtavirta, F. P. Adams, and V. N. Anghel, "Modeling of the SPERT transients using Serpent 2 with time-dependent capabilities," *Annals of Nuclear Energy*, vol. 125, p. 80–98, 2019.
- [8] D. Ferraro, M. Garcia, U. Imke, V. Valtavirta, J. Leppänen, and V. Sanchez-Espinoza, "Serpent/SCF pin-level multiphysics solutions for the VERA Fuel Assembly benchmark," *Annals of Nuclear Energy*, vol. 128, p. 102–114, 2019.
- [9] D. Legrady, G. Tolnai, T. Hajas, E. Pazman, T. Parko, and I. Pos, "Full Core Pin-Level VVER-440 Simulation of a Rod Drop Experiment with the GPU-Based Monte Carlo Code GUARDYAN," *Energies*, vol. 15, no. 8, p. 2712, 2022.
- [10] D. Mancusi, M. Faucher, and A. Zoia, "Monte Carlo simulations of the SPERT III E-core transient experiments," *The European Physical Journal Plus*, vol. 137, no. 1, p. 127, 2022.
- [11] T. E. Booth, "Computing the Higher k-Eigenfunctions by Monte Carlo Power Iteration: A Conjecture," *Nuclear Science and Engineering*, vol. 143, no. 3, p. 291–300, 2003.
- [12] P. Zhang, H. Lee, M. Lemaire, C. Kong, J. Choe, J. Yu, F. Khoshahval, and D. Lee, "Practical Monte Carlo simulation using modified power method with preconditioning," *Annals of Nuclear Energy*, vol. 127, p. 372–384, 2019.
- [13] T. Yamamoto, "Implementation of a frequency-domain neutron noise analysis method in a production-level continuous energy Monte Carlo code: Verification and application in a BWR," *Annals of Nuclear Energy*, vol. 115, p. 494–501, 2018.
- [14] A. Rouchon, A. Zoia, and R. Sanchez, "A new Monte Carlo method for neutron noise calculations in the frequency domain," *Annals of Nuclear Energy*, vol. 102, p. 465–475, 2017.

- [15] T. Viitanen and J. Leppänen, "Explicit Treatment of Thermal Motion in Continuous-Energy Monte Carlo Tracking Routines," *Nuclear Science and Engineering*, vol. 171, no. 2, p. 165–173, Oct 2012.
- [16] G. Yesilyurt, W. R. Martin, and F. B. Brown, "On-the-Fly Doppler Broadening for Monte Carlo Codes," *Nuclear Science and Engineering*, vol. 171, no. 3, p. 239–257, 2012.
- [17] S. P. Hamilton and T. M. Evans, "Continuous-energy Monte Carlo neutron transport on GPUs in the Shift code," *Annals of Nuclear Energy*, vol. 128, p. 236–247, 2019.
- [18] P. K. Romano, S. P. Hamilton, R. O. Rahaman, A. Novak, E. Merzari, S. M. Harper, P. C. Shriwise, and T. M. Evans, "A Code-Agnostic Driver Application for Coupled Neutronics and Thermal-Hydraulic Simulations," *Nuclear Science and Engineering*, vol. 195, no. 4, p. 1–21, 2020.
- [19] L. L. Carter, E. D. Cashwell, and W. M. Taylor, "Monte Carlo Sampling with Continuously Varying Cross Sections Along Flight Paths," *Nuclear Science and Engineering*, vol. 48, no. 4, p. 403–411, 1972.
- [20] D. Legrady, B. Molnar, M. Klausz, and T. Major, "Woodcock tracking with arbitrary sampling cross section using negative weights," *Annals of Nuclear Energy*, vol. 102, p. 116–123, 2017.
- [21] A. Rouchon, "Analyse et développement d'outils numériques déterministes et stochastiques résolvant les équations du bruit neutronique et applications aux réacteurs thermiques et rapides," Ph.D. dissertation, Université Paris-Saclay, 2016.
- [22] E. Brun, F. Damian, C. Diop, E. Dumonteil, F. Hugot, C. Jouanne, Y. Lee, F. Malvagi, A. Mazzolo, O. Petit, J. Trama, T. Visonneau, and A. Zoia, "TRIPOLI-4®(R), CEA, EDF and AREVA reference Monte Carlo code," *Annals of Nuclear Energy*, vol. 82, p. 151–160, 2015.
- [23] H. Belanger, "Continuous Media Monte Carlo Transporter," Jun. 2020. [Online]. Available: <https://doi.org/10.5281/zenodo.3898594>
- [24] P. K. Romano, N. E. Horelik, B. R. Herman, A. G. Nelson, B. Forget, and K. Smith, "OpenMC: A state-of-the-art Monte Carlo code for research and development," *Annals of Nuclear Energy*, vol. 82, p. 90–97, 2015.
- [25] J. Leppänen, M. Pusa, T. Viitanen, V. Valtavirta, and T. Kaltiaisenaho, "The Serpent Monte Carlo code: Status, development and applications in 2013," *Annals of Nuclear Energy*, vol. 82, p. 142–150, 2015.
- [26] H. Belanger, "Papillon Nuclear Data Library," 2021. [Online]. Available: <https://github.com/HunterBelanger/papillon-ndl>
- [27] A. Sood, R. A. Forster, and D. K. Parsons, "Analytical Benchmark Test Set For Criticality Code Verification," *Progress in Nuclear Energy*, vol. 42, no. 1, p. 55–106, 2003.
- [28] U. Kasemeyer, R. Früh, J. M. Paratte, and R. Chawla, "Benchmark on Kinetic Parameters in the CROCUS Reactor," OECD/NEA, Tech. Rep.
- [29] A. Zoia, Y. Nauchi, E. Brun, and C. Jouanne, "Monte Carlo analysis of the CROCUS benchmark on kinetics parameters calculation," *Annals of Nuclear Energy*, vol. 96, p. 377–388, 2016.
- [30] M. Chadwick, P. Obložinský, M. Herman, N. Greene, R. McKnight, D. Smith, P. Young, R. MacFarlane, G. Hale, S. Frankle, A. Kahler, T. Kawano, R. Little, D. Madland, P. Moller, R. Mosteller, P. Page, P. Talou, H. Trellue, M. White, W. Wilson, R. Arcilla, C. Dunford, S. Mughabghab, B. Pritychenko, D. Rochman, A. Sonzogni, C. Lubitz, T. Trumbull, J. Weinman, D. Brown, D. Cullen, D. Heinrichs, D. McNabb, H. Derrien, M. Dunn, N. Larson, L. Leal, A. Carlson, R. Block, J. Briggs, E. Cheng,

H. Huria, M. Zerkle, K. Koziar, A. Courcelle, V. Pronyaev, and S. v. d. Marck, "ENDF/B-VII.0: Next Generation Evaluated Nuclear Data Library for Nuclear Science and Technology," *Nuclear Data Sheets*, vol. 107, no. 12, p. 2931–3060, 2006.

- [31] K. Tada, Y. Nagaya, S. Kunieda, K. Suyama, and T. Fukahori, "Development and verification of a new nuclear data processing system FRENDY," *Journal of Nuclear Science and Technology*, vol. 54, no. 7, p. 1–12, 2017.
- [32] C. R. Harris, K. J. Millman, S. J. v. d. Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. v. Kerkwijk, M. Brett, A. Haldane, J. F. d. Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, p. 357–362, 2020.
- [33] H. Belanger, "MGMC," 2022, v0.2.0. [Online]. Available: <https://github.com/HunterBelanger/mgmc>

Part I

Particle Tracking Methods

2 - Particle Tracking in Continuous Media

Monte Carlo codes devoted to particle transport problems rely on sampling an ensemble of random particle trajectories through the traversed multiplying or non-multiplying media. Starting from the source event, each sampled trajectory consists of a series of random flights separated by collisions with the nuclei composing the material: the trajectory is terminated when the particle is killed upon reaching a leakage boundary, when its statistical importance decreases below some predefined threshold, or when a sterile capture event is sampled (if the simulation is analog). Determining the distance that a particle will travel along a given direction before undergoing a collision is therefore an integral operation in Monte Carlo transport codes. In this chapter we will illustrate how Monte Carlo codes traditionally deal with this operation, and how flight sampling must be modified and extended in order to take into account two relevant classes of materials often encountered in real-world applications: *random media*, and *spatially-continuous media*.

In Sec. 2.1, we will first overview the standard “homogeneous medium” assumption, which is used in virtually all production Monte Carlo transport codes, and the tracking methods which are typically employed to move particles through the problem domain. Sec. 2.2 provides a concise introduction to the sampling methods needed for particle tracking in random media, and in particular for the class of Markov geometries based on Poisson tessellations. Finally, in Sec. 2.3 the distinct challenges of spatially-continuous media properties in the context of flight sampling are presented, with a brief discussion concerning which existing tracking methods would allow for particle transport simulations in such a scenario.

2.1 . Overview of Existing Tracking Methods

Virtually all general-purpose production Monte Carlo codes today employ the “homogeneous medium” assumption, which assumes that the cross sections are piece-wise constant functions of spatial position. This corresponds to every fundamental volume (often referred to as a “cell” in Monte Carlo transport jargon) of the simulated model containing a single material, with a uniform composition, density, and temperature [1–4]. For an infinite homogeneous medium, the probability density function (PDF) for a particle to have its first collision at a distance s is written as

$$f(s) = \Sigma_t(E) \exp(-\Sigma_t(E)s), \quad (2.1)$$

where E is the energy of the particle, and $\Sigma_t(E)$ is the total macroscopic cross section (typically in units of cm^{-1}) in the homogeneous medium, at energy E [5]. The exponential functional form in Eq. (2.1) stems from the assumption that the collision centers (i.e., the nuclei) in the traversed medium are uniformly distributed, hence the number of particle-nuclei collision events over a flight path ds is Poisson-distributed with parameter $\Sigma_t(E) ds$. Integrating Eq. (2.1) with respect to s , we may obtain the explicit cumulative density function (CDF) for the flight distance d :

$$F(d) = \int_0^d f(s) ds = \int_0^d \Sigma_t(E) \exp(-\Sigma_t(E)s) ds = 1 - \exp(-\Sigma_t(E)d). \quad (2.2)$$

Sampling a random variable $\xi \sim \mathcal{U}(0, 1)$ and setting $F(d) = \xi$ by virtue of the inverse transform theorem [5], from Eq. (2.2) it is therefore possible to analytically obtain the corresponding flight distance

$$d = -\frac{\ln(1 - \xi)}{\Sigma_t(E)}. \quad (2.3)$$

Often, the $1 - \xi$ in Eq. (2.3) is simply written as ξ , as both terms are uniformly distributed on the unit interval, and are therefore statistically equivalent [5].

Observe that, in the homogeneous media assumption, the macroscopic cross section is allowed to be a continuous function of the particle energy, E . Cross sections which are continuous in energy pose no problem for Monte Carlo tracking of neutral particles (neutrons and photons), as the energy of the particle remains constant over a flight, and can only change during a collision. As such, basically all of the existing production Monte Carlo codes use cross section representations which are continuous in energy [1–4]. This is in contrast to deterministic transport codes, which must use the multi-group approximation, assuming that the cross section for a nuclide is constant over a given energy interval [6, 7]. Multi-group cross sections require great care, as they must be properly self-shielded and homogenized to ensure that reaction rates are preserved [7]. Conversely, sampling from continuous-energy cross sections foregoes the need for any such approximations and constitutes a prominent advantage of Monte Carlo codes. In the following, we will refer to continuous media to denote the possibility of having fundamental volumes where the material is not necessarily uniform, but demonstrates some spatial dependence. This idea will be outlined in detail in Sec. 2.3. The remainder of this section will review the possible tracking methods which are able to treat homogeneous media.

2.1.1 . Surface Tracking

The derivation of Eq. (2.3) assumes that our system is comprised of a single uniform and infinitely large volume. Considering the behavior of d as $\xi \rightarrow 1$, we note that

$$\lim_{\xi \rightarrow 1} -\frac{\ln(1 - \xi)}{\Sigma_t(E)} = \infty. \quad (2.4)$$

Therefore, the theoretical maximum flight distance of a particle is $d = \infty$. Most systems of interest, however, are not infinitely large, and are composed of several different volumes, each with a different material. Any flight distance sampled from Eq. (2.3) is only valid for the volume with the cross section which was used in computing d ; if a distance is sampled that would cause the particle to move into a different volume during the flight, we are only allowed to displace the particle to the boundary of this new volume, where the cross section for this new volume is then used to sample a new flight distance.¹ This process is repeated until a flight distance is sampled which would not take the particle into a different material [5].

Determining where a particle will leave one volume and enter another can be a complex task. Most Monte Carlo codes use a form of constructive solid geometry (CSG), where primitive volumes (or surface half-spaces) are combined using Boolean operators [8, 9]. For these primitives, there are known algorithms to compute the distance to the boundary of a volume, d_s , for a given starting position \mathbf{r}_0 , and direction of travel $\hat{\Omega}$ [8]. This method of comparing a sampled flight distance to the distance to the boundary of a volume is often called *surface tracking* or *ray tracing*. Ray tracing is an equivalent term that is more commonly used in the computer graphics communities, where the image rendering problem is identical to the problem of determining the distance to the boundary of a volume in the application of Monte Carlo methods for reactor physics problems [8, 10, 11]. The basic structure of the surface tracking algorithm is presented in Alg. 2.1. It assumes that a particle starts a flight at position \mathbf{r}_0 , traveling in direction $\hat{\Omega}$ at energy E .

Surface tracking is likely the most ubiquitous tracking method, and some variants of this algorithm are used in TRIPOLI-4[®], Serpent, MCNP, OpenMC, and many other Monte Carlo codes devoted to particle transport [1–4]. One reason for this is that surface tracking allows for the use of both collision estimators and track length estimators for reaction rates. If one wishes to estimate the reaction rate for the reaction channel α via the collision estimator, a particle of weight w will contribute

$$\frac{\Sigma_\alpha(\mathbf{r}, E)}{\Sigma_t(\mathbf{r}, E)} w \quad (2.5)$$

¹It is also possible that the distance sampled could be larger than the distance to the nearest boundary condition. If a “vacuum” boundary is encountered, the particle is killed, and can no longer contribute any statistics. For a “reflective” boundary, the particle is advanced to the boundary, and has its direction modified according to the law of reflection.

Algorithm 2.1: Surface Tracking Procedure

```
1 Sample random variable  $\xi \sim \mathcal{U}(0, 1)$ ;  
2  $d := -\ln(1 - \xi)/\Sigma_t(\mathbf{r}_0, E)$ ;  
3  $d_s :=$  find distance to volume boundary given  $\mathbf{r}_0$  and  $\hat{\Omega}$ ;  
4 if  $d < d_s$  then  
5    $\mathbf{r}_1 := \mathbf{r}_0 + d\hat{\Omega}$ ;  
6   Contribute to track length estimator scores;  
7   Contribute to collision estimator scores;  
8   Perform collision, changing  $E$  and  $\hat{\Omega}$ ;  
9    $\mathbf{r}_0 = \mathbf{r}_1$ ;  
10 else  
11    $\mathbf{r}_1 := \mathbf{r}_0 + d_s\hat{\Omega}$ ;  
12   Contribute to track length estimator scores;  
13    $\mathbf{r}_0 = \mathbf{r}_1$ ;  
14   Find new volume;  
15   Update  $\Sigma_t(\mathbf{r}_0, E)$  for current volume;  
16   Goto line 1;  
17 end
```

to the estimation of the reaction rate, at each collision site [5]. For the track length estimator, a particle will contribute

$$w \int_0^{|\mathbf{r}_1 - \mathbf{r}_0|} \Sigma_\alpha(\mathbf{r}_0 + s\hat{\Omega}, E) ds \quad (2.6)$$

to the estimation of the reaction rate, where \mathbf{r}_0 is the starting position of the flight and \mathbf{r}_1 is the position where the particle underwent another collision or moved into a different material [5]. Evaluating the integral in Eq. (2.6) is quite straightforward under the homogeneous media assumption; the integral simply becomes a sum of the cross sections for reactions α in volume i , multiplied by the distance d_i that the particle flew through volume i :

$$w \int_0^{|\mathbf{r}_1 - \mathbf{r}_0|} \Sigma_\alpha(\mathbf{r}_0 + s\hat{\Omega}, E) ds = w \sum_i \Sigma_{\alpha,i}(E) d_i, \quad (2.7)$$

where

$$\sum_i d_i = |\mathbf{r}_1 - \mathbf{r}_0|. \quad (2.8)$$

With surface tracking, the distance flown through each volume is always explicitly calculated, and therefore available for use in scores based on the track length estimator. The use of track length estimators is often preferred over the use of collision estimators, particularly when one is trying to estimate the reaction rates across a mesh, or within individual material volumes, as a particle does not need to have a collision within the region of interest in order to contribute to the estimation of the reaction rate.

2.1.2 . Delta Tracking

Delta tracking is a rejection sampling technique which can be used in place of surface tracking. Woodcock appears to have been the first to formalize the use of delta tracking, and promoted the use of this technique in the GEM Monte Carlo code [12]: for this reason, delta tracking is also called *Woodcock tracking*. The method has since become well-known, especially from its prominent use in the Serpent Monte Carlo code [13, 14].

A specific feature of delta tracking is that it requires a majorant cross section $\Sigma_{\text{maj}}(E)$ which, for each point in the tabulated energy grid, must be larger than or equal to the largest total cross section of all

Algorithm 2.2: Delta Tracking Procedure

```
1 Sample random variable  $\xi_1 \sim \mathcal{U}(0, 1)$ ;  
2  $d := -\ln(1 - \xi_1) / \Sigma_{\text{maj}}(E)$ ;  
3  $\mathbf{r}_1 := \mathbf{r}_0 + d\hat{\Omega}$ ;  
4 Find current volume;  
5 Compute  $\Sigma_t(\mathbf{r}_1, E)$ ;  
6 Sample random variable  $\xi_2 \sim \mathcal{U}(0, 1)$ ;  
7 if  $\xi_2 < \Sigma_t(\mathbf{r}_1, E) / \Sigma_{\text{maj}}(E)$  then  
8 |   Contribute to collision estimator scores;  
9 |   Perform a real collision, changing  $E$  and  $\hat{\Omega}$ ;  
10 |   $\mathbf{r}_0 = \mathbf{r}_1$ ;  
11 else  
12 |   Virtual collision (do nothing);  
13 |    $\mathbf{r}_0 = \mathbf{r}_1$ ;  
14 |   Goto line 1;  
15 end
```

materials in the problem domain. Mathematically, this can be summarized as

$$\Sigma_{\text{maj}}(E) \geq \max_{\mathbf{r}} \Sigma_t(\mathbf{r}, E), \quad (2.9)$$

where $\Sigma_t(\mathbf{r}, E)$ would be a piece-wise constant function in space, under the homogeneous media assumption. In general, $\Sigma_{\text{maj}}(E)$ must be prepared on initialization, as it depends on what materials are used in the problem [15]. From this majorant cross section, the distance to a tentative collision site is sampled using

$$d = -\frac{\ln(1 - \xi)}{\Sigma_{\text{maj}}(E)}. \quad (2.10)$$

We note that Eq. (2.10) only differs from Eq. (2.3) in that the total cross section has been replaced with this new majorant cross section. It is not necessary to keep track of where the particle will move from one volume into another in this method. Regardless of the geometry of the system, the particle will be moved a distance of d . At this tentative collision site, we look up what volume we are in, and find the total cross section at the present position. Then, with probability $\Sigma_t(\mathbf{r}, E) / \Sigma_{\text{maj}}(E)$, a real collision is performed; the particle may contribute to reaction rates using the standard collision estimator, and then undergo the collision physics. Alternatively, with probability $1 - \Sigma_t(\mathbf{r}, E) / \Sigma_{\text{maj}}(E)$, the particle will undergo what is called a virtual collision, where the energy and direction of the particle do not change (hence the name “delta” tracking), and the particle does not contribute to any scores with the collision estimator. This process is repeated with the sampling of a new distance to a new tentative collision site, until a real collision is sampled, or the particle leaks from the system. The delta tracking algorithm is summarized in Alg. 2.2.

From just a cursory inspection, it is not evident that this algorithm is unbiased for systems with more than one material. A very thorough analysis of the method, with a mathematical proof of its unbiasedness, has been provided by Coleman [16]. A distinct advantage of the delta tracking method over surface tracking is that it is no longer necessary to compute the distance to surface intersections, which can be computationally intensive. However, when $\Sigma_t(\mathbf{r}, E) / \Sigma_{\text{maj}}(E) \ll 1$, many virtual collisions must be performed before a real collision is sampled, and statistics for the collision estimator are only generated at real collisions.² This issue is known as the “localized heavy absorber problem”, and has been shown to greatly reduce the efficiency of delta tracking [13]. If a geometrically small portion of the problem domain contains a material which has a very large cross section (which is then used as the majorant), then the probability of a real collision

²An alternative version of the collision estimator, which generates statistics at both real and virtual collisions, exists and is discussed in Sec. 4.3.1.

is greatly reduced in the majority of the problem domain. The Serpent code has chosen to avoid these performance penalties by switching to surface tracking in areas where the probability of a real collision is below some user-defined threshold [13]. Another disadvantage of delta tracking is that it is impossible to use track length estimators: without having calculated all the surface intersections along the flight path, we do not know what distance the particle traveled through each volume³.

2.1.3 . Negative-Weighted Delta Tracking

While delta tracking allows forgoing the calculation of surface crossings, the requirement of using a majorant cross section can at times be burdensome. The construction of a majorant cross section becomes difficult when considering performance implications, or in combination with the use of the Target Motion Sampling algorithm as a method of performing on-the-fly Doppler broadening for the cross sections [15, 17, 18]. Additionally, the localized heavy absorber problem might impede its efficiency for problems of interest to reactor physics [15]. It is therefore reasonable to conjecture that relaxing the requirement of a majorant cross section could be beneficial in some cases. In the literature, it appears that a tracking method very similar to delta tracking but not requiring the use of a strict majorant cross section has been developed on two separate occasions. The first such implementation was proposed by Carter et al. in 1972 [19]. A similar tracking method was independently proposed by Legrady et al. in 2017 [20], and was later shown to be a generalization of the method proposed by Carter et al. [21]. In this manuscript, we refer to this family of methods collectively as *negative-weighted delta tracking*, as these transport methods are closely related to delta tracking (using a system of rejection sampling with real and virtual collisions), and allow the statistical weights of particles to become negative. We will outline this method in the more general framework used by Legrady et al., and will then specify under what conditions the variant proposed by Carter et al. can be retrieved.

As proposed by Legrady et al., the negative-weighted delta tracking algorithm makes use of two free parameters. The first of these parameters is a sampling cross section, $\Sigma_{\text{smp}}(\mathbf{r}, E)$. This cross section is not required to satisfy the definition of a majorant cross section provided by Eq. (2.9). All that is required is that it be greater than zero. The second parameter is the probability of sampling a real collision, denoted as q , which may be a function of any of the phase space parameters involved in the flight. All that is required is that q represent a valid probability (i.e. $q \in [0, 1]$). Using the sampling cross section, the distance to a tentative collision site is sampled (in the same manner that the majorant is used in delta tracking). If a particle started a flight at \mathbf{r}_0 , the tentative collision site is then $\mathbf{r}_1 = \mathbf{r}_0 + d\hat{\Omega}$. A real collision occurs with probability q : should this event be sampled, the particle's weight is then updated as

$$w = w \frac{\Sigma_t(\mathbf{r}_1, E)}{q(\mathbf{r}_1, E) \Sigma_{\text{smp}}(\mathbf{r}_0, E)}. \quad (2.11)$$

Should a virtual collision be sampled (which occurs with complementary probability $1 - q$), the particle's weight is instead updated as

$$w = w \frac{1 - \frac{\Sigma_t(\mathbf{r}_1, E)}{\Sigma_{\text{smp}}(\mathbf{r}_0, E)}}{1 - q(\mathbf{r}_1, E)}. \quad (2.12)$$

This algorithm is presented in Alg. 2.3 [20]. Upon inspection of Eq. (2.12), if a virtual collision is sampled at a tentative collision site where the sampling cross section is not a majorant, i.e. where $\Sigma_t(\mathbf{r}_1, E) > \Sigma_{\text{smp}}(\mathbf{r}_0, E)$, the weight multiplier is negative and the particle's weight will therefore change sign.

The variant of negative-weighted delta tracking proposed by Carter et al. can be retrieved by choosing q to be [21]

$$q(\mathbf{r}_0, \mathbf{r}_1, E) = \frac{\Sigma_t(\mathbf{r}_1, E)}{\Sigma_t(\mathbf{r}_1, E) + |\Sigma_{\text{smp}}(\mathbf{r}_0, E) - \Sigma_t(\mathbf{r}_1, E)|}. \quad (2.13)$$

³While it is not possible to use track length estimators for reaction rates with delta tracking, it is possible to use the track length estimator for the flux over a mesh, for example. Only the distance traveled through each mesh element is needed in this case, which can be calculated from mesh parameters and the flight start and end points.

Algorithm 2.3: Negative-Weighted Delta Tracking Procedure

```
1 Sample random variable  $\xi_1 \sim \mathcal{U}(0, 1)$ ;  
2  $d := -\ln(1 - \xi_1) / \Sigma_{\text{smp}}(\mathbf{r}_0, E)$ ;  
3  $\mathbf{r}_1 := \mathbf{r}_0 + d\hat{\Omega}$ ;  
4 Find current volume;  
5 Compute  $\Sigma_t(\mathbf{r}_1, E)$ ;  
6 Sample random variable  $\xi_2 \sim \mathcal{U}(0, 1)$ ;  
7 if  $\xi_2 < q(\mathbf{r}_1, E)$  then  
8    $w := w \frac{\Sigma_t(\mathbf{r}_1, E)}{q(\mathbf{r}_1, E)\Sigma_{\text{smp}}(\mathbf{r}_0, E)}$ ;  
9   Contribute to collision estimator scores;  
10  Perform a real collision, changing  $E$  and  $\hat{\Omega}$ ;  
11   $\mathbf{r}_0 := \mathbf{r}_1$ ;  
12 else  
13    $w := w \frac{1 - \frac{\Sigma_t(\mathbf{r}_1, E)}{\Sigma_{\text{smp}}(\mathbf{r}_0, E)}}{1 - q(\mathbf{r}_1, E)}$ ;  
14   Virtual collision (do nothing);  
15    $\mathbf{r}_0 := \mathbf{r}_1$ ;  
16   Goto line 1;  
17 end
```

Observe that negative-weighted delta tracking reduces to regular delta tracking, should $\Sigma_{\text{smp}}(\mathbf{r}, E)$ satisfy the requirement of a majorant cross section given by Eq. (2.9), and if one chooses a real collision probability of

$$q(\mathbf{r}_0, \mathbf{r}_1, E) = \frac{\Sigma_t(\mathbf{r}_1, E)}{\Sigma_{\text{smp}}(\mathbf{r}_0, E)}. \quad (2.14)$$

While a majorant is no longer required, two pieces of information must now be provided for the negative-weighted delta tracking algorithm to work: a sampling cross section, and a real collision probability. Although the variant of Legrady et al. has demonstrated the potential to yield performance improvements over standard delta tracking, its performance is highly dependent on the choice of these two free parameters [20]. Currently, there is no clear method to determine the optimum choice of these parameters for any given system, without performing some sort of optimization study beforehand [20, 21]. We stress the fact that both variants require the presence of negative particle weights in the simulation: many authors have previously observed that Monte Carlo simulations involving mixtures of positive and negative particle weights could potentially lead to serious issues with numerical convergence and/or large variances [22–24].

2.1.4 . Regionalization

As was previously mentioned in Sec. 2.1.2, the efficiency of delta tracking can sometimes be diminished in the presence of the localized heavy absorber problem. When this occurs, the majorant cross section is determined by a small volume fraction of the model domain, where the probability of sampling a real collision is unity. Conversely, in the vast majority of the problem domain a large number of virtual collisions would be required, each demanding the determination of the particle's current volume and total cross section, both of which can be time-consuming operations. This situation is quite common in reactor physics problems, where each fuel assembly might have a few control rods, or burnable poison rods, which are likely to become the majorant cross section at thermal energies.

As mentioned, switching to surface tracking whenever the real collision probability is lower than some user-defined threshold is a way of overcoming this issue [13]. An alternative solution to the localized heavy absorber problem was proposed by Guo and Chen, which we refer to as *regionalization* [25]. In their approach, the problem domain is broken into several different regions, each with its own majorant cross section. Doing this means that we are required to calculate the distance a particle can fly along its current direction, until

entering the next region. If a particle samples a distance to a new tentative collision site which is larger than the distance to the next region, then the particle may only be moved as far as the region boundary. Here, the majorant is updated, and then a new distance can be sampled; at this point, the regionalization method begins to become very resemblant of surface tracking. As long as

- it is fast to determine the region containing a given position;
- it is fast to evaluate the distances to the surfaces which enclose a region;
- the mean free path of particles in a region is shorter than or comparable to the average linear size of the region,

then it is reasonable to hope for improved performance from a regionalized version of delta tracking. Regionalization might be applied to negative-weighted delta tracking as well, and is similarly expected to potentially improve its performance. These criteria can be met by using a simple subdivision technique, such as a regular or irregular Cartesian mesh. Alternatively, more advanced structures from the field of computer graphics, such as voxels and octrees, might be used to generate regionalizations which can be better adapted to the geometry and material composition of each specific problem [8, 26, 27].

2.2 . Transport in Stochastic Geometries

In many applications occurring in particle transport problems for reactor physics, the geometry and material compositions are known only statistically. Stochastic geometries, or problems where material regions are organized in a random manner, emerge for instance in the investigation of neutron multiplication in random fuel-cladding-moderator mixtures which can form during severe accidents in nuclear reactors, leading to core degradation and corium formation [28], or in the analysis of neutron shielding by randomly dispersed absorber lumps embedded in concrete plates [29]. Another well-known example in reactor physics concerns next-generation reactor designs: TRISO fuel, for example, takes the form of small spherical particles which are randomly dispersed within larger fuel elements [30]. Transport problems for inertial confinement fusion experiments involve photon propagation through turbulent Rayleigh-Taylor instabilities, which can also be described in terms of random media [31]. For all of these cases, the material properties at a given location are not known exactly, and some probability $p_\alpha(\mathbf{r})$ is typically assigned for sampling material phase (“color”) α at position \mathbf{r} . Different mathematical methods and models have been introduced in the literature in order to describe the statistical properties of random media. Such examples include the random placement of bodies (typically spheres) in a background matrix (aligning with the TRISO fuel case), and Poisson or Voronoi tessellations, where a portion of the viable domain is randomly partitioned by planes (aligning with the corium case) [32]. For illustration, realizations of spherical inclusions in a background matrix and a Poisson tessellation are shown in Figure 2.1.

The Monte Carlo method is a natural choice for examining particle transport in stochastic geometries. Due to the complex and disordered nature of random media, analyzing such a system with a deterministic transport code would require a highly refined spatial mesh, which could raise issues in convergence, run time, and memory usage. Since no geometry discretization is required in the Monte Carlo approach, some of these problems can be avoided or mitigated. Nonetheless, stochastic media used in most realistic particle transport simulations pose distinct challenges to Monte Carlo particle tracking routines. Often, such problems contain several hundreds of thousands (if not millions or billions) of volumes. This makes standard surface tracking quite inefficient, or even practically impossible, as a huge number of surface intersection calculations might be required for each particle history: whenever a particle moves from one volume to another, it might have to check hundreds of thousands of volumes before finding the correct one. Due to these inherent difficulties, most Monte Carlo simulations of stochastic geometries rely on either surface tracking with neighbor maps [33] (which will be detailed Sec. 2.2.2) or delta tracking [34]. No comparison has been previously attempted between these two strategies: in order to establish a firmer ground for performing particle transport in random media, and thus enhance the reliability of these Monte Carlo methods for complex real-world applications,

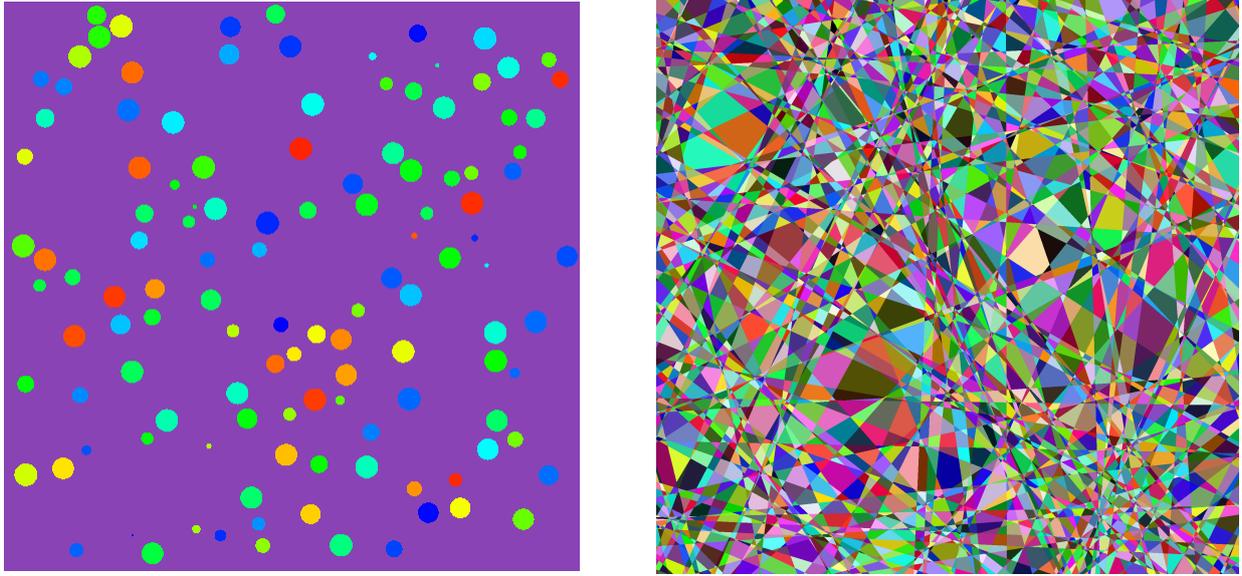


Figure 2.1: On the left is a realization of stochastic spherical inclusion placed in a background matrix. On the right is a realization of a Poisson tessellation. Both configurations correspond to two-dimensional cuts of three-dimensional random media sampled using the CASTOR Monte Carlo code developed at CEA.

in this thesis we will examine which of these two approaches is most efficient, and further discuss other techniques which might speed up particle tracking. As a case-study, for the remainder of this work we shall consider the case of Poisson tessellations, whose properties are subsequently outlined in Sec. 2.2.1. The considered acceleration techniques are presented in Sec. 2.2.2. A comprehensive investigation is conducted in Chapter 3.

2.2.1 . Properties of Poisson Tessellations

Isotropic and homogeneous Poisson tessellations are stochastic partitions of space into disjoint polyhedra and belong to the class of Markov media [32]. Consider an infinitely large medium, composed of an infinite number of different convex volumes. We trace an arbitrary line through this geometry, and compute the chord lengths determined by intersecting the volumes of the medium with the line. The medium is said to be Markovian if the chord lengths are exponentially distributed. Infinitely large Poisson tessellations can be shown to have the Markov property [35]. The properties of isotropic and homogeneous Poisson tessellations depend on a single parameter ρ , called the tessellation density, having units of inverse length. By construction, the distribution describing the chord length is exponential, with parameter ρ , namely

$$P(l) = \rho e^{-\rho l}. \quad (2.15)$$

This then indicates an average chord length of $\Lambda = \langle l \rangle = 1/\rho$. An algorithm leading to an explicit construction procedure to sample isotropic and spatially homogeneous Poisson tessellations of a finite region embedded in a D -dimensional space has recently been implemented [35]. This algorithm is based on the formalism of integral geometry [32], and has been extended to the broader cases of non-isotropic [36] and spatially non-homogeneous [37] Poisson tessellations. For the sake of simplicity, here we will exclusively focus on the isotropic and spatially homogeneous case. For Poisson tessellations in D spatial dimensions, the average number of volumes within an D -dimensional cube with side length L scales as

$$\langle N_V \rangle \propto (\rho L)^D. \quad (2.16)$$

Figure 2.2 demonstrates the effect of increasing the tessellation density for a box of a fixed size. On average, each volume within the tessellation will have

$$\langle N_n \rangle = 2D \quad (2.17)$$

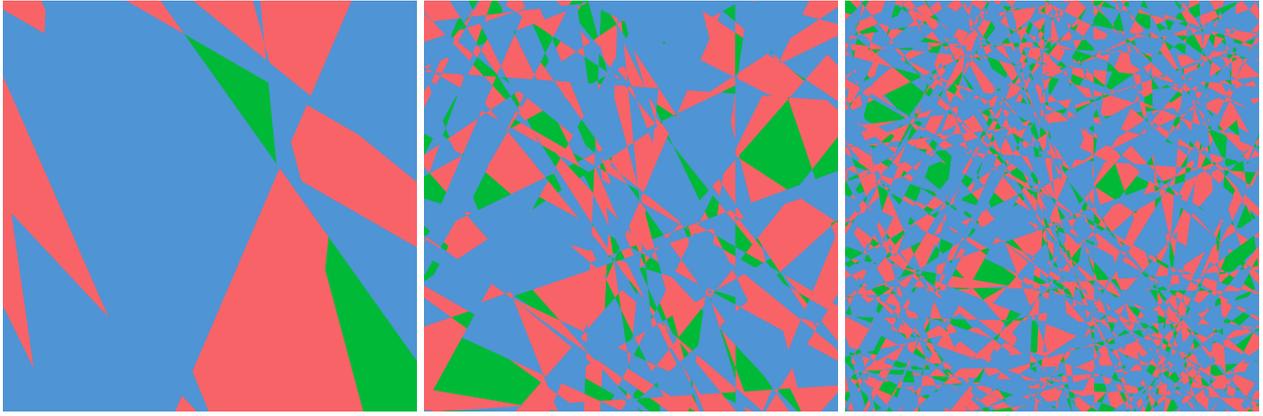


Figure 2.2: All three images depict a $50 \text{ cm} \times 50 \text{ cm}$ slice of a 3D Poisson tessellation used in Chapter 3 and sampled using the CASTOR code. The realization on the left has a tessellation density of $\rho = 0.1 \text{ cm}^{-1}$, the realization in the center has $\rho = 0.8 \text{ cm}^{-1}$, and the realization on the right has $\rho = 2 \text{ cm}^{-1}$. Blue represents water, red represents UOX fuel, and green represents cladding. The probability of each material is provided in Sec. 3.4, and is based of the material ratio of an idealized fuel pin cell.

sides, which means that the average number of neighboring volumes is also $2D$, independently of the tessellation density ρ . In three spatial dimensions, $\langle N_n \rangle = 6$, and it has been shown that the associated variance is $(13\pi^2 + 336)/12 - 6^2 \simeq 2.6921$ [35, 38]: the dispersion $\sigma[N_n] \simeq 1.641$ is small and the average value is thus representative of a typical cell of the tessellation. Due to the complexity of Poisson tessellations, even for the simpler case of isotropic and spatially homogeneous geometries, only a limited number of exact results are known [32]. Moreover, these properties are only strictly valid for an infinite Poisson tessellation. For a finite tessellation, i.e., a tessellation restricted to a region of finite size, such as a cube or a sphere, these properties are not exact, especially in the vicinity of the tessellation boundary. However, for a sufficiently large tessellation ($\rho L \approx 100$), the deviations from the ideal formulas at the boundary of the tessellation are minimal [35].

Previously, the CASTOR code (Construction and Analysis of STOchastic Realizations) has been developed to generate Poisson tessellations for use with the Monte Carlo transport codes at CEA [38]. Once the tessellation has been constructed, it is necessary to associate material properties with each volume in the tessellation. Typically, one only knows the probability p_α of material α within the tessellation, which corresponds to the fraction of the volume occupied by material α . With the material probabilities known in advance, the Switzer procedure is used to attribute a material to each volume by sampling the colors independently from p_α [39]. The resulting “colored” geometry is still Markovian, having scaled chord length distributions

$$P_\alpha(l) = \rho_\alpha e^{-\rho_\alpha l}, \quad (2.18)$$

where $\rho_\alpha = \rho(1 - p_\alpha)$ is the tessellation density for material α . This yields a colored average chord length of $\Lambda_\alpha = \Lambda/(1 - p_\alpha)$. With respect to tracking, only the uncolored tessellation is seen by the particle; the material-specific quantity ρ_α (and equivalently Λ_α) plays no role in particle tracking.

2.2.2 . Particle Tracking in Random Media

We will now examine the distinct issues related to Monte Carlo particle transport in Poisson tessellations, and propose some strategies to overcome these difficulties. When using surface tracking, each time a particle leaves a cell, it is moved to the boundary of the cell, and then a search is performed to determine the new cell that the particle is about to enter. In a naive implementation, this typically requires iterating through all cells available to the particle in the geometry, and checking if the particle is located inside that cell. Tracking can only continue once the new cell is found. Having to iterate through all possible cells becomes

quite time-consuming when the number of volumes is large, as can be the case for Poisson tessellations with high ρ . Most of the acceleration techniques which will be outlined in this section focus on speeding up this process of determining in which cell a particle finds itself.

Not only is it time-consuming to determine which cell a particle is in, but surface tracking has the added disadvantage of needing to calculate all of the surface intersections to determine at what distance a particle will leave its current cell. If the material and geometric properties are such that a particle will typically fly through several cells before undergoing a collision, it is reasonable to envision that surface tracking might become very inefficient. For this reason, some researchers suggest the use of delta tracking when performing transport in stochastic media [34]. If the majorant cross section is very close to the total cross section, few virtual collisions will be performed. Since the cell must be found at each tentative collision site, virtual collisions can be particularly costly with stochastic media, as a cell search must be performed even for events that do not generate any statistics. Techniques which accelerate the cell search process could potentially provide a benefit to delta tracking as well.

Evaluation of Simulation Performance

The performance of the acceleration techniques which are examined in Chapter 3 is done using the Figure of Merit (FOM). To compute the FOM, a specific quantity is typically chosen for the comparison, and it's FOM is can be calculated as

$$\text{FOM} = \frac{1}{T\sigma^2}, \quad (2.19)$$

where σ^2 is the variance of the estimation of the tally, and T is the wall-clock run time that was required to obtain the estimate. This quantity is used as a reference, as it is generally know that for a quantity estimated by Monte Carlo methods,

$$\sigma^2 \propto \frac{1}{N}, \quad (2.20)$$

where N is the number of independent replicas (i.e. generations or batches) [5]. Additionally, it is generally true that

$$T \propto N. \quad (2.21)$$

Through Eqs. (2.20) and (2.21), we can infer that Eq. (2.19) should be a constant (for a given code, computer system, parallelization scheme, and quantity). A larger FOM indicates a better calculation efficiency. For the simulations performed in Chapter 3, we chose k_{eff} as the variable used to compute the FOM.

Neighbor Maps

Neighbor maps, sometimes also referred to as connectivity maps, have been successfully used to accelerate surface tracking. In its most basic form, a neighbor map is simply a list of all cells which touch another cell [40]. Some codes are able to construct this neighbor map on-the-fly during the course of the simulation, while others read them as an input parameter [1, 2]. Regardless of how this map is obtained (which can nonetheless affect the initialization time of the specific code), the usage and results are the same. When a particle leaves a cell, the code will first iterate through all of the cells in its neighbor map. If the map was properly generated and is complete, the particle is then guaranteed to be in one of the cells in the neighbor map; otherwise, the code performs a linear search through the rest of the cells, as if the neighbor map did not exist. Connectivity maps can be further improved during the simulation by also taking into account the frequency at which the neighboring cells are visited by the particles.

In 3D Poisson tessellations, it is noted From Eq. (2.17) that each cell will have an average of 6 neighbors. Only needing to check the ≈ 6 neighboring cells is much more efficient than needing to iterate through the, say, $\approx 10^5$ cells which could be in the tessellation. A thorough analysis is presented in Chapter 3. Simulations performed with TRIPOLI-4[®] have shown that surface tracking can be sped up by a factor of 100 when using a neighbor map with Poisson tessellations [33, 38]. Due to the very large run times which

would have been encountered, the analysis of different acceleration techniques in Chapter 3 only compares surface tracking with neighbor maps to delta tracking.

While neighbor maps are commonly used with surface tracking, no literature has been found referencing the use of neighbor maps with delta tracking. The reason for this is likely due to the fact that with the delta tracking algorithm it is possible for the particle to cross through several cells during a single flight, so that the particle is not guaranteed to be located in a neighboring cell at the end of the flight. However, if the mean free path length λ is smaller than or of the same order of magnitude as the average chord length ($\Lambda = 1/\rho$), there will reasonably be a large probability that, if the particle leaves its current cell, its flight will end in a neighboring cell. If true, delta tracking should also benefit from the use of a neighbor map when $\Lambda > \lambda$. This hypothesis is thoroughly examined in Chapter 3: we will show that, if no cell search mesh (outlined below) is used, then neighbor maps are beneficial for delta tracking, with the largest improvements being observed for the regime $\Lambda > \lambda$. For the cases of $\Lambda \approx \lambda$ and $\Lambda < \lambda$, neighbor maps provide little to no performance improvements for delta tracking.

Cell Search Mesh

Another acceleration technique is the use of a *cell search mesh*, taking the form of a regular rectilinear mesh that is imposed on top of the Poisson tessellation. From a particle's position, it is fast to calculate the corresponding tuple of indices in the mesh using the pitch along each direction. Each element of this mesh contains a list of all cells which are located within the corresponding cuboid region. The cell corresponding to the particle's position is guaranteed to be in the list of cells contained in the mesh element. If the mesh has a sufficient number of elements along each coordinate axis, then the number of cells which need to be checked is greatly reduced, speeding up the cell search process.

The cell search mesh is only beneficial to speed up "where am I?" queries. For this reason, surface tracking would not generally benefit from the use of a cell search mesh if a neighbor map is also being used, except at particle initialization. When particles are stored in the bank while waiting to be transported, in several production-level Monte Carlo codes only their position and direction are typically recorded [2, 3]. The reasoning for this is multi-faceted, but it means that the cell of a banked particle must be determined when a particle is initialized for transport. Surface tracking can then benefit from the cell search mesh when initializing particles for transport, where it would otherwise need to iterate through all possible cells. Once this first cell has been determined, however, surface tracking can simply use the neighbor map for all other cell look-ups. On the contrary, delta tracking can benefit more from the use of a cell search mesh. At every tentative collision site, the cell could quickly be obtained using the cell search mesh, and this could greatly reduce the penalty imposed by virtual collisions, where the cell must be determined but statistics are not generated. In Chapter 3 we make extensive use of the cell search mesh with both surface and delta tracking. The performance of the two methods is examined in relation to the refinement of the search mesh. If no cell search mesh is used, then surface tracking with neighbor maps always outperforms delta tracking variants. If a cell search mesh is utilized, surface tracking seems to have superior efficiency when $\Lambda > \lambda$, while delta tracking has the best performance when $\Lambda < \lambda$. It is also noted that, with a cell search mesh, delta tracking with a neighbor map is slightly less efficient than vanilla delta tracking. This is because of the slight penalty that is incurred on the occasions when the particle does not fly into a neighboring cell.

Geometry Kernel

In most general-purpose Monte Carlo codes, the geometry is described not just with cells, but also universes and lattices [2–4]. A cell has a defined volume region, which can be filled with either a material, universe, or lattice. Universes are simply a collection of cells which are all present within the same coordinate system. These cells could in turn contain other universes, or lattices, or could terminate with a material. A lattice contains many universes which are organized in a repetitive pattern, typically a regular rectilinear mesh. This concept is useful when modeling structures such as fuel assemblies, where different pins are repeated in a regular manner. With universes nested in lattices or cells, and lattices nested in cells, the entire

geometry model can be represented as a directed acyclic graph (DAG), where cells filled with materials are the terminating leaves. For large nuclear reactor models, using this DAG structure can reduce the memory footprint by three orders of magnitude [41]. The exact cell at a given position is then represented as the path through the DAG which describes the geometry. In this thesis, we shall refer to this path through the geometry DAG as the “*geometry kernel*”. The geometry kernel is often stored with a particle, since it is needed for tracking purposes, but it is not stored with banked particle information. Instead, the geometry kernel must be determined at particle initialization. For traditional nuclear reactor problems (such as PWRs and BWRs), this is not a costly operation, as the use of lattices in the geometry definition typically makes this operation relatively fast.

When performing particle transport in stochastic media, on the contrary, we will not be able to profit from lattices, as the involved geometry structures are not repetitive: all of the random material cells are generally placed within the same universe. The initial geometry kernel of a particle is built when it undergoes initialization. As this particle undergoes transport, it may create secondary particles through reactions like fission or $(n, 2n)$ collisions. These secondary particles are then added to the bank to be transported at a later time, but without this geometry kernel which is known thanks to the parent particle. For standard reactor problems, this discarded information is not generally worth the added memory requirements, as it is very cheap to recalculate. In order to avoid recalculating quantities which have already been determined, it might be a good idea to store the geometry kernel with banked particles, theoretically speeding up particle initialization. Storing the geometry kernel also poses the disadvantage of increased time spent in MPI communications, where particles are transferred between nodes. There are also implementation details which must be considered if one is going to store the geometry kernel with banked particle information. In general, the maximum size of the geometry kernel depends on how the geometry was defined by the user. Allowing an arbitrary geometry kernel size would require the kernel to be stored on the heap, complicating the transfer of particles between nodes using MPI. Limiting the size of the geometry kernel is a possible solution to the problem, but restricts the generality of the code, and the geometries that could be modeled. These design choices and the disadvantage of slower MPI communications would have to be balanced with the possibility of faster simulations for stochastic media. In Chapter 3 we examine the effects of storing the geometry kernel as an acceleration technique. In general, storing the kernel only leads to performance improvements for surface tracking with neighbor maps when no cell search mesh is used. The kernel is only used to speed up the initialization of the particle’s history. Once completed, the neighbor map can be used to facilitate speedy searches for the next cell. While storing the geometry kernel would also speed up the initialization of histories for delta tracking, that method requires many frequent cell searches. Removing just one of these cell searches reduces the time spent performing cell searches by just a fraction of the time that would otherwise be required. When the cell search mesh is used, storing the geometry kernel always degrades performance for all methods, due to the added costs in MPI communications.

2.3 . Transport in Continuous Media

The overview of tracking methods in Sec. 2.1 only considered material regions where, for a given volume V and energy E , the total macroscopic cross section is spatially constant for all positions in V . The case of random media modeled using Poisson tessellations in Sec. 2.2, while posing distinct challenges in terms of the number of cells and the lack of regular (periodic) geometric structures, still relied on the assumption that the material properties were region-wise constant within each cell of the tessellation.

For a material comprised of many different isotopes, the total macroscopic cross section can be computed as

$$\Sigma_t(\mathbf{r}, E) = \sum_i N_i(\mathbf{r})\sigma_{t,i}(T(\mathbf{r}), E), \quad (2.22)$$

where N_i is the concentration of isotope i , $\sigma_{t,i}$ is the microscopic total cross section for isotope i , and T is the temperature [42]. The microscopic cross section for an isotope is generally given as a function of energy and temperature. However, temperature is generally a function of position, so we will simply

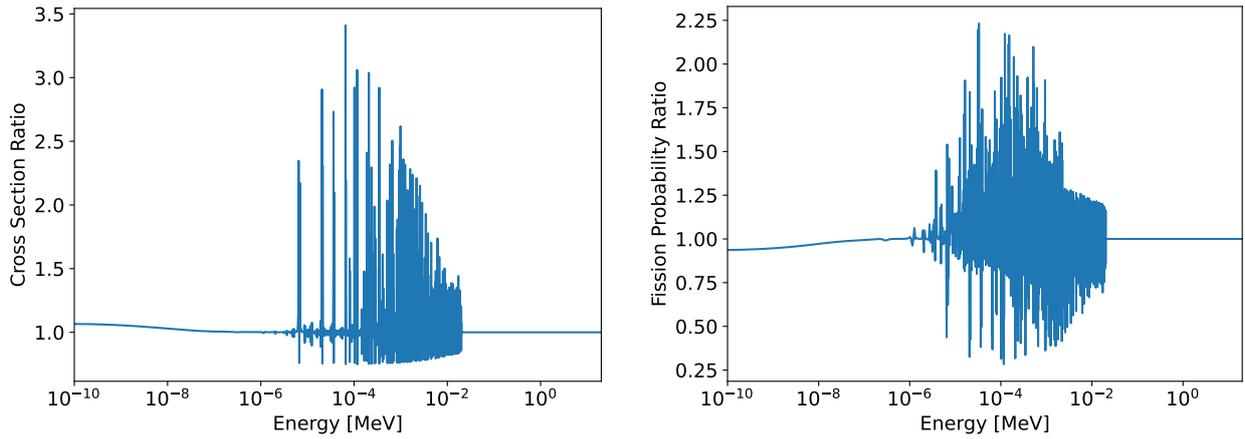


Figure 2.3: Ratio of the total cross section (left) and fission probability (right) of ^{238}U w/o enriched UO_2 , at 1500 K and 800 K.

consider the cross section to be a function of position: $\sigma_{t,i}(\mathbf{r}, E)$. This implies that the homogeneous medium assumption used previously in Sec. 2.1 and 2.2 (and commonly adopted in almost all existing general-purpose Monte Carlo codes) is only true when the isotopic concentrations and the temperature are constant within a given material volume. In real-world applications, however, this is rarely ever the case, and a more faithful description would inherently require the use of a spatially-continuous representation for material properties.

Let us first consider a simple example of a single fuel pin from a PWR reactor. The temperature at the center of a fuel pin in a PWR might be approximately 1500 K, while the temperature on the outer surface of the fuel might be approximately 800 K [43, 44]. Over a distance of less than 4 mm, the temperature can vary by approximately 700 K. The effects of this temperature variation can lead to large differences in the total cross section and thus the fission probability within the fuel. Figure 2.3 demonstrates that, within the resolved resonance region, the total cross section can be more than 3 times larger at the center of the fuel than at the fuel's outer surface. This subsequently produces large changes in the fission probability as well. For reactor eigenvalue calculations at steady state, these phenomena are often ignored, and an effective temperature is used for the entire fuel region.

The consideration of spatially-continuous media reaches far beyond reactor physics simulations at steady state. Recently, Monte Carlo neutronics solvers have been coupled with various thermal-hydraulics and/or thermo-mechanics solvers to perform various types of multi-physics simulations [45–49]. These studies vary greatly in scope, as some are examining operational or accidental power transients with Doppler broadening and moderator density feedback, while others are performing fuel burn-up analysis. The coupling strategy also varies, and is often accomplished through chained input files and calls to different physics solvers [48, 49]. Other multi-physics code drivers have been developed to perform the transfer of data between physics solvers in memory [50]. With such utilities, one can consider the possibility of being able to use, in particle transport, the spatially-continuous temperature and density fields that are computed by the thermal-hydraulics and thermo-mechanics solvers. Particles could then look up the temperature and density for their given position in space, and use that information when computing cross sections. Being able to make use of these spatially-continuous fields would increase the fidelity of Monte Carlo simulations for multi-physics applications.

Another use-case of spatially-continuous cross sections would be the spatially-continuous representation of isotope concentrations for reactor depletion simulations. Currently, fuel pins must be discretized into many different rings when performing depletion simulations, to account for the spatial self-shielding effects within the pin [51]. If this is not taken into account, inaccurate results will be obtained; conversely, increasing the pin discretization also increases the memory footprint of the simulation. Instead of using many discrete rings to model the fuel pellets, a higher fidelity option would be to model the isotope concentrations continuously,

using for instance Zernike polynomials [52]. Depending on the resolution required, it is also possible that a continuous representation could require less computation time, and have a smaller memory footprint [52].

It is thus clear that the treatment of spatially-continuous cross sections in Monte Carlo simulations can contribute to increased fidelity of results. We will now overview the difficulties that arise when incorporating spatially-continuous cross sections, the existing approaches to treat such cases, and possible strategies with potential to improve the efficiency and/or the accuracy of these methods. A thorough analysis will be then presented in Chapter 4.

2.3.1 . Sampling Flight Distances with Spatially-Continuous Cross Sections

If the total macroscopic cross section is known for all positions and energies in the problem domain, then the PDF for the particle to have its first interaction at $s\hat{\Omega} + \mathbf{r}_0$ when starting at position \mathbf{r}_0 and traveling in the direction of $\hat{\Omega}$ at energy E is [5]:

$$f(s) = \Sigma_t(s\hat{\Omega} + \mathbf{r}_0, E) \exp\left(-\int_0^s \Sigma_t(u\hat{\Omega} + \mathbf{r}_0, E) du\right). \quad (2.23)$$

The PDF may be integrated to produce the CDF, which is used to obtain the flight distance d :

$$F(d) = \int_0^d \Sigma_t(s\hat{\Omega} + \mathbf{r}_0, E) \exp\left(-\int_0^s \Sigma_t(u\hat{\Omega} + \mathbf{r}_0, E) du\right) ds. \quad (2.24)$$

When Σ_t is constant over the spatial variables, we retrieve the CDF used in Eq. (2.2) for the homogeneous media assumption. In general, the two integrals in Eq. (2.24) make it extremely cumbersome or even impossible to solve for the flight distance d using analytical inversion, as was done in Eq. (2.3). Instead, different sampling methods which do not rely on inversion should be used. Delta tracking and negative-weighted delta tracking are two tracking methods which can be used without modification for continuously-varying cross sections. Another method, referred to as “direct sampling”, has been proposed to treat spatially-continuous cross sections [53]. We will briefly outline the use of these techniques with spatially-continuous cross sections.

Delta Tracking

As delta tracking is a rejection sampling technique, no numerical integration of the macroscopic cross section is required to sample the flight distance. All that is needed is a valid majorant cross section to sample the distance to tentative collision sites. Unfortunately, determining this majorant in the context of coupled multi-physics simulations could be quite difficult. The dependence of the microscopic cross section on temperature is not trivial, and can be time-consuming to compute [54]. It is possible that alternative Doppler-broadening methods such as Target Motion Sampling could eliminate this problem, as work has already been conducted on how to compute the majorant cross section when temperature effects must also be considered [17, 18]. As the temperature and density fields are updated by different solvers, these quantities can change over the course of the simulation, making the determination of the majorant more difficult. Depending on the internal representation of the temperature field in the solvers being used, it might not even be possible to obtain the maximum and minimum temperatures for a material region in advance. Due to these difficulties, approximations might be made when determining a majorant, and Monte Carlo practitioners might be tempted to multiply this majorant by a large “safety factor”, in an effort to protect themselves from not having a true majorant cross section. On the surface, this seems like a good idea, but could lead to needlessly increasing the majorant, and therefore reducing the probability of a real collision, which in turn increases run times. Even if a “tight” majorant could be obtained, delta tracking could still suffer from the localized heavy absorber problem, and be very inefficient. Switching to surface tracking in such regions, as done in Serpent, is not an option with spatially-continuous cross sections. These limitations inherent to delta tracking lead us to consider the use of negative-weighted delta tracking, which does not require the use of a strict majorant cross section.

Negative-Weighted Delta Tracking

Both of the variants for negative-weighted delta tracking outlined in Sec. 2.1.3 have the advantage of not requiring a majorant cross section. It is therefore possible to forego the aforementioned difficulties in determining a majorant cross section for delta tracking, at the cost of introducing a mixture of positive and negative statistical weights to the simulation. The presence of a combination of positive and negative weights will in general increase the variance of scores in the simulation, which is undesirable at best and might prevent the simulation from achieving a satisfactory uncertainty at worst. Another potential difficulty in using the variant proposed by Legrady et al. is that a real collision probability must be provided, and the efficiency of the method can vary greatly depending on this parameter [20, 21]. In view of these considerations, in Chapter 4 one of our main objectives is to evaluate the trade-off between not needing a majorant, and the introduction of negative weights into the scores. Delta tracking, and both negative-weighted delta tracking variants are tested using several different cross section shapes, in 1D fixed-source benchmark problems for particle transport. The different functional forms of cross sections are given in Sec. 4.3, and the corresponding results are presented in Sec. 4.4. To compare two tracking methods, the FOM was used, but the time variable T in Eq. (2.19) is replaced with the number of cross section evaluations required for the simulation. Since cross section look-ups account for almost 80% of the run time in continuous energy codes, it is appropriate to assume that the run time is proportional to this quantity [55]. The findings from these simulations indicate that the variant of Legrady et al. has inconsistent performance, at times outperforming delta tracking, and at others performing much worse. It seems that the efficiency of this method is highly dependent on the choice of the real collision probability q , making the method less robust. Conversely, the variant of Carter et al. is shown to demonstrate performance which is comparable to, if not better than delta tracking for fixed-source problems, and seems to be more robust than the variant of Legrady et al.. Nonetheless, in Part II we will demonstrate that negative-weighted delta tracking displays severe instabilities when used for k -eigenvalue power iteration problems. We will analyze in detail the cause of these numerical difficulties, and propose a solution to this problem in Chapters 6, 7, and 8.

Direct Sampling

Direct sampling is a method which was proposed by Brown and Martin for performing transport in continuous media [53]. It is based on the use of the optical depth, defined as

$$\tau(x) = \int_0^x \Sigma_t(\mathbf{r}_0 + s\hat{\Omega}, E) ds. \quad (2.25)$$

Starting from position \mathbf{r}_0 and looking along the direction $\hat{\Omega}$, one first determines the distance d_s to the surface where the particle would leak (either from the current cell, or from the problem domain). The probability of the particle not undergoing a collision and leaking is then

$$P_{NC} = \exp(-\tau(d_s)). \quad (2.26)$$

Therefore, with probability P_{NC} the particle will either be moved to the next cell at a distance d_s , or leak from the problem domain and be killed. With probability $1 - P_{NC}$, the particle will undergo a collision at some distance $d < d_s$. It is then possible to calculate the distance to the collision using

$$\tau(d) = -\ln[1 - (1 - P_{NC})\xi] = \int_0^d \Sigma_t(\mathbf{r}_0 + s\hat{\Omega}, E) ds, \quad (2.27)$$

with ξ being a uniform random number in $[0, 1)$. To solve Eq. (2.27), it is possible to use Newton's method, or a bisection search to obtain d [53]. The weakness of this method is that for both determining P_{NC} and determining d through iterative methods one must be able to integrate the macroscopic cross section over any parameterized ray through 3D space. If limited to a particular subset of continuous media, where

the material density is the only quantity that varies spatially (and relative isotopic concentrations remain constant), then performing the required integration is reasonably feasible in a 3D setting, and has been demonstrated in conjunction with functional expansion tallies [56]. While this method has been used for depletion studies where relative isotopic concentrations were allowed to vary spatially, the optical depths were estimated by evaluating the cross section at several points along the flight path [52]. A large number of cross section evaluations might be needed to ensure that the accuracy of this integration is adequate to properly determine leakage rates. Varying isotopic concentrations are also easier to treat, as the macroscopic cross section is linear with respect to this quantity, and the microscopic cross section would only need to be evaluated once per nuclide. A spatially continuous temperature field is more difficult to treat, as the relationship between temperature and the microscopic cross section is not linear, and can be very costly to calculate [54]. Even if faster methods to describe the Doppler broadening effects are used [57, 58], due to the non-linear relationship, many cross section evaluations would be needed to perform integrals. We evaluate direct sampling in Chapter 4 alongside delta tracking and negative-weighted delta tracking, for the same set of 1D fixed-source benchmark problems. As outlined in Sec. 4.4, direct sampling has very poor performance compared to delta tracking and negative-weighted delta tracking, despite the very generous evaluation criteria, which considered a cross section integration to take the same amount of time as a cross section evaluation.

References

- [1] E. Brun, F. Damian, C. Diop, E. Dumonteil, F. Hugot, C. Jouanne, Y. Lee, F. Malvagi, A. Mazzolo, O. Petit, J. Trama, T. Visonneau, and A. Zoia, "TRIPOLI-4[®], CEA, EDF and AREVA reference Monte Carlo code," *Annals of Nuclear Energy*, vol. 82, p. 151–160, 2015.
- [2] P. K. Romano, N. E. Horelik, B. R. Herman, A. G. Nelson, B. Forget, and K. Smith, "OpenMC: A state-of-the-art Monte Carlo code for research and development," *Annals of Nuclear Energy*, vol. 82, p. 90–97, 2015.
- [3] J. Leppänen, M. Pusa, T. Viitanen, V. Valtavirta, and T. Kaltiaisenaho, "The Serpent Monte Carlo code: Status, development and applications in 2013," *Annals of Nuclear Energy*, vol. 82, p. 142–150, 2015.
- [4] T. Goorley, M. James, T. Booth, F. Brown, J. Bull, L. J. Cox, J. Durkee, J. Elson, M. Fensin, R. A. Forster, J. Hendricks, H. G. Hughes, R. Johns, B. Kiedrowski, R. Martz, S. Mashnik, G. McKinney, D. Pelowitz, R. Prael, J. Sweezy, L. Waters, T. Wilcox, and T. Zukaitis, "Initial MCNP6 Release Overview," *Nuclear Technology*, vol. 180, no. 3, p. 298–315, 2012.
- [5] I. Lux and L. Koblinger, *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations*. CRC Press, 1991.
- [6] G. Bell and S. Glasstone, *Nuclear Reactor Theory*. Van Nostrand Reinhold Company, 1970.
- [7] A. Hébert, *Applied Reactor Physics*, 3rd ed. Presses Internationales Polytechnique, 2020.
- [8] S. D. Roth, "Ray casting for modeling solids," *Computer Graphics and Image Processing*, vol. 18, no. 2, p. 109–144, 1982.
- [9] "MCNP-A General Monte Carlo N-Particle Transport Code, Version 5," Los Alamos National Laboratory, Tech. Rep. LA-UR-03-1987, 2003.
- [10] P. Hanrahan, "Ray Tracing Algebraic Surfaces," *Computer Graphics*, vol. 17, no. 3, p. 83–90, 1983.
- [11] R. L. Cook, T. Porter, and L. Carpenter, "Distributed Ray Tracing," *Computer Graphics*, vol. 18, no. 3, p. 137–145, 1984.

- [12] E. R. Woodcock, T. Murphy, P. J. Hemmings, and T. C. Longworth, "Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry," Argonne National Laboratory, Tech. Rep., 1965, ANL-7050.
- [13] J. Leppänen, "Performance of Woodcock delta-tracking in lattice physics applications using the Serpent Monte Carlo reactor physics burnup calculation code," *Annals of Nuclear Energy*, vol. 37, no. 5, p. 715–722, 2010.
- [14] —, "On the use of delta-tracking and the collision flux estimator in the Serpent 2 Monte Carlo particle transport code," *Annals of Nuclear Energy*, vol. 105, p. 161–167, 2017.
- [15] —, "Two practical methods for unionized energy grid construction in continuous-energy Monte Carlo neutron transport calculation," *Annals of Nuclear Energy*, vol. 36, no. 7, p. 878–885, 2009.
- [16] W. A. Coleman, "Mathematical Verification of a Certain Monte Carlo Sampling Technique and Applications of the Technique to Radiation Transport Problems," *Nuclear Science and Engineering*, vol. 32, no. 1, p. 76–81, 1968.
- [17] T. Viitanen and J. Leppänen, "Explicit Treatment of Thermal Motion in Continuous-Energy Monte Carlo Tracking Routines," *Nuclear Science and Engineering*, vol. 171, no. 2, p. 165–173, Oct 2012.
- [18] —, "Temperature Majorant Cross Sections in Monte Carlo Neutron Tracking," *Nuclear Science and Engineering*, vol. 180, no. 2, p. 209–223, 2015.
- [19] L. L. Carter, E. D. Cashwell, and W. M. Taylor, "Monte Carlo Sampling with Continuously Varying Cross Sections Along Flight Paths," *Nuclear Science and Engineering*, vol. 48, no. 4, p. 403–411, 1972.
- [20] D. Legrady, B. Molnar, M. Klausz, and T. Major, "Woodcock tracking with arbitrary sampling cross section using negative weights," *Annals of Nuclear Energy*, vol. 102, p. 116–123, 2017.
- [21] B. Molnar, G. Tolnai, and D. Legrady, "Variance Reduction and Optimization Strategies in a Biased Woodcock Particle Tracking Framework," *Nuclear Science and Engineering*, vol. 190, no. 1, p. 1–17, 2018.
- [22] D. M. Arnow, M. H. Kalos, M. A. Lee, and K. E. Schmidt, "Green's function Monte Carlo for few fermion problems," *The Journal of Chemical Physics*, vol. 77, no. 11, p. 5562–5572, 1982.
- [23] T. Yamamoto, "Monte Carlo method with complex weights for neutron leakage-corrected calculations and anisotropic diffusion coefficient generations," *Annals of Nuclear Energy*, vol. 50, p. 141–149, 2012.
- [24] —, "Monte Carlo method with complex-valued weights for frequency domain analyses of neutron noise," *Annals of Nuclear Energy*, vol. 58, p. 72–79, 2013.
- [25] Q. Guo and Z. Chen, "Multi-Regional Delta-Tracking Method for Neutron Transport Tracking in Monte Carlo Criticality Calculation," *Sustainability*, vol. 10, no. 7, p. 2272, 2018.
- [26] D. Meagher, "Geometric modeling using octree encoding," *Computer Graphics and Image Processing*, vol. 19, no. 2, p. 129–147, 1982.
- [27] A. S. Glassner, "Space subdivision for fast ray tracing," *IEEE Computer Graphics and Applications*, vol. 4, no. 10, p. 15–24, 1984.
- [28] C. Larmier, A. Zoia, F. Malvagi, E. Dumonteil, and A. Mazzolo, "Neutron multiplication in random media: Reactivity and kinetics parameters," *Annals of Nuclear Energy*, vol. 111, p. 391–406, 2018.

- [29] W. B. Doub, "Particle self-shielding in plates loaded with spherical poison particles," *Nuclear Science and Engineering*, vol. 10, no. 4, pp. 299–307, 1961.
- [30] K. Terrani, B. Jolly, M. Trammell, G. Vasadevamurthy, D. Schappel, B. Ade, G. Helmreich, H. Wang, A. M. Rossy, B. Betzler, and A. Nelson, "Architecture and properties of TCR fuel form," *Journal of Nuclear Materials*, vol. 547, p. 152781, 2021.
- [31] G. C. Pomraning, *Linear Kinetic Theory and Particle Transport in Stochastic Mixtures*. Singapore: World Scientific, 1991.
- [32] R. Schneider and W. Weil, *Stochastic and Integral Geometry*. Berlin, Germany: Springer, 2008.
- [33] C. Larmier, F.-X. Hugot, F. Malvagi, A. Mazzolo, and A. Zoia, "Benchmark solutions for transport in d-dimensional markov binary mixtures," *Journal of Quantitative Spectroscopy and Radiative Transfer*, vol. 189, p. 133–148, 2017.
- [34] Olson, Aaron, Prinja, Anil, and Franke, Brian, "Radiation transport in random media with large fluctuations," *EPJ Web Conf.*, vol. 153, p. 06014, 2017. [Online]. Available: <https://doi.org/10.1051/epjconf/201715306014>
- [35] C. Larmier, E. Dumonteil, F. Malvagi, A. Mazzolo, and A. Zoia, "Finite-size effects and percolation properties of Poisson geometries," *Physical Review E*, vol. 94, no. 1, p. 012130, 2016.
- [36] C. Larmier, A. Marinosci, and A. Zoia, "Chord length distribution in d-dimensional anisotropic Markov media," *Journal of Quantitative Spectroscopy and Radiative Transfer*, vol. 224, pp. 403–414, 2019.
- [37] M. A. Kowalski, C. Larmier, F. Madiot, J. Durand, S. Lemaire, and A. Zoia, "Particle transport in Markov media with spatial gradients: Comparison between reference solutions and Chord Length Sampling," *Journal of Quantitative Spectroscopy and Radiative Transfer*, vol. 286, p. 108185, 2022.
- [38] C. Larmier, "Stochastic particle transport in disordered media: beyond the Boltzmann equation," Ph.D. dissertation, Université Paris-Saclay, 2018.
- [39] P. Switzer, "A random set process in the plane with a Markovian property," *Annals of Mathematical Statistics*, vol. 36, pp. 1859–1863, 1965.
- [40] P. Shen, X. Guo, K. Li, S. Huang, L. Zheng, Q. Pan, and K. Wang, "Research on global neighbor list method in Monte Carlo code RMC," *Annals of Nuclear Energy*, vol. 167, p. 108861, 2022.
- [41] D. Lax, W. Boyd, N. Horelik, B. Forget, and K. Smith, "A Memory Efficient Algorithm for Classifying Unique Regions in Constructive Solid Geometries," in *PHYSOR 2014*, 2014.
- [42] J. R. Lamarsh, *Introduction to Nuclear Reactor Theory*. Addison-Wesley, 1966.
- [43] A. Scolaro, I. Clifford, C. Fiorina, and A. Pautz, "The OFFBEAT multi-dimensional fuel behavior solver," *Nuclear Engineering and Design*, vol. 358, p. 110416, 2020.
- [44] J. J. Duderstadt and L. J. Hamilton, *Nuclear Reactor Analysis*. John Wiley & Sons, 1976.
- [45] B. L. Sjenitzer, J. E. Hoogenboom, J. J. Escalante, and V. S. Espinoza, "Coupling of dynamic Monte Carlo with thermal-hydraulic feedback," *Annals of Nuclear Energy*, vol. 76, p. 27–39, 2015.
- [46] D. J. Kelly, A. E. Kelly, B. N. Aviles, A. T. Godfrey, R. K. Salko, and B. S. Collins, "MC21/CTF and VERA multiphysics solutions to VERA core physics benchmark progression problems 6 and 7," *Nuclear Engineering and Technology*, vol. 49, no. 6, p. 1326–1338, 2017.

- [47] M. Ellis, D. Gaston, B. Forget, and K. Smith, "Preliminary Coupling of the Monte Carlo Code OpenMC and the Multiphysics Object-Oriented Simulation Environment for Analyzing Doppler Feedback in Monte Carlo Simulations," *Nuclear Science and Engineering*, vol. 185, no. 1, p. 184–193, 2017.
- [48] C. Castagna, E. Cervi, S. Lorenzi, A. Cammi, D. Chiesa, M. Sisti, M. Nastasi, and E. Previtali, "A Serpent/OpenFOAM coupling for 3D burnup analysis," *The European Physical Journal Plus*, vol. 135, no. 6, p. 433, 2020.
- [49] D. Mancusi, M. Faucher, and A. Zoia, "Monte Carlo simulations of the SPERT III E-core transient experiments," *The European Physical Journal Plus*, vol. 137, no. 1, p. 127, 2022.
- [50] P. K. Romano, S. P. Hamilton, R. O. Rahaman, A. Novak, E. Merzari, S. M. Harper, P. C. Shriwise, and T. M. Evans, "A Code-Agnostic Driver Application for Coupled Neutronics and Thermal-Hydraulic Simulations," *Nuclear Science and Engineering*, vol. 195, no. 4, p. 1–21, 2020.
- [51] D. She, Y. Liu, K. Wang, G. Yu, B. Forget, P. K. Romano, and K. Smith, "Development of burnup methods and capabilities in Monte Carlo code RMC," *Annals of Nuclear Energy*, vol. 51, p. 289–294, 2013.
- [52] M. S. Ellis, "Methods for Including Multiphysics Feedback in Monte Carlo Reactor Physics Calculations," Ph.D. dissertation, Massachusetts Institute of Technology, 2017.
- [53] F. B. Brown and W. R. Martin, "Direct Sampling of Monte Carlo Flight Paths in Media with Continuously Varying Cross-Sections," in *M&C 2003*, 2003, LA-UR-02-6530.
- [54] D. E. Cullen and C. R. Weisbin, "Exact Doppler Broadening of Tabulated Cross Sections," *Nuclear Science and Engineering*, vol. 60, no. 3, p. 1999–229, 1976.
- [55] Y. Wang, E. Brun, F. Malvagi, and C. Calvin, "Competing Energy Lookup Algorithms in Monte Carlo Neutron Transport Calculations and their Optimization on CPU and Intel MIC Architectures," *Procedia Computer Science*, vol. 80, p. 484–495, 2016.
- [56] F. B. Brown, D. Griesheimer, and W. R. Martin, "Continuously Varying Material Properties and Tallies for Monte Carlo Calculations," in *PHYSOR 2004*, 2004, IA-UR-04-0732.
- [57] G. Yesilyurt, W. R. Martin, and F. B. Brown, "On-the-Fly Doppler Broadening for Monte Carlo Codes," *Nuclear Science and Engineering*, vol. 171, no. 3, p. 239–257, 2017.
- [58] C. Josey, P. Ducru, B. Forget, and K. Smith, "Windowed multipole for cross section Doppler broadening," *Journal of Computational Physics*, vol. 307, p. 715–727, 2016.

3 - Optimization of Particle Tracking Methods for Stochastic Media

This chapter has previously appeared as:

H. Belanger, C. Larmier, D. Mancusi, and A. Zoia, "Optimization of Particle Tracking Methods for Stochastic Media," In *Proceedings of the International Conference on Physics of Reactors 2022 (PHYSOR 2022)*, May 2022, Pittsburgh, PA, p. 294-303.

Optimization of Particle Tracking Methods for Stochastic Media

Hunter Belanger, Coline Larmier, Davide Mancusi, and Andrea Zoia

Université Paris-Saclay, CEA, Service d'Études des Réacteurs et de Mathématiques Appliquées,
91191, Gif-sur-Yvette, France
hunter.belanger@cea.fr, coline.larmier@cea.fr, davide.mancusi@cea.fr, andrea.zoia@cea.fr

ABSTRACT

Random media emerge in several applications involving particle transport, encompassing e.g. photon propagation through Rayleigh-Taylor instabilities in fuel pellets for inertial confinement fusion, or neutron multiplication problems related to the assessment of re-criticality risk following severe accidents with fuel degradation. Reference calculations in such material configurations by means of Monte Carlo transport codes are particularly challenging, since high-density stochastic media might involve several hundreds of thousands of volumes and thus make particle tracking routines extremely cumbersome. In order to cope with these issues, two distinct strategies have been proposed so far: the use of neighbor maps, or the use of delta tracking. In this work we will compare these methods and illustrate their specific merits and drawbacks, as taken both alone and in combination with each other.

KEYWORDS: Random media, Monte Carlo, Delta tracking, Neighbor map

1. INTRODUCTION

Particle transport in stochastic media is key in several technological applications in nuclear science, including e.g. the evaluation of re-criticality risk due to neutron multiplication in reactor cores following severe accidents with fuel degradation [1], photon propagation through turbulent material layers with Rayleigh-Taylor instabilities for inertial confinement fusion [2], or next-generation reactor design involving randomly dispersed fuel particle [3]. In all such situations, the material properties are known only statistically, and a specific stochastic model is needed in order to assign the cross sections at a given point. For this purpose, several approaches have been suggested, encompassing random inclusions, where objects of fixed or varying shape are randomly placed within a background matrix, or stochastic tessellations, where a given domain is randomly partitioned using planes [4]. In this work we will focus on the latter class of stochastic media, and adopt Markov media as a prototype example for our numerical investigations. Markov media are a widely used model, originally proposed for particle transport through stratified slab-like configurations in one dimension [2], and have recently been extended to three dimensions [5, 6].

Since three-dimensional Markov media will result in complex geometrical structures, it is natural to resort to Monte Carlo simulations (as opposed to deterministic methods) in order to perform particle transport. Monte Carlo circumvents the need for spatial meshes and the associated discretization bias, at the expense of increased computer time for the individual tracking of particle histories. In real-world applications, the number of volumes of each realization of Markov media might exceed several hundreds of thousands, which would make a naive approach to particle tracking in the best case extremely cumbersome; in the worst case, such calculations would simply be unfeasible, due to the overwhelming computer time which would be required. In order to speed up particle tracking in three-dimensional random media, two strategies have been introduced: the use of neighbor maps [6], or the use of delta tracking [7]. Neighbor maps associate each volume of the tessellation with a list of neighbors, so that a particle located within a volume performing surface-tracking during a flight only need to determine the next crossed volume from the list of neighbours

instead of the full list of volumes in the tessellation [8]: since the polyhedral volumes composing three-dimensional Markov media have on average six faces [4], the use of neighbor maps turns out to be highly effective in decreasing the cost of particle-tracking routines [6]. Delta tracking is based on replacing surface tracking by a collision-based rejection procedure whereupon either the particle flight stops or a new flight in the same direction is re-sampled [9]. Since this sampling procedure does not involve the tracking of the next surface in the direction of flight, the computer cost is greatly reduced and this method has been also shown to be very effective for particle transport in random media [7].

In this work, we will assess the specific merits and drawbacks of neighbor maps and delta tracking in simple benchmark configurations consisting in Markov media inspired by re-criticality problems in fuel assemblies with severe material degradation [1]. The two methods will be used both alone and in combination with each other, and the performances will be assessed as a function of the tessellation density. To the best of our knowledge, similar comparisons have never been attempted before, and thus shed light on the behaviour of such strategies for real-life applications. In addition, we will also probe the effectiveness of using superimposed meshes in order to further accelerate particle tracking. This paper is organized as follows: in Sec. 2 we will describe Markov media and the procedure to sample them. Then, in Sec. 3 we will detail the algorithms used for particle tracking. We will introduce a benchmark transport problem in Sec. 4 and present the corresponding simulation results in Sec. 5. Conclusions will be finally drawn in Sec. 6.

2. MARKOV MEDIA

Markov media can be produced by using a two-step procedure. The first step consists in using a Poisson hyperplane tessellation in order to partition a domain of a D -dimensional space into randomly shaped polyhedra by sampling hyperplanes from an auxiliary Poisson process [4]. The sampling of homogeneous and isotropic Poisson stochastic geometries depends on a single free parameter ρ , the so-called tessellation density, that influences the average number of resulting planes per unit volume. Homogeneous and isotropic Poisson tessellations satisfy a Markov property: for domains of infinite size, arbitrary drawn lines will be cut by the hyperplanes of the tessellation into segments whose lengths are exponentially distributed, with an average chord length $\Lambda = 1/\rho$ [4]. The quantity Λ intuitively defines the correlation length, i.e. the typical linear size of a volume composing the random tessellation. The second step consists of assigning each polyhedron of the tessellation a material property i with a given probability p_i , using the Switzer's coloring procedure [8]. The coloring probabilities p_i are typically chosen so as to preserve material volume ratios (i.e., the total amount of material that is present on average per unit volume). After the coloring procedure, the chord length distribution in each material i is still exponential, with a corresponding average $\Lambda_i = \Lambda/(1 - p_i)$, whence the name of Markov media. For illustration, examples of un-colored and colored Markov media are provided in Fig. 1.

One key property of finite-size Poisson geometries is the number of resulting polyhedral volumes, which is a random quantity for each realization and depends on ρ : the higher the tessellation density, the more polyhedra in the tessellation. For a Poisson tessellation applied to a D -dimensional box of side L , it has been shown that the average number of volumes scales as $(\rho L)^D$ and the associated standard deviation scales as $(\rho L)^{D-1/2}$ [8]. For infinite-size Poisson tessellations, the lowest-order moments and correlations (and in rare cases the full distributions) of a few significant polyhedral features are exactly known [4]; finite-size effects come into play when the tessellated domain is bounded [5]. In particular, the average number of D -dimensional faces per polyhedron, which intuitively represents the connectivity degree of the tessellation, reads $\langle C_D \rangle = 2D$ in infinite domains [4], which means that the average number of neighbors for a volume in dimension $D = 3$ will be $\langle C_3 \rangle = 6$. To the best of our knowledge, the associated distribution is not known, although numerical estimates have been provided by Monte Carlo methods for $D = 2$ and $D = 3$ [8]: in three-dimensional Poisson geometries, the number of neighbors per volume ranges typically between 4 and 10 with a probability close to 0.99, and falls off very rapidly after 11.

A code capable of sampling three-dimensional Poisson tessellations has been recently developed at CEA [5, 6]. During the generation of each stochastic realization, care is taken so as to store the indexes of the

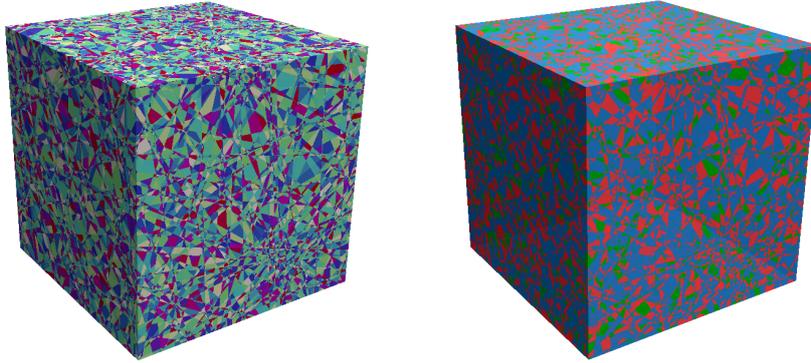


Figure 1: Realizations of a Poisson tessellation for $\rho = 2$ and $L = 50$, before (left) and after (right) the Switzer's coloring procedure. The number of volumes is about 6.5×10^5 . The coloring probabilities have been chosen in agreement with the benchmark specifications provided in Sec. 4: red corresponds to UOX fuel, blue to moderator and green to cladding.

neighboring volumes. The code can export not only the surface-based description of the volumes (including the assignation of material properties to each volume) but also the neighbor map, i.e. a map associating each volume index to the list of indices of the neighboring volumes. The format of the neighbor map can be easily read by Monte Carlo transport solvers. Moreover, an additional feature implemented in the generator allows superposing a regular Cartesian mesh (with arbitrary number of cells $n_x \times n_y \times n_z$ along the axes) over the realization and storing the list of polyhedral volumes present in each mesh cell. These lists are pre-computed by the generator and can be also easily exported to Monte Carlo transport solvers.

3. PARTICLE TRACKING METHODS

We detail now the algorithms that have been used to probe the efficiency of particle tracking routines for Markov media. All of the algorithms and methods discussed in this section have been implemented in a modified version of the MGMC Monte Carlo code [10], developed at CEA for R&D purposes. The new version of MGMC has been extended to not only handle three-dimensional multi-group problems, but also perform calculations in continuous energy, with modeling of all continuous-energy physics, scattering distributions, and thermal scattering laws. In addition to the original shared memory parallelization with OpenMP, distributed memory parallelization has been added with MPI, allowing effective use of cluster computing resources available at CEA. Use of this code facilitates quick development, and provides a rather modular setting where different tracking methods and simulation modes can quickly be examined, in a fairly realistic environment. The code represents general three-dimensional geometries using a Constructive Solid Geometry (CSG) model, which is quite standard in Monte Carlo codes; volumes are defined relative to different surface sides, and can be grouped together in universes. Universes may be placed inside of lattices, allowing for efficient replication of the same geometry portion. This structure then resembles a tree, and the exact location of a position in a geometry is defined by the path through the tree structure [11].

3.1. Surface Tracking

Surface tracking (ST) is the most common particle tracking method, used in most Monte Carlo neutral particle transport codes [12–14]. With this technique, the distance d_s to the nearest surface in the geometry along the direction of flight is determined, and compared to the sampled distance for the next collision site d . The distance to a collision site is calculated using the standard exponential distribution, which is true for volume-wise homogeneous material properties. In the event that $d > d_s$, then the particle is moved to the

Algorithm 1: Surface Tracking

```
1 Sample  $\xi \in \mathcal{U}(0, 1)$ 
2  $d := -\ln(\xi)/\Sigma_t$ 
3  $d_s :=$  distance to closest surface
4  $\mathbf{r} = \mathbf{r} + \min(d, d_s)\boldsymbol{\Omega}$ 
5 if  $d < d_s$  then
6 |   Perform collision
7 else
8 |   Find new volume and update  $\Sigma_t$ 
9 |   Goto 1
10 end
```

Algorithm 2: Delta Tracking

```
1 Sample  $\xi \in \mathcal{U}(0, 1)$ 
2  $d := -\ln(\xi)/\Sigma_{\text{maj}}(\mathbf{r}, E)$ 
3  $\mathbf{r} = \mathbf{r} + d\boldsymbol{\Omega}$ 
4 Find new volume and determine  $\Sigma_t$ 
5 Sample  $\xi' \in \mathcal{U}(0, 1)$ 
6 if  $\xi' < \Sigma_t(\mathbf{r}, E)/\Sigma_{\text{maj}}(\mathbf{r}, E)$  then
7 |   Perform real collision
8 else
9 |   Goto 1
10 end
```

corresponding surface, and the next material volume must be determined. Once the new material volume is found, the procedure starts again, and repeats until $d < d_s$, at which point the particle is advanced by d and a collision occurs. The general methodology is provided in Algorithm 1.

3.2. Delta Tracking

Delta tracking (DT), also known as Woodcock tracking, is an alternative to ST, and is the principal tracking method in the Serpent Monte Carlo code [9]. Use of this method requires a majorant cross section $\Sigma_{\text{maj}}(\mathbf{r}, E)$, which fulfills

$$\Sigma_{\text{maj}}(\mathbf{r}, E) \geq \Sigma_t(\mathbf{r}, E). \quad (1)$$

Here, the distance to a tentative collision site d is calculated using the typical exponential distribution, but Σ_t is replaced by Σ_{maj} . Once at the new location, we determine what volume the particle lies in and we sample whether the particle undergoes a real collision or a virtual collision. A real collision occurs with probability $P_r = \Sigma_t(\mathbf{r}, E)/\Sigma_{\text{maj}}(\mathbf{r}, E)$, and the particle correspondingly undergoes an interaction. A virtual collisions occurs with probability $1 - P_r$, and correspondingly neither the direction nor energy of the particle is changed. The particle then samples a new distance to a tentative collision site, and continues this process until a real collision is sampled. DT is outlined in Algorithm 2. With this method, one is not required to perform many surface intersection calculations, which can be quite costly. However, in the presence of small volumes having very large cross sections compared to the other volumes, the efficiency of delta tracking decreases, as many virtual collisions will be performed: this issue is known as the ‘localized heavy absorber problem’ [9].

3.3. Neighbor Maps

A neighbor map takes the form of a list for each volume, which provides a reference or handle to all other volumes in the geometry which touch the volume in question: this method provides a means of speeding up the volume search during the transport algorithm [15]. For ST, this acceleration technique is applied at line 8 of Algorithm 1. If a particle was previously in volume A , and has crossed the boundary of A to enter an immediate neighbor of A , then the particle only needs to look through the list of neighbors for volume A , and not the entire list of volumes, greatly speeding up the search for the new volume. The potential gain is particularly large in problems involving particle transport in random media with millions of volumes, as on average a volume in a Poisson tessellation has only six neighbors: for such large systems, use of a neighbor map becomes mandatory in order to keep the computational burden within acceptable limits.

Neighbors maps are used in combination with surface tracking in TRIPOLI-4[®], the production Monte Carlo code developed at CEA [13], and have been recently implemented in MGMC. We are not aware of examples of the use of neighbor maps with delta tracking in the literature. This is almost certainly due to the

reasoning that with delta tracking one is not guaranteed to move to a neighboring volume, but could instead traverse multiple volumes on a single flight. Therefore, it might at first seem as though no improvements could be made in using a neighbor map. However, in the context of particle transport in random media, if one considers the case where the mean free path length λ is of the same order of magnitude as the mean chord length Λ , then it is reasonable to assume that there is a large probability that a particle will travel to a neighbor of its last known volume in most cases. If this is true, one should be able to see performance gains when using a neighbor map with delta tracking, under certain conditions. For a quantitative assessment, we will explore different regimes of the ratio Λ/λ when considering the use of neighbor maps with delta tracking. The neighbor map is computed and provided by the stochastic geometry generator, and the map is then read and used by the MGMC transport code.

3.4. Volume Search Mesh

Another tracking acceleration option is the use of a volume search mesh. This takes the form of a regular Cartesian mesh imposed on top of the geometry. Each cell of this mesh contains a list of all volumes that are present within the mesh cell. It is very fast to determine in which mesh cell a particle is located. If a sufficiently fine mesh is used, there are much fewer volumes per mesh cell than volumes in the problem, so it should be much faster to determine in which volume a particle is located by using the list of volumes in the particle's current mesh cell. The Poisson tessellation generator used in this work is able to create this search mesh, which is then read and used by the MGMC Monte Carlo code.

3.5. Geometry Kernel Memory

When performing power iteration simulations, generated fission particles are typically stored in a bank, to be processed in the subsequent generation. While the exact position of the daughter particles and their paths in the CSG tree is known when the daughter particles are created, typically the only information that is stored is the position, direction, energy, and weight of the particle [9, 12]. This reduces the memory requirements for storing fission particles, and reduces the amount of data that needs to be sent between computer nodes between generations. For stochastic geometries, however, it can be costly to re-construct this path through the CSG tree. We have thus implemented a version of the MGMC code where the exact path through the CSG tree is saved with banked particles, and also transferred between nodes when synchronising fission banks between generations. Such an operation comes at the added cost of requiring more memory to store banked fission particles, and demands the transfer of more data with MPI calls between generations. Unlike in TRIPOLI-4[®], which uses a flat geometry structure and thus enables the information concerning the volume to be kept in memory with a single integer [13], MGMC uses a nested structure with universes, lattices, and volumes. Representing this unique path requires the storage of sets of several local coordinates, due to the possibility of nested lattices [11]. We therefore modified the code to only allow for up to 10 coordinate layers, effectively limiting users to have at most two nested lattices. This choice was made to facilitate an easier implementation, as having a maximum number of local coordinates greatly simplifies the logic for carrying out the synchronization between MPI nodes. We will refer to the structure containing the path through the CSG tree as the geometry kernel.

4. BENCHMARK SPECIFICATIONS

In this paper, we consider three different test systems. All three are realizations obeying ternary Markov mixtures, obtained from the Poisson geometry generator described in Sec. 2. These configurations were all cubical, with reflective boundary conditions on all sides. For the Switzer's coloring procedure, each volume of the tessellation was randomly assigned one of the following compositions: fuel, moderator or cladding. The respective coloring probabilities, provided in Tab. I, are defined so that the ensemble-averaged volume ratio of each material corresponds to those obtained in a square pin-cell of side length $\delta = 1.262082$ cm, with a fuel pin having an outer radius of $R_2 = 0.47436$ cm, surrounded by borated light water as a moderator. The fuel pin has two portions: UOX fuel at the center, with a radius $R_1 = 0.41266$ cm, and

Material	Isotopes	Concentration (atoms $\times 10^{24} \times \text{cm}^{-3}$)	Color probability
UOX fuel	U235	8.4148×10^{-4}	$\frac{\pi R_1^2}{\delta^2} \approx 0.335861$
	U238	2.1625×10^{-2}	
	O16	4.4932×10^{-2}	
Moderator	H1	4.7716×10^{-2}	$1 - \frac{\pi R_2^2}{\delta^2} \approx 0.556196$
	O16	2.3858×10^{-2}	
	B10	3.9724×10^{-6}	
	B11	1.5890×10^{-5}	
Cladding	ZR90	2.2060×10^{-2}	$\pi \frac{R_2^2 - R_1^2}{\delta^2} \approx 0.107943$
	ZR91	4.8107×10^{-3}	
	ZR92	7.3532×10^{-3}	
	ZR94	7.4518×10^{-3}	
	ZR96	1.2005×10^{-3}	

Table I: Material compositions for the UOX pin cell used for the benchmark configurations.

Case	ρ (cm^{-1})	Λ (cm)	λ (cm)
1	0.10	10.00	1.156
2	0.80	1.25	1.403
3	2.00	0.50	1.548

Table II: Description of the benchmark parameters.

then cladding between R_1 and R_2 . The respective material compositions for the fuel, the cladding and the moderator are provided in Tab. I. The proposed compositions correspond to fresh (Beginning Of Life) fuel. All materials were assumed to be at a uniform temperature of $T = 294$ K. These choices were inspired by an idealized model of fuel assembly with random material fragmentation following a severe accident [1].

The goal of our simulations is to assess the performance of the different tracking modes as a function of the ratio between the average chord length Λ of the tessellation and the mean free path λ . For this purpose, we will consider configurations corresponding to three regimes: $\Lambda \ll \lambda$, $\Lambda \sim \lambda$, and $\Lambda \gg \lambda$. Three realizations of the Markov media satisfying these regimes were sampled by using the tessellation densities ρ provided in Tab. II. Since λ is not known in advance and must be determined by a transport simulation, the parameters were determined by trial and error. For each realization, the side of the cube was then rescaled so that $L = 100/\rho$. Since the number of polyhedral volumes in the tessellation only depends on the product $L\rho$, the number of volumes is the same for the three realizations and is equal to 652628. A depiction of the configuration is presented in Fig. 1. For each of the three configurations, ST with the neighbor map, DT without the neighbor map, and DT with the neighbor map were considered. A second version of these nine cases was run, where the geometry kernel was stored with banked particles, giving 18 distinct test cases. A volume search mesh was used for almost all simulations, since computation times could otherwise exceed the allowed time limits on the computer cluster. To probe the effects of the mesh, we took each of the 18 test cases to be a function of n , representing the number of cells of the mesh on each axis of the box.

5. SIMULATION RESULTS

The calculations are performed with the Monte Carlo particle transport code MGMC described in Sec. 3, in criticality mode. For all configurations, we compute the multiplication factor k (based on a collision estimator) and the associated variance σ^2 , and measure the corresponding simulation wall clock time T .

The figure of merit (FOM) is computed as follows: $FOM = 1/(\sigma^2 T)$. Each transport simulation was run with 16 MPI ranks, each using 32 OpenMP threads, distributed over 256 CPUs of type EPYC 7281 2.1 GHz. All simulations were performed using 10^5 particles and 2500 batches (the first 500 of which were discarded to allow for source convergence).

The simulation times are displayed in Fig. 2, and the corresponding FOM values in Fig. 3. From these results, it becomes evident that there are two different regimes with respect to n : when $1 \leq n \leq 10$, increasing the size of the volume search mesh decreases the run time exponentially. Once the search mesh reaches a size $n \times n \times n = 10^3$, however, a further increase in n does little to improve performance, and we see that run times level off. For $n > 10$, we are therefore dominated by the volume search mesh. The right hand columns of Figs. 2 and 3 provide a better view of the asymptotic behavior of this regime. For case 1, when $\rho = 0.1$, ST has a higher FOM than the DT variants; as ρ increases, the DT variants become more effective for $\Lambda \ll \lambda$. A counter-intuitive observation for the $n > 10$ regime is that DT without the neighbors map consistently performs better than DT with the neighbor map, in all cases. During DT, when a particle is not in its last known volume, it first searches the neighbor map (if one is present), and if this fails then it will use the mesh. With DT, one is not guaranteed to be in a neighbor of the last known volume, but one is guaranteed to be in one of the volumes provided by the volume search mesh. This means that systematically using the neighbor map first leads to extra searches for the case where a particle is not in a neighbor. When the number of neighbors and the number of volumes provided by the search mesh are somewhat similar, it will in general take more time to look at the neighbors first, although there exist cases where a particle is more likely to have flown into a neighboring volume. One important note is that all simulation cases that transferred the geometry kernel with the banked particle took longer to run than the version of the code not keeping the geometry kernel. This can be explained due to the efficiency of the search mesh when $n > 10$, making it very fast to re-generate the geometry kernel when starting a new history. Keeping this geometry kernel with the banked particles leads to added MPI costs, and this is what can explain the longer run times for the trials that kept the geometry kernels.

When looking at the $1 < n < 10$ regime, in the left-hand columns of Figs. 2 and 3, we first see that ST with a neighbor map is always more efficient than either of the DT variants. This can be explained by the fact that with ST a particle is always guaranteed to fly into a neighbor; if the neighbors are known for each volume, the time required to determine the next volume is minimal. Examining the two DT variants, it is seen that when $\Lambda \gg \lambda$ the neighbor map version is faster. This is logical, as in this case it is less likely that a particle will fly out of a volume at all. Furthermore, in the event that it does, it is very likely to have only flown at most one volume away (i.e. into a neighbor). As the ratio of Λ/λ decreases, the improvements that the neighbor map provides also decrease, but always seem to yield a slightly lower run time. Such behavior is the converse of that observed in the asymptotic regime of $n > 10$. When the number of volumes provided by the volume search mesh is very large (compared to the average number of neighbors a volume has), then it is always beneficial to look in the neighbors first. It is still more likely that a particle is to fly into a neighboring volume than into a volume being farther away. Therefore, it is more efficient to check the (on average) 6 neighboring volumes first, and then start looking at the thousands of volumes which may have been provided by the search mesh. Keeping the geometry kernel with the banked particles turns out to provide a very reasonable gain in efficiency for ST when the volume search mesh is not used or is not effective, and reduces run times by approximately a factor of 2. While DT with the neighbor map demonstrated some improvements when storing the geometry kernel for case 1, it appears that in general keeping the geometry kernel is not particularly advantageous for either of the DT methods.

6. CONCLUSIONS

In this work we have examined several different transport methods for performing Monte Carlo particle transport in stochastic geometries. Surface tracking (ST) and delta tracking (DT) were both tested using neighbor maps, and volume search meshes of varying refinement. We also looked at the possibility of storing the particles position in the CSG tree between generations. In general, if a volume search mesh is not used or not optimally refined, ST with a neighbor map is always better than DT, with or without a

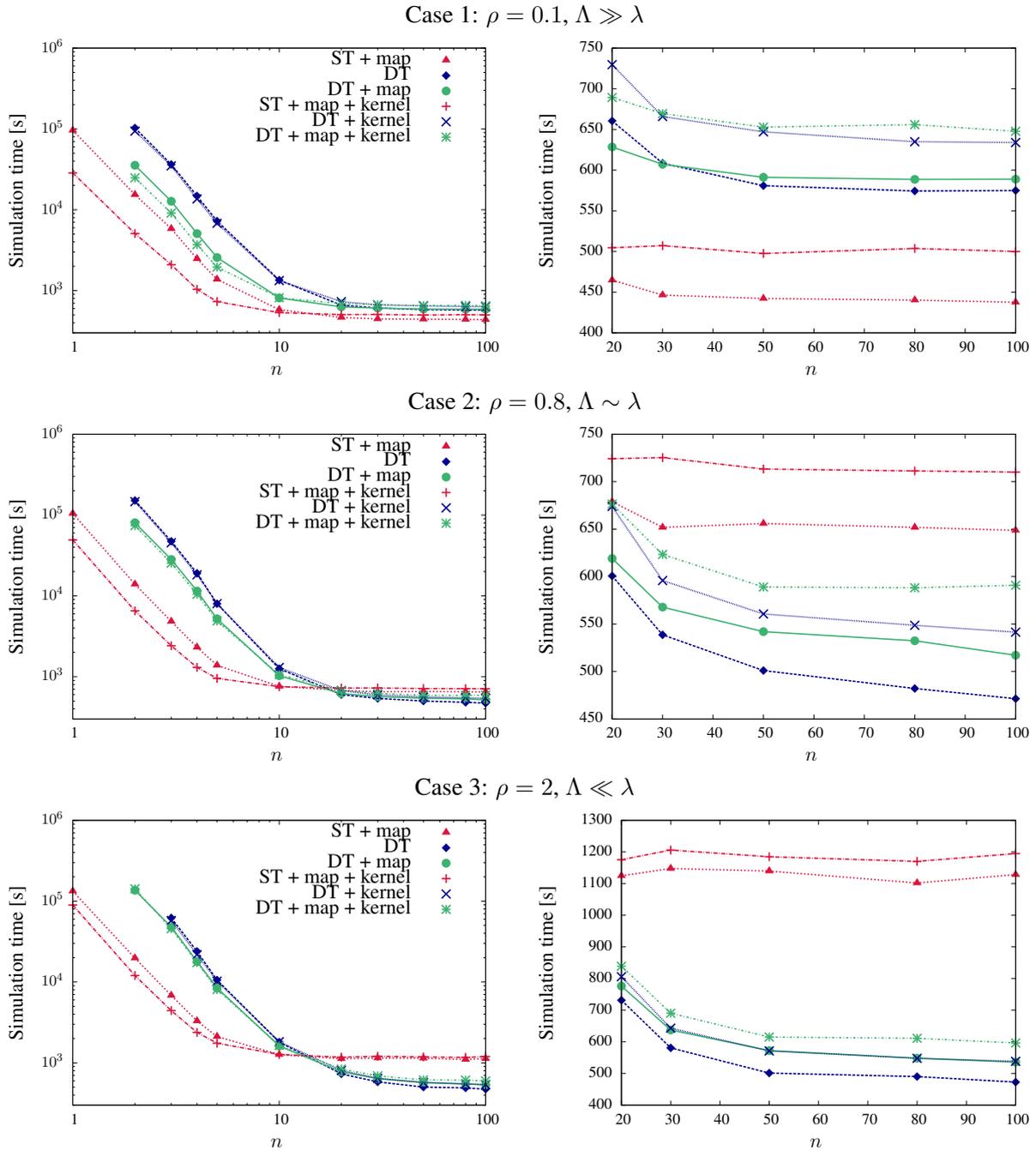


Figure 2: Left. Simulation time (in seconds) as a function of the number n of cells in the mesh along each axis, for the following benchmark configurations: $\rho = 0.1$ (top), $\rho = 0.8$ (middle) and $\rho = 2$ (bottom). Each plot displays different tracking modes: red triangles (resp. pluses) refer to ST with neighbor map without (resp. with) storing the geometry kernel, blue diamonds (resp. crosses) to DT without neighbor map without (resp. with) storing the geometry kernel and green circles (resp. stars) to DT with neighbor map without (resp. with) storing the geometry kernel. Right. Same data, displayed with a focus on the range $n \in [20, 100]$.

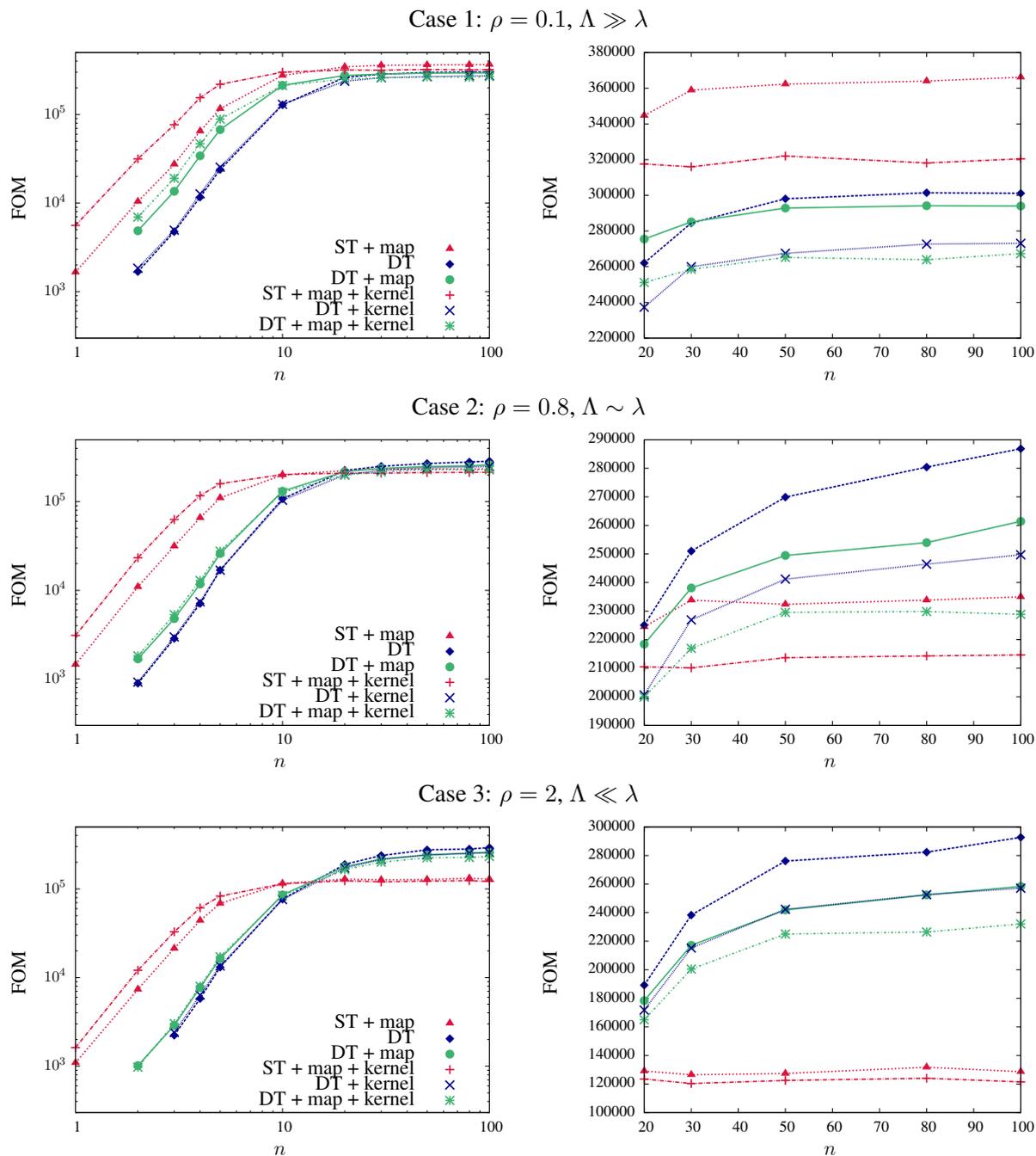


Figure 3: Left. Figure of merit (FOM) as a function of the number n of cells in the mesh along each axis, for the following benchmark configurations: $\rho = 0.1$ (top), $\rho = 0.8$ (middle) and $\rho = 2$ (bottom). Each plot displays different tracking modes: red triangles (resp. pluses) refer to ST with neighbor map without (resp. with) storing the geometry kernel, blue diamonds (resp. crosses) to DT without neighbor map without (resp. with) storing the geometry kernel and green circles (resp. stars) to DT with neighbor map without (resp. with) storing the geometry kernel. Right. Same data, displayed with a focus on the range $n \in [20, 100]$.

neighbor map. If a volume search mesh is used and is sufficiently refined, then DT can be more efficient than ST when the mean chord length is larger than the mean free path. Storing the CSG tree in memory does reduce run times for ST when the volume search mesh is not used or is ineffective, but is not significantly beneficial for DT. If an optimal volume search mesh is used, storing the CSG tree is detrimental to the run time for all methods, due to the increased MPI overhead. While this study only analyzed performance with regard to the global parameter k_{eff} , a topic of further investigation would be an analysis of performance on scores for local reaction rates, where DT is only able to use a collision estimator, while ST is able to use a track-length estimator.

REFERENCES

- [1] C. Larmier, A. Zoia, F. Malvagi, E. Dumonteil, and A. Mazzolo. “Neutron multiplication in random media: Reactivity and kinetics parameters.” *Annals of Nuclear Energy*, **volume 111**, pp. 391–406 (2018).
- [2] G. C. Pomraning. *Linear Kinetic Theory and Particle Transport in Stochastic Mixtures*. World Scientific, Singapore (1991).
- [3] F. B. Brown and W. R. Martin. “Stochastic geometry capability in MCNP5 for the analysis of particle fuel.” *Annals of Nuclear Energy*, **volume 31**(17), p. 2039–2047 (2004).
- [4] R. Schneider and W. Weil. *Stochastic and Integral Geometry*. Springer, Berlin, Germany (2008).
- [5] C. Larmier, E. Dumonteil, F. Malvagi, A. Mazzolo, and A. Zoia. “Finite-size effects and percolation properties of Poisson geometries.” *Physical Review E*, **volume 94**, p. 012130 (2016).
- [6] C. Larmier, F.-X. Hugot, F. Malvagi, A. Mazzolo, and A. Zoia. “Benchmark solutions for transport in d -dimensional Markov binary mixtures.” *J Quant Spectrosc Radiat Transfer*, **volume 189**, p. 133–148 (2017).
- [7] A. J. Olson, A. K. Prinja, and B. C. Franke. “Woodcock Monte Carlo Transport Through Karhunen-Loeve Reconstructed Binary Stochastic Media.” In *ANS Winter Meeting* (2014).
- [8] C. Larmier. *Stochastic particle transport in disordered media: beyond the Boltzmann equation*. Ph.D. thesis, Université Paris-Saclay (2018).
- [9] J. Leppänen. “Performance of Woodcock delta-tracking in lattice physics applications using the Serpent Monte Carlo reactor physics burnup calculation code.” *Annals of Nuclear Energy*, **volume 37**(5), p. 715–722 (2010).
- [10] H. Belanger. “MGMC.” (2021). URL <https://github.com/HunterBelanger/mgmc>. v0.2.0-alpha.
- [11] D. Lax, W. Boyd, N. Horelik, B. Forget, and K. Smith. “A Memory Efficient Algorithm for Classifying Unique Regions in Constructive Solid Geometries.” In *PHYSOR 2014* (2014).
- [12] P. K. Romano, N. E. Horelik, B. R. Herman, A. G. Nelson, B. Forget, and K. Smith. “OpenMC: A state-of-the-art Monte Carlo code for research and development.” *Annals of Nuclear Energy*, **volume 82**, p. 90–97 (2015).
- [13] E. Brun et al. “TRIPOLI-4@, CEA, EDF and AREVA reference Monte Carlo code.” *Annals of Nuclear Energy*, **volume 82**, p. 151–160 (2015).
- [14] T. Goorley et al. “Features of MCNP6.” *Annals of Nuclear Energy*, **volume 87**, pp. 772–783 (2016).
- [15] P. Shen, X. Guo, K. Li, S. Huang, L. Zheng, Q. Pan, and K. Wang. “Research on global neighbor list method in Monte Carlo code RMC.” *Annals of Nuclear Energy*, **volume 167**, p. 108861 (2022).

4 - Review of Monte Carlo Methods for Particle Transport in Continuously-Varying Media

This chapter has previously appeared as:

H. Belanger, D. Mancusi, and A. Zoia, "Review of Monte Carlo methods for particle transport in continuously-varying media," *European Physical Journal Plus*, vol. 135, no. 11, p. 877, 2020, doi: [10.1140/epjp/s13360-020-00731-y](https://doi.org/10.1140/epjp/s13360-020-00731-y).

Review of Monte Carlo methods for particle transport in continuously-varying media

Hunter Belanger, Davide Mancusi, and Andrea Zoia

DES-Service d'études des réacteurs et de mathématiques appliquées (SERMA), CEA,
Université Paris-Saclay, 91191 Gif-sur-Yvette, France

Abstract. Monte Carlo techniques for solving the Boltzmann neutron transport equation have become increasingly popular in recent years, due to their exact nature and to the increase in computational power. Monte Carlo makes it possible to solve transport problems without discretizing the geometry or energy domain. Current mainstream methods, however, do not allow for media with continuous properties, such as spatially-varying cross sections due to density and temperature variations. Being able to perform simulations under such conditions is a desirable feature for a complete Monte Carlo implementation in the contexts of multi-physics and depletion simulations. Several methods can currently be used to conduct transport in materials having non-uniform cross sections. This paper aims to compare some of these methods (substepping, direct sampling, delta tracking, and several delta tracking variants) in a simplified one-dimensional system with spatially-varying cross sections, to examine their respective properties. Performance (as measured by the Figure of Merit), robustness, and ease of implementation are considered in evaluating the methods. Such results will help determine which transport methods would be most suitable for implementation in a general-purpose Monte Carlo code.

1 Introduction

The Monte Carlo method has been in use since the dawn of computational nuclear engineering to solve the Boltzmann neutron transport equation(s), and has become increasingly popular, due to the fact that solutions are exact (no approximation and discretization are introduced), and its large computational requirements are offset by the rapid development of modern super-computers and by the relatively simple implementation of massively parallel Monte Carlo calculation strategies, making this method increasingly accessible. Not only are static criticality problems for full reactor cores possible, but also depletion problems, kinetics, and even multi-physics problems coupled with thermo-hydraulics solvers can nowadays be tackled using Monte Carlo neutronics solvers [1–4].

This new realm of problems poses particular challenges that have not traditionally been considered in Monte Carlo transport codes. When considering coupling with thermo-hydraulics solvers, cross sections have to be prepared at many different temperatures. Depending on the temperature resolution required to adequately simulate the system and the number of different nuclides present, a prohibitively large amount of memory could be required to hold the necessary data. This limitation fostered the development of algorithms that avoid cross-section preprocessing, such as on-the-fly Doppler broadening of cross sections [5] or target motion sampling [6]. Such methods may mitigate the memory problem in the context of multi-physics simulations, but fail to address the fact that the temperature within a given cell, be it a fuel pin, or moderator channel, is not uniform but has a spatial dependence, therefore imposing a spatial dependence on the cross section as well. The remedy to this issue has traditionally been to homogenize the problem domain by giving each cell a uniform temperature, which is valid at every point in the cell. This temperature is typically selected so as to best represent the physical properties of the system to be simulated [7].

The magnitude of the variation of the cross section within the cell is highly dependent on the neutron energy, the material composition for the given cell, and the variations in temperature which can be expected. For example, a typical fuel pin in a PWR reactor might have an outer surface temperature of approximately 500 K, and a core temperature near 1500 K [8]. As an order-of-magnitude estimate of the extent of the possible cross-section variation, we consider the ratio of the total cross section for ^{238}U at 1500 K to the total cross section at 500 K. The choice of ^{238}U for this example is motivated by the fact that it is a primary component of the fuel. Figure 1 shows that it is possible to have up to an order of magnitude in the value of the cross section between the center and outer edge of a pin, for neutron energies in the resolved resonance region. Even in the thermal region, the ratio deviates from unity by approximately 18%. The effects of temperature for materials such as water, however, are somewhat milder, as illustrated by the total cross section ratio for ^1H at 600 K to 300 K, shown in Figure 2. For energies greater than

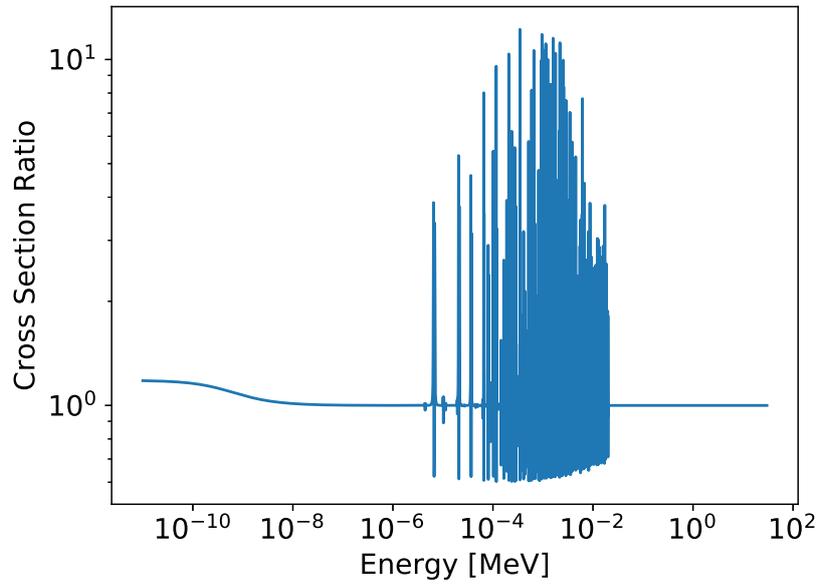


Fig. 1: Ratio of the total cross section at 1500 K to 500 K for ^{238}U . Cross sections have been extracted from the ENDF-B.VIII library [9] and preprocessed with NJOY2016 [10].

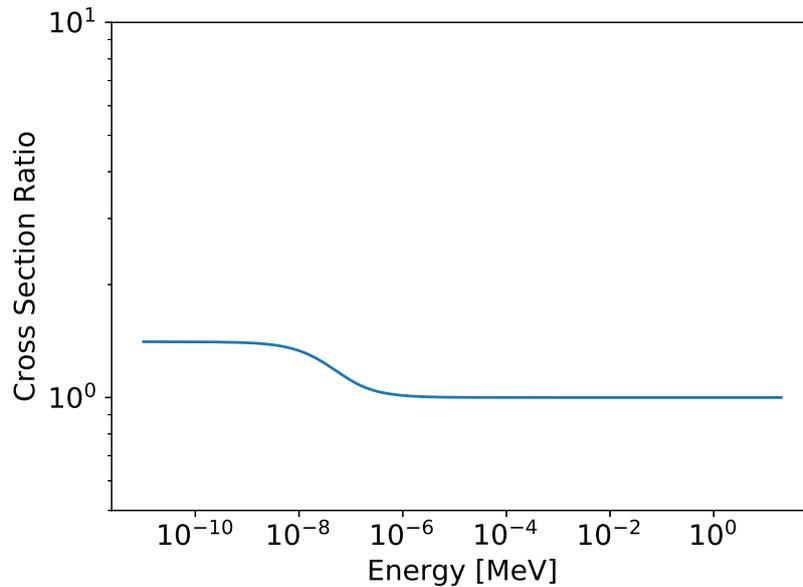


Fig. 2: Ratio of the total cross section at 600 K to 300 K for ^1H . Cross sections have been extracted from the ENDF-B.VIII library [9] and preprocessed with NJOY2016 [10].

100 eV, the cross sections are very nearly equal, while in the thermal region the 600 K cross section is approximately 40% larger. Ignoring these spatial variations leads to a biased sampling of the collision sites for neutrons during a Monte Carlo simulation.

Temperature gradients may also affect the branching ratios of reaction channels. Figure 3 shows the ratio of the fission probability at 1500 K to that at 500 K for ^{235}U . Most of the variation here is again observed within the resonance region, where neutrons spend a small part of their time, but the probability for fission can still be as much as 2.5 times greater at 1500 K than for 500 K. Effects due to this branching are lost under the assumption of homogeneous media.

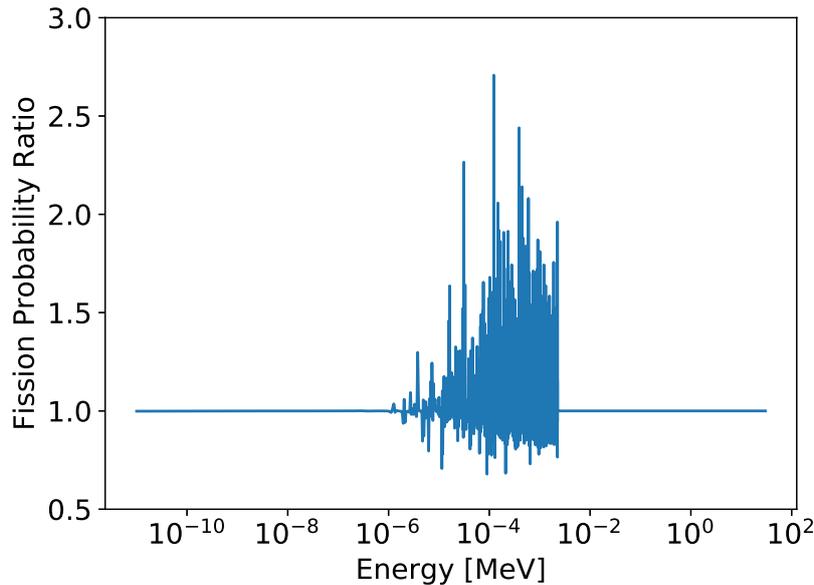


Fig. 3: Ratio of the fission probability at 1500 K to 500 K for ^{235}U .

One solution to this problem has been to divide regions into smaller portions, each with their own material properties. This method however would require a very large number of subdivisions to produce accurate results in certain cases, and is not exact, but is the only feasible option for the standard method of ray-tracing, used in most general-purpose codes such as MCNP [11] and TRIPOLI-4[®] [12]. Delta tracking [13], on the contrary, is an exact sampling method for flights in continuously varying media. Delta tracking is based on rejection sampling of collision sites and on the definition of a *majorant* cross section, i.e. a cross section that must always be larger than the total cross section. Determining a proper value for the majorant cross section is the main difficulty of this method. Use of a value which is not a true majorant can lead to events having a probability greater than unity: if this occurs, the resulting collision sites are no longer valid. In an attempt to avoid this situation, it could be tempting to overestimate the majorant cross section, ensuring global statistical validity. Overestimating the majorant however will decrease the efficiency of the method and increase the computation time [14]. This is also congruous to the well-known local heavy absorber problem, also reducing the efficiency of delta tracking [15]. If one were to consider the density and temperature to be non-constant, then ensuring a proper majorant at all locations could become a cumbersome task. Vittanen and Leppänen have previously developed a method of determining the majorant cross section for use in the target motion sampling algorithm [6]. While this problem is distinct from the currently considered issue of varying temperatures, one could envision a similar process for the determination of majorant cross sections. A variant of delta tracking has previously been used to perform transport calculations with varying cross sections due to density [16].

Other tracking methods for continuous media have also been proposed, such as the direct integration approach introduced by Brown and Martin [17]. While several different algorithms and strategies have been investigated in recent years, there has been little work done to apply and compare these methods to transport problems where cross sections are spatially varying, such as one might find in a multi-physics context.

The aim of this work is thus to shine light on a few existing methods by applying them to a simple continuous transport problem, and comparing their advantages and disadvantages. We propose three evaluation endpoints: performance, robustness, and simplicity. Any new method should have a performance at least equal to the current standard. Since we are considering continuous media, we have chosen to consider the reference method to be delta tracking, such as implemented in the Monte Carlo code Serpent [18]. New methods must be robust, in the sense that they do not break easily, and have few conditions attached. If an algorithm cannot easily handle real-world simulations, or requires that too many other conditions be met, it will serve little use. Lastly, a simple method should be preferred over a more complicated method. This would allow it to be implemented more efficiently in existing codes, and would likely also lend itself to being a more robust method. An evaluation of current methods in this context will hopefully illuminate a direction for better implementing more accurate multi-physics Monte Carlo simulations.

This paper is structured as follows: Sec. 2 will provide an overview of the current available methods to perform neutral particle transport in continuous media. Sec. 3 outlines the benchmark system used to compare various methods,

and specifies the examined observables. A discussion of the performance of the investigated tracking methods is provided in Sec. 4, followed by the concluding remarks in Sec. 5.

2 Particle tracking in continuous media

We start by reviewing the theory of neutral particle tracking in continuous media. Let E be the particle energy. The probability that a particle at position \vec{r} has a collision within an infinitesimal distance ds is known to be

$$dP = \Sigma_t(\vec{r}, E)ds \quad (1)$$

where Σ_t is the total macroscopic cross section as a function of the particle position and energy. The probability distribution function (PDF) for the particle traveling from its current location at \vec{r}_0 in the direction $\hat{\Omega}$, and having its first interaction at $s\hat{\Omega} + \vec{r}_0$ is

$$f(s) = \Sigma_t(s\hat{\Omega} + \vec{r}_0, E) \exp \left[- \int_0^s \Sigma_t(u\hat{\Omega} + \vec{r}_0, E)du \right]. \quad (2)$$

Upon integration of the PDF, the cumulative distribution function (CDF) for the flight distance d is obtained:

$$F(d) = \int_0^d \Sigma_t(s\hat{\Omega} + \vec{r}_0, E) \exp \left[- \int_0^s \Sigma_t(u\hat{\Omega} + \vec{r}_0, E)du \right] ds. \quad (3)$$

In order to then sample the flight distance, a uniform random variable $\xi \in \mathcal{U}(0, 1)$ would be generated, and set equal to the CDF, thereby providing the sampled flight distance d as [19]

$$F^{-1}(\xi) = d. \quad (4)$$

2.1 Ray tracing in piecewise homogeneous media

Solving for the inverse of Eq. (3) is hardly feasible, except under certain conditions. Traditionally, the common assumption used in most Monte Carlo transport codes is that of piecewise homogeneous media. This requires that the macroscopic total cross section be constant within a given region or volume. With the enforcement of this requirement, Eqs. 3 and 4 become the much tamer, and more familiar

$$1 - e^{-\Sigma_t(E)d} = \xi \quad \Rightarrow \quad d = -\frac{\ln(\xi)}{\Sigma_t(E)}. \quad (5)$$

Under this assumption, the tracking process then follows an algorithm that we shall refer to in this paper as ray tracing (the term *surface-to-surface tracking* is also commonly used in the literature). The general idea of this procedure is outlined in Algorithm 1. With this approach, care must be taken to ensure that the boundary crossings are properly handled. The flight distance sampling is only valid for the current material; as such, should a distance larger than the distance to the closest surface along the direction of flight be chosen, the particle may only move to that surface intersection. The cell and material are then changed to take into account this new region, and a new flight distance must then be selected with the new total cross section.

The main limitation of this method is the fact that material cells must have piecewise constant cross sections. Thus, ray tracing cannot faithfully represent systems with continuously-varying cross sections; nevertheless, it is possible to adapt ray tracing by dividing cells into many smaller cells, each with a constant cross section, in a method known as substepping [17]. Depending on the problem being considered, many subdivisions could be required, potentially using large amounts of computer memory, and being computationally inefficient. Such a method would still only ever provide an approximate solution to the problem of interest, however, and could only yield the true solution in the limit of the number of subdivisions tending to infinity.

To date, several alternative tracking methods have been proposed, each with their own advantages and shortcomings. Some of these methods make use of cells and ray tracing in addition to other techniques, while others do not make use of these classic tools. A brief explanation of each of these methods is provided in the rest of this section.

Algorithm 1: Ray-tracing procedure.

```

1 Sample uniform random variable  $\xi$ ;
2  $d := -\ln(\xi)/\Sigma_t(E)$ ;
3  $d_s :=$  distance to closest surface;
4 if  $d < d_s$  then
5   | Move particle by  $d$ ;
6   | Perform real collision;
7   | Tally real collision estimator;
8 else
9   | Move particle by  $d_s$  to new cell;
10  | Update  $\Sigma_t(E)$  for current cell;
11  | Goto line 1;
12 end

```

2.2 Delta (Woodcock) tracking

Delta tracking (also referred to as Woodcock tracking) is a rejection sampling technique, first made popular by Woodcock in the 1960's [13]. Subsequently, variants of this technique have been implemented in many Monte Carlo codes, most notably in Serpent [15]. In this method, a majorant cross section $\Sigma_{maj}(E)$ is chosen at each energy in the unionized energy grid, under the constraint that

$$\Sigma_{maj}(E) \geq \max_{\vec{r}} \Sigma_t(\vec{r}, E). \quad (6)$$

Here $\Sigma_t(E, \vec{r})$ may be a piecewise constant function in space, in which case Eq. (6) reduces to the maximum over all the cells of the total cross section at the given energy. Using the majorant cross section, a tentative collision site is selected by sampling a flight distance with

$$d = -\frac{\ln(\xi)}{\Sigma_{maj}(E)}. \quad (7)$$

Once at the new location $\vec{r}_1 = d\hat{\Omega} + \vec{r}_0$, there are two possibilities: a normal collision (called a *real* collision) may occur with probability $P_{real} = \Sigma_t(\vec{r}_1, E)/\Sigma_{maj}(E)$, or a *virtual* collision may occur with probability $P_{virt} = 1 - P_{real}$. Should a virtual collision occur, the particle is just displaced to the collision site, leaving both its flight direction and energy unchanged. A new flight distance is sampled using Eq. (7), and the process starts again. This continues until the particle undergoes a real collision. The procedure is outlined in Algorithm 2.

Algorithm 2: Delta tracking procedure.

```

1 Sample uniform random variable  $\xi_1$ ;
2  $d := -\ln(\xi_1)/\Sigma_{maj}(E)$ ;
3  $\vec{r}_1 := d\hat{\Omega} + \vec{r}_0$ ;
4 Tally total collision estimator;
5 Sample random variable  $\xi_2$ ;
6 if  $\xi_2 < \Sigma_t(\vec{r}_1, E)/\Sigma_{maj}(E)$  then
7   | Perform real collision;
8   | Tally real collision estimator;
9 else
10  |  $\vec{r}_0 := \vec{r}_1$ ;
11  | Virtual collision; goto line 1;
12 end

```

This method has the advantage of not needing to calculate surface intersections at each flight, which are often computationally expensive. There are known issues with this method, however, such as its performance in the presence of localized heavy absorbers, which lead to many virtual collisions [15]. Such inefficiencies are often encountered in reactor problems, where a control rod can have a much larger total cross section compared to other materials in the core, despite taking up a very small portion of the total volume. When a particle finds itself in a material with a cross section which is much smaller than the majorant, many virtual collisions will occur, as many short flight distances will be sampled without the particle having a real collision (see the previous definition of P_{real}) [15]. Also problematic is

the requirement of ensuring a majorant cross section, which can become difficult in the context of continuous media, where variations in the position change both the density and temperature, in turn changing the macroscopic total cross section. If one were to drastically overestimate the total cross section in an effort to safely ensure a majorant, the same inefficiencies such as those observed in the local heavy absorber problem would be encountered, as needless virtual collisions would be performed. Nonetheless, delta tracking has been proven to be a reliable way to solve problems with non-constant densities [16].

2.3 Direct sampling

The direct sampling method as proposed by Brown and Martin involves numerically integrating the macroscopic total cross section to obtain the flight distance [17]. The probability of the particle not having a collision in the cell, and leaking (either out of the problem or to another cell) can be defined as

$$P_{NC} = \exp[-\tau(d_s)] = \exp\left[-\int_0^{d_s} \Sigma_t(s\hat{\Omega} + \vec{r}_0, E)ds\right]. \quad (8)$$

This equation generally speaking implies numerical integration of the cross section to obtain the optical depth $\tau(d_s)$ to the cell boundary. There is then a probability P_{NC} that the particle moves to the next cell, and probability $1 - P_{NC}$ that it undergoes a collision within the current cell. Should a collision be sampled, then the optical depth at which the collision will occur may be sampled as

$$\tau(d) = -\ln[1 - (1 - P_{NC})\xi] = \int_0^d \Sigma_t(s\hat{\Omega} + \vec{r}_0, E)ds \quad (9)$$

and the exact value of d may be found using an iterative algorithm such as Newton's method, or the bisection method [17]. Direct sampling has been shown to be a viable method, producing the same results as substepping or delta tracking [20].

The accuracy of this method however hinges on the ability to obtain very accurate numerical integrations of the cross section, a requirement which may be difficult to meet in many cases. Notwithstanding, for cases where the cross section is easily represented analytically and direct integration is feasible and readily conducted, direct sampling provides an efficient and exact technique for sampling the flight distance of a spatially-varying cross section.

2.4 New (and old) delta tracking variants

2.4.1 Regional delta tracking

Methods to overcome what is known in the nuclear engineering field as the local heavy absorber problem have been developed in the field of computer graphics, where delta tracking is sometimes used in the rendering of high-fidelity images. Yue et al. used kd-trees to divide media of interest into different subdomains, each with their own majorant extinction coefficient (analogous to the total cross section) [21]. Another implementation which used supervoxels was proposed by Szirmay-Kalos et al. [22]. Both of these works demonstrated speed improvements with their algorithms, compared to standard delta tracking implementations. These methods work in the same manner as regular delta tracking, except that the system geometry is divided into arbitrary regions, each having its own majorant cross section. In doing so, areas of the geometry which have a high cross section relative to other areas can be separated, preventing their high cross sections from increasing the majorant globally, decreasing efficiency by increasing the probability of virtual collisions in the other parts of the geometry. More recently, Guo and Chen have used a similar concept to improve the delta tracking method in reactor transport problems [23]. In their paper, the method is applied to three different whole-core reactor systems, and each one was simulated faster using their multi-region technique.

Recently, Lemaire et al. have investigated the optimization of algorithms which represent piecewise deterministic Markov processes, a class of simulations to which delta tracking belongs. Their work examines three different methods for determining the majorant cross section for use in flight distance sampling [14]. The first and least efficient is the use of the global majorant, as we have previously outlined for the traditional delta tracking method. Second is the use of a local majorant, similar to this regional delta tracking variant. The local majorant demonstrated large improvements in efficiency compared to use of the global majorant [14]. Lastly, a third option is outlined, where the majorant is calculated by taking the majorant cross section along the flight path of the particle, and is the most efficient of the three methodologies [14], but is not examined in this work. This method essentially reduced the problem to a different one-dimensional system for each flight, but, as outlined in Sec. 3, the system that we used to benchmark these methods is already a one-dimensional rod geometry. In addition, the work of Lemaire et al. does not consider the

added computational cost that would be required to calculate all surface intersections along the flight path for each flight. Therefore, while this method would reduce the number of virtual collisions which occur, it is not necessarily the case that the computational time would be reduced; this method should be further investigated in future work to determine whether it could pose an advantage over other delta tracking variants.

2.4.2 Negative weighted delta tracking

A method proposed by Legrady et al. provides an option that is far more general than delta tracking, and does not require the sampling cross section to be a strict majorant [24]. This method is based on two free parameters: a sampling cross section, $\Sigma_{smp}(\vec{r}, E, \hat{\Omega}) > 0$, and a probability of having a real collision, $q(\vec{r}, E, \hat{\Omega}) \in]0, 1[$. Both of these parameters may be taken to be arbitrary functions of the phase space coordinates. For each flight, the distance is sampled using Eq. (7), replacing $\Sigma_{maj}(E)$ with $\Sigma_{smp}(\vec{r}, E, \hat{\Omega})$. Once the particle is moved to the tentative collision site, a real collision is performed with probability q , and a virtual collision with probability $1 - q$. A fair game is achieved by altering the weight of the particle (w) at both real and virtual collisions, as outlined in Algorithm 3 [25].

Algorithm 3: Negative weighted delta tracking procedure.

```

1 Sample uniform random variable  $\xi_1$ ;
2  $d := -\ln(\xi_1)/\Sigma_{smp}(\vec{r}_0)$ ;
3  $\vec{r}_1 := d\hat{\Omega} + \vec{r}_0$ ;
4 Sample random variable  $\xi_2$ ;
5 if  $\xi_2 < q$  then
6    $w := w \frac{\Sigma_t(\vec{r}_1, E)}{q \Sigma_{smp}(\vec{r}_0)}$ ;
7   Perform real collision;
8   Tally real collision estimator;
9   Tally total collision estimator;
10 else
11    $w := w \frac{1 - \frac{\Sigma_t(\vec{r}_1, E)}{\Sigma_{smp}(\vec{r}_0)}}{1 - q}$ ;
12   Virtual collision; goto line 1;
13   Tally total collision estimator;
14 end
    
```

Upon examination of Algorithm 3, it is evident that the sign of the particles weight will change when a virtual collision is sampled, and the cross section is smaller than the sampling cross section. In spite of the appearance of negative weights, which usually lead to inflated variances for the target observables, Legrady [24] and Molnár [25] showed that it is possible for this method to perform just as well, if not better than standard delta tracking, provided that Σ_{smp} and q are properly chosen for the problem at hand. The choice of these parameters is non-trivial, and highly dependent on the geometric and material properties of the problem to be examined. Also, the best choice of Σ_{smp} and q in one portion of the geometry may not be a good choice in another portion [24].

2.4.3 Carter, Cashwell, and Taylor tracking

An older, and little discussed variant of delta tracking is that proposed by Carter, Cashwell, and Taylor in 1972, which we shall simply refer to as CCT tracking (or CCTT) in this work [26]. This method also allows for a flexible sampling cross section, which need not be a strict majorant, and which can be cast as a function of the phase space coordinates. Contrary to negative weighted delta tracking, no other free parameter is introduced. A key element of this method (outlined in Algorithm 4), is that the magnitude of the particle weight increases every time a (real or virtual) collision is attempted at a position where the sampling cross section underestimates the actual total cross section. Additionally, on virtual collisions, the sign of the weight is changed in conjunction with its magnitude.

It should be noted that, if the sampling cross section Σ_{smp} is a majorant, the delta tracking algorithm is recovered, preventing the sign of the particle from changing, or the magnitude of the weight from increasing. In regions of the phase space where Σ_{smp} is not a majorant, however, the factor F (which is always greater than unity, as defined in Algorithm 4), multiplies the particle weight. The algorithm has the property that, if the sampling cross section overestimates the total cross section by a small amount, then the weight increase due to the factor F is only marginally

Algorithm 4: CCT Tracking procedure.

```

1 Sample uniform random variable  $\xi_1$ ;
2  $d := -\ln(\xi_1)/\Sigma_{smp}$ ;
3  $\vec{r}_1 := d\hat{\Omega} + \vec{r}_0$ ;
4 if  $\Sigma_t(\vec{r}_1) < \Sigma_{smp}(\vec{r}_0)$  then
5 |   Use delta tracking algorithm;
6 else
7 |   Tally total collision estimator;
8 |    $D := \frac{\Sigma_t(\vec{r}_1)}{2\Sigma_t(\vec{r}_1) - \Sigma_{smp}(\vec{r}_0)}$ ;
9 |    $F := 2\frac{\Sigma_t(\vec{r}_1)}{\Sigma_{smp}(\vec{r}_0)} - 1$ ;
10 |   Sample random variable  $\xi_2$ ;
11 |   if  $\xi_2 < D$  then
12 | |    $w := w \cdot F$ ;
13 | |   Perform real collision;
14 | |   Tally real collision estimator;
15 |   else
16 | |    $w := -w \cdot F$ ;
17 | |   Virtual collision; goto line 1;
18 |   end
19 end

```

greater than unity, with F approaching unity as Σ_{smp} approaches Σ_t from below. This is a highly desirable feature, as it guarantees that small underestimations of the actual total cross section will result in small increases in the variance.

At first glance, this method appears to be rather different from the previously outlined negative weighted delta tracking. Despite this, and the fact that CCTT was developed much earlier, it is in fact a special case of negative weighted delta tracking. Molnar et al. have pointed out that the selection of $q = \Sigma_t/(\Sigma_t + |\Sigma_{smp} - \Sigma_t|)$ in negative weighted delta tracking yields exactly the algorithm for CCTT [25]. With this choice of q , CCTT and NWDT reduce to the same method if they employ the same sampling cross section. Compared to the general case of negative weighted delta tracking, where q is an independent variable, CCTT is easier to use, as there is only one independent variable which must be determined (Σ_{smp}). Determination of Σ_{smp} is easier in CCTT as well, as it is known that the closer the sampling cross section is to being the true majorant, the fewer negative particles will be born. Conversely, there are no straightforward criteria to determine Σ_{smp} and q in negative weighted delta tracking for any given system.

3 Benchmarking in a 1D System

In order to test these different transport methods in the context of continuous media, either due to density or temperature variations, a series of one-dimensional rod systems are proposed. All of these systems have a length of 2 cm ($x \in [0, 2]$), with vacuum boundary conditions at either end. Particles can only move in the forward or the backward direction; scattering is isotropic (forwards or backwards with equal probability), the systems are non-multiplying, and the one-speed approximation is used. For simplicity, only the total cross section is allowed to vary spatially, while the probabilities for scattering and absorption remain constant at all locations in the rod, with values of 0.7 and 0.3 respectively. For these fixed-source problems, all particles begin at $x = 0$, traveling into the rod. The total cross section was allowed to vary spatially according to several different functional forms, which are provided in Table 1 and are inspired (but different) from those used by Brown and Martin [17]. Care is taken to ensure that the cross sections are all positive over the system domain. For each transport method and cross section profile combination, 10^3 batches of 10^5 particles were used to generate the reported data. The C++ source code for the program used to carry out the simulations described in this section, as well as the numeric results, have been made publicly available under the CeCILL v2.1 license [27].

3.1 Examined Transport Methods

The series of systems previously described are solved using several different transport methods, based on the theory detailed in Sec. 2. To provide a baseline, the substepping method (SS) is used, assuming a constant cross section over

Table 1: Functional forms of the total cross section (in units of cm^{-1}) as a function of position x (in units of cm, $x \in [0, 2]$), in the rod.

Linearly Increasing (LI)	$\Sigma_t(x) = x$
Linearly Decreasing (LD)	$\Sigma_t(x) = 2 - x$
Exponentially Increasing (EI)	$\Sigma_t(x) = 0.1e^{2x}$
Exponentially Decreasing (ED)	$\Sigma_t(x) = e^{-3x}$
Sharp Gaussian (SG)	$\Sigma_t(x) = \sqrt{\frac{2}{\pi}} \exp\left[-\left(\frac{x - 1.23}{0.05}\right)^2\right] + 0.1$
Broad Gaussian (BG)	$\Sigma_t(x) = \sqrt{\frac{2}{\pi}} \exp\left[-(x - 1.23)^2\right] + 0.1$

each sub-interval (using the mid-point of the region). One hundred sub-intervals were used for all problems, providing a reasonable approximation for most of the proposed systems.

Direct sampling (DS), as described by Brown and Martin, is used, with the slight alteration of replacing Newton’s method with the bisection method. While it is claimed that Newton’s method could not fail in the particular case of cross sections which are guaranteed to be positive, numerical difficulties were encountered when the value of the cross section was close to zero, causing the method to fail¹. As such, the bisection method was used due to its simplicity, and the fact that it can never fail, despite it being somewhat less efficient.

Standard delta tracking (DT) was examined using the “best case scenario” of taking the maximum cross section for the system over the given domain as the majorant. In practice, this is rarely the case and usually the value used as the majorant is somewhat greater than the maximum cross-section value, leading to extra virtual collisions. Regional delta tracking (RDT) was applied by dividing the problem domain into 5 equi-spaced bins, and the majorant cross section in each region was taken to be the maximum value of the cross section over that portion of the domain. When a particle moves from one bin to another, it is stopped at the boundary where the majorant is updated for the new region, and a new flight distance is sampled. This is required to make sure the majorant used to sample the flight distance is never smaller than the true majorant in the region.

The performance of negative weighted delta tracking (NWDT) is dependent on the value of Σ_{smp} and the probability of a real collision (q), which are both free parameters chosen to have any reasonable value ($0 < q < 1$ and $\Sigma_{smp} > 0$). The optimal choice of these parameters is highly dependent on the problem geometry and material properties. The values of $q = 0.3$ and $p = \Sigma_{smp}/\Sigma_{maj} = 0.85$ have been chosen for these systems, somewhat arbitrarily. These values seem to perform reasonably well across all the examined cross section profiles. Further discussion as to the determination of the optimal values for use with this method is outside the scope of this work. Implementing the regional variation of negative weighted delta tracking (RNWDT) was done in a similar way to the traditional delta tracking, with the sampling cross section in each bin also being 85% of the maximum value of the cross section within the bin. Like with RDT, a particle is stopped at the boundary of a region, where the sampling cross section is updated and a new flight distance is sampled. This is not strictly necessary for the algorithm, but was chosen as it better matches the procedure required in RDT.

Carter, Cashwell, and Taylor Tracking (CCTT) was performed using the same ratio $\Sigma_{smp}/\Sigma_{maj} = 0.85$ as the negative weighted delta tracking methods, and likewise for its regional variant (RCCTT). RCCTT also stops particles at bin boundaries to update the sampling cross section and flight distance. A second variant of Regional Carter Tracking was also implemented, however, where the initial value of the sampling cross section is taken to be 85% of the maximum, but is progressively updated as locations are found where the total cross section is greater than the sampling cross section. This method is referred to as Improving Regional Carter Tracking (IRCCTT).

The fiducial quantities which shall be examined with these different transport methods are the leakage rate, the integral collision rate (average number of collisions per particle), and the collision density. In transport methods that have virtual collisions, such as delta tracking and its related methods, two estimators can be used to calculate the collision density and number of collisions per particle: one estimator which scores only at real collisions, and another which scores at both real and virtual collisions, previously described by Leppänen [28]. These shall be referred to as the real collision estimator, and the total collision estimator for the duration of this work. The quantities which were tallied for each family of tracking methods are presented in Table 2. Standard variance reduction techniques such as implicit capture and Russian Roulette are used for all methods. These techniques are applied to the negative particles by using the absolute value of their weight in calculations. The cutoff weight for roulette is $|w| = 0.6$, while the survival weight is 1 for positive particles, and -1 for negative particles. Several of the tested methods also provide

¹ It is well known that Newton’s method has the tendency to fail when the first derivative of the function of interest is close to zero, causing the method to diverge due to overcorrections to the estimate.

Table 2: Quantities which are tallied for the real and total collision estimators of the collision density for each family of transport method. These values are then normalized by the volume of the tally bin when recorded.

	Real	Total
SS / DS	w	-
DT	w	$\frac{w\Sigma_t(\vec{r})}{\Sigma_{maj}}$
NWDT	$\frac{w\Sigma_t(\vec{r})}{q\Sigma_{maj}}$	$\frac{w\Sigma_t(\vec{r})}{\Sigma_{smp}}$
CCTT	w	$\frac{w\Sigma_t(\vec{r})}{\Sigma_{smp}}$

the possibility of a particle weight being increased in magnitude. Where this is the case (NWDT- and CCT-based methods), particle splitting is also implemented: particles are split once they obtain a weight magnitude of 2.0 or greater.

3.2 Evaluation of Methods

Well known in the transport community is the Figure of Merit (FOM), a quantity used to compare the efficiency of different codes. One accepted definition is

$$FOM = \frac{1}{TR^2} \quad (10)$$

where T is the wall-clock time required to perform the simulation, and R is the relative standard error of the estimated quantity; in this paper, the standard error was estimated using independent replicas.

The problem with using this definition for a simple model such as the one proposed in this work is that the wall-clock time of our model is not necessarily representative of the behaviour of the tracking algorithm. In a real, continuous-energy MC calculation, about 80% of the time is spent looking up the values of cross sections [29,30]. (Conversely, in our simple model, the calculation time is overwhelmingly dominated by the pseudo random number generator). Given this fact, the amount of time required to perform a real simulation is highly correlated to the number of cross section searches required by a method. In an effort to obtain evaluations more representative of how methods might perform relative to one another in a production-level, continuous-energy MC transport code, the time parameter T in Eq. (10) has been replaced with the total number of times the actual total cross section has been looked up, which we will denote by N_{xs} .

For the direct sampling and substepping methods, the calculation of N_{xs} was slightly more nuanced. Direct sampling not only requires performing a search for the cross section, but must also integrate it along the line of flight. As the time to complete such an operation is highly implementation-dependent, it was chosen to count each integration of the cross section as only one cross section search, therefore incrementing N_{xs} by one. This choice is somewhat arbitrary and it is likely to yield unrealistically favorable evaluations for the method; as we shall see in the following, this choice actually has very little importance. Substepping is also peculiar, as each region has a constant cross section. As such, each time a particle moves from one region to another, and a new cross section must be sought for the region, N_{xs} is incremented by one.

4 Results and discussion

4.1 Integral collision rate and leakage

Part of the results of our calculations are presented in Table 3. For the sake of conciseness, we limit the content of this table to the Exponentially Increasing and the Broad Gaussian cross-section shapes (see Table 1), although our analysis is based on all the calculation results.

One of the first observations is that the RDT method always outperformed the traditional DT method, regardless of the cross section or the estimator being used. The extent to which the regional variant performed better than the standard method was slightly dependent on the form of the cross section being examined. RDT showed greater improvement for cross sections which were monotonically increasing, than for those which were monotonically decreasing. This can be seen in the exponential cross sections where, for exponentially increasing, RDT outperformed DT by a factor of more than 3.5 for the real collision estimator. For the exponentially decreasing cross section, the ratio of their respective FOMs was 2.9.

CCTT and RCCTT exhibit a behavior quite similar to that seen between DT and RDT. RCCTT always outperforms CCTT, though the difference between increasing and decreasing cross sections was not observed. This is likely due to the sampling cross section for CCTT and RCCTT being only 85% of the maximum cross section, either for the entire problem, or for the region in which a particle finds itself. In this case, at the location of the maximum, there is a possibility of the particle changing sign, and the magnitude of the particle weight will increase. The comparison of CCTT vs DT and RCCTT vs RDT is more difficult, however. There were some cases where CCTT outperformed DT, and others where DT outperformed CCTT. The same is true for the relationship between RCCTT and RDT. Taking the Broad Gaussian cross section as an example that might be more typical of many real world reactor problems, where particles see both an increase and a decrease in the cross section, it was observed that DT and RDT performed marginally better than CCTT and RCCTT. These values can be seen in Table 3. Interestingly, the IRCCTT method always performed just as well as the RDT method, and sometimes slightly better. As the simulation runs, IRCCTT decreases the probability of a particle having a collision in a location where the sampling cross section under-estimates the maximum, progressively reducing the probability of a collision occurring where the particles sign would change, and weight magnitude would increase, therefore pushing the method to behave more like RDT.

One striking difference between the CCTT family and the NWDT family is the difference in the amount of particle splitting which occurred in each method. NWDT could sometimes have to transport as many as 30% extra particles, as particle weights would increase drastically, causing high amounts of splitting. For the Broad Gaussian cross section, NWDT transported 30.1% extra particles, while CCTT only transported 0.0013% extra particles, a marginal increase over the 10^8 particles initially prescribed for each simulation. Despite this large number of extra particles, NWDT does not perform quite as many additional cross section evaluations. With the Broad Gaussian, NWDT performed 1.72 cross section look-ups per history, compared to 1.66 for DT. The near factor of 10 between their FOMs is then accounted for by the higher variance for NWDT. A higher variance is not surprising, as the large number of extra particles indicates large changes in the magnitude of particle weights.

RNWDT always outperformed NWDT as seen with DT and RDT as well as CCTT and RCCTT. NWDT and RNWDT exhibited poorer performance for all for all test cases compared to DT, RDT, CCTT, and RCCTT. DS typically had a FOM an order of magnitude lower than that of NWDT, and performed similarly to SS.

4.2 Collision density

The FOM for the collision density of four selected cross sections are provided in Figure 4. Several observations can be made with the FOM of the collision density which seem rather contradictory to the previously discussed results for integral collision rates. One such example in Figure 4a is that NWDT has a substantially higher FOM from $x = 0$ up to $x \approx 1.25$ compared to DT and CCTT. From $x = 0$ to $x \approx 0.5$, NWDT even performs better than the meshed variants. Figures 4b and 4c also have substantial portions of the problem domain where NWDT performed better than DT and CCTT when using the real collision estimator. However, in the regions having the largest number of collisions, DT and CCTT do perform better than NWDT. Our choice of the real collision probability, $q = 0.3$, leads to a larger number of real collisions in NWDT (compared to DT) in regions where the majorant cross section is much larger than total cross section. A larger number of real collisions in these regions will of course increase the FOM for the real estimator in that location, explaining why NWDT is able to outperform DT and CCTT in certain regions. This lack of real collisions in optically thin regions is congruous to the known inefficiencies in delta tracking, due to the exclusive use of the real collision estimator, and is recognized as one of the main disadvantages of the method [15].

Saw-tooth features are also observable in the FOM for the RNWDT and RCCTT methods, particularly in Figure 4a for the exponentially increasing cross section. These features always occur near the boundary of two regions, where the sampling cross section changes. For RCCTT, a dip in the FOM, observed near $x \simeq 1.6$ cm, is caused by the particles with negative weights being generated in this region; indeed, at the location of the dip, the sampling cross section underestimates the real cross section. The negative weights of course increase the variance in that region, causing this dip. This explanation is also the reason behind the teeth in the RNWDT FOM. These teeth are also present the other cross sections, but sometimes appear to be decreasing, as seen with the exponentially decreasing cross section in Figure 4b. For a monotonically decreasing cross section, negative particles are produced at the left side of the bin, decreasing the FOM. This is also why the teeth change from increasing to decreasing for the Gaussian cross sections.

In addition, despite the fact that the FOM of the RNWDT method for the integral collision rate is 2.7 times smaller than that of DT, RNWDT performs just as well as both DT and CCTT in estimating the spatial collision density. This surprising observation will be discussed further in Sec. 4.4.

Presented in Figure 5 are the FOM for the collision density calculated with the total collision estimator for the same cross sections. The main difference with the real collision estimator (Fig. 4) is the presence of saw-teeth in all of the regional methods. When moving from one region to another, the sampling cross section changes, which in turn alters the number of virtual collisions which will occur. For methods such as RCCTT, IRCCTT, and RDT, this leads to an increase in the FOM for the exponentially increasing cross section, because of the increased probability of a virtual collision, as seen in Figure 5a. NWDT also has a performance much more comparable to DT and CCTT

Table 3: Tabulated results of the average number of collisions per particle for the exponentially increasing and broad Gaussian cross sections. For monotonically increasing cross sections, the total collision estimators for delta tracking and Carter Tracking perform worse than the real collision estimators. This behavior is not seen in monotonically decreasing and non-monotonic cross sections.

Exponentially Increasing			
	Estimator	Avg. Collisions / Part.	FOM
SS	Real	1.701157 ± 0.000110	2.262×10^{-2}
	Total	-	-
DS	Real	1.701195 ± 0.000108	2.760×10^{-2}
	Total	-	-
DT	Real	1.701328 ± 0.000110	2.098×10^{-1}
	Total	1.701260 ± 0.000125	1.623×10^{-1}
RDT	Real	1.700965 ± 0.000114	7.745×10^{-1}
	Total	1.701021 ± 0.000113	7.762×10^{-1}
NWDT	Real	1.701128 ± 0.000321	3.657×10^{-2}
	Total	1.700933 ± 0.000285	4.632×10^{-2}
RNWDT	Real	1.701366 ± 0.000334	7.669×10^{-2}
	Total	1.701229 ± 0.000182	2.579×10^{-1}
CCTT	Real	1.701079 ± 0.000113	2.321×10^{-1}
	Total	1.701017 ± 0.000127	1.846×10^{-1}
RCCTT	Real	1.701154 ± 0.000127	7.006×10^{-1}
	Total	1.701237 ± 0.000121	7.742×10^{-1}
Broad Gaussian			
	Estimator	Avg. Collisions / Part.	FOM
SS	Real	1.088740 ± 0.000087	1.708×10^{-2}
	Total	-	-
DS	Real	1.088684 ± 0.000085	2.903×10^{-2}
	Total	-	-
DT	Real	1.088836 ± 0.000085	9.964×10^{-1}
	Total	1.088815 ± 0.000078	1.175×10^0
RDT	Real	1.088762 ± 0.000088	1.117×10^0
	Total	1.088732 ± 0.000086	1.171×10^0
NWDT	Real	1.088649 ± 0.000256	1.053×10^{-1}
	Total	1.088626 ± 0.000126	4.344×10^{-1}
RNWDT	Real	1.088408 ± 0.000273	1.022×10^{-1}
	Total	1.088612 ± 0.000131	4.424×10^{-1}
CCTT	Real	1.088732 ± 0.000108	6.875×10^{-1}
	Total	1.088766 ± 0.000096	8.763×10^{-1}
RCCTT	Real	1.088668 ± 0.000114	7.281×10^{-1}
	Total	1.088689 ± 0.000100	9.434×10^{-1}

through the entire problem geometry when using the total collision estimator. This is again in contradiction with the behavior observed with the integral collision rate estimated with the total collision estimator, as NWDT always had a worse FOM than DT and CCTT.

Both the SS and DS methods demonstrated much worse performance than the other methods, regardless of the cross section examined. The FOM for these methods has been left out of the plots in Figures 4 and 5, as values were indistinguishable from zero, given the scale imposed by the FOM for the other methods. It should be noted that our criteria for determining the FOM for these two methods was somewhat generous. For SS, several simulations would need to be done to ensure that enough steps are present to obtain an accurate result. Our reported values assume that the number of required steps is known, despite not having any knowledge of the problem solution. In the case of DS, an integration of the cross section was considered to be equivalent to the time required for one cross section look-up. Indeed, if the density of a material or its isotopic concentration is represented as a polynomial, with a constant material temperature, it is quite likely that the integration to determine P_{NC} from Eq. (8) can be carried out quite

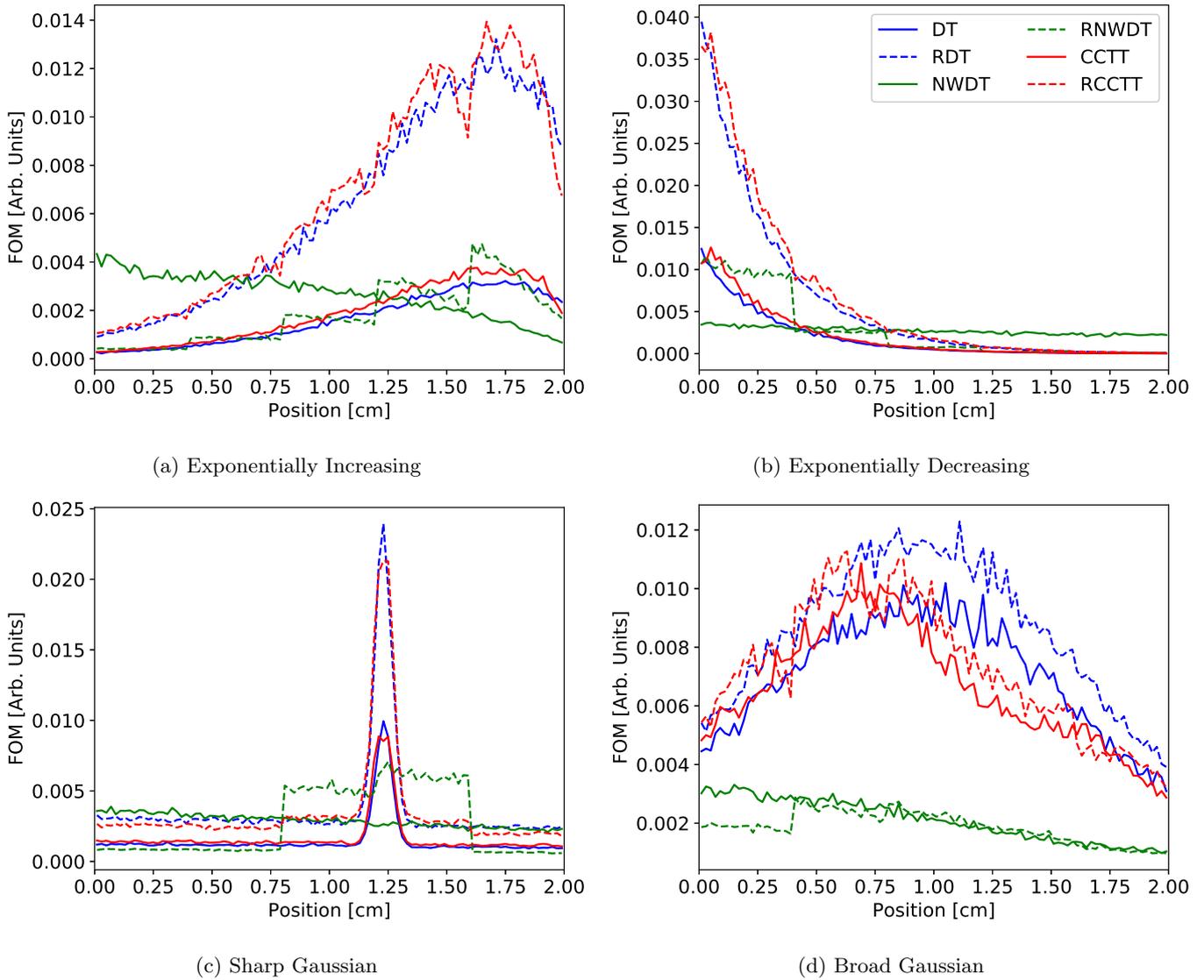


Fig. 4: Figure of Merit throughout the rod, using the real collision estimator, for the selected cross section profiles.

fast and efficiently. A continuous representation of the temperature (be it by polynomial or some other method) is likely to greatly hinder the performance of DS however. This is because there is no linear relationship between the temperature and the cross section. As such, many small steps along the flight path would need to be made; at each step the temperature would be evaluated, and the cross section at that location would then be Doppler broadened, in an effort to approximate the value of P_{NC} . Depending on the temperature profile, many small steps might need to be taken, and many evaluations the total cross section may need to be carried out, in order to obtain an accurate estimate of P_{NC} . It is arguably conceivable to transform a polynomial representation of the temperature field into a polynomial representation of the spatial dependence of the macroscopic cross section, which could make DS be truly competitive. However, such a transformation would likely be difficult to construct and expensive to update on the fly, during the course of a multi-physics simulation.

4.3 Regional splitting effects on FOM

From Figure 6, for the three examined regional tracking methods (RDT, RNWDT, RCCTT) it is always observed that increasing the number of regional divisions in the system leads to a decrease in the number of true cross section look-ups required to perform the simulation. This is due to the fact that, within a region, the initial flight distance is determined with a sampling cross section or majorant cross section which is specific to only that region. If the sampled

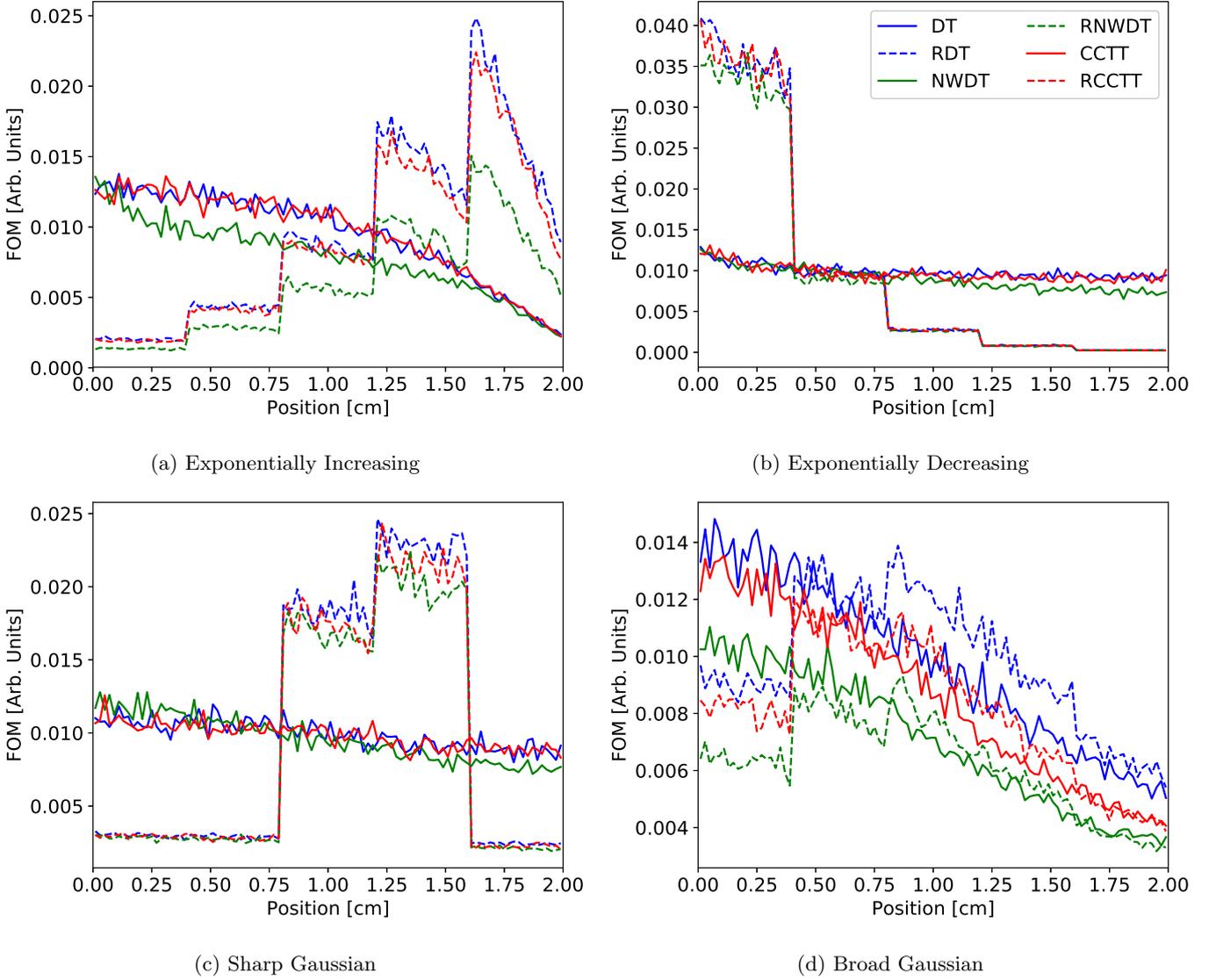


Fig. 5: Figure of Merit throughout the rod, using the total collision estimator, for the selected cross section profiles.

distance then places the particle in a new region, the particle is moved to the next region, and begins sampling a new distance with the sampling/majorant cross section of that region. This provides a fair way to transport particles while minimizing look-ups of the true cross section. These findings are in agreement with the results previously obtained by Guo and Chen [23].

In Figure 6, it is also apparent that there are diminishing returns on increasing the number of regions in a system. For all three methods, at least 85% of the total possible decrease to the number of cross section evaluations had been reached once six regions were used. Therefore, adding more than six regions for this system would not be beneficial. Such a response is important to note, as regionalization in a 3D fully continuous transport code with an analogous method such as voxelization, with each voxel needing to contain a set of sampling/majorant cross sections, could quickly increase the memory footprint of such a method. Such a method however, has recently been used in the GUARDYAN Monte Carlo neutron transport code [31]. Their voxelization technique not only divides the spatial domain into bins, but the energy domain as well. With this implementation, impressive speedup factors of up to 34–35 were obtained under certain conditions. One might also envision a modified version of these regional methods that treat each cell in the geometry as its own region, requiring no more resources than already required for current transport methods. Such an implementation has already been used in Serpent with nonuniform densities, and demonstrated very reasonable performance [16].

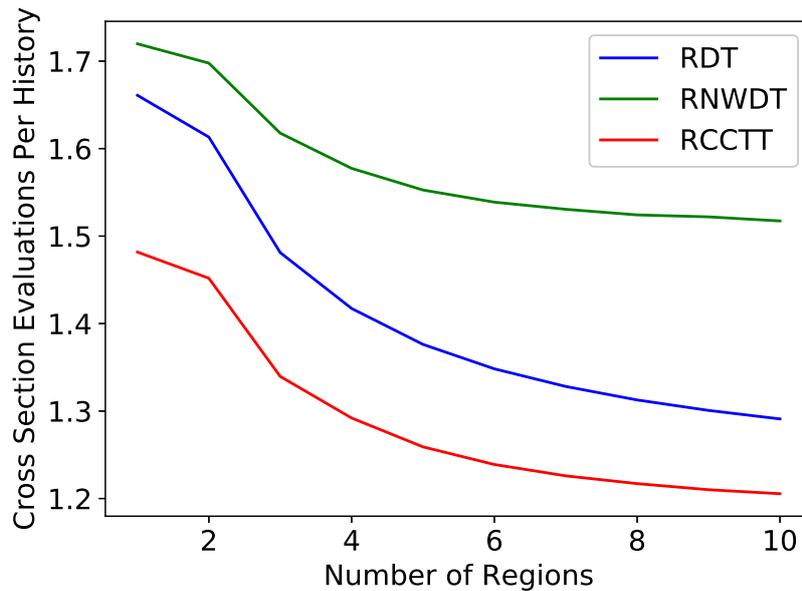


Fig. 6: The number of true cross section evaluations performed per particle. All batches were simulated in the Broad Gaussian system.

While both of these techniques seem promising, more work must be done to compare these two methods. Specifically, comparisons must be made in regards to the memory footprint of each method, and potential speedup which one might expect to gain, and the overall ease of implementation and use.

4.4 Collision estimators

Two different collision estimators were evaluated for each method (excluding substepping and direct sampling). The first is the standard collision estimator, which only scores at real collisions. The second is the total estimator, which scores at both real and virtual collisions. Table 3 presents a subset of the data for the average number of collisions per particle. These samples have been selected as they demonstrate the differences in performance between collision estimators, transport methods, and material properties.

Under the Exponentially Increasing cross section, delta tracking performed measurably better with the real collision estimator than with the total collision estimator, having a FOM which was 29% greater. Looking at delta tracking performance under the Broad Gaussian, the total collision estimator then performed better by 18%. This is very different from the regional delta tracking variant, which had a very similar performance for the two estimators under both cross section profiles. The lack of a discrepancy between the two estimators with this tracking method is due to each region having its own local majorant. Because the local majorant is usually much closer to the minimum value of the cross section in the region, the likelihood of a virtual collision is smaller than if the global majorant had been used. With fewer virtual collisions, the estimator behaves similarly to the real estimator, leading to a similar performance.

Conversely, negative weighted delta tracking performed consistently better when using the total estimator, as compared to using the real estimator. This is also attributed to the mechanics of the method, as the probability of a collision being virtual is constant, and not affected by the shape of the cross section along the particles path. Because of this, the total collision estimator would be expected to always outperform the real collision estimator, as more scores are provided when also counting virtual collisions.

Passing from the integrated collision rate to the spatial collision densities in Figures 4 and 5, different conclusions can be drawn. Figure 4a presents the FOM as a function of position for the Exponentially Increasing cross section, using the real collision estimator, and Figure 5a likewise but for the total estimator. Looking at the curves for delta tracking, it can be seen that the FOM never exceeds 0.004 within the problem domain while using the real estimator, but the FOM for the total collision estimator remains above 0.01 from $x = 0$ cm up to $x = 1.0$ cm, and then ends just below 0.005 at $x = 2.0$ cm. The total collision estimator is clearly much more efficient at determining the spatial collision density than the real collision estimator for this problem. This is the opposite of what was observed for the integrated collision tally, where the real collision estimator was generally less efficient. Further complicating this is the fact that for the Broad Gaussian cross section, presented in Figures 4d and 5d, the total collision estimator does

indeed perform better than the real collision estimator. Negative weighted delta tracking performed better with the total collision estimator in both cases, as seen in Table 3. These observations would suggest that the best estimator is dependent on the transport method, but more interestingly, such as with delta tracking, the estimator that performs better at tallying the integral reaction rate may perform worse at tallying the spatial reaction rate. One could choose to have more knowledge of the spatial reaction rate, at the expense of having less knowledge about the magnitude of the reaction rate, or vice-versa. A possible explanation of this finding is that the virtual collision estimator has a high covariance, as particles experiencing several virtual collisions could score in several bins, without changing direction or energy. The integrated collision rate (f) is the sum of the collision rates in each bin (f_i), which make up the collision density:

$$f = \sum_i f_i. \quad (11)$$

From general probability theory, the variance of quantity f can be described as:

$$\sigma_f^2 = \sum_i \sigma_i^2 + \sum_i \sum_{j, j \neq i} \rho_{ij} \sigma_i \sigma_j, \quad (12)$$

where σ_i^2 is the variance of f_i , and ρ_{ij} is the correlation coefficient between the f_i and f_j quantities. By increasing the correlation between bins, the correlation coefficient would increase. From Eq. (12), it is observed that, for given values of σ_i , the variance of the integrated collision rate increases with an increase of the correlation coefficients.

4.5 Discussion

One important result of these comparisons is that no method can be considered better than another for all, or even most cases when comparing FOM alone. It is quite evident that, for a given problem, one transport method might perform much better at calculating the integral collision rate, while that same method is the worst at calculating the spatial collision density for that same problem. Furthermore, the appropriateness of each method for a given case also changes with the geometry of the problem, and the material properties. Due to these factors, it is impossible to single out just one method which is dominant. From our observations, it does seem evident that substepping and direct sampling are the methods which provided the worst performance, and were consistently much less efficient than the other examined methods. Similarly contradicting results were observed regarding the use of the real collision estimator, or the total collision estimator. Neither proved to consistently have superior performance.

The implementation of regionalization for delta tracking, negative weighted delta tracking, and Carter Cashwell and Taylor tracking did lead to an increase in performance for all methods when looking at the spatial collision density, and integral collision rate tally. There is clearly the possibility to obtain gains in performance by regionalizing the problem geometry. Depending on how regionalization is implemented, however, this could potentially lead to difficulties with how to split a system, and extra memory requirements, should a large number of regions be desired. Further work on this subject is required to determine whether a method such as that used in computer graphics with voxelization would be better or worse than an approach using the material cells such as already examined by Leppänen [16].

5 Conclusions

The field of Monte Carlo simulation for neutron transport is quickly moving in the direction of attempting to solve coupled multi-physics problems. This paper has reviewed several methods that solve the problem of efficiently sampling the flight kernel in cases where the total macroscopic cross section is not piecewise constant, but varies spatially in a continuous manner. Numerical simulations have been carried out for a simple, yet significant, one-dimensional transport benchmark.

The three criteria initially considered in this paper for any new transport method were performance, robustness, and simplicity. Should the authors have to rank the examined methods based on these criteria, they would assign direct sampling with the lowest mark, due to the complexity required to compute the numeric integration under certain circumstances. Next would be the negative weighted delta tracking, and regional negative weighted delta tracking methods. While these algorithms do pose solutions to the problem of determining the true majorant cross section in continuous media, the authors are unaware of any straightforward criteria for choosing the real collision probability and sampling cross section for a problem. Also, despite the number of required cross section evaluations being similar to that of delta tracking, the amount of particle splitting which was observed lead to a serious need for synchronization in parallel implementations, when particles need to be added to the bank. Tied for first are delta tracking and Carter, Cashwell, and Taylor tracking. Delta tracking is of course well proven, and quite simple to implement, though hindered by the requirement of knowing the majorant cross section, and by the well-known local heavy absorber problem. CCT

tracking is not much more complicated than delta tracking, and does not require the knowledge of the true majorant. It does require the use of negative weights, however, which could cause problems with the variance in certain situations (which have not been encountered in our simulations). Despite this fact, CCT tracking performed just as well as delta tracking, and therefore provides a possible path forward.

Future work must be conducted to further evaluate these methods. One aspect yet to be explored is the performance of these methods in eigenvalue problems, to be solved by power iteration. Multiplying systems could potentially pose unique considerations when using methods such as CCT or NWDT, which produce negative-weighted particles. More work should also be conducted to explore the method (not considered here) which uses the majorant along the flight path, as described by Lemaire et al. [14]. This method should be evaluated in a three-dimensional Monte Carlo code to assess the possible benefits and drawbacks of using the majorant cross section along the flight path, and to see the true effects on computation time when surface intersections must be calculated along the entire trajectory.

References

1. F. Damian, E. Brun, *Ann. Nucl. Energy* **82**, 203 (2015)
2. M. Faucher, D. Mancusi, A. Zoia, *Ann. Nucl. Energy* **120**, 74 (2018)
3. C. Castagna, E. Cervi, S. Lorenzi, A. Cammi, D. Chiesa, M. Sisti, M. Nastasi, E. Previtali, *Eur. Phys. J. Plus* **135**, 433 (2020)
4. M. Ellis, D. Gaston, B. Forget, K. Smith, *Nucl. Sci. Eng.* **185**, 184 (2017)
5. G. Yesilyurt, W.R. Martin, F.B. Brown, *Nucl. Sci. Eng.* **171**, 239 (2012)
6. T. Viitanen, J. Leppänen, *Nucl. Sci. Eng.* **171**, 165 (2012)
7. A. Meister, A. Santamarina, *The effective temperature for Doppler broadening of neutron resonances in UO_2* , in *Proceedings of the International Conference on Nuclear Data for Science and Technology, Long Island, USA* (1998), pp. 233–239
8. J.J. Duderstadt, L.J. Hamilton, *Nuclear Reactor Analysis* (John Wiley & Sons, Inc., New York, 1976)
9. D. Brown, M. Chadwick, R. Capote, A. Kahler, A. Trkov, M. Herman, A. Sonzogni, Y. Danon, A. Carlson, M. Dunn et al., *Nucl. Data Sheets* **148**, 1 (2018)
10. R. Macfarlane, D.W. Muir, R.M. Boicourt, A.C. Kahler, III, J.L. Conlin, *The NJOY Nuclear Data Processing System, Version 2016*, Los Alamos National Laboratory (2019), LA-UR-17-20093
11. T. Goorley, M. James, T. Booth, F. Brown, J. Bull, L.J. Cox, J. Durkee, J. Elson, M. Fensin, R.A. Forster et al., *Nucl. Technol.* **180**, 298 (2012)
12. E. Brun, F. Damian, C. Diop, E. Dumonteil, F. Hugot, C. Jouanne, Y. Lee, F. Malvagi, A. Mazzolo, O. Petit et al., *Ann. Nucl. Energy* **82**, 151 (2015)
13. E. Woodcock, T. Murphy, P. Hemmings, S. Longworth, Tech. Rep. ANL-7050, Argonne National Laboratory, USA (1965)
14. V. Lemaire, M. Thieullen, N. Thomas, *J. Sci. Comput.* **75**, 1776 (2018)
15. J. Leppänen, *Ann. Nucl. Energy* **37**, 715 (2010)
16. J. Leppänen, *Nucl. Sci. Eng.* **174**, 318 (2013)
17. F.B. Brown, W.R. Martin, *Direct Sampling of Monte Carlo Flight Paths in Media with Continuously Varying Cross-Sections*, in *ANS Mathematics & Computation Topical Meeting* (2003), Vol. 2
18. J. Leppänen, M. Pusa, T. Viitanen, V. Valtavirta, T. Kaltiaisenaho, *Ann. Nucl. Energy* **82**, 142 (2015)
19. I. Lux, L. Koblinger, *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations* (CRC Press, Boca Raton, 1991)
20. F.B. Brown, D. Griesheimer, W.R. Martin, *Continuously Varying Material Properties and Tallies for Monte Carlo Calculations*, in *PHYSOR 2004: The Physics of Fuel Cycles and Advanced Nuclear Systems: Global Developments* (2004)
21. Y. Yue, K. Iwasaki, B.Y. Chen, Y. Dobashi, T. Nishita, *ACM Transactions on Graphics* **29**, 1 (2010)
22. L. Szirmay-Kalos, B. Tóth, M. Magdics, *Computer Graphics Forum* **30**, 85 (2011)
23. Q. Guo, Z. Chen, *Sustain.* **10**, 2272 (2018)
24. D. Legrady, B. Molnar, M. Klausz, T. Major, *Ann. Nucl. Energy* **102**, 116 (2017)
25. B. Molnar, G. Tolnai, D. Legrady, *Nucl. Sci. Eng.* **190**, 1 (2018)
26. L.L. Carter, E.D. Cashwell, W.M. Taylor, *Nucl. Sci. Eng.* **48**, 403 (1972)
27. H. Belanger, *Continuous Media Monte Carlo Transporter* (2020), <https://doi.org/10.5281/zenodo.3898594>
28. J. Leppänen, *Ann. Nucl. Energy* **105**, 161 (2017)
29. Y. Wang, E. Brun, F. Malvagi, C. Calvin, *Procedia Comput. Sci.* **80**, 484 (2016)
30. J. Leppänen, *Ann. Nucl. Energy* **36**, 878 (2009)
31. B. Molnar, G. Tolnai, D. Legrady, *Ann. Nucl. Energy* **132**, 46 (2019)

5 - Conclusions for Particle Tracking Methods

We recall here our main findings concerning particle tracking in stochastic media and in continuously-varying media.

5.1 . Acceleration Techniques for Stochastic Geometries

From the work presented in Chapter 3, focusing on the use of different acceleration techniques for performing transport in Poisson tessellations, the conclusions depended on whether or not a cell search mesh was used in the problem. If a cell search mesh is not used or is not properly refined, then the following trends were observed:

- Surface tracking with a neighbor map is always better than delta tracking or delta tracking with a neighbor map.
- Neighbor maps do improve the performance of delta tracking, with the largest improvement observed for cases where $\Lambda \gg \lambda$.
- Storing the geometry kernel with banked particles improves performance for surface tracking, but provides little improvement to delta tracking.

If a cell search mesh is used and is properly refined, then it was observed that:

- Surface tracking is better when $\Lambda \gg \lambda$, while delta tracking is better when $\Lambda \ll \lambda$.
- Neighbor maps reduce the performance of delta tracking.
- Storing the geometry kernel with banked particles reduces the performance.

Future work should focus on the use of these acceleration techniques for the class of stochastic inclusions (in particular the prominent example of spherical inclusions), whose geometrical properties and connectivity structure are inherently different from those of random tessellations. For these types of media, neighbor maps do not generally help: when a particle moves between random spheres, it must first pass through the common background matrix. Each sphere therefore has only one neighbor, and the background matrix neighbors all spheres, so neighbor maps would provide no benefit. In this case, it is unknown whether surface tracking or delta tracking is more efficient.

5.2 . Transport in Spatially Continuous Media

In Chapter 4 we have compared the performance of delta tracking, negative-weighted delta tracking, and direct sampling for several single speed, 1D, fixed-source transport problems. We have also considered the effects of regionalization on delta tracking and negative-weighted delta tracking. In general, the negative-weighted delta tracking variant of Carter et al. performed very similarly to delta tracking, when using a sampling cross section which was 85% of the majorant value of the cross section. In the examined fixed-source problems, the simultaneous presence of positive and negative weights did not appear to greatly degrade performance; on the contrary, Carter et al.'s method demonstrated slightly better performance on certain spatial scores, depending on the cross section shape, the position in question, and the estimator. Performance would certainly decrease, however, if an even lower sampling cross section were used, as this would generate more negative particles, further increasing the variance in scores. The performance of the negative-weighted delta tracking variant proposed by Legrady et al. typically demonstrated poorer performance than the other two methods. For certain cross section shapes, such as the exponentially increasing cross section for example, Legrady et al.'s variant had much better performance on spatial scores

when compared to the other two methods for the majority of the problem domain. For other problems, such as the broad Gaussian, it demonstrated worse performance at all positions. It appears that the choice of the real collision probability q has a large effect on the efficiency of the method for a given problem, and that the optimal choice is highly problem-dependent. This makes the application of Legrady et al.'s variant of negative-weighted delta tracking in a general-purpose Monte Carlo code quite difficult, as the method appears to be much less robust than either delta tracking, or Carter et al.'s variant. Direct sampling demonstrated very poor performance compared to the three other methods, despite the generosity it was afforded in the calculation of its performance criteria. Dividing the problem domain into regions always improved the efficiency of the delta tracking and negative-weighted delta tracking algorithms, but needlessly increasing the number of regions demonstrated diminishing returns. These results indicate that either delta tracking or Carter et al.'s variation of negative-weighted delta tracking would be the best candidates for treating spatially continuous cross sections in a general-purpose Monte Carlo code, as both demonstrated excellent performance, and appeared to be rather robust in general.

We stress that our analysis conducted in Chapter 4 only considered fixed-source transport problems. In Part II, we will consider the use of delta tracking and Carter et al.'s method for eigenvalue problems solved by power iteration, in view of their significance to reactor physics applications. In Chapters 7 and 8, we will see that, while Carter et al.'s variant performs well for fixed-source problems, it leads to severe instabilities when applied to k -eigenvalue power iteration problems. We will propose some novel strategies based on weight cancellation methods, and demonstrate their potential to counter these issues through several numerical applications on benchmark configurations.

Part II

Weight Cancellation

6 - Negative-Weighted Delta Tracking with Weight Cancellation

In Part I of this manuscript, we have focused on the evaluation of existing transport methods which can treat spatially continuous cross sections. In Chapter 4, these methods were compared using one-dimensional single-speed fixed-source benchmark problems, with total macroscopic cross sections that exhibited different types of spatial behavior. From this study, we have determined that the best candidates for treating spatially continuous cross sections in a robust, general-purpose Monte Carlo code would be either delta tracking, or the negative-weighted delta tracking variant proposed by Carter et al. [1]. Since Chapter 4 only considered fixed-source transport problems, it is of utmost importance to evaluate the chosen particle-tracking methods in the context of k -eigenvalue power iteration, which is the most frequently used method for solving eigenvalue problems of interest in reactor physics. In Sec. 6.1, we will therefore address the modifications which must be made to the standard power iteration algorithm to facilitate the use of negative weights, and show that computational instabilities might arise due to their use. We will also outline why *weight cancellation*, a procedure by which the positive and negative statistical weights carried by the particles are “summed up” by exact or approximate methods, is a viable solution to these instabilities. Section 6.2 will first present an approximate cancellation technique, as a possible strategy to remedy the previously mentioned instabilities. An exact cancellation method will subsequently be developed in Sec. 6.3. While the majority of Part II will be devoted to the use of weight cancellation to solve the instabilities encountered while using negative-weighted delta tracking for power iteration, these findings have inspired other possible applications of weight cancellation for Monte Carlo simulations in the field of nuclear reactor physics, which will be then considered in Part III.

6.1 . Failure of Negative-Weighted Delta Tracking with Power Iteration

The delta tracking method is well-known in the Monte Carlo community, and part of its success stems from the fact that it is routinely used in the Serpent Monte Carlo code, which is primarily intended for reactor physics applications [2–4]. Delta tracking has therefore already been proven in power iteration simulations; it requires no special treatment or modifications to the general algorithm for power iteration [5]. To the best of our knowledge, the only existing transport code which uses any variant of negative-weighted delta tracking is GUARDYAN, developed at the Budapest University of Technology [6]. GUARDYAN is unique in that it was written to only run on Graphics Processing Units (GPUs), and is devoted to dynamic (time-dependent) Monte Carlo simulations for the study of reactor transients. The code adopts Legrady et al.’s variant of negative-weighted delta tracking, always with a majorant cross section for a sampling cross section, therefore preventing the production of negative-weighted particles according to Alg. 2.3 [6]. It appears that negative-weighted delta tracking has never been tested with power iteration on a problem where a non-majorant sampling cross section was used (i.e. where negative-weighted particles can be produced). We will subsequently outline the modifications which we made to the power iteration algorithm to accommodate negative weights, and then discuss the behavior of the resulting algorithm. In particular, we will show that instabilities may appear, seriously affecting the convergence of this scheme.

6.1.1 . Accommodating Negative Weights in Power Iteration

There are two possible methods to handle the production of fission particles during power iteration. In the TRIPOLI-4[®] Monte Carlo code, after nuclide i has been sampled, the number of fission particles born at a collision site is sampled from

$$n = \left\lfloor \frac{\nu_{i,f}(E)\sigma_{i,f}(E)}{\sigma_{i,t}(E)} + \xi \right\rfloor, \quad (6.1)$$

where $\xi \sim \mathcal{U}(0, 1)$, and each fission neutron inherits the statistical weight w from its parent [7]. At the beginning of the next generation, these particles can be normalized, so that the net weight entering each generation is kept constant, ensuring the stability of the particle population [5]. This approach to handling fission production in power iteration is able to treat negative weights without modification. If a particle with a negative weight produces fission particles, its progeny will inherit the negative weight of the parent neutron.

Other codes, like OpenMC,¹ use an alternative method: the number of fission particles born at a collision site is sampled from

$$n = \left\lfloor \frac{w}{k_{\text{eff}}^{(g-1)}} \cdot \frac{\nu_{i,f}(E)\sigma_{i,f}(E)}{\sigma_{i,t}(E)} + \xi \right\rfloor, \quad (6.2)$$

where $k_{\text{eff}}^{(g-1)}$ is the estimated multiplication factor from the previous generation [9].² In this framework, each particle is born with the same weight, and the weight entering each generation is kept constant. If N particles enter a generation, and the simulation began with an initial net weight of W , then each particle will be initialized with a weight of W/N [8]. This methodology is not able to handle particles with negative weights; Eq. (6.2) would imply the creation of a negative number of particles, which is of course not possible. Instead, we modify the algorithm in the following manner. At each collision site,

$$n = \left\lfloor \frac{|w|}{k_{\text{eff}}^{(g-1)}} \cdot \frac{\nu_{i,f}(E)\sigma_{i,f}(E)}{\sigma_{i,t}(E)} + \xi \right\rfloor \quad (6.3)$$

fission particles are sampled. If the parent of the fission particles was positive, they are all born with a weight of 1; if the parent was instead negative, their weights are all initialized to be -1 . Once the current generation has finished, all of the fission source particles must be normalized before they begin the next generation. To normalize their weights, we first calculate the *net weight* of the particles entering the next generation as

$$W_{\text{net}} = \sum_i w_i. \quad (6.4)$$

We then can determine a coefficient

$$C_w = \frac{W}{W_{\text{net}}} \quad (6.5)$$

which will multiply all of the fission particles' weights. This enforces $W_{\text{net}} = W$ at beginning of each generation. Due to the inclusion of negative weights in the simulation, the number of particles entering a generation no longer plays a role in determining the initial weight of each particle, contrary to the case of the original OpenMC/MCNP implementation which assumed all weights to be positive [8].

For all of our studies on the behavior of negative-weighted delta tracking in the context of power iteration, we decided to use the modified MCNP approach for treating fission production. This choice was admittedly entirely arbitrary. Carter et al.'s variant of negative-weighted delta tracking will first be tested on the "broad Gaussian cross section" benchmark problem, which was used in Chapter 4. The problem will be slightly modified, adding a fission cross section. While the total macroscopic cross section varies spatially, the scattering, capture, and fission probabilities will be constant within the rod, and have values of $\Sigma_s/\Sigma_t = 0.7$, $\Sigma_c/\Sigma_t = 0.1$, and $\Sigma_f/\Sigma_t = 0.2$ respectively, and the number of neutrons born per fission will be $\nu_f = 2.5$. This simple test problem is presented in detail in both Chapters 7 and 8. Our key finding is that the simulation was not able to complete normally. System analysis tools indicated that, as the simulation ran, its memory footprint would grow without bound, and at an exponential rate with respect to the number of generations. Each fission generation would also take more and more time to complete. Eventually, the memory consumption would reach a point where the operating system would terminate the simulation. With memory consumption growing uncontrollably, and the time required to complete each generation also

¹MCNP uses the same formula, but replaces the microscopic cross sections for the isotope with the macroscopic cross sections for the material [8].

²For the first generation, a "guess" for the initial value of k_{eff} is used.

becoming longer, it was hypothesised that the particle population was growing without bound (despite a reported eigenvalue of $k_{\text{eff}} \approx 0.336$, according to the collision estimator for the generations which were able to complete). To gain insight into this peculiar (and unexpected) behavior, we have carefully examined the evolution of the number of simulated particles in the modified power iteration, as well as their statistical weights. For this purpose, we have defined several observables: N_+ is the number of positive-weighted particles, and N_- is the number of negative-weighted particles. From these two values, we can define the total number of particles as

$$N_{\text{tot}} = N_+ + N_-, \quad (6.6)$$

and the net number of particles as

$$N_{\text{net}} = N_+ - N_-. \quad (6.7)$$

A symmetric set of quantities can be defined for the weight as well, where the positive weight in the system is

$$W_+ = \sum_i \max(w_i, 0), \quad (6.8)$$

the negative weight in the system is

$$W_- = - \sum_i \min(w_i, 0), \quad (6.9)$$

the total weight is

$$W_{\text{tot}} = \sum_i |w_i| = W_+ + W_-, \quad (6.10)$$

and the net weight is

$$W_{\text{net}} = \sum_i w_i = W_+ - W_-. \quad (6.11)$$

These eight quantities were calculated at the beginning of each generation of the modified power iteration, and displayed as a function of the generation in Fig. 6.1. We observe that, while N_{net} and W_{net} remain constant with generation, N_+ , N_- , N_{tot} , W_+ , W_- , and W_{tot} are all increasing exponentially.³ This explains the increase in memory consumption and run times which was previously observed. Many other population control techniques such as particle splitting and combing were applied in an attempt to control the particle population and thus prevent the simulation from being abnormally terminated [5, 10]; unfortunately, none of these techniques provided any relief with respect to aforementioned instabilities. It is important to note that, although not explicitly displayed in this thesis, the inability to control the particle population when using negative-weighted delta tracking for power iteration was also observed when using the TRIPOLI-4[®] approach to treating the fission source sampling.

6.1.2 . Coupled Transport Equations

To better understand the mechanisms at play in eigenvalue problems involving positive and negative weights, which render the normalization of the particle population impossible, we will explicitly consider the flux from the positive particles, φ_+ , and the flux from the negative particles, φ_- , separately. This approach will be first taken in Chapter 7, and again in Chapter 8 with much more detail. Section 8.8 will provide our detailed derivation of a system of Boltzmann-like equations for φ_{\pm} , for negative-weighted delta tracking as outlined in Alg. 2.3. The resulting set of coupled transport equations can be shown to be:

$$\begin{aligned} \left[\hat{\Omega} \cdot \nabla + \Sigma_{\text{smp}}(\mathbf{r}, E) \right] \varphi_{\pm}(\mathbf{r}, \hat{\Omega}, E) = & \\ & \iint \nu_s(\mathbf{r}, E') \Sigma_s(\mathbf{r}, E') f_s(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) \varphi_{\pm}(\mathbf{r}, \hat{\Omega}', E') dE' d\hat{\Omega}' \\ & \frac{1}{k} \iint \nu_f(\mathbf{r}, E') \Sigma_f(\mathbf{r}, E') f_f(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) \varphi_{\pm} dE' d\hat{\Omega}' \\ & \Delta (\Sigma_{\text{smp}}(\mathbf{r}, E) - \Sigma_t(\mathbf{r}, E)) \varphi_{\pm}(\mathbf{r}, \hat{\Omega}, E) + \Delta (\Sigma_t(\mathbf{r}, E) - \Sigma_{\text{smp}}(\mathbf{r}, E)) \varphi_{\mp}(\mathbf{r}, \hat{\Omega}, E), \quad (6.12) \end{aligned}$$

³The “kink” observed in N at generation 1 in Fig. 6.1 is due to the initial guess of k_{eff} being too high compared to the actual value.

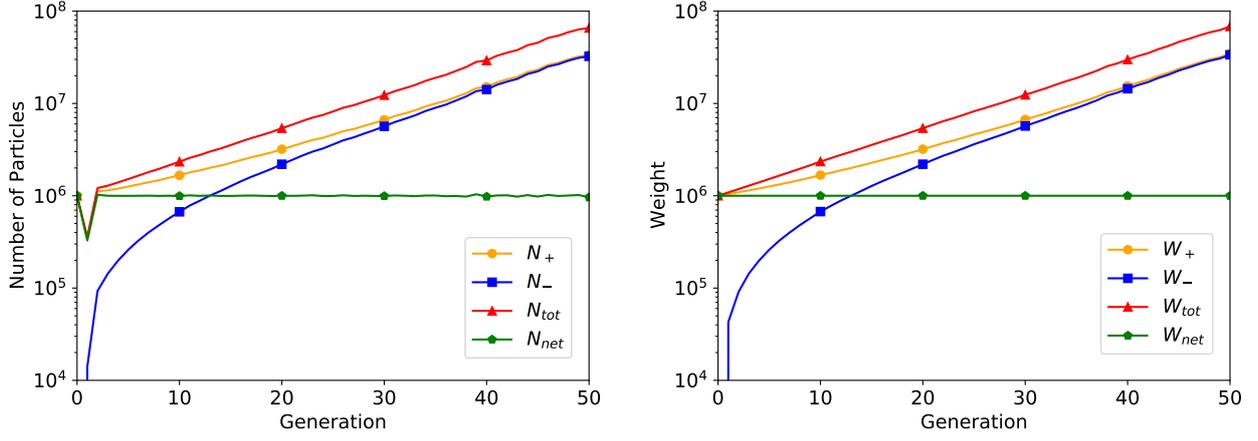


Figure 6.1: Behavior of the number of particles and weight in the simulation, given as a function of generation, for a power iteration simulation performed using negative-weighted delta tracking.

where the function $\Delta(x)$ is given by

$$\Delta(x) = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases}. \quad (6.13)$$

The terms involving Δ describe the coupling between positive and negative particles when virtual collisions occur at locations where $\Sigma_{\text{smp}}(\mathbf{r}, E) < \Sigma_t(\mathbf{r}, E)$.⁴ In this scheme, both φ_+ and φ_- are strictly positive quantities (when considering the fundamental eigenstate). The physical flux, corresponding to the fundamental eigenmode of the eigenvalue formulation of the Boltzmann equation, can be retrieved from these two quantities via $\varphi = \varphi_+ - \varphi_-$. However, Eq. (6.12) is associated to a general eigenvalue k , which does not necessarily coincide with the k_{eff} eigenvalue of the Boltzmann equation. At this point, to simplify notation it becomes useful to introduce the scattering operator

$$\mathcal{S}\varphi(\mathbf{r}, \hat{\Omega}, E) = \iint \nu_s(\mathbf{r}, E') \Sigma_s(\mathbf{r}, E') f_s(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) \varphi(\mathbf{r}, \hat{\Omega}', E') dE' d\hat{\Omega}' \quad (6.14)$$

and the fission operator

$$\mathcal{F}\varphi(\mathbf{r}, \hat{\Omega}, E) = \iint \nu_f(\mathbf{r}, E') \Sigma_f(\mathbf{r}, E') f_f(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) \varphi(\mathbf{r}, \hat{\Omega}', E') dE' d\hat{\Omega}'. \quad (6.15)$$

Summarizing the analysis of this coupled system which will be performed in Sec. 8.3.1, we write the coupled system of transport equations in Eq. (6.12) in operator form as

$$\mathbf{A}\zeta = \frac{1}{k}\mathbf{F}\zeta, \quad (6.16)$$

where the desired eigenstate is

$$\zeta = \begin{bmatrix} \varphi_+ \\ \varphi_- \end{bmatrix} \quad (6.17)$$

and the operators \mathbf{A} and \mathbf{F} are taken to be

$$\mathbf{A} = \begin{bmatrix} \hat{\Omega} \cdot \nabla + \Sigma_{\text{smp}} - \mathcal{S} - \Delta(\Sigma_{\text{smp}} - \Sigma_t) & -\Delta(\Sigma_t - \Sigma_{\text{smp}}) \\ -\Delta(\Sigma_t - \Sigma_{\text{smp}}) & \hat{\Omega} \cdot \nabla + \Sigma_{\text{smp}} - \mathcal{S} - \Delta(\Sigma_{\text{smp}} - \Sigma_t) \end{bmatrix} \quad (6.18)$$

and

$$\mathbf{F} = \begin{bmatrix} \mathcal{F} & 0 \\ 0 & \mathcal{F} \end{bmatrix}. \quad (6.19)$$

⁴The following identities may be useful to the reader: $\Delta(x) + \Delta(-x) = |x|$ and $\Delta(x) - \Delta(-x) = x$.

Upon inspection of Eq. (6.16), we notice that we can define a parity operator

$$\mathbf{P} = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}, \quad (6.20)$$

I being the identity operator. We can easily verify that $\mathbf{PAP} = \mathbf{A}$ and $\mathbf{PFP} = \mathbf{F}$. The quantity \mathbf{P} is also an involutive operator, meaning that $\mathbf{P} = \mathbf{P}^{-1}$. From these two properties, we can demonstrate that, if $\bar{\zeta}$ is an eigenstate of Eq. (6.16) with associated eigenvalue \bar{k} , then so is $\mathbf{P}\bar{\zeta}$:

$$\begin{aligned} \mathbf{A}(\mathbf{P}\bar{\zeta}) &= (\mathbf{P}\mathbf{P})\mathbf{A}(\mathbf{P}\bar{\zeta}) \\ &= \mathbf{P}(\mathbf{PAP})\bar{\zeta} \\ &= \mathbf{P}\mathbf{A}\bar{\zeta} \\ &= \frac{1}{\bar{k}}\mathbf{P}\mathbf{F}\bar{\zeta} \\ &= \frac{1}{\bar{k}}(\mathbf{P}\mathbf{F})(\mathbf{P}\mathbf{P})\bar{\zeta} \\ &= \frac{1}{\bar{k}}(\mathbf{P}\mathbf{F}\mathbf{P})(\mathbf{P}\bar{\zeta}) \\ &= \frac{1}{\bar{k}}\mathbf{F}(\mathbf{P}\bar{\zeta}). \end{aligned} \quad (6.21)$$

For $\bar{\zeta}$ to be nondegenerate, it must be the case that

$$\mathbf{P}\bar{\zeta} = \pm\bar{\zeta}, \quad (6.22)$$

where either $\varphi_-(\mathbf{r}, \hat{\Omega}, E) = -\varphi_+(\mathbf{r}, \hat{\Omega}, E)$, or $\varphi_+(\mathbf{r}, \hat{\Omega}, E) = \varphi_-(\mathbf{r}, \hat{\Omega}, E)$. Eq. (6.21) indicates that any solution of Eq. (6.16) is also a solution of Eq. (6.22), i.e. the k -eigenvectors have a well-defined parity; there is a family of odd-parity states ζ_o with dominant eigenvalue k_o , and a family of even-parity states ζ_e with dominant eigenvalue k_e . Choosing to write ζ_o as

$$\zeta_o = \varphi(\mathbf{r}, \hat{\Omega}, E) \begin{bmatrix} 1 \\ -1 \end{bmatrix} \quad (6.23)$$

and substituting into Eq. (6.16), we obtain

$$\hat{\Omega} \cdot \nabla \varphi + \Sigma_t \varphi = \mathcal{S} \varphi + \frac{1}{k_o} \mathcal{F} \varphi. \quad (6.24)$$

This is actually the Boltzmann k -eigenvalue equation for the physical system of interest: φ is the fundamental flux eigenmode which we are trying to estimate, and k_o is the multiplication factor k_{eff} . Therefore, the odd solutions of Eq. (6.16) correspond to the solutions of the physical Boltzmann k -eigenvalue equation. Writing ζ_e as

$$\zeta_e = \eta(\mathbf{r}, \hat{\Omega}, E) \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad (6.25)$$

and substituting into Eq. (6.16), a different transport equation is obtained:

$$\hat{\Omega} \cdot \nabla \eta + [\Sigma_{\text{smp}} - |\Sigma_{\text{smp}} - \Sigma_t|] \eta = \mathcal{S} \eta + \frac{1}{k_e} \mathcal{F} \eta. \quad (6.26)$$

The even eigenstate has an effective total cross section on the left hand side, whereas the scattering and fission operators are identical to those found in the regular transport equation. Since $\Sigma_{\text{smp}} - |\Sigma_{\text{smp}} - \Sigma_t| \leq \Sigma_t$, and the equation for η has the same Σ_s and Σ_f as in the equation for φ , then the equation for η must have a smaller capture cross section. We may therefore conclude that the dominant eigenvalues of Eqs. (6.24) and (6.26) must satisfy

$$k_e \geq k_o = k_{\text{eff}}. \quad (6.27)$$

By nature of the power-iteration algorithm, the simulation will always converge to the eigenstate corresponding to the eigenvalue with the largest magnitude. Therefore, if $\Sigma_{\text{smp}}(\mathbf{r}, E) < \Sigma_t(\mathbf{r}, E)$ at any position or energy within the problem domain, power iteration will not converge on the physical flux, φ , but will converge on the non-physical eigenstate η . Moreover, by keeping the net weight of the system constant, be it with Eq. (6.3) or with the TRIPOLI-4[®] algorithm, the total weight is left free to increase by a factor k_e/k_o at each generation. This explains the exponential growth observed in Fig. (6.1). This analysis, further developed in Chapters 7 and 8, sheds light on the failure of the negative-weighted delta tracking algorithm when applied to eigenvalue problems, and supports our numerical findings on the observed instabilities. No population control mechanism, or normalization techniques can be applied to the fission source to prevent this convergence toward the non-physical even parity eigenstate. This was indeed observed to be the case when different population control mechanisms were applied to our test problem.

6.1.3 . Necessity of Weight Cancellation

In light of the insights gained through the analysis in Sec. 6.1.2, we might wonder whether negative-weighted delta tracking has any hope of being used for power iteration without a majorant cross section. As a matter of fact, there are several other occurrences of Monte Carlo methods involving mixtures of positive and negative particle weights, in the field of neutronics and beyond. A literature survey suggests that many of these different types of simulations and systems benefit from, or even strictly require, the use of “weight cancellation” techniques in order to ensure proper convergence [11–16]. By weight cancellation, it is essentially meant that a particle with a positive statistical weight must “find” a particle that has a negative statistical weight: these two particles must “combine” in some fashion, so that their weights can be summed. In a continuous phase space, the probability that any two particles actually “meet” at exactly the same point is vanishingly small, which suggests that the application of weight cancellation methods will generally demand extreme care. Based on these considerations, in Sec. 8.3.2, we will demonstrate that the application of a weight cancellation procedure can indeed allow negative-weighted delta tracking with power iteration to successfully converge to the fundamental eigenstate, thus overcoming the aforementioned issues. The derivation of this scheme is also summarized here, as this result is one of the key elements of this thesis.

While the solution to the transport equation in Eq. (6.24) is an element of a vector space \mathcal{V} , the solution to the coupled system in Eq. (6.16) is an element of $\mathcal{V} \times \mathcal{V}$. We can define an operator \mathbf{D} which yields the physical flux $\varphi = \varphi_+ - \varphi_-$ when applied to ζ . Therefore, \mathbf{D} maps $\mathcal{V} \times \mathcal{V} \rightarrow \mathcal{V}$. A suitable definition for \mathbf{D} is then

$$\mathbf{D} = \begin{bmatrix} I & -I \end{bmatrix}. \quad (6.28)$$

A second operator is also required, which instead provides the mapping $\mathcal{V} \rightarrow \mathcal{V} \times \mathcal{V}$, and we shall denote it as \mathbf{E} . There are many suitable choices for \mathbf{E} ; we only require that

$$\mathbf{D}\mathbf{E} = I, \quad (6.29)$$

since mapping from \mathcal{V} up to $\mathcal{V} \times \mathcal{V}$, and back down to \mathcal{V} , should not change the original function which $\mathbf{D}\mathbf{E}$ was acting on. Several different options for \mathbf{E} will be proposed in Chapter 8, but we do not elaborate on \mathbf{E} anymore here, as the exact form is not necessarily important.

A *cancellation operator* \mathbf{C} can be constructed with \mathbf{D} and \mathbf{E} :

$$\mathbf{C} = \mathbf{E}\mathbf{D}. \quad (6.30)$$

\mathbf{C} first takes the difference of the positive and negative flux, which conceptually corresponds to summing positively and negatively weighted particles that exist at the exact same point in phase space (the action of \mathbf{D}). Once the weights have been cancelled, the result is then moved back into the expanded vector space by the action of \mathbf{E} . By virtue of the action of \mathbf{D} , this cancellation operator annihilates all even-parity vectors, leaving only odd-parity vectors. We can therefore apply cancellation to the fission source, as

$$\mathbf{A}\zeta = \frac{1}{k}\mathbf{C}\mathbf{F}\zeta. \quad (6.31)$$

In order to verify that this has not altered the odd eigenstates associated with the physical transport equation, we apply D to both sides, resulting in

$$DA\zeta = \frac{1}{k}DCF\zeta = \frac{1}{k}DF\zeta, \quad (6.32)$$

which simplifies to the physical transport equation in Eq. (6.24). The physical eigenstates and corresponding eigenvalues are therefore left intact by cancellation.

The removal of the even eigenstates from the fission source should prevent power iteration from converging on the non-physical solutions. In Sec. 8.3.3, we will devise a method to model cancellation in a deterministic solver, where we demonstrate how cancellation is indeed able to permit convergence of power iteration. However, a more important result which will result from this deterministic model is the fact that ***there is a minimum amount of cancellation which is required to permit convergence of power iteration on the physical eigenstate.*** Although we have yet to discuss how cancellation might be carried out in a Monte Carlo simulation, it is possible that (independently of the algorithm applied) the removal of all of the negative weight from the fission source might be unfeasible. If the amount of cancellation that is attainable is not sufficient to ensure $k_o > k_e$, it would simply not be possible to use negative-weighted delta tracking in the simulation. Even if it is the case that $k_o > k_e$, this does not necessarily indicate that running the simulation with negative-weighted delta tracking is feasible. In the event that the dominance ratio k_e/k_o is very close to unity, the resulting W_{tot} of the system might be so large as to prevent the simulation from running on most computers, or to prevent adequate convergence of scores.

6.2 . Approximate Regional Weight Cancellation

There are several different methods which could be implemented in a Monte Carlo code to perform weight cancellation of positive and negative particles. One such method is an approximate regional cancellation method, which has been used extensively by Yamamoto, for a variety of problems [12–15]. In his approach, Yamamoto scores the fission rate in cancellation regions that are pre-defined by the user in some manner. Positive particles bring a positive contribution to the fission rate in a cancellation region, whereas negative particles bring a negative contribution, leading to a cancellation of weight. At the end of a generation, the net intensity of the fission source in each cancellation region can be calculated, and used to sample the fission source particles which will initiate the next generation. The new fission source particles are distributed uniformly within the cancellation region, which leads to an approximation. An additional disadvantage of this method is that fission particles need to be re-sampled: re-sampling fission particles in continuous energy is somewhat difficult to accomplish, as the spectrum for the fission energy is typically dependent on the incident neutron energy [17].

Another approximate regional cancellation method has been proposed by Zhang et al. and has been demonstrated on the calculation of the first sixty eigenstates and eigenvalues for the BEAVRS benchmark reactor [18, 19]. In their approach, a mesh is imposed on top of the geometry, dividing the domain into many cancellation regions. Over the course of the simulation, fission particles are sampled in the standard manner, and stored in the fission bank. At the end of the generation, the fission particles are sorted into the cancellation regions created by the cancellation mesh. With all particles sorted into the cancellation regions, the average weight of all the particles in the region is then calculated, and assigned to each of the particles in the region. Figure 6.2 shows the effective result of this operation. This technique has several advantages over Yamamoto's method. First, it requires fewer modifications to an existing code, and is slightly easier to implement. Additionally, it does not require the re-sampling of fission particles, as these are sampled in the standard manner, with the incident energy of their parent. While the net weight in the cancellation region is then distributed amongst all of the contributing particles, the positions in phase space of the sampled fission sites do not change, which should be seen as an advantage. This method is also fast to run from a computational standpoint. Unfortunately, approximate regional cancellation will introduce a bias in the fission source, which will propagate to all the simulation results. The bias can be reduced by using smaller

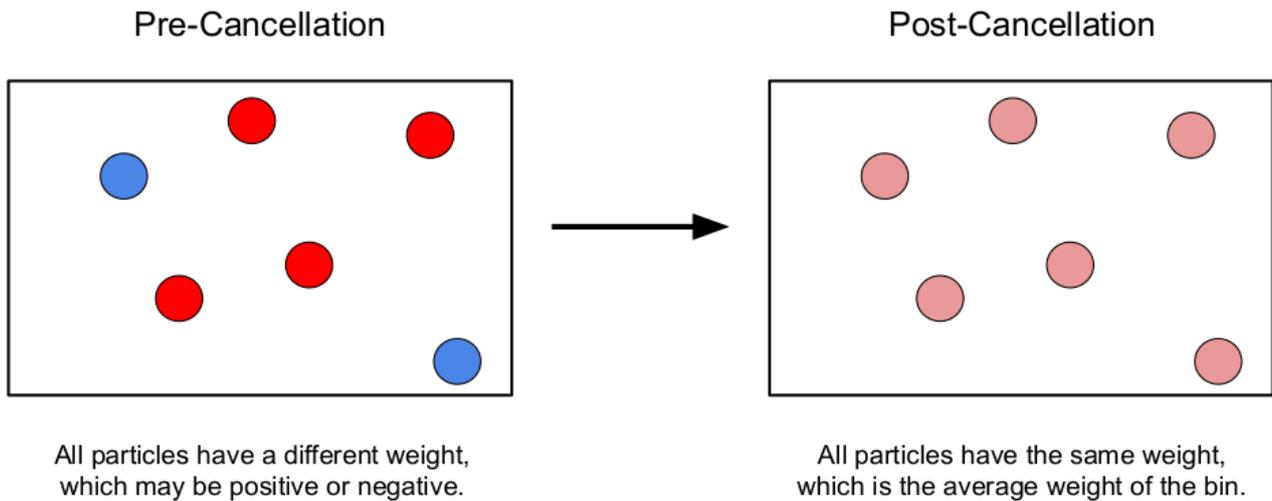


Figure 6.2: Depiction of the approximate regional cancellation method, where the average weight is assigned to all particles. After this operation, all particle weights in the same region will have the same sign.

cancellation regions; as a consequence however, there will also be fewer particles per cancellation region, reducing the cancellation efficiency of the method.

Considering the trade-offs of the two approximate techniques, we have chosen to use the method proposed by Zhang et al. as our approximate cancellation method of choice. For the rest of this work, this will be the method which is being referred to when approximate regional cancellation is mentioned. In Chapter 7, we attempt to use negative-weighted delta tracking to solve the C5G7 benchmark problem. C5G7 is a three dimensional multi-group benchmark (7 groups), consisting of a 1/8th simplified core, with four 17×17 fuel assemblies [20]. The published reference value for the multiplication factor of the C5G7 benchmark is $k_{\text{eff}} = 1.183810 \pm 0.000036$ [20]. This problem does not contain spatially continuous cross sections, but instead assumes homogeneous media, as it is standard in today's Monte Carlo simulations. The sampling cross section used for the problem was the majorant cross section for all but the 1st energy group, where the sampling cross section was taken to be 90% of the majorant for that group. Section 7.4 again demonstrates that the particle population diverges; without weight cancellation, it was not possible to complete the C5G7 simulation using negative-weighted delta tracking. A $34 \times 34 \times 10$ cancellation mesh was then imposed over the four assemblies in the problem. Each fuel pin was then given its own cancellation mesh region in the x - y plane, and was split into 10 different axial cancellation regions along the z -axis. Using approximate regional weight cancellation, N_{tot} no longer increased exponentially, but was easily controlled, allowing the simulation to finish uninterrupted. Despite a cancellation mesh that might be considered "too coarse", the estimated multiplication factor was $k_{\text{eff}} = 1.18378 \pm 0.00002$, well within the statistical error of the reference value. The results from Chapter 7 indicate that not only is weight cancellation a solution to perform power iteration with negative-weighted delta tracking, approximate regional weight cancellation is also an easily implemented method to accomplish weight cancellation. Initial results seem to indicate that the imposed bias on the fission source is small enough so as to not be of great importance for many reactor physics problems.

6.3 . Exact Regional Weight Cancellation

The choice to use a Monte Carlo code over a deterministic code is often at least partially driven by the fact that it requires far fewer approximations in its solution strategy. An added sense of fidelity to the physics at play is often attractive to code users. It is therefore firmly implanted within the Monte Carlo mindset that exact physics, or an exact algorithm, should be used whenever possible (or at least be made available). Despite the success of the approximate regional cancellation method which will be outlined in

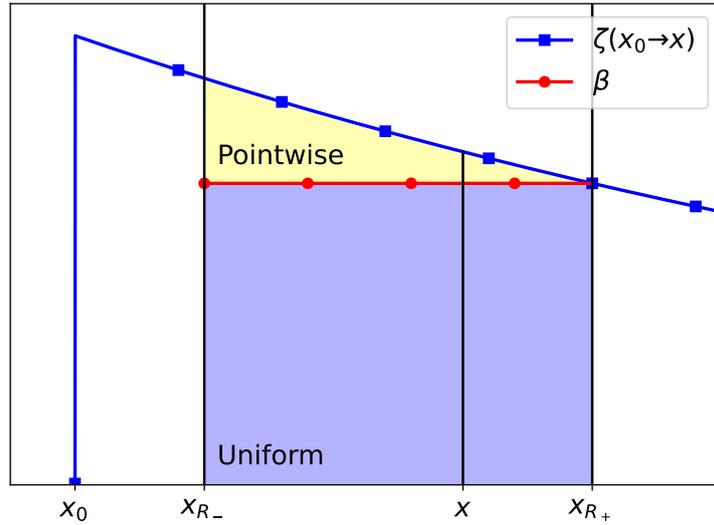


Figure 6.3: Depiction of a one dimensional example of exact regional cancellation. The cancellation region is delimited by the positions $x_{\mathcal{R}_-}$ and $x_{\mathcal{R}_+}$. A particle beginning a flight at x_0 would then contribute at least β to the fission emission density at every position within the cancellation region. The sampled position of the emitted fission particle, x , then determines what fraction of the particle can be uniformly distributed within the region, and what fraction must remain at x .

Chapter 7, we will now examine possibilities for an *exact cancellation* method, which would impose no bias on simulation results.

Only a handful of works from the literature appear to mention possible weight cancellation techniques which are exact. Booth initially proposed a point-wise cancellation scheme, which estimated the fission density contribution from all particles, at the collision sites of all the other particles [11]. This method had a computational complexity of $\mathcal{O}(N_{\text{tot}}^2)$, which would make it very time consuming in a general-purpose Monte Carlo code. Later, Booth and Gubernatis proposed an *exact regional cancellation* method, which is linear in complexity with respect to the number of particles [21]. We will focus only on exact regional cancellation in this work, as its $\mathcal{O}(N_{\text{tot}})$ complexity makes the method much more appealing for implementation in a general Monte Carlo code.

6.3.1 . Conceptual Introduction

Regional cancellation makes use of a fission emission density function, which we shall denote as $\zeta(P' \rightarrow P)$. This is interpreted as the expected contribution to the fission emission density at the generalized phase space point $P = (\mathbf{r}, \hat{\Omega}, E)$, given an event that occurred at the phase space point P' . Booth and Gubernatis do not give a precise definition of $\zeta(P' \rightarrow P)$ in their work [21], but their proposed concept is sufficient for the time being. A particle entering either a flight or a collision at the point P' could potentially contribute to the fission emission density at the point P , after undergoing a fixed number of flights and collisions. Now, we consider a “cancellation region” \mathcal{R} , which is a generalized phase space volume, with dimensions along all three spatial axes, both direction axes, and the energy axis. A particle can begin a series of flights/collisions at the point P' , and eventually produce a fission particle at the point P , which we take to be located within \mathcal{R} .

The following summarizes the concept of exact regional cancellation as described by Booth and Gubernatis [21], and as implemented in Chapter 7. From the point P , we are (at least in theory) able to look back at the particles history, and calculate $\zeta(P' \rightarrow P)$ for any previous point P' that occurred during the particle’s lifetime. If, for a given point P' , $\zeta(P' \rightarrow P'') \neq 0$ for all values of $P'' \in \mathcal{R}$, it is possible to envision splitting the fission particle that was emitted at P into two portions: one which will remain at

P , and another portion which will then be “smeared” across the cancellation region \mathcal{R} , contributing to the fission emission density uniformly at all points in \mathcal{R} . Figure 6.3 helps demonstrate this concept, with a simplified one-dimensional phase space example. The blue line indicates the value of the fission emission density for a particle beginning a flight at x_0 , and contributing to the fission emission density at x . The cancellation region \mathcal{R} is delimited by the vertical lines at $x_{\mathcal{R}_-}$ and $x_{\mathcal{R}_+}$. If the flight starting at x_0 produced a fission particle at x , we could consider the possibility of $(\zeta(x_0 \rightarrow x) - \beta)/\zeta(x_0 \rightarrow x)$ of the particle being located exactly at x , and $\beta/\zeta(x_0 \rightarrow x)$ of the particle being uniformly distributed at all points between $x_{\mathcal{R}_-}$ and $x_{\mathcal{R}_+}$. If the original particle at x was born with a weight of w , then the point-wise portion of the particle which will remain located at x has a weight of

$$w_p = \frac{\zeta(x_0 \rightarrow x) - \beta}{\zeta(x_0 \rightarrow x)} w, \quad (6.33)$$

and the portion which can be uniformly distributed across \mathcal{R} has a weight of

$$w_u = \frac{\beta}{\zeta(x_0 \rightarrow x)} w. \quad (6.34)$$

Splitting the fission particles into a point-wise and uniform component does nothing to achieve cancellation in itself. Suppose however that multiple fission particles are born in the cancellation region \mathcal{R} , and particle i is able to contribute a uniform portion of $w_{u,i}$.⁵ These uniform portions can be summed into a variable

$$U = \sum_i w_{u,i}, \quad (6.35)$$

which is unique for each cancellation region. This sum of the uniform components has effectively led to a cancellation effect, as some of the particles could have been initially born with a positive weight, and some with a negative weight. Once the uniform portions of all particles which were born in \mathcal{R} have been summed, the uniform weight contained in U must now be somehow distributed within the region. To do this,

$$n = \lceil |U| \rceil \quad (6.36)$$

new particles will be sampled uniformly within \mathcal{R} , and each will be assigned a weight of U/n .

Here, we have assumed that β is the minimum value of $\zeta(x_0 \rightarrow x)$ for all points $x \in \mathcal{R}$. This facilitates the visualization, but Booth and Gubernatis have shown that β can actually be taken to be any value, without biasing the fission source [21]. This cancellation method is tested on a one-dimensional, single-speed system in Sec. 7.3, using negative-weighted delta tracking. For this test case, we use a fission emission density function of

$$\zeta(x_0 \rightarrow x) = \Sigma_f(x) \exp(-\Sigma_{\text{smp}}|x - x_0|), \quad (6.37)$$

which is interpreted as the probability of a particle to fly from a previous collision at x_0 to x , and induce a fission. This approach was shown to work well at performing weight cancellation, and allowed power iteration to converge without issue. Nonetheless, many questions remain, such as what form the fission emission density function should take, how cancellation might be performed in more than one dimension and in continuous energy, and how should β be determined.

6.3.2 . Conditions for the Unbiasedness of Regional Cancellation

Having outlined the general principle of exact regional cancellation in Sec. 6.3.1, we expand upon the foundation laid by Booth and Gubernatis. Chapter 9 provides a detailed derivation for how exact regional cancellation can be performed in three spatial dimensions, and in continuous energy. The results from Sec. 9.3 are summarised very briefly here.

We make use of the integral form of the Boltzmann k -eigenvalue transport equations for this analysis, introducing the collision density

$$\Psi(P) = \Sigma_t(\mathbf{r}, E) \varphi(\mathbf{r}, \hat{\Omega}, E), \quad (6.38)$$

⁵Each particle i would have a unique β_i , used to calculate its uniform portion.

and the emission density, which we denote as $\chi(P)$. These two quantities are coupled through the relations

$$\Psi(P) = \int T(P' \rightarrow P) \chi(P') dP' \quad (6.39)$$

$$\chi(P) = \int C(P' \rightarrow P) \Psi(P') dP', \quad (6.40)$$

where $T(P' \rightarrow P)$ is the flight kernel and $C(P' \rightarrow P)$ is the collision kernel [5]. For our purposes here, we will use

$$T(P' \rightarrow P) = \frac{\Sigma_t(\mathbf{r}, E')}{|\mathbf{r} - \mathbf{r}'|^2} \exp\left(-\int_0^{|\mathbf{r}-\mathbf{r}'|} \Sigma_t(\mathbf{r}' + u\hat{\Omega}', E') du\right) \delta\left(\hat{\Omega} - \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|}\right) \delta(\hat{\Omega}' - \hat{\Omega}) \delta(E - E') \quad (6.41)$$

for the flight kernel, and

$$C(P' \rightarrow P) = \bar{\nu}(\mathbf{r}', E') \bar{f}\left(\hat{\Omega}, E|\mathbf{r}', \hat{\Omega}', E'\right) \delta(\mathbf{r} - \mathbf{r}') \quad (6.42)$$

for the collision kernel, $\bar{\nu}$ being the average number of particles emitted per collision, and \bar{f} being the average transfer function. We also introduce the fission emission density

$$\chi_f(P) = \frac{1}{k} \int C_f(P' \rightarrow P) \varphi(P') dP', \quad (6.43)$$

where $C_f(P' \rightarrow P)$ is the fission kernel, taking the form of

$$C_f(P' \rightarrow P) = \frac{\nu_f(\mathbf{r}', E') \Sigma_f(\mathbf{r}', E')}{k \Sigma_t(\mathbf{r}', E')} f_{\text{fiss}}\left(\hat{\Omega}, E|\mathbf{r}', \hat{\Omega}', E'\right) \delta(\mathbf{r} - \mathbf{r}'). \quad (6.44)$$

Now consider a particle which enters a collision at P_1 . Upon application of the collision kernel by Eq. (6.40), the particle will leave the collision at P_2 , and enter a flight. Application of the flight operator through Eq. (6.39) moves the particle to P_3 . Subsequently applying the fission kernel as in Eq. (6.44) yields a fission particle which is emitted at P_4 . This series of events leads us to then consider the fission emission density at P_4 , for all possible trajectories which begin at P_1 where a particle enters a collision. The subsequent application of the collision, flight, and fission kernels on the collision density yields the transfer function

$$\zeta(P_1 \rightarrow P_4) = \int dE_3 \frac{\nu_f(\mathbf{r}_4, E_3) \Sigma_f(\mathbf{r}_4, E_3) f_{\text{fiss}}\left(\hat{\Omega}_4, E_4|\mathbf{r}_4, \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3\right) \bar{f}\left(\frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3|\mathbf{r}_1, \hat{\Omega}_1, E_1\right)}{k|\mathbf{r}_4 - \mathbf{r}_1|^2} \times \exp\left(-\int_0^{|\mathbf{r}_4 - \mathbf{r}_1|} \Sigma_t\left(\mathbf{r}_1 + u \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3\right) du\right). \quad (6.45)$$

$\zeta(P_1 \rightarrow P_4)$ is the expected contribution to the fission emission density at point P_4 , for a particle that undergoes a collision-flight-fission, starting at P_1 . From here, we wish to develop an estimator for the fission emission density at phase space coordinate $Q \in \mathcal{R}$, which acts on fission particles which were emitted at $P_4 \in \mathcal{R}$, and started a collision-flight-fission at P_1 . If this estimator, $\vartheta(P_1 \rightarrow P_4|\mathcal{R}, Q)$, is to be unbiased, it must conserve the expected fission emission density at Q , i.e. $\vartheta(P_1 \rightarrow P_4|\mathcal{R}, Q)$ must satisfy

$$\int \zeta(P_1 \rightarrow P_4) \vartheta(P_1 \rightarrow P_4|\mathcal{R}, Q) dP_4 = \zeta(P_1 \rightarrow Q) \quad (6.46)$$

to be unbiased. The choice of

$$\vartheta(P_1 \rightarrow P_4|\mathcal{R}, Q) = \delta(P_4 - Q) \quad (6.47)$$

corresponds to the standard estimator for the fission emission density, where a particle only contributes to the fission emission density at the exact phase space coordinate where it was emitted. Using the intuition gained from Booth and Gubernatis' work, in Sec. 9.3.2 it is shown that

$$\vartheta(P_1 \rightarrow P_4 | \mathcal{R}, Q) = \frac{\zeta(P_1 \rightarrow P_4) - \beta}{\zeta(P_1 \rightarrow P_4)} \delta(P_4 - Q) + \frac{\beta}{\zeta(P_1 \rightarrow P_4)} \frac{1}{V_{\mathcal{R}}} \quad (6.48)$$

is also an unbiased estimator for the fission density, where

$$V_{\mathcal{R}} = \int_{\mathcal{R}} dP. \quad (6.49)$$

This estimator contributes $1 - \beta/\zeta(P_1 \rightarrow P_4)$ of the fission particle's weight to the fission emission density at P_4 , and $\beta/\zeta(P_1 \rightarrow P_4)$ of the fission particle's weight uniformly across all phase space coordinates in \mathcal{R} . For Eq. (6.46) to hold true upon insertion of Eq. (6.48), it is required that β have no functional dependence on P_4 . We are therefore permitted to use any information from the sampled history at P_1 , P_2 , and P_3 in the calculation of β , and regional cancellation will be unbiased. The transition kernel $\zeta(P_1 \rightarrow P_4)$ must also be non-zero for all points in \mathcal{R} , which implies that the entire cancellation region must be fissile, and "accessible" for histories starting at P_1 .

The transition kernel that we have developed in Eq. (6.45) is somewhat problematic, however, since performing the integral over E_3 is likely not feasible (and potentially even impossible) in a general continuous-energy Monte Carlo code. To remove the integral over E_3 from the transition kernel, we must instead not start from P_1 , but from an intermediate point between P_1 and P_2 . This concept is developed in Secs. 9.3.3 and 9.3.4, where additional details can be found. Briefly, we must start not at the beginning of the collision (P_1), but at the point mid-way through the collision where the energy E_2 has been sampled, but the new direction, $\hat{\Omega}_2$, is still unknown. This is elucidated from the form of the collision kernel which is typically used in continuous-energy Monte Carlo codes, where a nuclide is first sampled, then a scattering channel, then an energy, and finally a direction. Such a collision kernel can be written as

$$C(P' \rightarrow P) = C_s(P' \rightarrow P) + \frac{1}{k} C_f(P' \rightarrow P), \quad (6.50)$$

where the scattering kernel is now given as

$$C_s(P' \rightarrow P) = \frac{\delta(\mathbf{r} - \mathbf{r}')}{\Sigma_t(\mathbf{r}', E')} \sum_i N_i(\mathbf{r}') \sum_{\substack{m \\ m \neq \text{fiss}}} \nu_{i,m}(E') \sigma_{i,m}(\mathbf{r}', E') f_{i,m}(E | \hat{\Omega}', E') f_{i,m}(\hat{\Omega} | \hat{\Omega}', E', E) \quad (6.51)$$

and the fission kernel is given as

$$C_f(P' \rightarrow P) = \frac{\delta(\mathbf{r} - \mathbf{r}')}{\Sigma_t(\mathbf{r}', E')} \sum_i N_i(\mathbf{r}') \nu_{i,\text{fiss}}(E') \sigma_{i,\text{fiss}}(\mathbf{r}', E') f_{i,\text{fiss}}(E | \hat{\Omega}', E') f_{i,\text{fiss}}(\hat{\Omega} | \hat{\Omega}', E', E). \quad (6.52)$$

In this formalism, the scattering law for isotope i and reaction channel m has been decomposed into

$$f_{i,m}(\hat{\Omega}, E | \hat{\Omega}', E') = \underbrace{f_{i,m}(E | \hat{\Omega}', E')}_{\text{Marginal PDF for } E} \times \underbrace{f_{i,m}(\hat{\Omega} | \hat{\Omega}', E', E)}_{\text{Conditional PDF for } \hat{\Omega}}. \quad (6.53)$$

Creating a transition kernel which starts at the partial collision point between P_1 and P_2 where E_2 has been sampled generates

$$\zeta(P_1 \rightarrow P_4 | i, m, E_3) = \frac{\nu_f(\mathbf{r}_4, E_3) \Sigma_f(\mathbf{r}_4, E_3) f_{\text{fiss}}(\hat{\Omega}_4, E_4 | \mathbf{r}_4, \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3) f_{i,m}(\frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|} | \hat{\Omega}_1, E_1, E_3)}{k |\mathbf{r}_4 - \mathbf{r}_1|^2} \times \exp\left(-\int_0^{|\mathbf{r}_4 - \mathbf{r}_1|} \Sigma_t\left(\mathbf{r}_1 + u \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3\right) du\right). \quad (6.54)$$

This is a general transition kernel which could possibly be used to perform exact regional cancellation in a three dimensional continuous-energy problem. However, this kernel can be simplified under certain conditions. From Eq. (6.46), any terms in the transition kernel which do not depend on P_4 can be removed from the integral, and then cancel with the identical term on the other side. If our cancellation region \mathcal{R} is perfectly homogeneous, then terms such as $\nu_f(\mathbf{r}_4, E_3)$ and $\Sigma_f(\mathbf{r}_4, E_3)$ will be constant within \mathcal{R} , and can be neglected. Similarly, if fission is perfectly isotropic, and the distribution for the fission energy E_4 is independent of the incident energy, we do not need to perform cancellation along the dimensions of $\hat{\Omega}_4$ or E_4 , and the term $f_{\text{fiss}}\left(\hat{\Omega}_4, E_4|\mathbf{r}_4, \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3\right)$ can also be neglected. These conditions are true for the multi-group benchmark problems which are used to test exact regional cancellation in Chapters 8 and 9. For these multi-group problems, it is possible to reduce the transition kernel to just

$$\zeta(P_1 \rightarrow P_4|E_3) = \frac{\exp\left(-\int_0^{|\mathbf{r}_4 - \mathbf{r}_1|} \Sigma_t\left(\mathbf{r}_1 + u\frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3\right) du\right)}{|\mathbf{r}_4 - \mathbf{r}_1|^2}. \quad (6.55)$$

This transition kernel is tested in a modified version of the C5G7 multi-group benchmark (although with some slight modifications, covered in Sec. 9.3.6, for the accommodation of delta tracking and negative-weighted delta tracking). Exact regional cancellation is demonstrated to be unbiased for this three dimensional multi-group problem, and allows power iteration to converge on the fundamental eigenstate.

6.3.3 . Optimization of Exact Regional Cancellation

Section 9.3 informs us as to how the fission emission density function (transition kernel) can be determined, and what restrictions must be observed when calculating β , for cancellation to be unbiased. The choice of β is a complex topic, however, since Sec. 9.3.2 shows that β can take *any* value and result in an unbiased fission emission density estimator, so long as the value was not calculated using P_4 . Each particle k in a cancellation region will have its own cancellation parameter β_k . Combining Eqs. (6.10), (6.33), (6.34), and (6.35), we can write the total weight for the region \mathcal{R} after cancellation as

$$\Gamma = \sum_{k \in \mathcal{R}} \left| \frac{\zeta_k - \beta_k}{\zeta_k} w_k \right| + \left| \sum_{k \in \mathcal{R}} \frac{\beta_k}{\zeta_k} w_k \right|, \quad (6.56)$$

where ζ_k is a shorthand for $\zeta(P_{1,k} \rightarrow P_{4,k})$. Our initial implementation of cancellation used the minimum value of ζ_k within the cancellation region for β , as this was easy to determine, and seemed to be a relatively safe choice as it prevents the point-wise portion from changing sign (see Sec. 8.5). Nonetheless, it is entirely likely that there is an alternative method for calculating β_k , which results in a smaller value Γ when compared to using the minimum value of ζ_k . If we can minimize the total weight in each region, we will also minimize W_{tot} for the entire problem, therefore maximizing the efficiency of cancellation. Equation (6.56) is unfortunately quite unmalleable, due to the absolute values.

In Sec. 9.4, we propose to minimize the total weight in a cancellation region by using the alternative form

$$\Gamma_2 = \sum_{k \in \mathcal{R}} \left(\frac{\zeta_k - \beta_k}{\zeta_k} w_k \right)^2 + \left(\sum_{k \in \mathcal{R}} \frac{\beta_k}{\zeta_k} w_k \right)^2. \quad (6.57)$$

While replacing $|\cdot|$ with $(\cdot)^2$ has given us a differentiable function which could in theory be minimized, any resulting values for β_k would be functions of ζ_k , and therefore $P_{4,k}$, which is not allowed if cancellation is to be unbiased. In order to remove the functional dependence on $P_{4,k}$, two different possibilities are proposed in Secs. 9.4.1 and 9.4.2. The first proposition replaces $\zeta(P_{1,k} \rightarrow P_{4,k})$ with

$$\langle \zeta_k \rangle = \frac{\int_{\mathcal{R}} \zeta(P_{1,k} \rightarrow P_{4,k}) dP_{4,k}}{\int_{\mathcal{R}} dP_{4,k}}, \quad (6.58)$$

resulting in a straightforward minimization, as presented in Secs. 9.4.1 and 9.8. However, we show in Sec. 9.6.1 that for the modified C5G7 benchmark, this approach to choosing β_k results in a value of $W_{\text{tot}} \approx 8 \cdot 10^6$, while using the minimum of ζ_k for β_k results in $W_{\text{to}} \approx 4.9 \cdot 10^6$. It is not known why this approach fails to yield a better cancellation efficiency than using the minimum value. The fact that this method is almost half as efficient as using the minimum of ζ_k might indicate that simply replacing ζ_k with $\langle \zeta_k \rangle$ and then minimizing is not sufficient, and that similar results might be observed with other problems, although this has yet to be observed.

A second approach to optimizing the cancellation efficiency instead builds on Eq. (6.57), and minimizes the quantity

$$\langle \Gamma_2 \rangle = \frac{\int_{\mathcal{R}} \Gamma_2 \prod_{k \in \mathcal{R}} \zeta_k dP_{4,k}}{\int_{\mathcal{R}} \prod_{k \in \mathcal{R}} \zeta_k dP_{4,k}}. \quad (6.59)$$

The details of this optimization are provided in Sec. 9.9, and the resulting solution for β_k is given in Sec. 9.4.2. For this method,

$$\beta_k = \langle \zeta_k \rangle c_k \left(1 - \frac{S}{w_k} \right), \quad (6.60)$$

where

$$c_k = \left(2 \langle \zeta_k \rangle \left\langle \frac{1}{\zeta_k} \right\rangle - 1 \right)^{-1} \quad (6.61)$$

and

$$S = \frac{\sum_{k \in \mathcal{R}} c_k w_k}{1 + \sum_{k \in \mathcal{R}} c_k}. \quad (6.62)$$

We test this method in Sec. 9.6.1 and show that it yields a value of $W_{\text{tot}} = 2.7 \cdot 10^6$, which is much lower than the total weight when using the minimum value of ζ_k .

It is important to mention how $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$ are determined in the simulation. Performing the necessary integrations analytically or numerically would likely be too computationally expensive to be done in a general-purpose Monte Carlo code. Instead, we choose to use a Monte Carlo estimate for these quantities, where random points \tilde{P}_i are sampled uniformly within the phase space region \mathcal{R} . The needed quantities are then estimated as

$$\langle \zeta_k \rangle \approx \frac{1}{N_s} \sum_{i=1}^{N_s} \zeta(P_{1,k} \rightarrow \tilde{P}_i) \quad (6.63)$$

and

$$\left\langle \frac{1}{\zeta_k} \right\rangle \approx \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{1}{\zeta(P_{1,k} \rightarrow \tilde{P}_i)}, \quad (6.64)$$

N_s being the number of random points used. In this manner, we are able to easily treat non-cuboid cancellation regions by performing rejection sampling on the points \tilde{P}_i . This advantage makes exact regional cancellation much more robust, as we will demonstrate in Sec. 9.6.3 on the true C5G7 benchmark, with cylindrical fuel pins as per original specifications. Using exact regional cancellation, power iteration is able to converge on the fundamental eigenstate, and does not appear to demonstrate any bias in the resulting estimates for k_{eff} , or the scalar flux.

References

- [1] L. L. Carter, E. D. Cashwell, and W. M. Taylor, "Monte Carlo Sampling with Continuously Varying Cross Sections Along Flight Paths," *Nuclear Science and Engineering*, vol. 48, no. 4, p. 403–411, 1972.

- [2] J. Leppänen, M. Pusa, T. Viitanen, V. Valtavirta, and T. Kaltiaisenaho, "The Serpent Monte Carlo code: Status, development and applications in 2013," *Annals of Nuclear Energy*, vol. 82, p. 142–150, 2015.
- [3] J. Leppänen, "Performance of Woodcock delta-tracking in lattice physics applications using the Serpent Monte Carlo reactor physics burnup calculation code," *Annals of Nuclear Energy*, vol. 37, no. 5, p. 715–722, 2010.
- [4] —, "On the use of delta-tracking and the collision flux estimator in the Serpent 2 Monte Carlo particle transport code," *Annals of Nuclear Energy*, vol. 105, p. 161–167, 2017.
- [5] I. Lux and L. Koblinger, *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations*. CRC Press, 1991.
- [6] B. Molnar, G. Tolnai, and D. Legrady, "A GPU-based direct Monte Carlo simulation of time dependence in nuclear reactors," *Annals of Nuclear Energy*, vol. 132, p. 46–63, 2019.
- [7] A. Zoia, "Simulation Monte-Carlo pour le transport de particules: notice théorique," CEA, Tech. Rep. SERMA/LTSD/NT/2018-63255/B, 2019.
- [8] "MCNP-A General Monte Carlo N-Particle Transport Code, Version 5," Los Alamos National Laboratory, Tech. Rep. LA-UR-03-1987, 2003.
- [9] P. K. Romano, N. E. Horelik, B. R. Herman, A. G. Nelson, B. Forget, and K. Smith, "OpenMC: A state-of-the-art Monte Carlo code for research and development," *Annals of Nuclear Energy*, vol. 82, p. 90–97, 2015.
- [10] T. E. Booth, "A Weight (Charge) Conserving Importance-Weighted Comb for Monte Carlo," Los Alamos National Laboratory, Tech. Rep. LA-UR-96-0051, 1996.
- [11] —, "Computing the Higher k-Eigenfunctions by Monte Carlo Power Iteration: A Conjecture," *Nuclear Science and Engineering*, vol. 143, no. 3, p. 291–300, 2003.
- [12] T. Yamamoto, "Convergence of the second eigenfunction in Monte Carlo power iteration," *Annals of Nuclear Energy*, vol. 36, no. 1, p. 7–14, 2009.
- [13] —, "Monte Carlo method with complex weights for neutron leakage-corrected calculations and anisotropic diffusion coefficient generations," *Annals of Nuclear Energy*, vol. 50, p. 141–149, 2012.
- [14] —, "Monte Carlo algorithm for buckling search and neutron leakage-corrected calculations," *Annals of Nuclear Energy*, vol. 47, p. 14–20, 2012.
- [15] —, "Monte Carlo method with complex-valued weights for frequency domain analyses of neutron noise," *Annals of Nuclear Energy*, vol. 58, p. 72–79, 2013.
- [16] J. S. Spencer, N. S. Blunt, and W. M. Foulkes, "The sign problem and population dynamics in the full configuration interaction quantum Monte Carlo method," *The Journal of Chemical Physics*, vol. 136, no. 5, p. 054110, 2012.
- [17] A. Trkov, M. Herman, and D. A. Brown, "ENDF-6 Formats Manual," Brookhaven National Laboratory, Tech. Rep. BNL-203218-2018-INRE, 2018.
- [18] P. Zhang, H. Lee, and D. Lee, "A general solution strategy of modified power method for higher mode solutions," *Journal of Computational Physics*, vol. 305, p. 387–402, 2016.

- [19] P. Zhang, H. Lee, M. Lemaire, C. Kong, J. Choe, J. Yu, F. Khoshahval, and D. Lee, "Practical Monte Carlo simulation using modified power method with preconditioning," *Annals of Nuclear Energy*, vol. 127, p. 372–384, 2019.
- [20] E. E. Lewis, M. A. Smith, N. Tsoulfanidis, G. Palmiotti, T. A. Taiwo, and R. N. Blomquist, "Benchmark specification for Deterministic 2-D/3-D MOX fuel assembly transport calculations without spatial homogenisation (C5G7 MOX)," NEA/NSC, Tech. Rep. JT00105087, Mar 2001.
- [21] T. E. Booth and J. E. Gubernatis, "Exact Regional Monte Carlo Weight Cancellation for Second Eigenfunction Calculations," *Nuclear Science and Engineering*, vol. 165, no. 3, p. 283–291, 2010.

7 - Solving Eigenvalue Transport Problems with Negative Weights and Regional Cancellation

This chapter has previously appeared as:

H. Belanger, D. Mancusi, and A. Zoia, "Solving Eigenvalue Transport Problems with Negative Weights and Regional Cancellation," In *Proceedings of the The International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering 2021 (M&C 2021)*, October 2021, p. 46-55.

SOLVING EIGENVALUE TRANSPORT PROBLEMS WITH NEGATIVE WEIGHTS AND REGIONAL CANCELLATION

Hunter Belanger, Davide Mancusi, and Andrea Zoia

Université Paris-Saclay, CEA, Service d'Études des Réacteurs et de Mathématiques Appliquées,
91191, Gif-sur-Yvette, France

hunter.belanger@cea.fr, davide.mancusi@cea.fr, andrea.zoia@cea.fr

ABSTRACT

There are several lesser-known variants of delta tracking which rely on the use of negative particle weights to transport particles, without necessarily knowing the majorant cross section of the system. To the authors' knowledge, these methods have never been applied to k-eigenvalue power iteration problems. This work does just that, and outlines a new problem with the weight normalization of the particles which arises when the particle weights are allowed to switch sign. While the net weight of the particles in the system will remain stable using traditional population control techniques, the total weight magnitude of the system will increase without bound, overwhelming computer memory, and requiring more time to complete each generation of the simulation. Two systems are examined: a 1D single-speed rod, with a spatially varying cross section, and the 3D multigroup C5G7 benchmark. Weight cancellation techniques previously developed for calculation of the second harmonic of the Boltzmann transport equation have been used to successfully control the particle population. In the 1D problem, an exact regional cancellation method previously proposed by Booth and Gubernatis is used, while in the 3D problem, an approximate technique previously outlined by Zhang et al. is implemented. Both methodologies are shown to adequately control the particle population, allowing power iteration simulations to be conducted.

KEYWORDS: Monte Carlo, delta tracking, negative weights, cancellation, eigenvalue

1. INTRODUCTION

Negative weights have been used in Monte Carlo particle simulations for a large variety of reasons. One such use of negative weights has been to study the effects of neutron noise [1]. Methods developed by Booth and Gubernatis make use of negative weights to obtain the second harmonic of the Boltzmann transport equation [2,3]. A variation on the standard delta tracking [4] was proposed by Carter et al. in the early 1970s, to solve transport problems with continuously-varying cross sections [5]. More recently, Légrády et al. have generalized Carter et al.'s tracking algorithm by embedding it into a broader framework [6,7]. These methods make use of a sampling cross section $\Sigma_{\text{smp}}(P)$, which is a function of the phase-space coordinates $(\mathbf{r}, \hat{\Omega}, E)$, denoted as P . The sampling cross section is used to determine the flight distance of a particle to a tentative collision site. At a tentative collision site, a real collision is performed with probability $q(P)$, and a virtual

collision is performed with probability $1 - q(P)$. This quantity, $q(P)$ may be defined to be an arbitrary function of the phase-space coordinates, so long as it remains a valid probability over the domain of P (i.e. $0 \leq q(P) \leq 1 \quad \forall P$). In the event of a real collision, the particle weight (w) is multiplied by $\Sigma_t(P') / [\Sigma_{\text{smp}}(P)q(P)]$, with $\Sigma_t(P')$ being the true cross section at the particle's new position. For a virtual collision, the weight is multiplied by $[1 - \Sigma_t(P') / \Sigma_{\text{smp}}(P)] / [1 - q(P)]$ [7]. One may note here, that for a virtual collision where $\Sigma_t(P') < \Sigma_{\text{smp}}(P)$, the particle weight will be multiplied by a negative factor, changing its sign. Therefore, depending on the choice of $\Sigma_{\text{smp}}(P)$, particles may switch sign during the random walk process.

There is a marked difference between the use of negative weights for obtaining the second harmonic, and the use of negative weights in the two delta tracking-like methods. In the former, a particle is either born as having a positive or negative weight, and the sign cannot change, and is also kept for all of the particle's progeny as well [2]. In the latter two, a particle's weight may change sign at a virtual collision. These methods however permit one to use an arbitrary cross section (Σ_{smp}) for sampling flight distances. This leniency makes the methods proposed by Légrády et al. and Carter et al. very attractive for situations where it could be difficult to determine the majorant cross section, such as in the context of spatially continuously varying cross section. In such cases, it could be simple to look-up the exact total cross section at a given location and energy, but ensuring a majorant might be difficult (in the case of polynomial isotopic densities for example).

Belanger et al. have previously examined these two methodologies, along with several others, using a simple 1D-rod, single speed test system, with different cross section shapes, as a function of position within the rod [8]. Only fixed source problems were considered, and an examination of power iteration problems was left for a future work. This paper has expanded on upon those results, adding fission and power iteration capabilities to their original code, to solve k_{eff} eigenvalue problems. When sampling the number of new fission neutrons, the algorithm used by OpenMC and MCNP was implemented [9,10], with a slight modification to accommodate negative weights. The number n of generated fission neutrons for any given collision is

$$n = \left\lfloor |w| \cdot \frac{\nu \Sigma_f}{\Sigma_t} \cdot \frac{1}{k_{\text{eff}}^{(G-1)}} + \xi \right\rfloor, \quad (1)$$

where w is the particle weight, $k_{\text{eff}}^{(G-1)}$ is the value of the effective multiplication factor calculated for the previous generation, and ξ is a random variable sampled from $\mathcal{U}(0, 1)$. Each of the n new neutrons born at this fission site starts with weight 1 for the case of $w > 0$, or -1 for the case of $w < 0$. This leads to the next generation starting with N_+ positive particles, and N_- negative particles. Using this straightforward extension of this population control scheme for fission neutrons, it was observed that the net number of particles will remain stable ($N_{\text{net}} = N_+ - N_-$), while the total number of particles to be transported ($N_{\text{tot}} = N_+ + N_-$) will increase without bound, increasing simulation times and eventually overwhelming the computer memory.

This issue has, to our knowledge, never been previously reported. Attempts were made to use other population control methods, such as weight combing [11], but it was soon realized that it is not possible to solve this conundrum with such a methodology. The only viable strategy is to perform a cancellation of the negative and positive weights [3]. Similar problems arise in the context of determining higher harmonics of the transport equation using Monte Carlo methods, as discussed by Booth [2], where a point-wise cancellation scheme is suggested. Booth and Gubernatis have

also proposed an exact regional cancellation method [3]. This paper applies the exact regional cancellation method to the aforementioned extension to the systems examined by Belanger et al. [8], to remedy the problem of population normalization when conducting transport simulations with methods which involve both negative and positive weights. Approximate regional cancellation, as outlined by Zhang et al. [12], is then be used to address the same issue in a 3D test problem (the C5G7 benchmark [13]).

2. TRANSPORT WITH NEGATIVE PARTICLES

We can formalize transport problems involving negative particles by writing two coupled transport equations; one describes the angular flux of the positive particles (φ_+), and another describes the angular flux of the negative particles (φ_-). The transport equations are provided in Equation 2:

$$\begin{aligned} \hat{\Omega} \cdot \nabla \varphi_{\pm}(P) + \Sigma_{\text{smp}}(P) \varphi_{\pm}(P) = & \\ & \int \varphi_{\pm}(P') \left[\Sigma_s(P' \rightarrow P) + \frac{\chi(E) \nu(P')}{4\pi k_{\text{eff}}} \Sigma_f(P') \right] \delta(\mathbf{r}' - \mathbf{r}) dP' \\ & + \varphi_{\pm}(P) \Theta(\Sigma_{\text{smp}}(P) - \Sigma_t(P)) [\Sigma_{\text{smp}}(P) - \Sigma_t(P)] \\ & + \varphi_{\mp}(P) \Theta(\Sigma_t(P) - \Sigma_{\text{smp}}(P)) [\Sigma_t(P) - \Sigma_{\text{smp}}(P)], \quad (2) \end{aligned}$$

where $\Theta(x)$ is the Heaviside step function. For the sake of conciseness, we omit the derivation of these equations; it must be noted that these equations are equally valid for both the negative-weighted delta tracking as presented by Légrády et al. [6], and the method presented by Carter et al. [5]. The system of equations represented by Equation 2 may be re-written as an eigenvalue equation of the form

$$\hat{A} \begin{bmatrix} \varphi_+(P) \\ \varphi_-(P) \end{bmatrix} = \frac{1}{k_{\text{eff}}} \hat{F} \begin{bmatrix} \varphi_+(P) \\ \varphi_-(P) \end{bmatrix}. \quad (3)$$

From Equation 2, it is determined that the operators are symmetric under exchange of φ_+ and φ_- , therefore the eigenstates must have either odd or even parity, and be of the form

$$\eta_o(P) \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \text{ and } \eta_e(P) \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \quad (4)$$

The odd eigenstates correspond to the spectrum of the true Boltzmann transport equation, which is dominated by the k_{eff} eigenvalue. The even eigenstates correspond to a different equation, and are associated with a different spectrum of eigenvalues, with a dominant that we denote as k_{tot} .

Thus, the introduction of negative weights manifests itself with the appearance of an additional set of eigenvalues. Unfortunately, the new k_{tot} eigenvalue dominates the effective multiplication factor, k_{eff} , which can no longer be determined by power iteration. The power iteration converges towards a symmetric state, characterized by the same number of positive and negative particles. Since $k_{\text{tot}} > k_{\text{eff}}$, the population control of Equation 1 will not stabilize the total number of neutrons, which will grow without bound. Note that applying population control with k_{tot} instead will still lead the power iteration to converge to the wrong eigenstate. The only solution is to annihilate positive and negative weights.

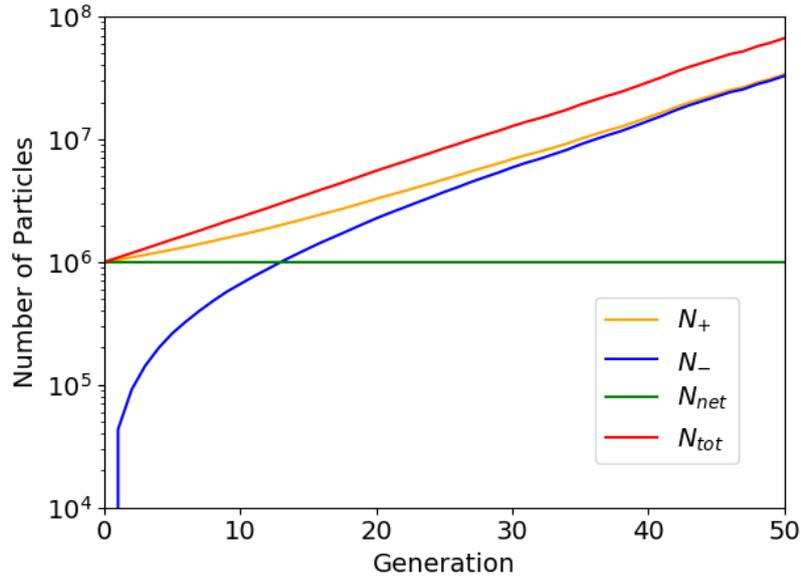


Figure 1: Sizes of the populations in the 1D rod, without weight cancellation, for each generation.

3. EXACT REGIONAL CANCELLATION: 1D BENCHMARK

The first system to be analyzed is a 1D rod, single speed problem, taken from Belanger et al. where the total cross section is a function of position within the rod [8]. As a general representative case, the Broad Gaussian cross section profile has been used, where the cross section is given as

$$\Sigma_t(x) = \sqrt{\frac{2}{\pi}} \exp\left[-(x - 1.23)^2\right] + 0.1 \quad \forall x \in [0, 2], \quad (5)$$

and leakage boundary conditions are applied at $x = 0$ and $x = 2$. While the total interaction probability varies spatially, the reaction channel probabilities have been kept constant throughout the rod, with the absorption probability being $\Sigma_a/\Sigma_t = 0.3$, and fission probability being $\Sigma_f/\Sigma_t = 0.2$. The number of neutrons born per fission is $\nu = 2.5$, and scattering is isotropic (in a 1D system, this means a particle may either scatter forwards, or backwards, both with equal probability). The tracking method of Carter et al. was employed, which implies $q = \Sigma_t/(\Sigma_t + |\Sigma_{\text{smp}} - \Sigma_t|)$ [7]. The sampling cross section was chosen to be a constant value of $\Sigma_{\text{smp}} = 0.85\Sigma_t(x = 1.23)$. This choice of sampling cross section allows positive particles to become negative when they have a virtual collision within the interval $[x \approx 0.8, x \approx 1.66]$. Since this tracking method allows particle weight magnitudes to increase, the variance reduction techniques of roulette and splitting were utilized, on both positive and negative particles. Roulette was performed when $|w| < 0.6$, and the survival weight was $w = \pm 1$, ensuring that the particle weight kept its sign. Particles were split if $|w| \geq 2$. The reference value for this system, obtained using standard delta tracking, is $k_{\text{eff}} = 0.33573 \pm 0.00002$.

These simulation parameters, in conjunction with the production of new fission neutrons described

by Equation 1, were then used to conduct a power-iteration k_{eff} eigenvalue simulation using the tracking method of Carter et al.. The particle populations as a function of generation are presented in Figure 1. Immediately apparent is the exponential growth of the N_+ , N_- , and N_{tot} populations, while N_{net} remains stable. This leads to an increase in computation time, and quickly exhausts the computer's memory, as more particles need to be transported.

In order to solve this problem, the exact weight cancellation algorithm proposed by Booth and Gubernatis was implemented [3]. Despite its apparent spatial discretization, this method performs the cancellation of weights amongst all particles within a spatial region in a manner which is exact. This is done by determining the uniformly distributed portion for each particle which is born in the region, and distributing this uniform weight equally amongst all the particles in the region. To implement this algorithm, the minimum expected fission density in the region for each particle which does fission in the region must be calculated. Consider particle p , which is born in region R at x_p , and whose parent, p' , previously came from $x_{p'} \notin R$. The quantity required is the minimum of the expected fission density of particle p' , within region R , which we shall denote as β . In the 1D case, with the previously described Gaussian cross section, this minimum fission density will always occur at one of the boundaries of R , x_{R_1} or x_{R_2} :

$$\beta = \min \left[\Sigma_f(x_{R_1}) \exp(-\Sigma_{\text{smp}}|x_{R_1} - x_{p'}|), \Sigma_f(x_{R_2}) \exp(-\Sigma_{\text{smp}}|x_{R_2} - x_{p'}|) \right]. \quad (6)$$

The true fission density corresponding to the birth of particle p is then found, replacing x_R with x_p in the previous equation:

$$f = \Sigma_f(x_p) \exp(-\Sigma_{\text{smp}}|x_p - x_{p'}|). \quad (7)$$

This methodology ensures that $\beta < f$ always. With these two quantities, it is possible to determine what portion of p may be uniformly distributed within the region. If p has a weight w , then the uniform portion w_u and static portion w_s are

$$w_u = w \frac{\beta}{f} \quad w_s = w \frac{f - \beta}{f}. \quad (8)$$

The weight of particle p is now set to w_s , and the sum of all w_u for all particles in the region is then tallied in a variable we shall call U_R . Once all new particles in the region have been treated, the uniform weight for the region, U_R , must be distributed. This is accomplished by sampling $\zeta = \lceil U_R \rceil$ new particles, which will be uniformly placed within the region, each having a weight $w = U_R/\zeta$. It is reiterated that this approach to exact regional cancellation is only valid when $x_{p'}$ is not in the same region as x_p . This is because the probability of a particle flying from $x_{p'}$ to x_p and having an interaction is zero, thus making the minimum fission density in the region zero (i.e. $\beta = 0$).

The previously outlined system was again simulated, this time with exact regional cancellation, using 10 equally spaced regions over the problem domain. Figure 2a presents the population sizes with this exact regional cancellation. An eigenvalue of $k_{\text{eff}} = 0.33575 \pm 0.00002$ was obtained using this transport method, in agreement with the that obtained by delta tracking. Initially, there are no negative particles, but after only two generations, the number of negative particles has increased to its equilibrium value near 1100, and oscillates near this value in a stable manner for the duration of the simulation. Figure 2b shows an enlarged portion of Figure 2a, showing how N_+ increased by the same amount as N_- . N_{tot} must therefore increase by twice the amount that N_+

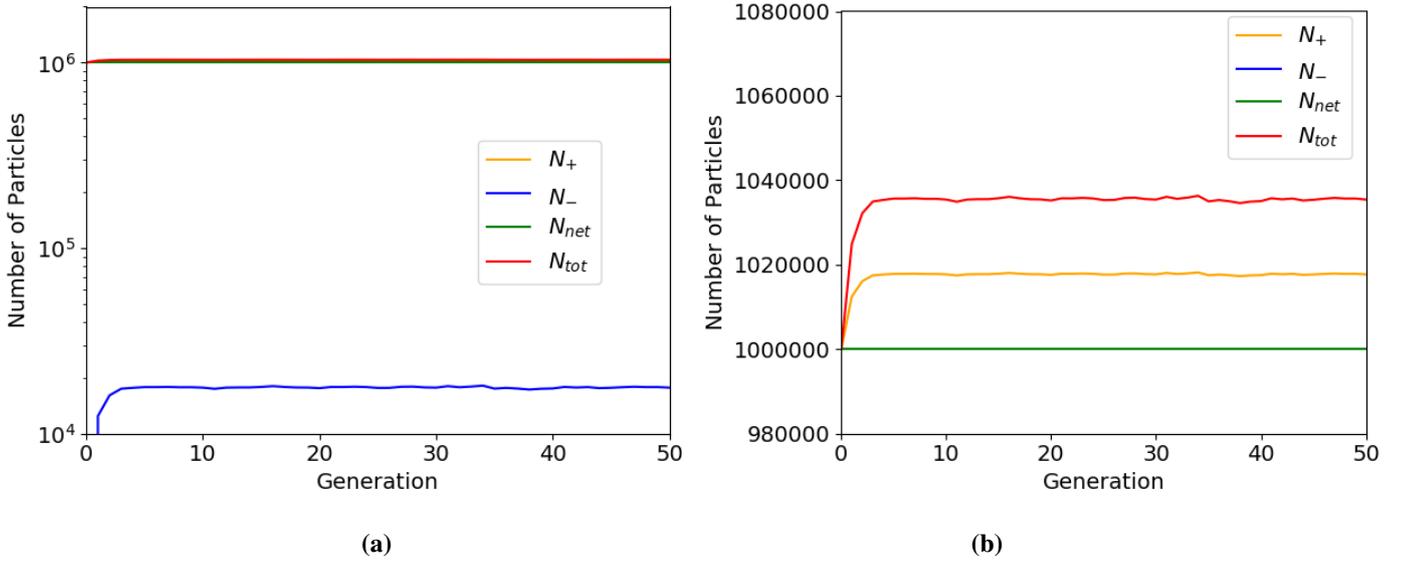


Figure 2: Sizes of the populations in the 1D rod, with exact regional weight cancellation, for each generation.

and N_- increased. While more particles must be transported than initially prescribed, the system is now stable, and a power iteration problem of any number of generations can be conducted. This also demonstrates that weight cancellation is indeed a solution to this problem where the number of particles increases when solving k_{eff} eigenvalue problems using coupled negative and positive particle weights. This exact regional weight cancellation scheme as described by Booth and Gubernatis, and implemented in this paper, is only valid for 1D systems. The reason for this is that 1D systems have the advantage that all points within the cancellation region R are along the line of flight for a particle leaving $x_{p'}$, flying in the direction of x_p . This is of course, not the case in higher dimensions. If one were to consider a flight through a 2D region, the weight cancellation could only occur along the ray through the region, imposed by the particle flight. There is of course a probability of zero that another particle would have exactly the same ray through the region, and the uniform weight portions could then never be canceled with the uniform weight portions of another particle.

4. APPROXIMATE REGIONAL CANCELLATION: C5G7

Performing weight cancellation in 3D, and in an exact manner is a non-trivial task. Recently, works by Zhang et al. have implemented an approximate method to conduct weight cancellation [14,12]. This methodology has proven to work well in their subsequent works, obtaining higher harmonics of the transport equation for a full 3D reactor core benchmark [15]. The method works by imposing a mesh over the fissile portions of the geometry. After all particles of a generation have been transported, there is a new set of particles which represents the subsequent neutron source after an application of the transport operators. These new particles should all be within the the meshed fissile regions. The particles are then sorted into the mesh bin corresponding to their

positions. Once all particles in a bin have been found, the total weight of the bin is calculated, as the simple sum of the weight of all particles in the bin. From this, the average weight of a particle in the bin is calculated, and then all of the particle weights are changed to this average value.

This is only an approximate approach to weight cancellation. As the cancellation mesh becomes more resolved (with each bin being smaller), this approximation becomes more accurate. Using a finer mesh however can start to reduce the efficiency of the cancellation at a certain point, as the bins can become so small, there may be no other particles in the bin to average with, leading to very little, or no cancellation. A thorough analysis of the optimization and behavior of different meshes is outside the scope of this work, but general characteristics as to the performance and accuracy of fine vs. coarse meshes will be touched upon subsequently.

The C5G7 is a multi-group benchmark, typically used to evaluate deterministic solvers. In 3D, it consists of a 1/8th core, with two UO₂ assemblies, and two MOX assemblies. 7 energy groups are used for the macroscopic cross sections. Reflective boundary conditions are applied on the three boundaries coinciding with the fuel assemblies, and vacuum boundary conditions are used on the three outer surfaces. The reference eigenvalue for the system is $k_{\text{eff}} = 1.183810 \pm 0.000036$ [13]. This system was chosen as it provides a geometry which is rather representative of general reactor physics applications, and allows for a basic investigation of the application of methods to energy dependent cross sections, while remaining easy to implement, change, and verify.

A multigroup Monte Carlo code, with basic support for 3D general geometries was written to conduct all of these tests. For validation of the code, standard delta-track was first implemented, to ensure agreement with the reference values for k_{eff} and normalized pin powers. The tracking method of Carter et al. was then added to the code. In a multi-group representation, each energy group has its own sampling cross section $\Sigma_{g,\text{smp}}$, where $g \in [1, 7]$. For this test, the sampling cross section for groups 2 through 7 was chosen to be the true majorant for those groups. The sampling cross section for group 1 was set to $\Sigma_{1,\text{smp}} = 0.9\Sigma_{1,\text{maj}}$. This particular choice leads to an underestimation of the cross section in all fissile regions of the system, for the first energy group. Under these conditions, neutrons in the 1st group which experience a virtual collision in a fuel region will experience a sign change. Also, once the neutron has scattered down to a higher energy group, it can no longer change sign, as the sampling cross section is the majorant. Any particles with a negative weight are stuck with it, and naturally can only produce negative progeny. Variance reduction methods of Russian roulette, and particle splitting were implemented, utilizing the same parameters as in the previous 1D benchmark.

Figure 3 shows the particle populations at the end of each generation. The observations are very similar to those previously observed for the 1D rod. Asymptotically, the the positive (N_+) and negative (N_-) populations increase exponentially, while maintaining a constant difference. The increase of N_{tot} in this 3D benchmark is much sharper however compared to the 1D rod problem, and it was only possible to run 35 generations in a reasonable amount of time.

To evaluate the accuracy and efficiency of the approximate cancellation method, two different meshes were used to perform the cancellation, one being relatively "fine", with the other being relatively "coarse". For both meshes, a 34x34 grid was defined for the x-y plane, so that each fuel pin had only one x-y bin. The fine mesh used 100 bins along the length of the fuel pins along the z-axis, while the coarse mesh only used 10 bins. The particle populations as a function of the

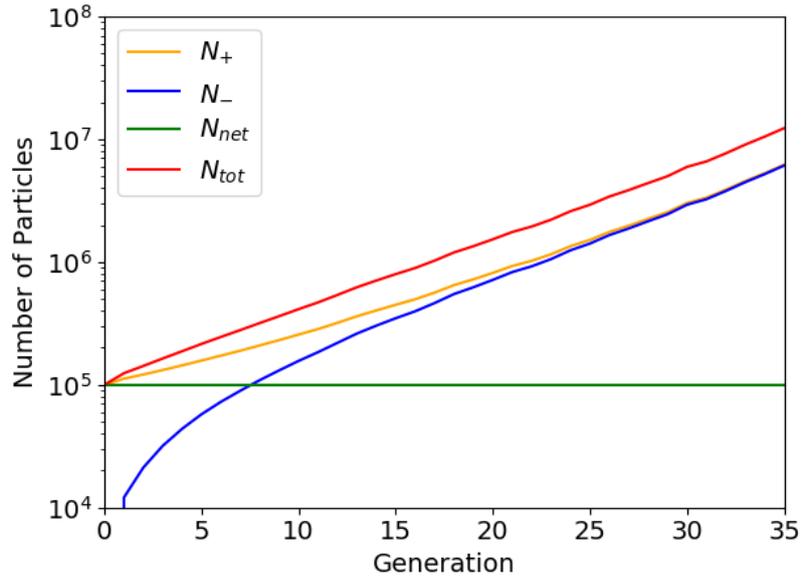


Figure 3: Sizes of the populations in the C5G7 simulation, without weight cancellation, for each generation.

generation for these two mesh configurations are presented in Figure 4. With the coarse mesh, it is observed (in Figure 4a) that as the fission source converges, the cancellation becomes more effective, and by the time 100 generations have been completed, there are only ever two or three negative particles in the system, per generation. For the fine cancellation mesh (in Figure 4b), it is similarly noted that the cancellation becomes more efficient as the source becomes more converged, though not as efficient as with the coarse mesh. Here, there were typically 850 negative particles in the system, per generation. This is to be expected with smaller mesh bins, as there will be fewer particles present in each bin. The fission source required approximately 200 generations to converged under both problems. For both problems, 2200 generations were run, with the first 200 being discarded, and both resulted in an eigenvalue of $k_{\text{eff}} = 1.18378 \pm 0.00002$, in a agreement with the reference eigenvalue for the problem.

5. CONCLUSIONS

This work has established the context in which a Monte Carlo simulation will fail to stabilize, when solving a k-eigenvalue transport problem with positively and negatively weighted particles. As a remedy to these instabilities in one dimension, the concept of exact regional cancellation, as proposed by Booth and Gubernatis, was built upon. Weight cancellation was demonstrated to adequately stabilize the particle populations, and allow the simulation to be conducted in an exact manner. As a solution to the same problem in three dimensions, an approximate weight cancellation scheme was used, where a Cartesian mesh is imposed over the fissile domain, and the average particle weight within a mesh bin is assigned to all particle in that bin. This methodology was applied to the C5G7 multigroup benchmark, as a proof of concept. While not exact, both

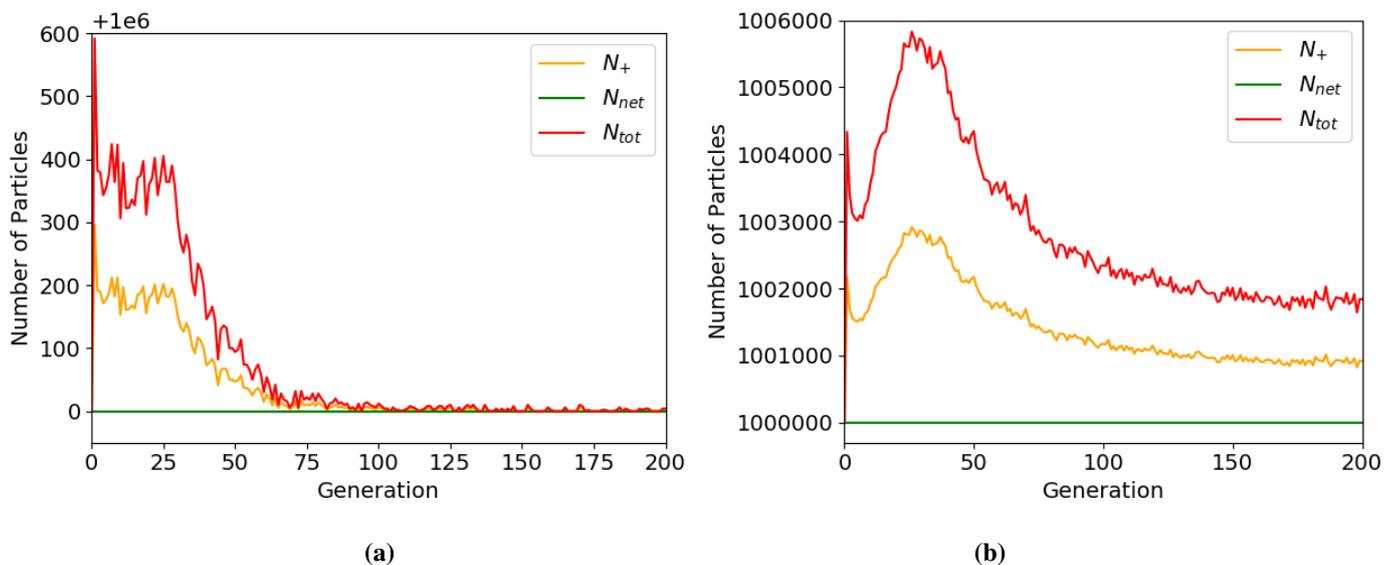


Figure 4: Sizes of the populations in the C5G7 simulation, with weight cancellation. Figure a was produced using a coarse cancellation mesh, while Figure b was produced with a fine cancellation mesh.

meshes produced results in good agreement with the accepted eigenvalue for the problem. The approach was shown to be quite efficient at canceling positive and negative weights, with both a coarse and fine meshes resulting in stable simulations. More work must be done to further examine the properties of such systems, such as the effects of using different choices of Σ_{smp} , and different cancellation meshes. For the 3D case, exact methods of weight cancellation should also be investigated, as an alternative to the approximate one used in this paper.

REFERENCES

- [1] T. Yamamoto. “Monte Carlo method with complex-valued weights for frequency domain analyses of neutron noise.” *Annals of Nuclear Energy*, **volume 58**, p. 72–79 (2013).
- [2] T. E. Booth. “Computing the Higher k-Eigenfunctions by Monte Carlo Power Iteration: A Conjecture.” *Nuclear Science and Engineering*, **volume 143**(3), pp. 291–300 (2003).
- [3] T. E. Booth and J. E. Gubernatis. “Exact Regional Monte Carlo Weight Cancellation for Second Eigenfunction Calculations.” *Nuclear Science and Engineering*, **volume 165**(3), pp. 283–291 (2010).
- [4] E. Woodcock, T. Murphy, P. Hemmings, and S. Longworth. “Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry.” (1965). ANL-7050.
- [5] L. L. Carter, E. D. Cashwell, and W. M. Taylor. “Monte Carlo Sampling with Continuously Varying Cross Sections Along Flight Paths.” *Nuclear Science and Engineering*, **volume 48**(4), pp. 403–411 (1972).

- [6] D. Legrady, B. Molnar, M. Klausz, and T. Major. “Woodcock tracking with arbitrary sampling cross section using negative weights.” *Annals of Nuclear Energy*, **volume 102**, pp. 116–123 (2017).
- [7] B. Molnar, G. Tolnai, and D. Legrady. “Variance Reduction and Optimization Strategies in a Biased Woodcock Particle Tracking Framework.” *Nucl Sci Eng*, **volume 190**(1), pp. 1–17 (2018).
- [8] H. Belanger, D. Mancusi, and A. Zoia. “Review of Monte Carlo methods for particle transport in continuously-varying media.” *European Physical Journal Plus* (2020). Accepted.
- [9] P. K. Romano, N. E. Horelik, B. R. Herman, A. G. Nelson, B. Forget, and K. Smith. “OpenMC: A state-of-the-art Monte Carlo code for research and development.” *Annals of Nuclear Energy*, **volume 82**, pp. 90 – 97 (2015).
- [10] T. Goorley, M. James, T. Booth, F. Brown, J. Bull, L. J. Cox, J. Durkee, J. Elson, M. Fensin, R. A. Forster, J. Hendricks, H. G. Hughes, R. Johns, B. Kiedrowski, R. Martz, S. Mashnik, G. McKinney, D. Pelowitz, R. Prael, J. Sweezy, L. Waters, T. Wilcox, and T. Zukaitis. “Initial MCNP6 Release Overview.” *Nuclear Technology*, **volume 180**(3), pp. 298–315 (2012).
- [11] T. E. Booth. “A weight (charge) conserving importance-weighted comb for Monte Carlo.” (1996). LA-UR 96-0051.
- [12] P. Zhang, H. Lee, and D. Lee. “A general solution strategy of modified power method for higher mode solutions.” *Journal of Computational Physics*, **volume 305**, pp. 387–402 (2016).
- [13] E. E. Lewis, M. A. Smith, N. Tsoufanidis, G. Palmiotti, T. A. Taiwo, and R. N. Blomquist. “Benchmark specification for Deterministic 2-D/3-D MOX fuel assembly transport calculations without spatial homogenisation (C5G7 MOX).” (2001).
- [14] P. Zhang, H. Lee, and D. Lee. “Stabilization technique of modified power iteration method for Monte Carlo simulation of neutron transport eigenvalue problem.” In *ANS Mathematics & Computation Topical Meeting* (2015).
- [15] P. Zhang, H. Lee, M. Lemaire, C. Kong, J. Choe, J. Yu, F. Khoshahval, and D. Lee. “Practical Monte Carlo simulation using modified power method with preconditioning.” *Annals of Nuclear Energy*, **volume 127**, p. 372–384 (2019).

8 - Exact Weight Cancellation in Monte Carlo Eigenvalue Transport Problems

This chapter has previously appeared as:

H. Belanger, D. Mancusi, and A. Zoia, "Exact weight cancellation in Monte Carlo eigenvalue transport problems," *Physical Review E*, vol. 104, no. 1, p. 015306, 2021, doi: 10.1103/physreve.104.015306.

The preprint version appearing in this manuscript can be found at <https://arxiv.org/abs/2103.13891>.

Exact weight cancellation in Monte Carlo eigenvalue transport problems

Hunter Belanger,^{*} Davide Mancusi,[†] and Andrea Zoia[‡]

Université Paris-Saclay, CEA, Service d'Études des Réacteurs et de Mathématiques Appliquées, 91191, Gif-sur-Yvette, France

Random walks are frequently used as a model for very diverse physical phenomena. The Monte Carlo method is a versatile tool for the study of the properties of systems modelled as random walks. Often, each walker is associated with a statistical weight, used in the estimation of observable quantities. Weights are typically assumed to be positive; nonetheless, some applications require the use of positive and negative weights or complex weights, and often pose particular challenges with convergence. In this paper, we examine such a case from the field of nuclear reactor physics, where the negative particle weights prevent the power iteration algorithm from converging on the sought fundamental eigenstate of the Boltzmann transport equation. We demonstrate how the use of weight cancellation allows convergence on the physical eigenstate. To this end, we develop a novel method to perform weight cancellation in an exact manner, in three spatial dimensions. The viability of this algorithm is then demonstrated on a reactor physics problem.

I. INTRODUCTION

Many physical processes can be represented by the random movement of particles, or “walkers”, through phase space. Such phenomena include radiation transport, propagation of active molecules in living bodies, or the spread of epidemics [1–4]. Quite often, it is desirable to compute the properties of such systems with the aid of Monte Carlo simulations. This is certainly the case in neutron transport, where Monte Carlo techniques are used to sample the random walk process and thus solve the Boltzmann neutron transport equation [5]. In this context, the Monte Carlo method may be preferred to (faster) deterministic methods because it requires very few (if any) approximations: Monte Carlo simulations are typically used to obtain reference solutions to which deterministic solutions may be compared.

The random walkers have an associated statistical “weight”, which is typically positive and real. This weight is used to estimate the observable quantities which are sought in the simulation. Certain forms of transport problems, however, require the use of positive and negative weights, or even complex weights. Some examples include diffusion quantum Monte Carlo [6], the solution of neutron noise equations in the frequency domain [7], the determination of the second harmonic of the Boltzmann eigenvalue equation [8], and the determination of transmittance in graphics rendering [9]. Monte Carlo simulations that use negative or complex weights are notoriously difficult, as they often do not converge to the desired solution [6, 8], or have a very high variance in the observed quantities [7]. Most literature recognizes that applying weight cancellation, where walkers carrying positive and negative weights may annihilate with one another, is highly beneficial for the solution of these problems [6–8, 10, 11].

Several methods for performing weight cancellation have been proposed. One method, stemming from neutron transport, achieves weight cancellation by pairing walkers individually (as opposed to pooling weights of walkers within a defined region) [8]. While in theory this algorithm is exact (i.e. it does not introduce any approximation), it is difficult to implement, and has a quadratic computational complexity in the number of random walkers being simulated. Another linear, exact regional cancellation algorithm has been proposed [11], but is only valid in 1D geometries. One alternative approach consists of using an approximate technique of averaging the weight of all walkers within the same region [7]. This is easily applied to 3D, and has linear complexity, but does not provide an exact solution. Finally, it must be mentioned that not all random walker problems are posed in a manner which makes cancellation difficult. In the full configuration interaction quantum Monte Carlo technique, for example, cancellation is straightforward, as walkers explore a discrete state space, as opposed to a continuous state space; when two walkers of different signs land in the same state, they may immediately annihilate [12].

In this paper, we review some of the existing weight cancellation methods and propose a novel exact 3D regional cancellation algorithm, that has linear complexity in the number of walkers, and that can be easily generalized to any number of dimensions. The development of this method has been motivated by certain neutron transport problems that require the use of negative walker weights in conjunction with the power iteration technique (for obtaining the dominant eigenstate), and that fail to converge to the correct solution without weight cancellation; a simplified example of such a system is presented in Section II. Subsequently, in Section III, we will develop a mathematical model to explain why the power iteration fails in the presence of negative weights, and why weight cancellation can resolve this problem. We will demonstrate how the existing 1D regional cancellation algorithm allows successful convergence of our 1D power iteration problem in Section IV, and we will develop our novel, 3D version of the cancellation algorithm

^{*} hunter.belanger@cea.fr

[†] davide.mancusi@cea.fr

[‡] andrea.zoia@cea.fr

in Section V, testing it on a reactor physics benchmark in Section VI. Finally, we will present our conclusions Section VII.

II. POWER ITERATION DEBACLE

In neutron transport, a main quantity of interest is the fundamental eigenpair of the Boltzmann k -eigenvalue equation for neutrons, which we write here in its one-speed form, as a function of position \mathbf{r} and direction $\hat{\Omega}$:

$$\hat{\Omega} \cdot \nabla \varphi_k(\mathbf{r}, \hat{\Omega}) + \Sigma_t(\mathbf{r})\varphi_k(\mathbf{r}, \hat{\Omega}) = \int_{4\pi} \Sigma_s(\mathbf{r}, \hat{\Omega}' \rightarrow \hat{\Omega})\varphi_k(\mathbf{r}, \hat{\Omega}')d\hat{\Omega}' + \frac{\nu(\mathbf{r})\Sigma_f(\mathbf{r})}{4\pi k} \int_{4\pi} \varphi_k(\mathbf{r}, \hat{\Omega}')d\hat{\Omega}'. \quad (1)$$

Here φ_k is the eigenfunction, k is the eigenvalue, Σ_t is the total cross section, Σ_s is the scattering cross section, Σ_f is the fission cross section, and ν is the average number of new particles produced per fission. While it does not explicitly appear in Eq. (1), there is an implicitly defined capture cross section $\Sigma_c = \Sigma_t - \Sigma_s - \Sigma_f$, which results in the death of neutrons. In general, Eq. (1) admits several eigenvalue-eigenfunction pairs; the fundamental (largest) k -eigenvalue is the multiplication factor, which is typically written as k_{eff} . The corresponding eigenfunction φ_0 is known as the angular neutron flux, and it represents the average number of neutrons crossing a unit surface area per unit time [5].

In order to obtain k_{eff} and φ_0 , a numerical technique known as power iteration is often employed [5, 13]. Although Eq. (1) is not written in such a form, we may assume that it represents an eigenvalue problem of the type $Lv_0 = \lambda_0 v_0$, with v_0 being the fundamental eigenstate, and λ_0 being the fundamental eigenvalue. In relation to Monte Carlo transport problems, the operator L can be interpreted as the propagation of particles through a system from one fission event to another. With power iteration, one may obtain the fundamental state v_0 from any state b such that the inner product $\langle b, v_0 \rangle \neq 0$, by repeated application of the operator L :

$$\lim_{n \rightarrow \infty} \frac{L^n b}{|L^n b|} = v_0. \quad (2)$$

Therefore, starting from almost any initial state, it is possible to converge to the fundamental mode [13].

We wish to use Monte Carlo to yield a solution to Eq. (1), in the sense that the average density of walkers in phase space should satisfy Eq. (1), and represent the fundamental eigenmode. In the context of Monte Carlo, the idea of the power iteration technique needs to be adapted as follows. Starting from an arbitrary set of walkers, we sample random walks for all walkers until their death. Along the walk, a walker may be randomly killed if the magnitude of its weight becomes too low, or

split if the magnitude becomes too large [14]. During the random walks of this first generation, new particles will be born from fission. These fission particles are stored in a bank and are attributed to the second generation. Once the first generation has finished, the banked particles undergo population control, where those with small weights may randomly be killed, those with large weights may be split into multiple particles, and the net weight of all particles is normalised. This is done to keep the net weight of all the particles at the beginning of a generation constant, and to keep the number of particles in the simulation constant on average [14]. This culled and normalized particle bank is subsequently used as the source for the second generation. These particles then undergo the random walk, producing fission particles which will belong to the third generation. This application of the random walk mechanics on a generation of particles may be continued indefinitely. After a number of a generations, the positions (and directions) of the fission particles will settle on an equilibrium distribution, representing the converged fission source for the problem [14]. The combined random walks of all particles from this converged fission source represent the dominant eigenfunction of Eq. (1). With the fission source converged, observable quantities (such as the angular flux and the multiplication factor) may be estimated in each generation. With an estimation of the angular flux and the multiplication factor obtained by each generation, an average may be calculated, although the estimation of the uncertainty is not trivial [15].

Within a generation, the random walk process is sampled as follows. A particle begins with an initial position and direction, which for the first generation may be sampled from a somewhat arbitrary distribution. The first task is to sample the distance the particle will fly before having a collision. In most applications of Monte Carlo simulation to reactor physics problems, the total cross section is usually assumed to be piece-wise constant within each macroscopic geometric region composing the modeled system. To sample a flight distance in a material region with a spatially constant cross section, a random variable $\xi \sim \mathcal{U}[0, 1)$ is drawn, and the cumulative distribution function for the flight distance must be inverted to obtain the distance to collision d_c ¹:

$$\xi = \int_0^{d_c} \Sigma_t \exp(-\Sigma_t s) ds \Rightarrow d_c = -\frac{\ln(1 - \xi)}{\Sigma_t}. \quad (3)$$

The sampled distance is only valid in the given material region. If the distance d_m to the next material boundary is less than d_c , the particle is only moved by d_m , the cross section is updated, and a new distance to collision is sampled. Once at a collision site, the collision mechanics may be simulated; the particle can be captured

¹ It is for this reason that the underlying random walk is called “exponential flight” [16].

with probability Σ_c/Σ_t , whereupon the history is terminated; alternatively, scattering events may occur with probability Σ_s/Σ_t , or the particle may produce new fission particles with probability Σ_f/Σ_t , which will be part of the next generation [14]. The multiplication factor may be estimated in several different manners: here we use the collision estimator. When a particle of weight w has a collision, we add the expected number of fission neutrons produced per collision at that location, namely $w\nu(\mathbf{r})\Sigma_f(\mathbf{r})/\Sigma_t(\mathbf{r})$, to an accumulator which we shall refer to as K . Once the generation has finished, the estimate for the multiplication factor for the generation G is

$$k_{\text{eff}}^{(G)} = \frac{K}{W}, \quad (4)$$

G being the generation number, and W being the sum of the weights of all the particles at the beginning of the generation (which is a constant, because of the normalization procedure described above). The accumulator K must of course be reset to zero before each new generation begins.

A more realistic description of the system would consist of relaxing the hypothesis of piece-wise constant cross sections, especially in view of multi-physics problems [17] where the cross sections for neutron transport depend on complex space-dependent physical feedback mechanisms such as temperature and material density fields. Recently, we examined different methods of sampling the flight distance for the case of spatially varying cross sections [18]. Such methodologies are highly desirable for the next generation of Monte Carlo transport codes, as they could allow for a better representation of the system being simulated, reduce memory requirements, and pair well with the reactor physics community's goals of taking into account a multi-physics approach [19]. In principle, sampling the distance to collision for spatially dependent cross sections would require inverting the following equation for d_c :

$$\xi = \int_0^{d_c} \Sigma_t(s\hat{\Omega} + \mathbf{r}_0) \exp\left(-\int_0^s \Sigma_t(u\hat{\Omega} + \mathbf{r}_0) du\right) ds. \quad (5)$$

In practice, this is quite difficult to accomplish. Our previous work considered several algorithms which could sample flight distances from spatially continuous cross sections, without needing to directly invert Eq. (5). In particular, we examined the traditional delta tracking [20, 21], and the negative weighted delta tracking methods [22, 23].

In order to illustrate how these algorithms work, we revisit a simple one-dimensional transport problem that was considered in our previous paper. In the so-called 'rod model', particles may only move in the forward or backward direction, along a line segment. For our purposes, this line segment has a finite length, and it is possible for particles to leak out of either end of the line. For our application we will set the boundaries of the segment at $x = 0$ and $x = 2$. As an example of a space-dependent

Algorithm 1: Negative weighted delta tracking

```

1 Sample uniform random variable  $\xi_1$ ;
2  $d := -\ln(\xi_1)/\Sigma_{\text{smp}}$ ;
3  $\mathbf{r}_1 := d\hat{\Omega} + \mathbf{r}_0$ ;
4 Sample random variable  $\xi_2$ ;
5 if  $\xi_2 < q$  then
6    $w := w \frac{\Sigma_t(\mathbf{r}_1)}{q\Sigma_{\text{smp}}}$ ;
7   Perform real collision;
8 else
9    $w := w \frac{1 - \frac{\Sigma_t(\mathbf{r}_1)}{\Sigma_{\text{smp}}}}{1 - q}$ ;
10  Virtual collision; goto line 1;
11 end

```

cross section, we will use the broad-Gaussian cross section from our earlier paper [18]. This cross section profile was chosen as it is not monotonic, and better represents the idea that particles may see both an increase and decrease in the cross section along their ray of flight. The chosen $\Sigma_t(x)$ has the form

$$\Sigma_t(x) = \sqrt{\frac{2}{\pi}} e^{-(x-1.23)^2} + 0.1 \quad \forall x \in [0, 2]. \quad (6)$$

The scattering, absorption and fission probabilities are chosen to be spatially constant, having values of $\Sigma_s/\Sigma_t = 0.7$, $\Sigma_c/\Sigma_t = 0.1$, and $\Sigma_f/\Sigma_t = 0.2$. The average number of neutrons born per fission is $\nu = 2.5$; both scattering and fission are isotropic (equal probability of emission forward or backward).

Transport is conducted using the negative weighted delta tracking (NWDT) method [22, 23]. The basic procedures for this method are presented in Alg. 1. It relies on the concept of real and virtual collisions. NWDT requires two parameters: a sampling cross section Σ_{smp} , which is used to sample the distance to a tentative collision site, and a probability q , which is used to determine whether a collision is real or virtual. At a real collision, the mechanics of a collision are used to change the direction of the particle accordingly. A virtual collision does not simulate the collision mechanics; instead, the direction of the particle is left unchanged, and the distance to a new tentative collision site is sampled, where the process begins again. Both Σ_{smp} and q are allowed to be functions of the position. According to the NWDT algorithm, the particle weight changes sign when a virtual collision is sampled, and $\Sigma_t > \Sigma_{\text{smp}}$. Here, we have chosen $\Sigma_{\text{smp}} = 0.85\Sigma_t(x = 1.23)$, which leads to particles changing sign within the region $0.8 \lesssim x \lesssim 1.66$. As for q , we have used $q = \Sigma_t/(\Sigma_t + |\Sigma_{\text{smp}} - \Sigma_t|)$, which corresponds to the strategy proposed by Carter, Cashwell and Taylor [23, 24]. This choice of q was motivated by our previous work, where the method demonstrated very reasonable performance compared to the standard of delta tracking [18].

Delta tracking is a special case of NWDT, where the

sampling cross section Σ_{smp} is taken to be a strict majorant cross section Σ_{maj} , with $\Sigma_{\text{maj}} \geq \Sigma_t$ and the acceptance probability q is taken equal to $\Sigma_t/\Sigma_{\text{maj}}$ [20, 21]². With these choices, particle weights never change sign; however, insisting on the use of delta tracking can raise a number of practical problems, because it could be very challenging to obtain a majorant when using spatially continuous cross sections in a more realistic context. If one were to conduct a simulation where both the isotopic density and the temperature of materials varied spatially, there is, in general, no way to exactly determine the majorant cross section for the system. It is certainly possible to probe the phase space of the problem, testing the cross section at each point. However, if the selected sampling cross section is not in fact a majorant, it will be impossible to handle collisions for which $\Sigma_t > \Sigma_{\text{smp}}$; even if such an event does not occur, the results will be slightly biased. Conversely, if Σ_{smp} is chosen by applying a large safety margin to the maximum known cross section (in an effort to ensure underestimation of the majorant does not occur), the method becomes quite inefficient, as many unnecessary virtual collisions will occur [21]. NWDT tolerates underestimations in the majorant cross section, and also allows one to avoid gross overestimations of the majorant which reduce efficiency.

With NWDT, not only may there be negatively weighted particles in the system, but the weights may also change in magnitude. These two traits can lead to an increase in variance and simulation time. To mitigate these effects, roulette is used on particles with $|w| < 0.6$, with the survival weight being $w = \pm 1$ (ensuring that the particle keeps its initial sign). Particles are also split if $|w| \geq 2$.

In our previous work, we neglected fission. In view of testing NWDT in the framework of power iteration for eigenvalue problems, fission has been added to our model. The number n of new fission neutrons generated at any collision is taken to be

$$n = \left\lfloor |w| \frac{\nu \Sigma_f}{\Sigma_t} \cdot \frac{1}{k_{\text{eff}}^{(G-1)}} + \xi \right\rfloor, \quad (7)$$

where $\xi \sim \mathcal{U}(0,1)$ is a uniform random variable, and $k_{\text{eff}}^{(G-1)}$ is the estimated value of k_{eff} for the previous generation of particles. Without dividing by $k_{\text{eff}}^{(G-1)}$, the number of particles in the simulation would either increase exponentially if the system is super-critical ($k_{\text{eff}} > 1$), or decrease exponentially if the system is sub-critical ($k_{\text{eff}} < 1$). This algorithm closely follows standard methods applied in Monte Carlo codes [28, 29], with slight modifications to accommodate fission with negative particles. All new fission particles are born with a weight

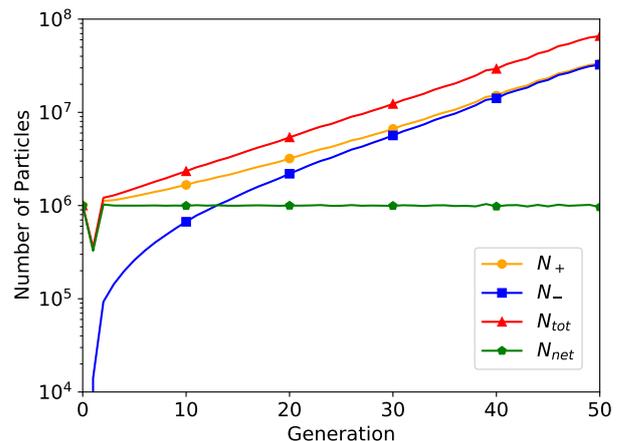


FIG. 1. The number of positive and negative particles per generation, when using the population control scheme provided in Eq. (7).

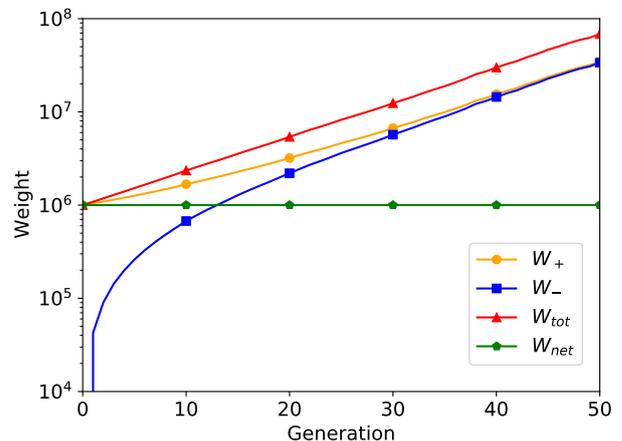


FIG. 2. The weight of positive and negative particles per generation, when using the population control scheme provided in Eq. (7).

$w = \pm 1$, keeping the sign of the weight of the particle inducing fission. Between particle generations, all particle weights are multiplied by a normalization coefficient to ensure that the net weight of the system (sum of all particle weights) is always a constant value.

Figure 1 shows the number of positive and negative particles per generation, when starting with 10^6 particles uniformly distributed within the rod. Also shown is the net number of particles $N_{\text{net}} = N_+ - N_-$, and the total number of particles $N_{\text{tot}} = N_+ + N_-$. Similarly, Figure 2 shows the total positive weight W_+ (the sum of the weights of all positive particles), the total negative weight W_- (the sum of the magnitude of the weights of all negative particles), together with the net weight $W_{\text{net}} = W_+ - W_-$ and the total weight $W_{\text{tot}} = W_+ + W_-$.

² Delta tracking is also sometimes referred to as self-scattering in the electron transport community [25], as thinning [26], or as the null-collisions method [27].

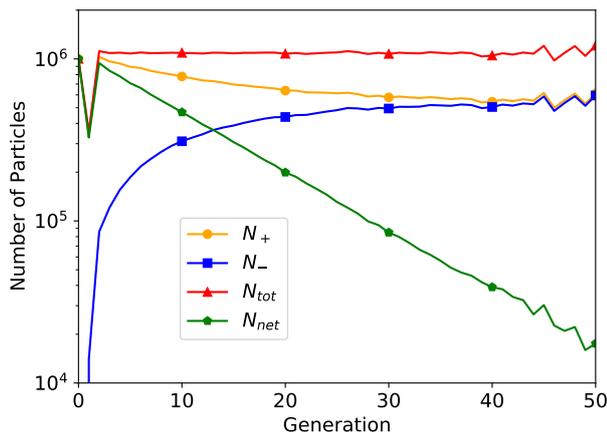


FIG. 3. The number of positive and negative particles per generation, when using the population control scheme provided in Eq. (8).

The number of particles is somewhat less meaningful than the weight, as one can use methods such as weight combing [30] to adjust how many particles are followed in the simulation, but this says nothing about the magnitude of their weight. Weight is the more natural quantity to examine, as its behavior is not controlled by the number of particles. It is instructive, however, to consider the interplay between weight and number of particles. While Eq. (7) keeps the *net* number of particles stable as expected, the *total* number of particles is left to increase exponentially. Eventually, this increase in the number of particles causes each generation to take a longer amount of time to process, and the memory requirements increase as well. At some point, the computer memory is overwhelmed, and the simulation fails. In an attempt to remedy this problem, we modified Eq. (7) by replacing $k_{\text{eff}}^{(G-1)}$; using the value of k_{eff} for the previous generation keeps the *net* number of particles constant on average, so a new quantity k_{tot} was defined, representing the increase in the *total* weight W_{tot} of the simulation. The quantity k_{tot} may be estimated in a similar manner to the multiplication factor, accumulating the collision estimator $|w|\nu(\mathbf{r})\Sigma_f(\mathbf{r})/\Sigma_t(\mathbf{r})$. Using $k_{\text{tot}}^{(G-1)}$ in Eq. (7), we obtain

$$n = \left[|w| \frac{\nu \Sigma_f}{\Sigma_t} \cdot \frac{1}{k_{\text{tot}}^{(G-1)}} + \xi \right]. \quad (8)$$

The effects on the particle populations from using this normalization technique are presented in Figure 3 and Figure 4. While the total number of particles indeed remains constant on average, the net number of particles now decreases exponentially. The net weight of course remains constant because we are normalizing it at the end of the generation. With fewer net particles to maintain the entire initial weight of the system, the magnitude

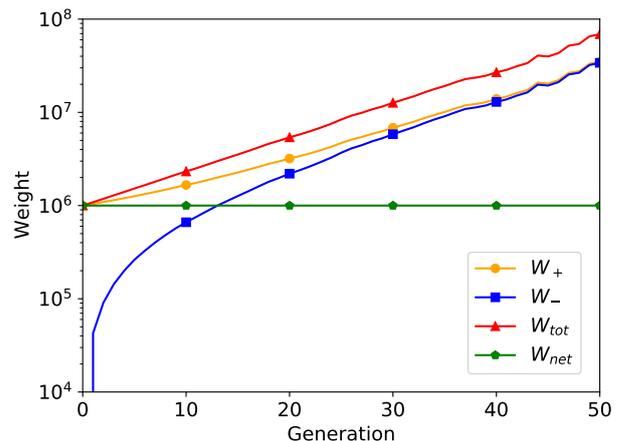


FIG. 4. The weight of positive and negative particles per generation, when using the population control scheme provided in Eq. (8).

of the weight of the particles increases drastically. This is indicated by the total weight of the system continuing to increase exponentially in Figure 4. This leads to near equal quantities of positive and negative particles, each with very large weight magnitudes, causing large fluctuations in the flux estimators, and therefore a larger variance in scores.

Other population control mechanisms were tested as well, hoping that they might stabilize the system. One such method was particle combing [30], conducted between each generation of particles. While combing did reduce the total number of particles going into each generation, the weight magnitude of each particle was quite large. Once they entered the simulation, the particles were split, causing an abrupt increase in memory consumption. After several generations, the simulation is killed by the operating system due to the large memory consumption.

Although the rod model example is admittedly simple, similar results have been obtained when using the NWDT algorithm in combination with power iteration on more realistic reactor physics problems in 3D, and with energy-dependent cross sections (see Sec. VI). It is clear that the power iteration with NWDT cannot be used to estimate the equilibrium distribution of walkers, and therefore the solution to Eq. (1). It is therefore pertinent to better understand why the power iteration is failing, and how the problem might be addressed.

III. NECESSITY OF WEIGHT CANCELLATION

A. The failure of power iteration

The Boltzmann equation presented in Eq. (1) is not capable of describing the interaction between the posi-

tive and negative particle populations in a simulation, only their combined average angular flux. It is fairly straightforward, however, to propose a formulation of the transport equation which is capable of representing both positively and negatively weighted particles. Considering Alg. 1, outlining the NWDT method, we may write a set of coupled transport equations, one for the angular flux φ_+ of the positive particles, and a second for the angular flux φ_- of the negative particles, resulting in the following coupled equations:

$$\begin{aligned} \hat{\Omega} \cdot \nabla \varphi_{\pm} + \Sigma_{\text{smp}} \varphi_{\pm} &= \mathcal{S} \varphi_{\pm} + \frac{1}{k} \mathcal{F} \varphi_{\pm} + \\ \Delta(\Sigma_{\text{smp}} - \Sigma_t) \varphi_{\pm} + \Delta(\Sigma_t - \Sigma_{\text{smp}}) \varphi_{\mp}. \end{aligned} \quad (9)$$

Here, \mathcal{S} is defined as

$$\mathcal{S} \varphi = \int_{4\pi} \Sigma_s(\mathbf{r}, \hat{\Omega}' \rightarrow \hat{\Omega}) \varphi(\mathbf{r}, \hat{\Omega}') d\hat{\Omega}' \quad (10)$$

and \mathcal{F} is defined as

$$\mathcal{F} \varphi = \frac{\nu(\mathbf{r}) \Sigma_f(\mathbf{r})}{4\pi} \int_{4\pi} \varphi(\mathbf{r}, \hat{\Omega}') d\hat{\Omega}'. \quad (11)$$

We have also made use of the function

$$\Delta(x) = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases}. \quad (12)$$

A rigorous derivation of this statement is given in the Appendix. There is a subtlety in this formulation: instead of having one species of particles which may have a weight that is positive or negative, we now have two species (positive and negative), and both are represented by particles having a strictly positive weight. In this framework, our unknown is now the set of eigenfunctions

$$\zeta = \begin{bmatrix} \varphi_+ \\ \varphi_- \end{bmatrix}, \quad (13)$$

which is the solution of the generalized eigenvalue problem

$$\mathbf{A} \zeta = \frac{1}{k} \mathbf{F} \zeta, \quad (14)$$

operator \mathbf{A} being

$$\mathbf{A} = \begin{bmatrix} \mathcal{A}_{11} & \mathcal{A}_{12} \\ \mathcal{A}_{12} & \mathcal{A}_{11} \end{bmatrix}, \quad (15)$$

with

$$\begin{aligned} \mathcal{A}_{11} &= \hat{\Omega} \cdot \nabla + \Sigma_{\text{smp}} - \mathcal{S} - \Delta(\Sigma_{\text{smp}} - \Sigma_t), \\ \mathcal{A}_{12} &= -\Delta(\Sigma_t - \Sigma_{\text{smp}}), \end{aligned} \quad (16)$$

and \mathbf{F} being defined as

$$\mathbf{F} = \begin{bmatrix} \mathcal{F} & 0 \\ 0 & \mathcal{F} \end{bmatrix}. \quad (17)$$

If we use \mathcal{V} to denote the vector space of the physical flux, then ζ is an element of the expanded vector space $\mathcal{V} \times \mathcal{V}$.

Given the previous definitions, the physical flux is interpreted to be

$$\varphi = \varphi_+ - \varphi_- \quad (18)$$

and may be retrieved through the application of $\mathbf{D} : \mathcal{V} \times \mathcal{V} \rightarrow \mathcal{V}$, defined as

$$\mathbf{D} = \begin{bmatrix} I & -I \end{bmatrix} \quad (19)$$

and whose action is

$$\mathbf{D} \begin{bmatrix} f \\ g \end{bmatrix} = f - g. \quad (20)$$

The eigenvalue equation presented in Eq. (14) has several interesting properties, which may be elegantly outlined by the introduction of the parity operator, namely

$$\mathbf{P} = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}. \quad (21)$$

Eq. (14) is invariant under the action of the parity operator: that is, if $\bar{\zeta}$ is a solution of Eq. (14) with eigenvalue \bar{k} , then so is $\mathbf{P}\bar{\zeta}$. It is easily verifiable that $\mathbf{P}\mathbf{A}\mathbf{P} = \mathbf{A}$ and $\mathbf{P}\mathbf{F}\mathbf{P} = \mathbf{F}$, and therefore

$$\mathbf{A}(\mathbf{P}\bar{\zeta}) = \frac{1}{\bar{k}} \mathbf{F}(\mathbf{P}\bar{\zeta}). \quad (22)$$

Thus, if the eigenvalue \bar{k} is non-degenerate, we must have $\mathbf{P}\bar{\zeta} = \pm \bar{\zeta}$. This partitions the eigenstates into two sets, depending on the sign of their eigenvalue with respect to \mathbf{P} . The odd eigenstates have the form

$$\zeta_o = \begin{bmatrix} \varphi \\ -\varphi \end{bmatrix}. \quad (23)$$

By inserting this ansatz into Eq. (14), we can verify that φ must solve the physical eigenvalue equation

$$\hat{\Omega} \cdot \nabla \varphi + \Sigma_t \varphi = \mathcal{S} \varphi + \frac{1}{k} \mathcal{F} \varphi, \quad (24)$$

which is just Eq. (1). We denote the eigenfunctions and eigenvalues of this equation as φ_i and $k_{\varphi,i}$, respectively, with $i \in \{0, 1, \dots\}$. The even eigenstates have the form

$$\zeta_e = \begin{bmatrix} \eta \\ \eta \end{bmatrix}, \quad (25)$$

where η must solve the modified Boltzmann equation

$$\hat{\Omega} \cdot \nabla \eta + \Sigma_{t,\eta} \eta = \mathcal{S} \eta + \frac{1}{k} \mathcal{F} \eta, \quad (26)$$

which is verified by substitution into Eq. (14). Here we have defined

$$\Sigma_{t,\eta} = \Sigma_{\text{smp}} - |\Sigma_{\text{smp}} - \Sigma_t|. \quad (27)$$

We denote the eigenfunctions and eigenvalues of Eq. (26) as η_i and $k_{\eta,i}$, respectively, with $i \in \{0, 1, \dots\}$.

Since we always have $\Sigma_{t,\eta} \leq \Sigma_t$, the equation for η has the same scattering and fission terms as Eq. (1), but capture has been decreased, namely

$$\Sigma_{c,\eta} = \Sigma_{t,\eta} - \Sigma_s - \Sigma_f \leq \Sigma_t - \Sigma_s - \Sigma_f. \quad (28)$$

Thus, on physical grounds, we correspondingly expect a larger dominant eigenvalue:

$$k_{\eta,0} \geq k_{\varphi,0} = k_{\text{eff}}. \quad (29)$$

Therefore, the dominant eigenvalue of Eq. (14) is not k_{eff} , but the nonphysical eigenvalue $k_{\eta,0}$. Applying the power iteration method to Eq. (14) will result in convergence towards the latter eigenstate. Additionally, the equilibrium distribution is even under the exchange of positive and negative particles, i.e. it contains the same amount of positive and negative particles, and thus zero net particles. Converging to a state which has zero net particles is incompatible with traditional population control mechanisms (such as combing, or normalizing by k_{eff}), which are designed to keep the net weight constant, leading to the divergence in particle populations which was observed in Sec. II.

It is worth stressing the similarity between our analysis and the study by Spencer et al. [12] on the origin of the sign problem in full configuration interaction quantum Monte Carlo without weight cancellation.

B. Modeling weight cancellation

Previous investigations have shown that weight cancellation can be very effective in dealing with Monte Carlo problems using particles of both positive and negative weights [6, 8, 12, 31]. In view of these considerations, in the following we will formally address the effect of cancellation on Eq. (14). Weight cancellation can be modeled in the following manner: we start with some function of the phase space variables $(\mathbf{r}, \hat{\Omega})$, i.e. an element of \mathcal{V} . This function may be embedded in $\mathcal{V} \times \mathcal{V}$ by applying a suitable mapping $\mathbf{E} : \mathcal{V} \rightarrow \mathcal{V} \times \mathcal{V}$. There is some latitude in the definition of \mathbf{E} ; one possible definition is

$$\mathbf{E}_1 f = \begin{bmatrix} f \\ 0 \end{bmatrix}, \quad (30)$$

but equally valid choices could be

$$\mathbf{E}_0 f = \begin{bmatrix} 0 \\ -f \end{bmatrix} \quad (31)$$

$$\mathbf{E}_{1/2} f = \begin{bmatrix} f/2 \\ -f/2 \end{bmatrix} \quad (32)$$

$$(33)$$

or the non-linear variant

$$\tilde{\mathbf{E}} f = \begin{bmatrix} \max(f, 0) \\ -\min(f, 0) \end{bmatrix}. \quad (34)$$

The only property that we require of \mathbf{E} is that it should be right-inverse to \mathbf{D} , viz.

$$\mathbf{D}\mathbf{E} = \mathbf{I}. \quad (35)$$

This property expresses the fact that lifting a function from \mathcal{V} into $\mathcal{V} \times \mathcal{V}$ (the action of \mathbf{E}) followed by collapsing back into the space \mathcal{V} of physical fluxes (the action of \mathbf{D}) should not change the function we started with. It is easily verifiable that the previously proposed definitions of \mathbf{E} satisfy this property. In matrix notation, the operators \mathbf{E}_1 , \mathbf{E}_0 , and $\mathbf{E}_{1/2}$ are given by the general formula

$$\mathbf{E}_z = \begin{bmatrix} zI \\ (z-1)I \end{bmatrix}. \quad (36)$$

It is not possible to express $\tilde{\mathbf{E}}$ as a matrix, as it is non-linear.

Let us now consider the operator product with the opposite ordering, $\mathbf{C} = \mathbf{E}\mathbf{D}$. This operator, $\mathbf{C} : \mathcal{V} \times \mathcal{V} \rightarrow \mathcal{V} \times \mathcal{V}$ is a projector, by virtue of Eq. (35):

$$\mathbf{C}^2 = \mathbf{E}\mathbf{D}\mathbf{E}\mathbf{D} = \mathbf{E}(\mathbf{D}\mathbf{E})\mathbf{D} = \mathbf{E}\mathbf{D} = \mathbf{C}. \quad (37)$$

By construction, the null space of \mathbf{C} coincides with the null space of \mathbf{D} , which is the space of even (unphysical) vectors:

$$\mathbf{C} \begin{bmatrix} \eta \\ \eta \end{bmatrix} = 0. \quad (38)$$

Therefore, the operator \mathbf{C} can be regarded as a model for perfect cancellation. We have shown that the even vectors represent the solutions of the nonphysical Boltzmann equation in Eq. (26). This is supported by the explicit form of \mathbf{C} which follows from the previous definitions for \mathbf{E} :

$$\mathbf{C}_z \begin{bmatrix} f_+ \\ f_- \end{bmatrix} = \mathbf{E}_z \mathbf{D} \begin{bmatrix} f_+ \\ f_- \end{bmatrix} = \begin{bmatrix} z(f_+ - f_-) \\ (1-z)(f_+ - f_-) \end{bmatrix} \quad (39)$$

$$\tilde{\mathbf{C}} \begin{bmatrix} f_+ \\ f_- \end{bmatrix} = \tilde{\mathbf{E}} \mathbf{D} \begin{bmatrix} f_+ \\ f_- \end{bmatrix} = \begin{bmatrix} \max(f_+ - f_-, 0) \\ -\min(f_+ - f_-, 0) \end{bmatrix}. \quad (40)$$

Note that only $\tilde{\mathbf{C}}$ can guarantee that both vector components are non-negative, but it requires that both \mathbf{C} and \mathbf{E} be non-linear. In matrix notation, \mathbf{C}_z reads

$$\mathbf{C}_z = \begin{bmatrix} zI & -zI \\ (z-1)I & -(z-1)I \end{bmatrix}. \quad (41)$$

Going back to our eigenvalue equation, Eq. (14) may be modified to include the application of cancellation on the fission source:

$$\mathbf{A}\zeta = \frac{1}{k} \mathbf{C}\mathbf{F}\zeta. \quad (42)$$

If ζ is even, $\mathbf{F}\zeta$ must also be even as $[\mathbf{F}, \mathbf{P}] = 0$. Since \mathbf{C} maps even vectors to 0, the cancellation operator in

Eq. (42) causes the nonphysical eigenmodes corresponding to Eq. (26) to vanish.

At this point, it remains to be shown that cancellation does not perturb the physical eigenvalues of Eq. (14), that were initially associated with the odd eigenstates. To accomplish this, we start with Eq. (42), and apply \mathbf{D} from the left on both sides, as this is the operator which performs the mapping $\mathcal{V} \times \mathcal{V} \rightarrow \mathcal{V}$:

$$\mathbf{D}\mathbf{A}\zeta = \frac{1}{k}\mathbf{D}\mathbf{C}\mathbf{F}\zeta. \quad (43)$$

From Eq. (35), we may substitute $\mathbf{D}\mathbf{C} = \mathbf{D}$. Using Eqs. (13), (15), and (18), this then simplifies to

$$(\mathcal{A}_{11} - \mathcal{A}_{12})\varphi = \frac{1}{k}\mathcal{F}\varphi. \quad (44)$$

Eq. (12) indicates that $\Delta(x) - \Delta(-x) = x$, which, when combined with Eq. (16), yields

$$[\hat{\Omega} \cdot \nabla + \Sigma_t]\varphi = \mathcal{S}\varphi + \frac{1}{k}\mathcal{F}\varphi. \quad (45)$$

This is the physical Boltzmann transport equation. Therefore, cancellation leaves the physical eigenstates unchanged, so long as Eq. (35) holds true, regardless of the form of \mathbf{E} .

In summary, the introduction of cancellation conserves all the (physical and nonphysical) eigenstates of Eq. (9); however, the nonphysical part of the spectrum is relegated to zero, while the physical eigenvalues are unperturbed and dominate the power iteration.

C. Deterministic proof of concept

In order to better understand the effects of cancellation, and how it positively affects the convergence of the power iteration, it is fruitful to visualize the spectrum of the associated equations. For this purpose, we have written a deterministic solver that computes the full spectrum of eigenvalues and eigenfunctions (using the LAPACK library [32]) for a discretized version of the one-dimensional system introduced in Sec. II. A deterministic solver was necessary, since Monte Carlo can only be used to estimate the fundamental mode by power iteration. This solver uses a finite-difference method, with cross section values being taken at the midpoint of each spatial bin; both the positive and negative particle fluxes are explicitly considered.

Before considering cancellation, it is helpful to observe the behaviour of the eigenvalue spectrum of Eq. (14) as a function of $\Sigma_{\text{smp}}/\Sigma_{\text{maj}}$. This quantity roughly measures of the amount of negative weight that is produced by NWDIT; the smaller the ratio is, the more negative weight is introduced. The spectrum is the union of the sets of eigenvalues $k_{\varphi,i}$ and $k_{\eta,i}$ and is presented in Figure 5. Naturally, the spectrum associated with the physical Boltzmann equation does not change with the ratio of $\Sigma_{\text{smp}}/\Sigma_{\text{maj}}$. The eigenvalues of the nonphysical

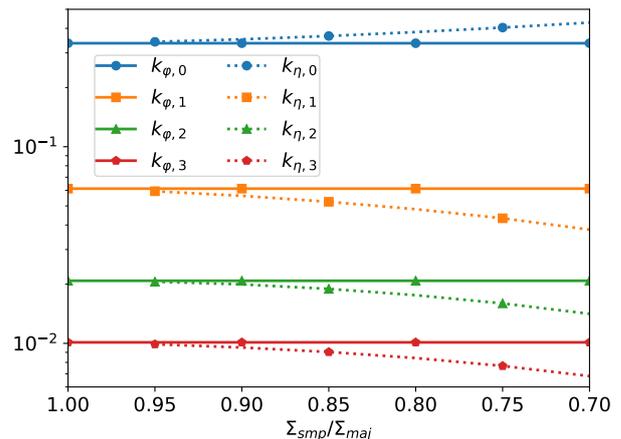


FIG. 5. Behavior of the eigenvalue spectrum as a function of the ratio of the sampling cross section to the majorant cross section. Eigenvalues associated with the physical Boltzmann equation have a solid line, while those associated with the nonphysical Boltzmann equation have a dotted line.

Boltzmann equation diverge from the physical values as this ratio decreases from unity. The dominant nonphysical eigenvalue $k_{\eta,0}$ continually increases above $k_{\varphi,0}$, while the other depicted eigenvalues decrease from their physical counterparts. In Eq. (29), we heuristically demonstrated that $k_{\eta,0} \geq k_{\text{eff}}$, and this is indeed the case in Figure 5. Our heuristics however, are not able to make any remarks as to the behavior of the higher eigenvalues, and while for this system $k_{\eta,i}$ for $i > 0$ always appears to decrease when $\Sigma_{\text{smp}}/\Sigma_{\text{maj}}$ decreases, we have observed other systems which do not exhibit this behavior.

When the cancellation operator \mathbf{C}_1 is added in accordance with Eqs. (41) and (42), we find that cancellation indeed suppresses the even eigenstates, leaving only the odd ones associated with the physical solutions, so long as $\Sigma_{\text{smp}}/\Sigma_{\text{maj}} > 0$. This is the case for a cancellation operator which is “perfect”, in that it is able to conduct cancellation in a manner which always completely neutralizes the negative particle population. While this is of course desired, and possible to implement in a deterministic solver, we will later show in Sec. IV that 100% cancellation efficiency is not necessarily achievable in a Monte Carlo approach. To mimic this fact, we introduce an imperfect cancellation operator $\mathbf{C}^{(\alpha)}$:

$$\mathbf{C}^{(\alpha)} = \alpha\mathbf{C}_1 + (1 - \alpha)\mathbf{I}. \quad (46)$$

Here α represents the cancellation efficiency: when $\alpha = 1$ there is perfect cancellation, and when $\alpha = 0$ there is no cancellation. This is a highly idealized approach to model imperfect cancellation, and is not necessarily a faithful model of the effect of cancellation in a Monte Carlo setting.

Figure 6 presents the effects of cancellation with varying levels of efficiency, for the case of $\Sigma_{\text{smp}}/\Sigma_{\text{maj}} = 0.7$.

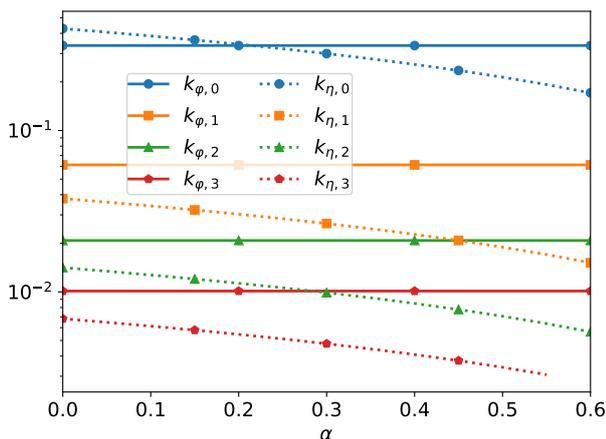


FIG. 6. Behaviour of the eigenvalue spectrum as a function of cancellation efficiency α , for the case of $\Sigma_{\text{smp}}/\Sigma_{\text{maj}} = 0.7$. Eigenvalues associated with the physical Boltzmann equation have a solid line, while those associated with the nonphysical Boltzmann equation have a dotted line.

For values of α between 1 and approximately 0.21, the cancellation operator remains efficient enough that the physical eigenstate is the dominant one. A critical point is reached near $\alpha \approx 0.21$, where the eigenvalues of the physical and nonphysical systems are equal. Further reducing the efficiency of cancellation leads to the nonphysical eigenstate being the dominant one. This example indicates that for our system, there is a *minimum amount of weight cancellation* required in order for power iteration to converge on the fundamental physical eigenstate. This is likely true for all systems which can be described by coupled transport equations for positive and negative particles.

Even in settings where enough cancellation is present to make the $k_{\phi,0}$ mode dominant, cancellation might still be unable to reduce $k_{\eta,0}$ to a level that allows for rapid statistical convergence towards the fundamental mode. To clarify this, we introduce the dominance ratio, which is defined as the ratio of the second-largest eigenvalue to the largest one. The dominance ratio is equal to $k_{\eta,0}/k_{\phi,0}$ for values of α where $k_{\phi,0} > k_{\eta,0} > k_{\phi,1}$. If instead α is such that $k_{\eta,0} < k_{\phi,1}$, then the dominance ratio is $k_{\phi,1}/k_{\phi,0}$, which is the dominance ratio of the physical system. The dominance ratio provides an indication of the rate of convergence of the power iteration. When it is very close to unity, more iterations are required to converge on the dominant eigenvalue. Should α be sufficiently large to ensure $k_{\eta,0} < k_{\phi,1}$, the convergence rate of the problem will no longer be bound by the efficiency of cancellation, but by the physical properties of the system being examined.

This model for cancellation in a deterministic context provides valuable insight as to the behavior of this coupled system of positive and negative particles, and to the possible behavior of implementing cancellation of parti-

cle weights in a Monte Carlo context. It is for this reason that we do not go beyond the provided surface-level analysis of the effects of cancellation efficiency or choice of Σ_{smp} on the convergence of the deterministic model. We will now continue by discussing techniques of weight cancellation in Monte Carlo simulations.

IV. AN EXACT REGIONAL CANCELLATION SCHEME FOR 1D PROBLEMS

As mentioned in the introduction, several cancellation strategies have been proposed in the past; one of these, devised by Booth and Gubernatis, is exact in 1D geometries [11]. This method works by partitioning all fissile domains of the problem domain into regions. Between each generation, weight cancellation amongst all new fission particles born in the region occurs. An important feature of this algorithm is that it has linear computational complexity with the number of particles partaking in cancellation, making it a good candidate for inclusion in general-purpose Monte Carlo transport codes. We have therefore chosen to focus on this algorithm. We will now provide a brief overview of how Booth and Gubernatis' method works, as it is essential for understanding our proposed 3D algorithm, which will be developed in Sec. V.

Booth and Gubernatis make use of a quantity referred to as the fission density [11]. In an effort to reproduce and expand upon their work, we have chosen the following definition for the expected fission density: considering a particle starting a flight at position x_0 and traveling in direction $\mu = \pm 1$, its expected fission density $f(x|x_0, \mu)$ is the expected value of the number of fission events per unit length around x . It can be written as the product of the probability density of flying from x_0 to x and having a collision at position x , and the probability of that collision being in the fission reaction channel. The exact form of $f(x|x_0, \mu)$ depends on the transport methodology being employed. For NWDT, this formula can be deduced by examination of Alg. 1. The probability density of flying from x_0 to x and having a real collision is

$$P_c(x|x_0, \mu) = \begin{cases} q \frac{\Sigma_t(x)}{q\Sigma_{\text{smp}}} \Sigma_{\text{smp}} e^{-\Sigma_{\text{smp}}|x-x_0|} & \frac{x-x_0}{|x-x_0|} = \mu \\ 0 & \frac{x-x_0}{|x-x_0|} \neq \mu \end{cases} \quad (47)$$

The $\Sigma_{\text{smp}} e^{-\Sigma_{\text{smp}}|x-x_0|}$ portion is the probability density of flying from x_0 to x , and having either a real or virtual collision. The factor q is the probability of the collision being real, while $\Sigma_t(x)/(q\Sigma_{\text{smp}})$ is the weight correction factor for real collisions. This must of course be combined with the fission probability

$$P_f(x) = \frac{\Sigma_f(x)}{\Sigma_t(x)}. \quad (48)$$

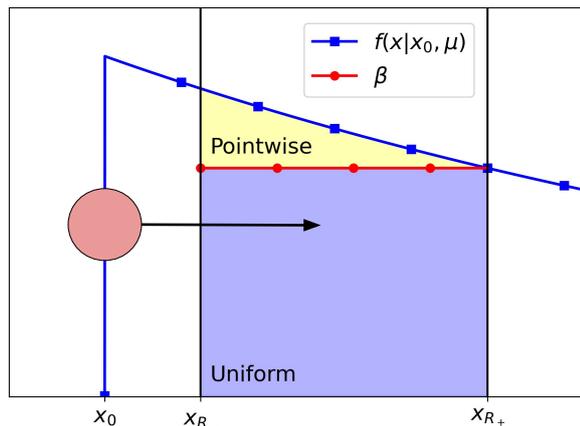


FIG. 7. Presented is a neutron (the colored circle), beginning a flight at x_0 , and flying in the $\mu = 1$ direction (direction of flight is indicated by the arrow). For this flight, the expected fission density, $f(x|x_0, \mu)$ has been plotted as a blue line. The limits of the region are denoted at x_{R-} and x_{R+} . The value of β displayed here is the minimum value of $f(x|x_0, \mu)$ within the region. Note that $f(x|x_0, \mu) = 0$ for positions that cannot be reached by the particle.

As $f(x|x_0, \mu) = P_f(x)P_c(x|x_0, \mu)$, for the case of NWDT we may write

$$f(x|x_0, \mu) = \begin{cases} \Sigma_f(x)e^{-\Sigma_{\text{amp}}(x_0)|x-x_0|} & \frac{x-x_0}{|x-x_0|} = \mu \\ 0 & \frac{x-x_0}{|x-x_0|} \neq \mu \end{cases}, \quad (49)$$

noting that the fission density is zero for all positions which cannot be reached by the particle during the considered flight.

With the expected fission density for a particle having been defined, let us consider a fission particle p belonging to the fission source and ready to start a random walk, at position x_p , located in an arbitrary region R (with bounds x_{R-} and x_{R+}). The parent of this particle (which was a member of the previous fission generation) was previously at position x_0 before flying to x_p , and producing the current particle of interest. Figure 7 illustrates the expected fission density of the parent particle, as well as the bounds of region R , and x_0 . The depicted value of β in the plot is the minimum value of the fission density within the region, for the flight of the parent particle:

$$\beta = \min_{x \in R} f(x|x_0, \mu). \quad (50)$$

The idea of Booth and Gubernatis' method is to estimate the fission density associated with the flight as the sum of a uniform component over the region R and a pointwise component concentrated at the actual fission

site. For a fission particle born in R , which was induced by a particle beginning a flight at x_0 and traveling in the direction μ , the bottom shaded portion in Fig. 7 represents the fraction of the fission density which is uniformly distributed in R , while the top portion represents the non-uniform fraction of the fission density, which depends on the position x_p where the parent lands and induces a fission. This indicates that, for the particle p , the fraction of the fission density concentrated at x_p is $(f(x_p|x_0, \mu) - \beta)/f(x_p|x_0, \mu)$, and the fraction uniformly distributed through region R is $\beta/f(x_p|x_0, \mu)$. As such, we can set the weight of p to be

$$w_p = w \frac{f(x_p|x_0, \mu) - \beta}{f(x_p|x_0, \mu)} \quad (51)$$

and at the same time create a uniform weight portion w_u of the particle, representing the portion of the particle which is evenly distributed throughout R , namely

$$w_u = w \frac{\beta}{f(x_p|x_0, \mu)}. \quad (52)$$

It is important to note that $w_p + w_u = w$: the net weight in the system has not been modified, we have simply displaced a portion of w , distributing it uniformly through region R .

Dividing particles into a pointwise and uniform portion does not in and of itself accomplish any weight cancellation. When there are many fission particles within the same region, however, all of their uniform weight portions may be combined into a single weight which represents the uniformly distributed portion of the fission source for the region. This quantity shall be denoted as U_R , and is the sum of all uniform weight portions for all particles born within the region, each coming with a sign. Negative particles will yield negative contributions to U_R , while positive particles will yield positive contributions, leading to a cancellation. Once all particles have contributed their uniform portion to U_R , n new particles are sampled uniformly within the region, where

$$n = [|U_R|]. \quad (53)$$

Each of these uniformly sampled particles within R then has a weight of U_R/n .

These newly sampled, uniformly distributed particles belong to the same generation as particle p , and behave exactly like traditionally generated particles from this point on. Under this scheme, a particle which was initially positive will produce a positive uniform portion, and keep a positive weight; conversely, negative particles will remain negative and produce negative uniform portions. In the case of x_0 being in the same region as x_p , the uniform weight portion w_u must be zero, as the minimum of the expected fission density "behind" the starting point of the flight is zero. Therefore, particles with x_0 and x_p in the same region effectively do not partake in the cancellation process. These two properties are ensured by β being the minimum value of f in the region. However,

it is possible to relax the requirement that β should be the minimum of the expected fission density of the region. Booth and Gubernatis show in their original work that any value of β may be used, while still producing an unbiased result [11]. In the event of $\beta > f(x_p|x_0, \mu)$, the sign of the particle's pointwise weight portion will change.

With this regional method, the efficiency of cancellation can never be 100%. Only the uniform weight portions contribute to cancellation. For negative weighted fission particles, the pointwise weight portions do not cancel, always leaving some residual negative weight. Given this information, one might think that using a value of β which is larger than the minimum fission density might be beneficial, as it would increase the magnitude of the weight which goes into the uniform portion. This is not necessarily the case, however, as it could lead to the pointwise portions of positive particles becoming negative, via Eq. (51), possibly defeating the purpose of cancellation. Sadly, this method of cancellation can not be modeled as a linear operator, and therefore we are unable to put it in the context of the framework which was put forth in Section III. This makes regional cancellation conceptually similar to (though not the same as) the \tilde{C} operator mentioned in Eq. (40). It is inconsequential if our cancellation is linear or not; so long as it corresponds to a choice of \mathbf{E} that preserves Eq. (35), the method will be exact and unbiased.

A. Results of the 1D methodology

We added the exact 1D cancellation method of Booth and Gubernatis to the power iteration problem outlined in Sec. II, performing the cancellation between generations, and before the normalization of the system weight occurs. Thirty evenly spaced cancellation regions were used to partition the rod, and β was always taken to be the minimum of the expected fission density within the region. With cancellation, the simulation was stable, and was able to complete without issue. An eigenvalue of $k_{\text{eff}} = 0.33577 \pm 0.00005$ was obtained after 120 generations (20 inactive generations). This is in excellent agreement with the value obtained for the system using delta tracking, which was $k_{\text{eff}} = 0.33573 \pm 0.00005$ after the same number of generations. The weights W_+ , W_- , W_{net} , and W_{tot} are shown in Figure 8. We see that the negative weight quickly rises to an equilibrium level, near 5000, and fluctuates then about that value for the duration of the simulation. The positive weight must naturally increase by the same amount to keep the net weight of the system constant. This leads to an increase in the total transported weight of approximately 10^4 (the net weight still being 10^6). The evolution of the number of positive and negative particles is presented in Figure 9. The number of particles in memory stabilizes near almost $2 \cdot 10^6$, twice as many particles as were initially used.

The cancellation algorithm itself has linear computa-

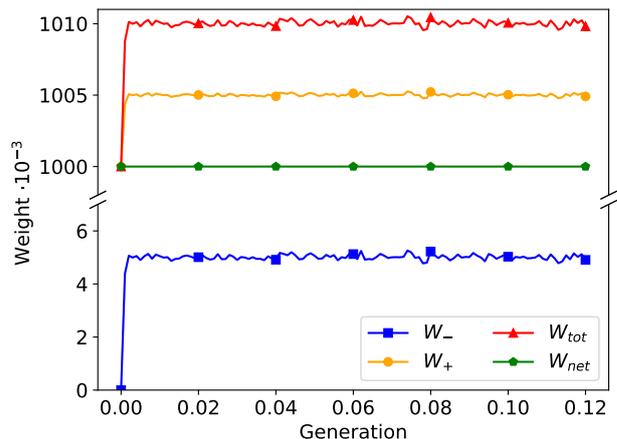


FIG. 8. The positive weight (W_+), negative weight (W_-), net weight (W_{net}), and total weight (W_{tot}) in the 1D rod system, when using 30 cancellation regions and 10^6 initial particles.

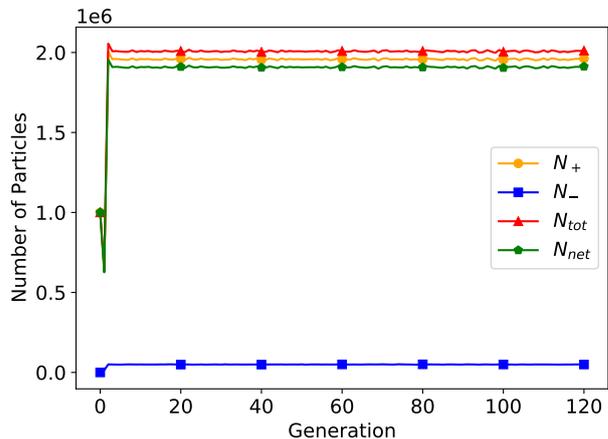


FIG. 9. The number of positive (N_+), negative (N_-), net (N_{net}), and total (N_{tot}) particles in the 1D rod system, when using 30 cancellation regions and 10^6 initial particles.

tional complexity with respect to the number of fission particles at any given generation. However, it is less obvious whether the total number of fission particles present once the simulation has settled to equilibrium is itself a linear function of the net weight of the system. It is evident that the total amount of weight transported must increase until equilibrium is reached, but is the increase linear in the net weight of the system? To examine this, we performed several runs with different initial values of W_{net} and we looked at the average value of W_-/W_{net} , as a function of W_{net} . The total weight can be deduced by looking at only the behavior of the negative weight, as the increase in the positive weight will mirror the negative weight. The result of this study is presented in Fig. 10. When too little net weight is injected in the

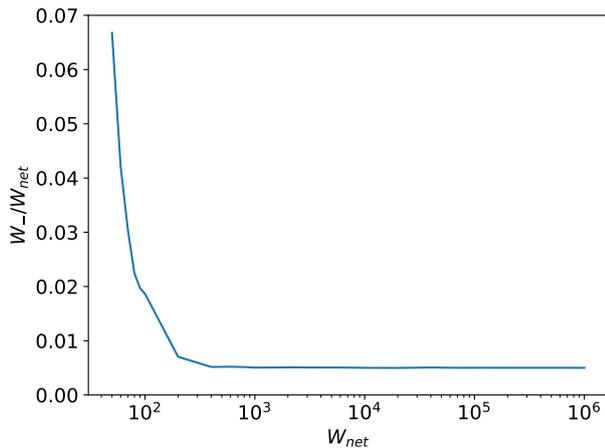


FIG. 10. Negative weight fraction (W_-/W_{net}) as a function of net weight (W_{net}) in the 1D rod system, when using cancellation with 30 cancellation regions.

system, a higher percentage of negative weight will be present in the system at equilibrium. There appears to be a critical point (in this case, near $W_{\text{net}} = 400$) above which the fraction of negative weight no longer decreases by adding more net weight. This might at first seem counter-intuitive, but positive particles are always being converted to negative particles during the random walk (because of the NWDT algorithm), and the cancellation process is not 100% efficient. Above this critical point, no matter how much positive weight is added to the system at the beginning of the simulation, the equilibrium amount of negative weight will on average be a set fraction of the net weight. In other words, the total equilibrium weight is a linear function of the starting weight only for sufficiently large values of the latter.

Another consideration is the effect of the number of cancellation regions on the amount of negative weight. When there are fewer regions, each one must become larger; this makes cancellation less efficient, as β will decrease if one is taking β to be the minimum of f over the region. Conversely, making regions too small will result in too few particles which can partake in cancellation, also reducing the efficiency. To examine this, we have plotted the negative weight fraction for the number of cancellation regions in Figure 11. Indeed, from the figure, there is a range for which the fraction of negative weight decreases (i.e. the efficiency of cancellation increases) by adding more regions. The number of regions where the minimum occurs depends on the net weight of the system. Here, this occurs near 6×10^2 regions for $W_{\text{net}} = 10^4$, 4×10^3 for $W_{\text{net}} = 10^5$ and 2×10^4 for $W_{\text{net}} = 10^6$, suggesting that the optimal number of regions does not quite scale linearly with the net initial weight. The three tested net weights all had very similar behavior, and the efficiency improved at the same rate when adding cancellation regions, until the mini-

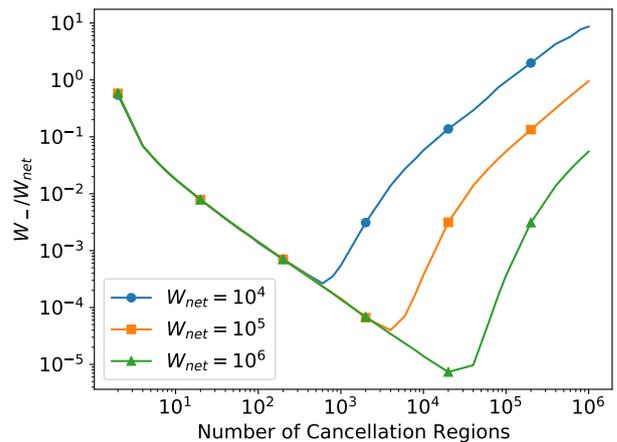


FIG. 11. Negative weight fraction (W_-/W_{net}) as a function of the number of cancellation regions, for different net weights.

mum value was reached. There is also a large range over which the negative weight fraction is less than 1% of the net weight. For all net weights, this range starts at approximately 20 regions, and goes up to 2×10^3 regions for $W_{\text{net}} = 10^4$, or 3×10^5 regions for $W_{\text{net}} = 10^6$.

This section has outlined the methodology behind Booth and Gubernatis' regional cancellation. The potential of the method was successfully demonstrated, having canceled enough negative weight to allow for the proper convergence of the power iteration algorithm when using NWDT. In light of this success, we move on to extending the method to work with more realistic problems in higher spatial dimensions.

V. EXACT 3D, MULTI-GROUP, REGIONAL CANCELLATION SCHEME

Unfortunately, it is not possible to directly implement the 1D regional cancellation scheme of Booth and Gubernatis in higher spatial dimensions. One-dimensional systems have the special property that lines coincide with volumes, ensuring the lines of flight of all particles traversing the region overlap, as well as their uniform weight portions. This fact allows us to add the uniform weight portions of all particles in the region, effectively leading to a cancellation of negative and positive weight. For higher dimensions, this is not the case. The uniform portion of the particle weights may only be distributed along their rays of flight within the cancellation region, but the flight rays will never overlap completely. It is then no longer possible to combine the uniform weight portions as before by simply taking their sum.

The problem with the method of calculating the fission density in Eq. (49) is that it is only valid along the ray of flight of the parent particle, after the flight direction has been sampled. To extend the regional cancellation to

higher dimensions, it is required to consider the probability of the parent particle scattering into the solid angle which is subtended by the cancellation region. This is done by examining $\hat{\Omega}'$, the direction of the parent particle at position \mathbf{r}_0 before the direction is changed by the scattering kernel. After the parent has been modified by the process of scattering, it then has a direction $\hat{\Omega}$, which must intersect the cancellation region (given that we are attempting to perform cancellation for a particle in the region which was induced by the parent). The scattering cosine for the interaction which the parent underwent at \mathbf{r}_0 is then $\mu = \hat{\Omega}' \cdot \hat{\Omega}$. For neutron transport, the azimuthal direction of a scatter is almost always isotropic, and only the scattering cosine is anisotropic (should there be any anisotropy) [5]. The probability density of scattering in direction $\hat{\Omega}$ from direction $\hat{\Omega}'$ is therefore

$$P(\hat{\Omega}|\hat{\Omega}') = \frac{P(\mu)}{2\pi}, \quad (54)$$

with $\int_{-1}^1 P(\mu)d\mu = 1$. Modifying Eq. (49) to use the flight kernel for 3D, and applying this factor to consider the scattering angle, we arrive at the expected fission density

$$f(\mathbf{r}|\mathbf{r}_0, \hat{\Omega}') = \frac{P(\mu)\Sigma_f(\mathbf{r})}{2\pi|\mathbf{r} - \mathbf{r}_0|^2} e^{-\Sigma_{\text{smp}}(\mathbf{r}_0)|\mathbf{r} - \mathbf{r}_0|}, \quad (55)$$

where

$$\mu = \frac{\mathbf{r} - \mathbf{r}_0}{|\mathbf{r} - \mathbf{r}_0|} \cdot \hat{\Omega}'. \quad (56)$$

Taking a closer look at Eq. (55), one may recognize it as being what Lux and Koblinger refer to as the next-event estimator, for the fission rate at \mathbf{r} given a collision at \mathbf{r}_0 [14, Sec. 6.IV.A]. Note that Eq. (55) represents the expected fission density over the scattering and the free flight following it; this should be contrasted with Eq. (49), which represents the expected fission density over the following free flight *only*.

While there is only one natural shape for a cancellation region in 1D, there are an infinite number of possible shapes in 3D which one could use. In this work, we will consider right rectangular prisms for our cancellation region. This is due to the fact that it is very simple to sample positions within a right rectangular prism uniformly, which is required in the cancellation processes. We will choose β to be the minimum value of the expected fission density within the cancellation region. If $\Sigma_f(\mathbf{r})$ is spatially varying within the cancellation region and scattering is anisotropic, it is difficult to determine the true minimum; when $\Sigma_f(\mathbf{r})$ is homogeneous within the region and scattering is isotropic, however, one may determine the true minimum of the expected fission density for the flight from \mathbf{r}_0 to \mathbf{r} by evaluating $f(\mathbf{r}|\mathbf{r}_0, \hat{\Omega}')$ at all eight corners of the prism, as the minimum must occur at one of those eight points. Figure 12 provides a depiction of

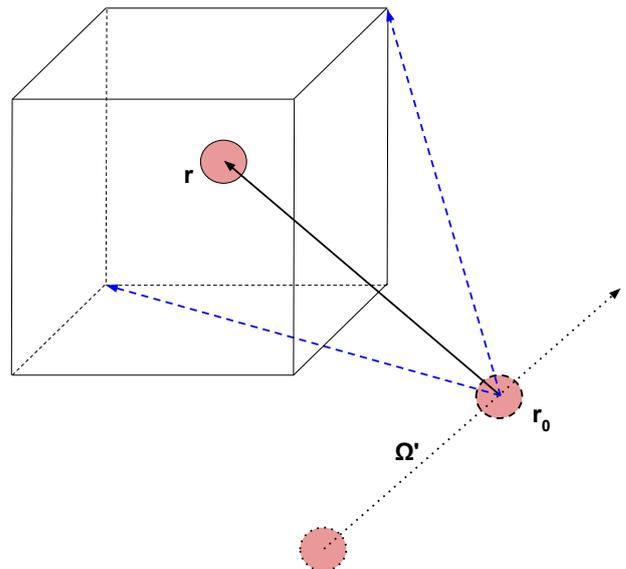


FIG. 12. Depiction of the 3D regional cancellation process. To find the uniform weight fraction of the fission particle at \mathbf{r} , β must first be determined. This is done by calculating the expected fission density at all eight corners of the cancellation region (though only two are depicted by the dashed arrows) from the parent particles previous position and direction (\mathbf{r}_0 and $\hat{\Omega}'$), and taking the minimum value.

the cancellation process. Using the minimum as the value of β for the flight, it is possible to calculate a pointwise weight portion and a uniform weight portion of the fission particle, as was done before in Eq. (51) and Eq. (52). Once the uniform portions of all fission particles in the region have been collected, one may take the net uniform weight of the region, and sample new particles to add to the fission bank uniformly within the prism.

Until now, for simplicity, we have only considered single-speed transport, with all cross sections being energy-independent. With no energy dependence, one does not need to consider how to sample the energy of the uniform particles. In all realistic applications in reactor physics, however, all cross sections are continuous functions of the neutron energy, and the emission spectrum for the energy of fission neutrons depends on the collided nuclide and on the incoming energy of the neutron which induced the fission [5]. Under this assumption, it would be impossible to collect uniform portions of all the fission particles together. Without this action, no cancellation occurs. This fact poses a difficulty in applying weight cancellation to continuous energy systems.

An approximation which is often used in the reactor physics community is the multi-group treatment of the energy variable in the phase space, by using cross sections which are piece-wise constant in energy. These energy intervals in which the cross section is constant are referred to as groups. We denote the g -th energy group as E_g , with $g \in \{1, 2, \dots, N_g\}$. By convention, the group corre-

sponding with the highest energy is E_1 , while the group

with the lowest energy is E_{N_g} [5]. The multi-group k -eigenvalue Boltzmann equation reads

$$\hat{\Omega} \cdot \nabla \varphi(\mathbf{r}, \hat{\Omega}, E_g) + \Sigma_t(\mathbf{r}, E_g) \varphi(\mathbf{r}, \hat{\Omega}, E_g) = \sum_{g'=1}^{N_g} \int_{4\pi} \Sigma_s(\mathbf{r}, \hat{\Omega}' \rightarrow \hat{\Omega}, E_{g'} \rightarrow E_g) \varphi(\mathbf{r}, \hat{\Omega}', E_{g'}) d\hat{\Omega}' + \frac{1}{4\pi k} \sum_{g'=1}^{N_g} \chi(E_{g'} \rightarrow E_g) \nu(\mathbf{r}, E_{g'}) \Sigma_f(\mathbf{r}, E_{g'}) \int_{4\pi} \varphi(\mathbf{r}, \hat{\Omega}', E_{g'}) d\hat{\Omega}' \quad (57)$$

The structure of Eq. (57) is formally that of a system of particles with N_g species, coupled with each other by means of scattering or fission.

Here, the fission spectrum $\chi(E_{g'} \rightarrow E_g)$ is written to show a dependence of the fission particle energy on the parent particle's incident energy. This dependence on the incoming particle energy is relatively weak and is often ignored in the multi-group approximation [5]. In this case, we may simply write $\chi(E_g)$, moving it outside the sum in Eq. (57). Under this approximation, we may collect the uniform portions of all fission neutrons in the region together; regardless of the energy of their parent, their energy spectra are all the same. However, one must change the definition of the expected fission density to take energy into account. Equation (55) is then modified to include the energy group E_g of the particle that induces fission:

$$f(\mathbf{r}|\mathbf{r}_0, \hat{\Omega}', E_g) = \frac{P(\mu, E_g) \Sigma_f(\mathbf{r}, E_g)}{2\pi |\mathbf{r} - \mathbf{r}_0|^2} \times e^{-\Sigma_{\text{smp}}(\mathbf{r}_0, E_g) |\mathbf{r} - \mathbf{r}_0|}. \quad (58)$$

A similar problem can occur for the direction of the fission particles as well. In continuous energy transport, both the direction and energy of the fission particles may be a function of the energy and direction of the incoming neutron. While the nuclear data representations allow for this, it is a very marginal occurrence³ in major nuclear-data evaluation libraries [33, 34]. Therefore, we do not need to make any special considerations for the angular distribution of fission particles in order to apply regional cancellation to multi-group problems.

We have seen that weight cancellation takes place at fission; however, one may wonder what is special about fission, and whether one could perform cancellation at scattering events, instead. At fission, the energy and angular distributions of the secondary fission particles are independent of the properties of the particle that induces the fission event. Fission represents the natural cancellation event because the three-dimensional distribution of

the expected fission density (Eq. (58)) encodes all the six-dimensional distribution of the fission emission density in phase space.

VI. IMPLEMENTATION AND RESULTS

To test our exact, 3D multi-group regional cancellation, we used a modified version of the C5G7 international benchmark, which we have depicted in Figure 13 [35]. The C5G7 represents a small 1/8th nuclear reactor core, with four fuel assemblies and 7 energy groups, that is customarily used to assess and compare deterministic transport codes. In its original specifications, the fuel pins within the assemblies are cylindrical, with a radius of 0.54 cm [35]. The use of cylindrical fuel pins however makes it difficult to use arbitrarily small cancellation regions while also being able to directly calculate the minimum value of Eq. (58), to be used as β . In order to facilitate cancellation, we have modified the benchmark so that the fuel pins are square in the x - y plane, with side lengths of 0.756 cm. This choice allows the fuel cells to be easily cut into an integral number of rectangular prisms to be used as cancellation regions, ensuring that no material other than fuel is present in the region. An important difference from the 1D rod system is that, in the C5G7 benchmark, the cross sections are piece-wise spatially constant, being homogeneous within a given material cell. Also, while we derived the 3D cancellation formulas for the general case of anisotropic scattering, all scattering in the C5G7 benchmark is assumed to be isotropic, simplifying Eq. (58) to

$$f(\mathbf{r}|\mathbf{r}_0, E_g) = \frac{\Sigma_f(\mathbf{r}, E_g)}{4\pi |\mathbf{r} - \mathbf{r}_0|^2} e^{-\Sigma_{\text{smp}}(\mathbf{r}_0, E_g) |\mathbf{r} - \mathbf{r}_0|}. \quad (59)$$

Under this assumption, we have no need to keep track of the parent particle's initial direction $\hat{\Omega}'$, as there is equal probability of scattering in any direction.

The C5G7 benchmark makes use of 3 reflective boundary conditions, as shown in Figure 13. One caveat of the regional cancellation method is that Eq. (58) is akin to a next-event point reaction rate estimator. Due to this, a trivial implementation of the method can not be used with reflective boundary conditions [28]. Use of this

³ One exception to this is the evaluation for ²³²Th which has anisotropic distributions for prompt fission neutrons in the laboratory frame, in both ENDF/B-VIII.0 and JEFF-3.3.

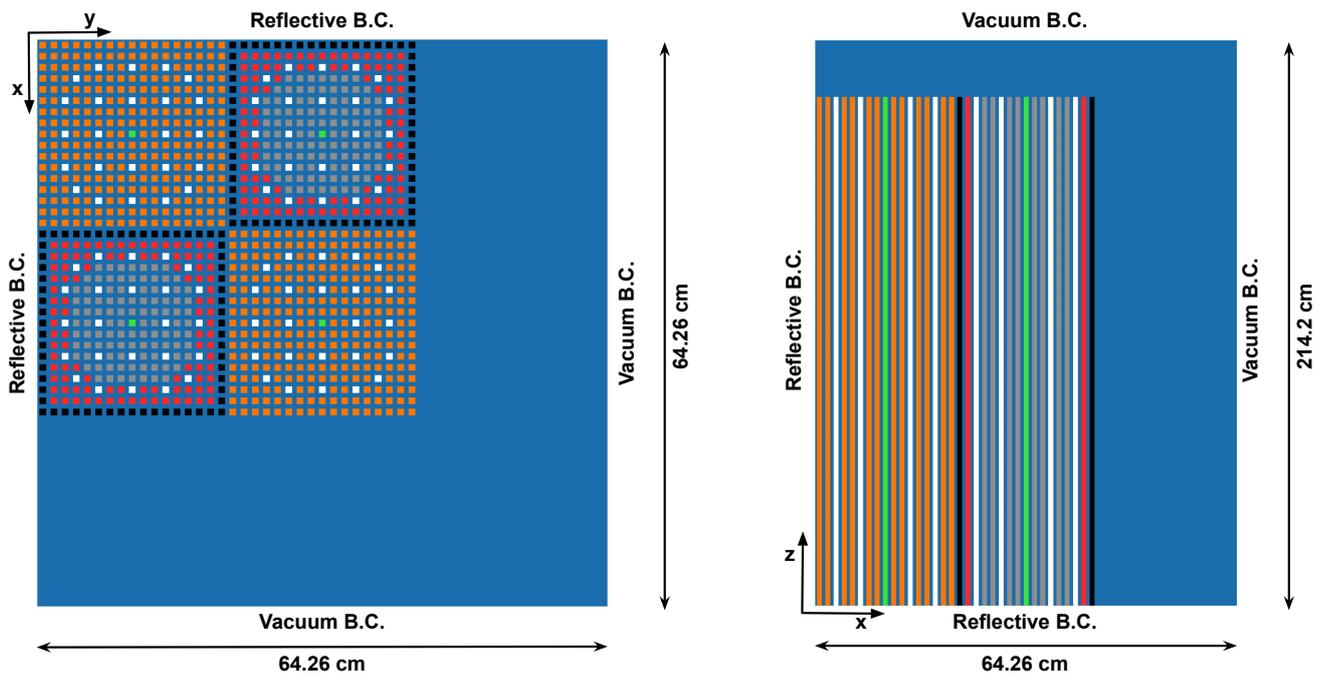


FIG. 13. Geometric configuration for our modified version of the C5G7 benchmark, with boundary conditions. Each color represents a material, associated with a unique set of cross sections.

method with reflective boundary conditions is only possible if special book-keeping is done to ensure a proper calculation of the expected fission density. If a particle begins at \mathbf{r}_0 and encounters a reflection before having a real collision at \mathbf{r} , then the distances which one would obtain by simply computing $|\mathbf{r} - \mathbf{r}_0|$ will not represent the actual distance traveled. It is also not enough to simply store the distance traveled by the particle, as one should ideally be able to calculate the value of f for any given \mathbf{r} in the region, in order to calculate β . To ensure a proper calculation of $|\mathbf{r} - \mathbf{r}_0|$, the initial position \mathbf{r}_0 of the parent particle is stored: upon each reflection, this position is transformed to the mirrored location on the other side of the plane of reflection. When doing this, the point will generally lie outside of the defined geometry for the system, but this should not pose a problem, as we only need to be able to calculate the flight distance between \mathbf{r}_0 and other points in the cancellation region. With delta tracking and NWDT, $\Sigma_{\text{maj}}(E_g)$ and $\Sigma_{\text{smp}}(E_g)$ do not depend on the position within the geometry, but only on the energy group. We therefore do not need to worry about trying to find what material this fictitious point would be located in, or the distance to the next surface.

A problem of this type is admittedly a simple case, where NWDT is not strictly necessary. One is able to trivially determine the majorant cross section so that delta tracking could be used, avoiding the problem of negative weights all together. As mentioned in Sec. II, however, in the context of multi-physics problems which are represented with spatially continuous cross sections, de-

termining the majorant exactly is likely to be difficult, or impossible. It would certainly be interesting to test this algorithm on a more complex problem, where delta tracking would not be possible (or would be very inefficient). It would be difficult to verify that the presented cancellation method is working with such a problem, however, as it could not easily be solved with existing methods. This is why we have chosen to examine this simpler problem.

To test and evaluate cancellation and NWDT, a multi-group Monte Carlo code, called MGMC, was written to solve k -eigenvalue power iteration problems. MGMC supports general geometries using traditional surface-based descriptions of volumes. Either delta tracking [20] or the variant of negative weighted delta tracking developed by Carter, Cashwell, and Taylor [22, 24] may be selected for transport. Scalar flux and the fission reaction rate may be scored over a rectilinear mesh, using collision estimators⁴. Monte Carlo estimates are saved as binary Numpy files [36] for easy analysis and plotting with Python. Cancellation regions are defined by a rectilinear mesh imposed on top of the problem, and can be used with both delta tracking and negative weighted delta tracking. Shared memory parallelism is implemented with OpenMP. Geometry, material properties, scores, cancellation, and simulation settings are all controlled with a YAML input file. MGMC is written in

⁴ Scalar flux is defined as $\int_{4\pi} \varphi(\mathbf{r}, \hat{\Omega}) d\hat{\Omega}$ [5].

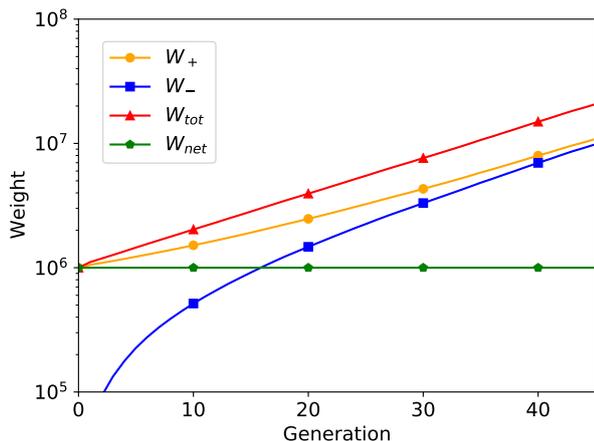


FIG. 14. The positive, negative, net, and total weights in the modified C5G7 benchmark, using negative weighted delta tracking and no weight cancellation.

C++17, and has been made available under the CeCILL-v2.1 license [37]. One will also find the necessary input files to replicate our results there.

All simulations were started with 10^6 particles uniformly distributed across the four fuel assemblies, all in the first energy group, and ran for 2200 generations, with the first 200 generations being discarded to allow for fission source convergence. Delta tracking was first run (without weight cancellation) to obtain a reference multiplication factor for the system, which was found to be $k_{\text{eff}} = 1.21912 \pm 0.00002$. When using the Carter, Cashwell, and Taylor variant of negative weighted delta tracking, the majorant cross section was used for the sampling cross section in all the energy groups except for the first; for the first energy group, $0.9\Sigma_{\text{maj}}(E_1)$ was used as the sampling cross section. With this choice of Σ_{smp} , the total cross section is underestimated in all of the fuel pins in the first energy group. Any virtual collisions which occur in these regions of the phase space will therefore result in the particle weight changing sign.

Running this simulation with NWDT, and without the use of any weight cancellation, causes the particle populations to diverge and the simulation to fail. The exponential increase in total weight is depicted in Figure 14. The behavior is nearly identical to the 1D case presented in Section II. Asymptotically, the negative weight will increase at the same rate as the positive weight, with the difference between the two remaining constant. The exponential increase in W_{tot} also leads to an exponential increase in the total number of particles in the simulation, overwhelming computer memory, exactly as in the 1D case. This supports the point made by Eq (29); weight cancellation will always be necessary when attempting to perform power iteration simulations using NWDT.

In order to implement regional cancellation, all fuel pins were divided into cubical cancellation regions with

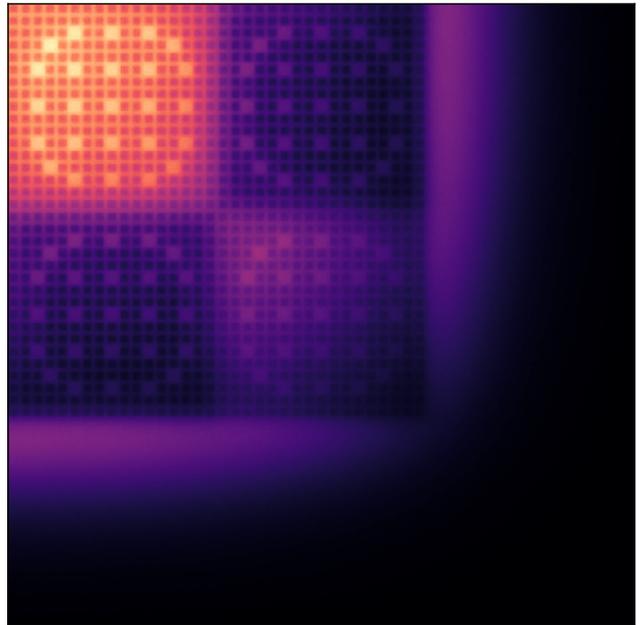


FIG. 15. Scalar flux of the 7th energy group in the center axial slice of the core, obtained using negative weighted delta tracking and 3D regional cancellation.

side lengths of 0.252 cm. This mesh was chosen by trial and error, as we have found no practical way to know in advance for any system the minimum number of cancellation regions required to stabilize the simulation. Cubical cancellation regions also seemed to be more efficient than regions which had aspect ratios much larger than one, as compact region shapes lead to a higher minimum value of f in the region, and therefore increases β .

Our 3D multi-group exact regional cancellation algorithm was able to stabilize the particle populations when using NWDT in conjunction with the previously outlined parameters, and an eigenvalue of $k_{\text{eff}} = 1.21915 \pm 0.00005$ was obtained. This is in very good agreement with the value of k_{eff} obtained from traditional delta tracking with no weight cancellation, as the two estimates differ by less than one standard deviation. The thermal (7th energy group) scalar flux which was tallied is shown in Figure 15. This was compared with the flux tally from the delta tracking simulation by performing a Student t -test in each mesh cell [38]. The plot of the absolute value of the Student t -statistic is provided in Figure 16. It can be seen there that the two flux estimates are also in good agreement with one another, with no apparent spatial dependence in their difference.

These two maps only show a small portion of the examined phase space. For a more thorough comparison, a histogram of the distribution of the Student t -statistic for the NWDT flux and the delta tracking flux is given in Figure 17 [38]. Large portions of the flux tally mesh have average values of zero, due to the large water reflectors. These elements were removed before producing

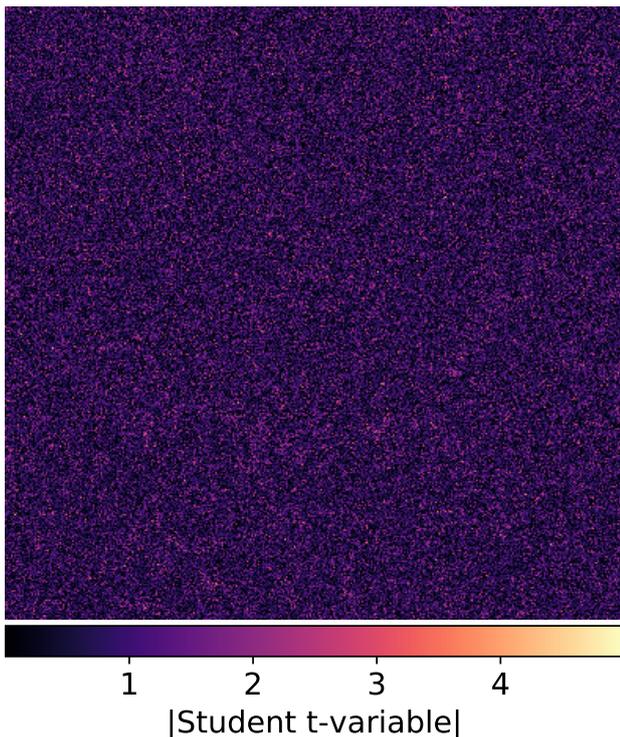


FIG. 16. Absolute value of the Student t -statistic for the difference between the delta tracking and NWDT flux estimates. The portion displayed is for the 7th energy group, at center axial slice.

Figure 17. Elements which had a relative error greater than 20% (in either the NWDT or the delta tracking score) were also removed, as the Student t -test is only applicable on normally distributed variables; therefore, we need to ensure that sufficient statistics are accumulated, so that the central limit theorem may apply. The 20% cut-off is admittedly somewhat arbitrary, but it is likely that bins with such a high relative error are not normally distributed. It can be seen that the histogram has very good agreement with the theoretical distribution, which we assume to be normal given the large number of degrees of freedom for the comparison.

Finally, Figure 18 shows the behaviour of the weights in the system for the first 100 generations in the simulation. As with the 1D case, there is initially no negative weight. This increases rapidly in the first 10-20 generations, before beginning to level out. By the time 80 generations have passed, the weights have reached their equilibrium values. This is much longer than the 1D case, which only required 2-3 generations before the weights reached equilibrium. The longer time to convergence is attributed to the C5G7 benchmark having a higher dominance ratio than the 1D case.

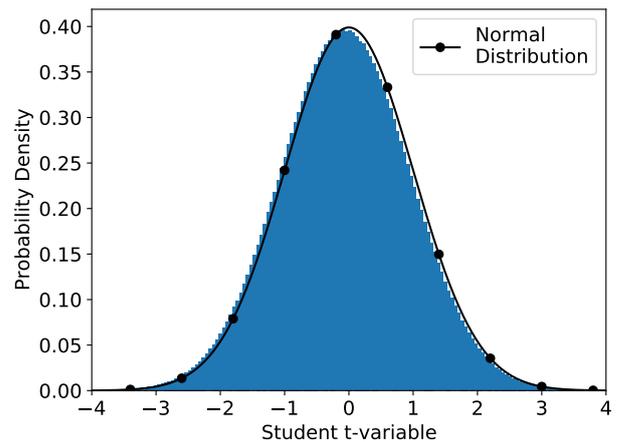


FIG. 17. Histogram of the Student t -statistic for the difference between the flux estimates obtained with delta tracking and NWDT. The curve plotted on top is the expected normal distribution.

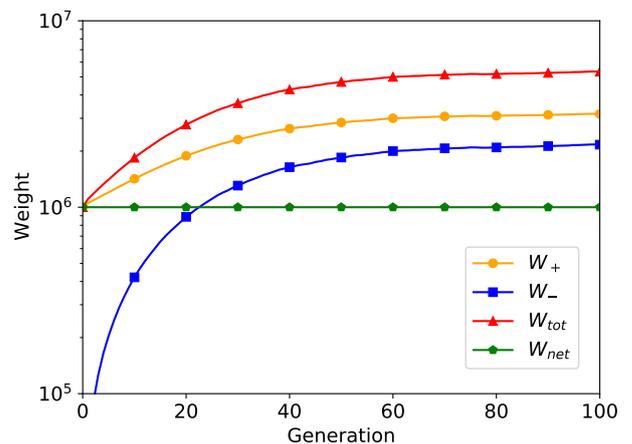


FIG. 18. The positive, negative, net, and total weights in the modified C5G7 benchmark, using negative weighted delta tracking with exact 3D regional weight cancellation.

VII. CONCLUSIONS

Over the course of this work, we have presented a previously undocumented population control problem which arises when attempting to run k -eigenvalue power iteration Monte Carlo simulations with both negative and positive weights in negative weighted delta tracking [22]. Modeling the transport process through a set of coupled Boltzmann transport equations for negative and positive particles, we were able to show that the power iteration technique applied to negative weighted delta tracking will always fail to converge, as the sought physical fundamental eigenvalue is not the dominant eigenvalue of the system. Instead, a fictitious eigenvalue, associated with a

system with a lower amount of absorption, is now dominant. We have formally developed a cancellation operator, demonstrating theoretically, and with a deterministic model, how particle weight cancellation can suppress the fictitious eigenstate and restore convergence of power iteration to the physical fundamental eigenvalue.

To demonstrate weight cancellation in a Monte Carlo context, we have implemented the exact 1D region cancellation algorithm of Booth and Gubernatis [11] in a 1D rod model. Weight cancellation did stabilize the particle populations through the fission generations, allowing the simulation to finish. As the method outlined by Booth and Gubernatis is only valid in 1D single-speed problems, we have developed an exact 3D regional cancellation method. We tested our 3D algorithm on a modified version of the C5G7 reactor-physics benchmark, using negative weighted delta tracking. Cancellation stabilized the particle populations, and also resulted in estimates for the multiplication factor and flux which were in agreement with the reference results obtained through delta tracking. Our exact 3D multi-group regional cancellation could potentially be useful for other applications in neutron transport as well. One such case is the use of Monte Carlo methods to obtain higher harmonics of the k -eigenvalue equation [8]. Our cancellation algorithm could also potentially prove useful in transport problems involving complex particle weights, such as the search for critical buckling or the solution of the neutron noise equations [7, 39].

Several questions remain to be settled in regard to the methodology of 3D cancellation. First, in this work we have required that a cancellation region consist of only one material. Being able to have multiple materials in a cancellation region would make it easier to perform cancellation on different geometric forms (cylindrical pins). Second, extension of this algorithm to continuous energy will require the application of new techniques, because in continuous-energy transport the fission spectrum depends on the incoming neutron energy. Finally, the efficient choice of β should be evaluated. While we have chosen to use the minimum value of the expected fission density in the region for β , Booth and Gubernatis made clear that any value of β results in an unbiased estimate of the fission density. However, the variance of the weights of the fission particles clearly depends on β . It is desirable to characterize which choices of β result in more efficient cancellation. All of these issues will require further investigation to improve the algorithm, and to probe its possible extension to continuous-energy problems.

Appendix: Derivation of coupled Boltzmann transport equations

We provide here the derivation of the coupled system of Boltzmann equations, where the populations of positive and negative particles are treated separately and are assumed to be governed by the rules of negative weighted

delta tracking (Alg. 1) [22]. The quantities that we wish to describe are the angular fluxes of positive and negative particles. These quantities are not physical observables, but they are useful to characterize the behaviour of the random walk. Consider any one of the usual Monte Carlo estimators for the (physical) angular flux; the unbiasedness condition requires any such estimator to be proportional to the particle weight w . Replacing the particle weight w in the estimator with

$$w_+ = \begin{cases} w & w > 0 \\ 0 & w \leq 0 \end{cases} \quad (\text{A.1})$$

results in a modified estimator; we define the angular flux of positive particles φ_+ to be the expected value of the modified estimator. Likewise, replacing the particle weight with

$$w_- = \begin{cases} 0 & w > 0 \\ -w & w \leq 0 \end{cases} \quad (\text{A.2})$$

results in a modified estimator, whose expected value is defined to be the angular flux of negative particles, φ_- .

A few properties are worth stressing. First, if all particle weights are positive, then $\varphi_+ = \varphi$ (the physical angular flux) and $\varphi_- = 0$. Second, since $w_+ - w_- = w$, then $\varphi_+ - \varphi_- = \varphi$. Finally, since $w_+ \geq 0$ and $w_- \geq 0$, then both φ_+ and φ_- are always non-negative.

With these postulations, we may commence our derivation of the angular flux of positive particles. We shall obey the rules for negative weighted delta tracking [22], presented in Alg. 1. The Boltzmann equation for neutron transport is canonically written in the form of a balance equation, where losses = gains. These losses and gains refer to the change in the neutron flux at the phase space point $(\mathbf{r}, \hat{\Omega})$. This can be referenced in Eq. (1), where the left hand side (LHS) of the equation represents losses, while the right hand side (RHS) represents gains. More detail as to the reasoning and derivation behind particular terms can be found in *Nuclear Reactor Theory*, by Bell and Glasstone [5].

Starting with the LHS, we first consider losses due to particle streaming, L_S . This term is identical to that found in Eq. (1), only replacing the physical angular flux with the angular flux of positive particles:

$$L_S = \hat{\Omega} \cdot \nabla \varphi_+(\mathbf{r}, \hat{\Omega}). \quad (\text{A.3})$$

Next, we consider losses due to collisions, L_C . In the case of NWDT, it is possible to be removed from the phase space point by having a collision at \mathbf{r} (real or virtual), which occurs with a cross section Σ_{smp} . We therefore consider the losses due to collisions with the term

$$L_C = \Sigma_{\text{smp}}(\mathbf{r}) \varphi_+(\mathbf{r}, \hat{\Omega}). \quad (\text{A.4})$$

In the event of a real collision, the particle direction will change, so there is indeed a removal from the phase space

point. If a virtual collision occurs and the positive particle changes sign to be a negative particle, this is also a loss. If a virtual collision does not result in a positive particle becoming negative, then it is not actually a loss. This case will be treated on the RHS by adding a gain term for virtual collisions which do not result in a sign flip. The only possible sources for loss is then L_S and L_C , and we then have a LHS of

$$L_S + L_C = \hat{\Omega} \cdot \nabla \varphi_+(\mathbf{r}, \hat{\Omega}) + \Sigma_{\text{smp}}(\mathbf{r}) \varphi_+(\mathbf{r}, \hat{\Omega}). \quad (\text{A.5})$$

Next, we may consider gains at the phase space point $(\mathbf{r}, \hat{\Omega})$. Gains may only come from particles which have had either a real or virtual collision, allowing them to enter the phase space point in question. First, we shall consider the gains from real collisions, G_R . A real collision occurs with probability $q(\mathbf{r})$, and when this happens, a weight modification also occurs, which is a multiplication of the factor $\Sigma_t(\mathbf{r})/(q(\mathbf{r})\Sigma_{\text{smp}}(\mathbf{r}))$:

$$G_R = q(\mathbf{r}) \frac{\Sigma_t(\mathbf{r})}{q(\mathbf{r})\Sigma_{\text{smp}}(\mathbf{r})} R, \quad (\text{A.6})$$

here R is the sum of the gains of all real collision channels.

$$\begin{aligned} G_R &= q(\mathbf{r}) \frac{\Sigma_t(\mathbf{r})}{q(\mathbf{r})\Sigma_{\text{smp}}(\mathbf{r})} \left[\frac{\Sigma_{\text{smp}}(\mathbf{r})}{\Sigma_t(\mathbf{r})} \int_{4\pi} \Sigma_s(\mathbf{r}, \hat{\Omega}' \rightarrow \hat{\Omega}) \varphi_+(\mathbf{r}, \hat{\Omega}') d\hat{\Omega}' + \frac{\Sigma_{\text{smp}}(\mathbf{r}) \nu(\mathbf{r})}{\Sigma_t(\mathbf{r})} \frac{1}{4\pi k} \int_{4\pi} \Sigma_f(\mathbf{r}) \varphi_+(\mathbf{r}, \hat{\Omega}') d\hat{\Omega}' \right] \\ &= \int_{4\pi} \Sigma_s(\mathbf{r}, \hat{\Omega}' \rightarrow \hat{\Omega}) \varphi_+(\mathbf{r}, \hat{\Omega}') d\hat{\Omega}' + \frac{\nu(\mathbf{r})}{4\pi k} \int_{4\pi} \Sigma_f(\mathbf{r}) \varphi_+(\mathbf{r}, \hat{\Omega}') d\hat{\Omega}' \\ &= \mathcal{S} \varphi_+ + \frac{1}{k} \mathcal{F} \varphi_+ \end{aligned} \quad (\text{A.9})$$

All that remains are gains from virtual collisions G_V . The probability of a virtual collision is $1 - q(\mathbf{r})$, and is accompanied by a weight modification of

$$\frac{\left| 1 - \frac{\Sigma_t(\mathbf{r})}{\Sigma_{\text{smp}}(\mathbf{r})} \right|}{1 - q(\mathbf{r})}. \quad (\text{A.10})$$

We take the absolute value here, as our particle weights may never become negative. Instead, a change in sign is modeled by the transfer of a particle from the positive population (corresponding to φ_+) to the negative population (corresponding to φ_-). Our virtual gains are

$$G_V = (1 - q(\mathbf{r})) \frac{\left| 1 - \frac{\Sigma_t(\mathbf{r})}{\Sigma_{\text{smp}}(\mathbf{r})} \right|}{1 - q(\mathbf{r})} V, \quad (\text{A.11})$$

V being the sum of gains due to virtual collisions. The first channel for virtual collisions is the previously mentioned case of a positive particle having a virtual collision, and remaining positive (V_+). This only occurs when $\Sigma_{\text{smp}}(\mathbf{r}) \geq \Sigma_t(\mathbf{r})$, and can be modeled with the Heaviside

We have two possible reaction channels which contribute to gains at $(\mathbf{r}, \hat{\Omega})$: particles having a collision at \mathbf{r} and scattering from direction $\hat{\Omega}'$ into direction $\hat{\Omega}$ (R_S), and particles having a collision at \mathbf{r} , inducing a fission particle which is born traveling in direction $\hat{\Omega}$ (R_F). Treating the scattering term, we must sum over all possible incoming directions, resulting in

$$R_S = \frac{\Sigma_{\text{smp}}(\mathbf{r})}{\Sigma_t(\mathbf{r})} \int_{4\pi} \Sigma_s(\mathbf{r}, \hat{\Omega}' \rightarrow \hat{\Omega}) \varphi_+(\mathbf{r}, \hat{\Omega}') d\hat{\Omega}'. \quad (\text{A.7})$$

We must multiply by the ratio $\Sigma_{\text{smp}}/\Sigma_t$ to account for the fact that we are conditioning on the collision being real. We now perform a similar operation for the fission channel, with the assumption that the distribution of the direction of fission particles is isotropic:

$$R_F = \frac{\Sigma_{\text{smp}}(\mathbf{r}) \nu(\mathbf{r})}{\Sigma_t(\mathbf{r})} \frac{1}{4\pi k} \int_{4\pi} \Sigma_f(\mathbf{r}) \varphi_+(\mathbf{r}, \hat{\Omega}') d\hat{\Omega}'. \quad (\text{A.8})$$

These being the only two real collision channels ($R = R_S + R_F$), we may combine the definitions of R_S , R_F , and G_R to obtain the result presented in Eq. (A.9), where we have employed our previous definitions for \mathcal{S} and \mathcal{F} from Eq. (10) and Eq. (11).

function

$$\Theta(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}. \quad (\text{A.12})$$

Our gains from virtual collisions of positive particles is then

$$V_+ = \Sigma_{\text{smp}}(\mathbf{r}) \Theta(\Sigma_{\text{smp}}(\mathbf{r}) - \Sigma_t(\mathbf{r})) \varphi_+(\mathbf{r}, \hat{\Omega}). \quad (\text{A.13})$$

In the event that $\Sigma_{\text{smp}}(\mathbf{r}) < \Sigma_t(\mathbf{r})$, negative particles will flip sign, joining the positive particles (V_-). This is modeled in a similar manner, simply flipping the argument in the Heaviside function, and replacing the positive angular flux with the negative angular flux:

$$V_- = \Sigma_{\text{smp}}(\mathbf{r}) \Theta(\Sigma_t(\mathbf{r}) - \Sigma_{\text{smp}}(\mathbf{r})) \varphi_-(\mathbf{r}, \hat{\Omega}). \quad (\text{A.14})$$

With these terms defined, we may combine our definitions of G_V and $V = V_+ + V_-$ to obtain the gains from virtual collisions, presented in Eq. (A.15), having used the function from Eq. (12).

All loss and gain terms have now been defined, leaving us with the final form of the equation, describing the angular flux of the positive particles in Eq. (A.16). By symmetry, the equation for negative particles must have the exact same form, and is given in Eq. (A.17). Re-

gardless of whether a particle is negative or positive, it is transported in the same manner, and by the same rules. These two equations are of course the form presented in Eq. (9), and demonstrate the coupling relationship between positive and negative particles.

$$\begin{aligned}
 G_V &= (1 - q(\mathbf{r})) \frac{\left|1 - \frac{\Sigma_t(\mathbf{r})}{\Sigma_{\text{smp}}(\mathbf{r})}\right|}{1 - q(\mathbf{r})} \Sigma_{\text{smp}}(\mathbf{r}) \left[\Theta(\Sigma_{\text{smp}}(\mathbf{r}) - \Sigma_t(\mathbf{r})) \varphi_+(\mathbf{r}, \hat{\Omega}) + \Theta(\Sigma_t(\mathbf{r}) - \Sigma_{\text{smp}}(\mathbf{r})) \varphi_-(\mathbf{r}, \hat{\Omega}) \right] \\
 &= \Theta(\Sigma_{\text{smp}}(\mathbf{r}) - \Sigma_t(\mathbf{r})) [\Sigma_{\text{smp}}(\mathbf{r}) - \Sigma_t(\mathbf{r})] \varphi_+(\mathbf{r}, \hat{\Omega}) + \Theta(\Sigma_t(\mathbf{r}) - \Sigma_{\text{smp}}(\mathbf{r})) [\Sigma_t(\mathbf{r}) - \Sigma_{\text{smp}}(\mathbf{r})] \varphi_-(\mathbf{r}, \hat{\Omega}) \\
 &= \Delta(\Sigma_{\text{smp}}(\mathbf{r}) - \Sigma_t(\mathbf{r})) \varphi_+(\mathbf{r}, \hat{\Omega}) + \Delta(\Sigma_t(\mathbf{r}) - \Sigma_{\text{smp}}(\mathbf{r})) \varphi_-(\mathbf{r}, \hat{\Omega}) \quad (\text{A.15})
 \end{aligned}$$

$$\hat{\Omega} \cdot \nabla \varphi_+ + \Sigma_{\text{smp}} \varphi_+ = \mathcal{S} \varphi_+ + \frac{1}{k} \mathcal{F} \varphi_+ + \Delta(\Sigma_{\text{smp}} - \Sigma_t) \varphi_+ + \Delta(\Sigma_t - \Sigma_{\text{smp}}) \varphi_- \quad (\text{A.16})$$

$$\hat{\Omega} \cdot \nabla \varphi_- + \Sigma_{\text{smp}} \varphi_- = \mathcal{S} \varphi_- + \frac{1}{k} \mathcal{F} \varphi_- + \Delta(\Sigma_{\text{smp}} - \Sigma_t) \varphi_- + \Delta(\Sigma_t - \Sigma_{\text{smp}}) \varphi_+. \quad (\text{A.17})$$

-
- [1] H. C. Berg, *Random Walks in Biology* (Princeton University Press, 1993).
- [2] M. Bartlett, *Stochastic Population Models in Ecology and Epidemiology*, Methuen's monographs on applied probability and statistics (Methuen, 1960).
- [3] M. M. R. Williams, *Random Processes in Nuclear Reactors*, 1st ed. (Pergamon Press Oxford, New York, 1974).
- [4] M. F. Modest, *Radiative Heat Transfer* (Academic press, 2013).
- [5] G. I. Bell and S. Glasstone, *Nuclear Reactor Theory* (Van Nostrand Reinhold Company, 1970).
- [6] D. M. Arnow, M. H. Kalos, M. A. Lee, and K. E. Schmidt, *J. Chem. Phys.* **77**, 5562 (1982).
- [7] T. Yamamoto, *Ann. Nucl. Energy* **58**, 72–79 (2013).
- [8] T. E. Booth, *Nucl. Sci. Eng.* **143**, 291 (2003).
- [9] L. Szirmay-Kalos, I. Georgiev, M. Magdics, B. Molnár, and D. Légrady, *Computer Graphics Forum* **36**, 9–19 (2017).
- [10] T. E. Booth and J. E. Gubernatis, *Phys. Rev. E* **80**, 046704 (2009).
- [11] T. E. Booth and J. E. Gubernatis, *Nucl. Sci. Eng.* **165**, 283 (2010).
- [12] J. S. Spencer, N. S. Blunt, and W. M. Foulkes, *J. Chem. Phys.* **136**, 054110 (2012).
- [13] M. H. Holmes, *Introduction to Scientific Computing and Data Analysis*, Texts in Computational Science and Engineering, Vol. 13 (Springer, 2016).
- [14] I. Lux and L. Koblinger, *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations* (CRC Press, 1991).
- [15] M. Nowak, J. Miao, E. Dumonteil, B. Forget, A. Onillon, K. S. Smith, and A. Zoia, *Ann. Nucl. Energy* **94**, 856–868 (2016).
- [16] A. Zoia, E. Dumonteil, and A. Mazzolo, *Phys. Rev. E* **83**, 041137 (2011).
- [17] C. Demazière, *Int. J. Nucl. Energy Sci. Technol.* **7**, 10.1504/IJNEST.2013.054368 (2013).
- [18] H. Belanger, D. Mancusi, and A. Zoia, *Eur. Phys. J. Plus* **135**, 877 (2020).
- [19] C. Demazière, *Modelling of Nuclear Reactor Multiphysics: From Local Balance Equations to Macroscopic Models in Neutronics and Thermal-hydraulics* (Academic Press, 2019).
- [20] E. Woodcock, T. Murphy, P. Hemmings, and S. Longworth, *Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry*, Tech. Rep. (Argonne National Laboratory, 1965) ANL-7050.
- [21] J. Leppänen, *Ann. Nucl. Energy* **37**, 715 (2010).
- [22] D. Legrady, B. Molnar, M. Klausz, and T. Major, *Ann. Nucl. Energy* **102**, 116 (2017).
- [23] B. Molnar, G. Tolnai, and D. Legrady, *Nucl. Sci. Eng.* **190**, 1 (2018).
- [24] L. L. Carter, E. D. Cashwell, and W. M. Taylor, *Nucl. Sci. Eng.* **48**, 403 (1972).
- [25] C. Jacoboni and L. Reggiani, *Rev. Mod. Phys.* **55**, 645–705 (1983).
- [26] V. Lemaire, M. Thieullen, and N. Thomas, *J. Sci. Comput.* **75**, 1776 (2018).
- [27] M. Galtier, S. Blanco, C. Caliot, C. Coustet, J. Dauchet, M. E. Hafi, V. Eymet, R. Fournier, J. Gautrais, A. Khuong, and et al., *J. Quant. Spectrosc. Radiat. Transfer* **125**, 57–68 (2013).
- [28] X-5 Monte Carlo Team, *MCNP — A General Monte Carlo N-Particle Transport Code, Version 5 Volume I: Overview and Theory* (2003), LA-UR-03-1987.
- [29] P. K. Romano, N. E. Horelik, B. R. Herman, A. G. Nelson, B. Forget, and K. Smith, *Ann. Nucl. Energy* **82**, 90 (2015).
- [30] T. E. Booth, *A weight (charge) conserving importance-weighted comb for Monte Carlo*, Tech. Rep. (Los Alamos National Laboratory, 1996) LA-UR 96-0051.

- [31] A. Rouchon, A. Zoia, and R. Sanchez, *Ann. Nucl. Energy* **102**, 465–475 (2017).
- [32] J. Demmel, in *IEEE Control Systems Society Workshop on Computer-Aided Control System Design* (1989) p. 1–7.
- [33] D. Brown, M. Chadwick, R. Capote, A. Kahler, A. Trkov, M. Herman, A. Sonzogni, Y. Danon, A. Carlson, M. Dunn, and et al., *Nucl. Data Sheets* **148**, 1 (2018).
- [34] A. J. M. Plompen, O. Cabellos, C. D. S. Jean, M. Fleming, A. Algora, M. Angelone, P. Archier, E. Bauge, O. Bersillon, A. Blokhin, and et al., *Eur. Phys. J. A* **56**, 181 (2020).
- [35] E. E. Lewis, M. A. Smith, N. Tsoulfanidis, G. Palmiotti, T. A. Taiwo, and R. N. Blomquist, *Benchmark specification for Deterministic 2-D/3-D MOX fuel assembly transport calculations without spatial homogenisation (C5G7 MOX)*, Tech. Rep. (NEA/NSC, 2001).
- [36] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, and et al., *Nature* **585**, 357–362 (2020).
- [37] H. Belanger, 10.5281/zenodo.4636567 (2021), <https://github.com/HunterBelanger/mgmc>.
- [38] J. Rice, *Mathematical Statistics and Data Analysis* (Duxbury Press, Belmont, California, 1995).
- [39] T. Yamamoto, *Ann. Nucl. Energy* **47**, 14–20 (2012).

9 - Unbiasedness and Optimization of Regional Weight Cancellation

This chapter has previously appeared as:

H. Belanger, D. Mancusi, and A. Zoia, "Unbiasedness and optimization of regional weight cancellation," *Physical Review E*, vol. 106, no. 2, p. 025302, 2022, doi: 10.1103/physreve.106.025302.

The preprint version appearing in this manuscript can be found at <https://arxiv.org/abs/2205.12062>.

Unbiasedness and Optimization of Regional Weight Cancellation

Hunter Belanger,^{*} Davide Mancusi,[†] and Andrea Zoia[‡]

Université Paris-Saclay, CEA, Service d'Études des Réacteurs et de Mathématiques Appliquées, 91191, Gif-sur-Yvette, France

The Monte Carlo method is often used to simulate systems which can be modeled by random walks. In order to calculate observables, in many implementations the “walkers” carry a statistical weight which is generally assumed to be positive. Some random walk simulations, however, may require walkers to have positive or negative weights: it has been shown that the presence of a mixture of positive and negative weights can impede the statistical convergence, and special weight-cancellation techniques must be adopted in order to overcome these issues. In a recent work we demonstrated the usefulness of one such method, exact regional weight cancellation, to solve eigenvalue problems in nuclear reactor physics in three spatial dimensions. The method previously exhibited had several limitations (including multi-group transport and isotropic scattering) and needed homogeneous cuboid cancellation regions. In this paper we lift the previous limitations, in view of applying exact regional cancellation to more realistic continuous-energy neutron transport problems. This extended regional cancellation framework is used to optimize the efficiency of the weight cancellation. Our findings are illustrated on a benchmark configuration for reactor physics.

I. INTRODUCTION

For day-to-day industrial needs in the field of nuclear reactor physics, deterministic codes are used to solve the neutron transport equation, estimating the reaction rates and the power distribution in the reactor core [1–3]. Deterministic methods have the advantage of running fast, but this speed comes at the cost of accuracy: many approximations are introduced, discretizing the phase space (position, direction and energy) and thus leading to a bias in the results. The gold-standard in reactor physics for solving the neutron transport equation is the Monte Carlo method, which does not need to introduce any discretization of the phase space, and is therefore free of any bias [4]. This high-fidelity simulation method comes at the cost of requiring extensive computer resources. Because of this computational cost, multi-physics simulations of a full-core nuclear reactor model, using Monte Carlo neutronics codes coupled with other state-of-the-art thermal-hydraulics and fuel performance codes, have become possible only very recently, mainly thanks to the increase in available computer power and to the development of efficient variance-reduction techniques [5, 6].

In these Monte Carlo simulations, the particles being simulated (typically neutrons or photons) carry a statistical weight, which is used to estimate observable quantities such as reaction rates and power distributions within the core of the nuclear reactor. For most applications involved in nuclear reactor physics or radiation shielding problems, these statistical weights are always positive. However, several types of Monte Carlo neutronics simulations require that the particles being sampled carry negative statistical weights (or complex weights, where each component is allowed to be negative). Problems

that require negative weights include the evaluation of the second harmonic of the flux, critical buckling, and neutron noise, as the quantities being estimated in these problems can be negative [7–9]. There are also special rejection sampling methods which allow negative weights, that could be used to treat spatially continuous material properties, even when the desired quantities should be positive [10]. Random walk problems using positive and negative statistical weights emerge more broadly in many applications outside of nuclear reactor physics, e.g. in quantum diffusion Monte Carlo [11], or in the Wigner Monte Carlo formalism [12]. Such simulations can be particularly challenging, as the summing of positive and negative contributions to estimate the observable quantities leads to very large variances in these tallies: it is often recognized that weight cancellation is mandatory to ensure convergence [7, 8, 13].

In a recent work, we have focused on the case of spatially continuous material properties for particle transport applications emerging in reactor physics. Material cross sections for neutron transport depend on the energy of the incident particle, as well as on the temperature and density of the material. Traditional neutronics codes (both Monte Carlo and deterministic) make the approximation that each material region in the reactor model has a constant temperature and density [1–3, 14–17]. In a real nuclear reactor, however, this is certainly not the case, as the temperature and density will depend continuously on position. The continual advances of high-performance computing resources allows us to consider new ways of improving the fidelity of our Monte Carlo codes. It is in this context that we have examined the possibility of treating spatially-continuous material temperatures and densities in Monte Carlo simulations in a previous work [10]. In particular, we have focused on assessing which particle-tracking methods might be best suitable to treat spatially-continuous cross sections for fixed-source transport problems, typically occurring in radiation shielding applications [10]. Among the possible choices, the delta-tracking [18, 19] and negative-

^{*} hunter.belanger@cea.fr

[†] davide.mancusi@cea.fr

[‡] andrea.zoia@cea.fr

weighted delta-tracking [20, 21] sampling strategies were deemed particularly attractive. Negative-weighted delta-tracking, although beneficial for dealing with spatially-continuous cross sections, has a potential drawback due to the statistical weights of the particles being allowed to become negative: in a subsequent study concerning k -eigenvalue problems, we have shown that the coupling of positive and negative particle weights prevents convergence of the power iteration method to the fundamental mode of the physical system being studied [22].

To overcome these issues, an exact regional weight cancellation method, originally proposed by Booth and Gubernatis in a 1D context [23], was extended to 3D and was shown to allow the convergence of power iteration with negative-weighted delta-tracking in a multi-group reactor physics benchmark [22]. Such a weight cancellation technique might be useful to improve the simulation methods of the other previously mentioned problems which have particles with negative statistical weights. While our previous work in Ref. 22 demonstrated potential for the method, many questions were left unanswered: under what conditions is regional cancellation unbiased? How might one extend cancellation from multi-group to continuous-energy material cross sections? Is it possible to maximize the efficiency of weight cancellation, for a given set of particles in a cancellation region? Our goal in this paper is to build upon our previous results in Ref. 22 and to start addressing these very questions.

Our manuscript is organised as follows. In Sec. II, we provide a brief summary of the exact regional cancellation technique which we presented in our previous paper. Section III will develop the mathematical theory behind the family of techniques for regional cancellation, for the most general continuous-energy case. We also examine what conditions must be satisfied to ensure that a regional cancellation method is unbiased. The theory presented in this section elucidates the concepts which are integral to the technique (for both multi-group and continuous-energy calculations), and blazes the trail for implementing exact regional cancellation in continuous-energy problems. The question of optimizing cancellation is then treated in Sec. IV, where two candidate optimization methods are proposed. Section V discusses the implementation of the two optimization strategies in our Monte Carlo code, and discusses how these two strategies allow us to deal with heterogeneous cancellation regions. The different optimization strategies are compared in Sec. VI, and we also assess the performances of our methods on a reactor physics benchmark with heterogeneous cancellation regions. Some concluding thoughts and remarks are provided in Sec. VII.

II. REGIONAL WEIGHT CANCELLATION

Previously, we have extended the 1D exact regional cancellation scheme of Booth and Gubernatis [23] to work in 3D multi-group neutron transport problems [22]. Here,

we shall briefly outline the mechanics of this method, in a general continuous-energy framework. For the case of k -eigenvalue problems, the fundamental mode and eigenvalue are sought by Monte Carlo methods using power iteration, which basically consists of following the neutron histories over fission generations [4]. When negative-weighted delta-tracking is used to sample particle flights, the transported neutron will have positive and negative weights, and weight cancellation will be mandatory to ensure the convergence of power iteration [22]. In this context, the regional cancellation operation is applied to neutrons born from fission. The fission particles are first sorted into user-defined cancellation regions, based on their position. Once all of the particles have been sorted into their cancellation regions, we may then consider each cancellation region independently for the cancellation procedure. In our previous work, a simple rectangular mesh was imposed on top of the problem geometry.

Consider cancellation region \mathcal{R} (which is assumed to be composed of only one fissile material), containing fission neutrons which have already been sampled. In addition to storing its own position (\mathbf{r}), energy (E), and direction ($\hat{\Omega}$), each fission particle also stores its parent's energy (E'), the position of the previous collision (\mathbf{r}'), and the direction of the parent's penultimate flight ($\hat{\Omega}''$).¹ From this information, we can calculate the “fission density function”, i.e. the expected fission density at \mathbf{r} due to a collision at \mathbf{r}' coming from direction $\hat{\Omega}''$, and a subsequent flight from \mathbf{r}' to \mathbf{r} at energy E' ; this is a key ingredient for the weight cancellation procedure. The exact form of the fission density function depends on the particle tracking method being used. For the case of negative-weighted delta-tracking,² as examined in our previous work, the fission density function was taken to be

$$\zeta(\mathbf{r}|\mathbf{r}', \hat{\Omega}'', E') = \frac{\mathcal{P}\left(\frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|} \cdot \hat{\Omega}''\right) \Sigma_f(\mathbf{r}, E')}{2\pi|\mathbf{r}-\mathbf{r}'|^2} e^{-\Sigma_{\text{smp}}(E')|\mathbf{r}-\mathbf{r}'|}. \quad (1)$$

In this notation, Σ_{smp} is the sampling cross-section required for negative-weighted delta tracking, Σ_f is the fission cross-section, and \mathcal{P} is the probability density function for the cosine of the scattering angle for the previous collision.³ Based on ζ , we are able to split each fission

¹ The direction of the parent's last flight is not explicitly stored, as it can be calculated as $\hat{\Omega}' = \frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|}$.

² We will only mention negative-weighted delta-tracking in the text, since that was the focus of our previous work, but Eq. (1) is also valid for regular delta-tracking, where Σ_{smp} would be the majorant cross section. This could be of use for neutron noise or critical buckling problems, which would not necessarily require the use of negative-weighted delta-tracking but nonetheless require weight cancellation.

³ While the symbol f was used for the fission density function in Ref. 22, we have instead chosen to use ζ in this paper, to avoid any confusion with other subsequent symbols.

particle in \mathcal{R} into two components: a point-wise component with weight w_p , and a uniform component of weight w_u . The point-wise portion, w_p , keeps the phase space coordinates $(\mathbf{r}, \hat{\Omega}, E)$ of the split fission particle. The uniform component, w_u , is spread uniformly over the region \mathcal{R} . To calculate the point-wise and uniform weights, we use

$$w_p = \frac{\zeta(\mathbf{r}|\mathbf{r}', \hat{\Omega}'', E') - \beta}{\zeta(\mathbf{r}|\mathbf{r}', \hat{\Omega}'', E')} w \quad (2)$$

$$w_u = \frac{\beta}{\zeta(\mathbf{r}|\mathbf{r}', \hat{\Omega}'', E')} w, \quad (3)$$

respectively, with w being the weight of the original fission particle [23]. Note that $w_p + w_u = w$, so that the net weight is conserved. The free parameter β can take any value, and in general is chosen independently for each particle in \mathcal{R} . Our previous work followed the recommendation of Booth and Gubernatis, and always took β to be the minimum value of $\zeta(\mathbf{r}''|\mathbf{r}', \hat{\Omega}'', E')$ over all possible $\mathbf{r}'' \in \mathcal{R}$, for the particle of interest. We demonstrated that, for the case of isotropic scattering and cuboid cancellation regions, one only needs to evaluate $\zeta(\mathbf{r}''|\mathbf{r}', \hat{\Omega}'', E')$ for the eight corners of the cuboid (taking \mathbf{r}'' to be the corner positions) to find the minimum value within \mathcal{R} .

With w_p and w_u having been calculated for each fission particle in \mathcal{R} , we then take the sum of all the uniform weight components

$$U = \sum_{i=1}^N w_{u,i}, \quad (4)$$

where the extra subscript i indicates the fission particle. This operation is effectively where the cancellation occurs: depending on the initial weights w_i of the fission particles, the individual uniform components $w_{u,i}$ will be positive or negative, and taking their sum cancels some of the positive and negative weight which was in the region \mathcal{R} . The uniform weight U must be distributed uniformly within \mathcal{R} . To do this, $n = \lceil |U| \rceil$ new fission particles are sampled within \mathcal{R} , each having a weight of U/n . The positions of the n uniform particles are sampled uniformly in \mathcal{R} . In our previous work, the direction was sampled from an isotropic distribution, as fission was assumed to be perfectly isotropic, and the energy was sampled from the fission spectrum of the material in \mathcal{R} , as it was assumed that the fission spectrum had no dependence on incident neutron energy. These n new uniform fission particles must be added to the fission bank, and will then be transported along with the other fission particles during the next fission generation.

The method proposed in Ref. 22 that we have recalled here, was demonstrated to work successfully and be unbiased on a simple reactor physics benchmark problem. While those results were very promising, the initial implementation admittedly had several limitations. First, cancellation regions must be homogeneous, containing

only a single fissile material. Second, fission must always be isotropic, and the fission spectrum must be independent of the incident energy. In general, even in continuous-energy transport, fission is almost always represented as isotropic, so this is not necessarily a large inconvenience. However, the fission energy spectrum is generally assumed to be dependent on the incident neutron energy. Furthermore, while Booth and Gubernatis argue that cancellation is unbiased for any value of the parameter β , the amount of canceled weight (and thus the efficiency of the method) clearly does depend on β . Using the minimum value of the fission density as β is not necessarily the most efficient choice for achieving the highest amount of cancellation. Nonetheless, taking β to be the minimum within \mathcal{R} guarantees that both w_p and w_u have the same sign as w : when β is larger than the minimum, the point-wise portion, w_p , can change sign, potentially leading to even more positive and negative weight in the region than there was initially. The cancellation operation does not change the net weight W_{net} in the bin, as

$$W_{\text{net}} = \sum_i^N w_i = \sum_i^N w_{u,i} + \sum_i^N w_{p,i} = U + \sum_i^N w_{p,i} \quad (5)$$

will still be located in the bin. However, cancellation does change the total weight, W_{tot} , defined as the sum of the absolute values of all weights. The total weight before cancellation is

$$W_{\text{tot}} = \sum_i^N |w_i|, \quad (6)$$

while the post-cancellation total weight is

$$W_{\text{tot,post}} = \sum_i^N |w_{p,i}| + |U| = \sum_i^N |w_{p,i}| + \left| \sum_i^N w_{u,i} \right|. \quad (7)$$

By using the triangle inequality, it is possible to show that

$$W_{\text{tot,post}} \geq |W_{\text{net}}|.$$

The more efficient cancellation is, the closer $W_{\text{tot,post}}$ will be to W_{net} , with 100% cancellation efficiency corresponding to $W_{\text{tot,post}} = W_{\text{net}}$ (i.e. all negative weight is removed). The optimal choice for β will maximize the cancellation efficiency, and therefore minimize $W_{\text{tot,post}}$. This optimal choice of β is clearly dependent on the other particles in the bin, and determining this optimal value is vital for improving the overall computational efficiency of the simulation.

III. UNBIASEDNESS OF CANCELLATION

In this section, a method for performing exact regional cancellation in general continuous-energy problems shall be developed, and it will be demonstrated

under what conditions such schemes lead to an unbiased fission source. For this purpose, it is mathematically beneficial to use the integral form of the transport equation, as opposed to the integro-differential form adopted in our previous work. We will begin by presenting the integral transport form for the eigenvalue transport problem in Sec. III A. Section III B makes a first attempt at developing an estimator for the fission emission density in a region, which averages over all possible collisions and subsequent flights which induce the fission. While this exact estimator is likely of little use to a practical application, we are able to use it to examine what requirements must be observed in order to have an unbiased fission emission density estimator. Section III C discusses how far back in a particle's history one must look, so that it is possible for it to have contributed to the fission emission density everywhere within the cancellation region. Section III D uses the ideas from Section III C to decompose the collision operator as is done in most Monte Carlo codes, to produce a fission emission density estimator which could potentially be used in an industrial code to achieve exact regional cancellation. Section III E outlines the possibility of distributing some of the fission emission density within the region according to a generic function, instead of distributing it uniformly. Finally, Section III F examines why delta-tracking algorithms are more suited to exact regional cancellation, and the peculiarities which can arise from delta-scatters.

A. Integral Formulation of the Transport Equation

We will start with the k -eigenvalue Boltzmann transport equation in integral form. Let $P = (\mathbf{r}, \hat{\Omega}, E)$ denote the coordinates of a point in phase space. The collision density $\psi(P) = \Sigma_t(\mathbf{r}, E)\varphi(\mathbf{r}, \hat{\Omega}, E)$ and the emission density $\chi(P)$ are related by [4]:

$$\psi(P) = \mathbb{T}\chi(P) \quad (8)$$

$$\chi(P) = \left[\mathbb{C}_s + \frac{1}{k}\mathbb{C}_f \right] \psi(P), \quad (9)$$

where Σ_t is the total macroscopic cross section, and φ is the angular neutron flux. In this notation, \mathbb{T} is the flight operator, defined as

$$\mathbb{T}g(P) = \int T(P' \rightarrow P)g(P')dP', \quad (10)$$

where we have made use of the flight kernel

$$T(P' \rightarrow P) = \frac{\Sigma_t(\mathbf{r}, E')}{|\mathbf{r} - \mathbf{r}'|^2} \exp\left(-\int_0^{|\mathbf{r} - \mathbf{r}'|} \Sigma_t(\mathbf{r}' + u\hat{\Omega}', E')du\right) \delta\left(\hat{\Omega} - \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|}\right) \delta(\hat{\Omega}' - \hat{\Omega}) \delta(E - E'). \quad (11)$$

We note that the flight kernel $T(P' \rightarrow P)$ is normalized, and can be interpreted as the probability density function (PDF) for a particle having a flight and landing at the phase space coordinate P , conditioning on its initial phase space coordinate being P' .

The scattering operator \mathbb{C}_s in Eq. (9) is defined as

$$\mathbb{C}_s g(P) = \int C_s(P' \rightarrow P)g(P')dP', \quad (12)$$

with the scattering kernel $C_s(P' \rightarrow P)$ being

$$C_s(P' \rightarrow P) = \frac{\nu_s(\mathbf{r}', E')\Sigma_s(\mathbf{r}', E')}{\Sigma_t(\mathbf{r}', E')} f_{\text{scat}}\left(\hat{\Omega}, E|\mathbf{r}', \hat{\Omega}', E'\right) \delta(\mathbf{r} - \mathbf{r}'), \quad (13)$$

where Σ_s is the macroscopic scattering cross section, ν_s is the average number of neutrons emitted from a scatter, and f_{scat} is the joint PDF for a neutron to scatter in direction $\hat{\Omega}$ at energy E . The fission operator \mathbb{C}_f is similar to the scattering operator in Eq. (12), but instead uses a fission kernel $C_f(P' \rightarrow P)$, defined as

$$C_f(P' \rightarrow P) = \frac{\nu_f(\mathbf{r}', E')\Sigma_f(\mathbf{r}', E')}{\Sigma_t(\mathbf{r}', E')} f_{\text{fiss}}\left(\hat{\Omega}, E|\mathbf{r}', \hat{\Omega}', E'\right) \delta(\mathbf{r} - \mathbf{r}'), \quad (14)$$

where Σ_f is the macroscopic fission cross section, ν_f is the average number of neutrons produced per fission, and f_{fiss} is the joint PDF for fission neutrons to be emitted in direction $\hat{\Omega}$ at energy E . The scattering and fission operators may be combined into a collision operator

$$\chi(P) = \mathbb{C}\psi(P), \quad (15)$$

which has a corresponding collision kernel

$$C(P' \rightarrow P) = C_s(P' \rightarrow P) + \frac{1}{k}C_f(P' \rightarrow P). \quad (16)$$

Here $C(P' \rightarrow P)$ can be interpreted as the average number of particles produced about the phase space coordinate P , from a collision induced by a particle at P' . Given this interpretation, it is also possible to rewrite $C(P' \rightarrow P)$ in a more concise form, using an average yield $\bar{\nu}(\mathbf{r}', E')$, and an average transfer function $\bar{f}(\hat{\Omega}, E|\mathbf{r}', \hat{\Omega}', E')$:

$$C(P' \rightarrow P) = \bar{\nu}(\mathbf{r}', E')\bar{f}\left(\hat{\Omega}, E|\mathbf{r}', \hat{\Omega}', E'\right) \delta(\mathbf{r} - \mathbf{r}'). \quad (17)$$

It is clear that Eq. (17) is true if

$$\bar{\nu}(\mathbf{r}', E') = \frac{\nu_s(\mathbf{r}', E')\Sigma_s(\mathbf{r}', E')}{\Sigma_t(\mathbf{r}', E')} + \frac{\nu_f(\mathbf{r}', E')\Sigma_f(\mathbf{r}', E')}{k\Sigma_t(\mathbf{r}', E')} \quad (18)$$

and

$$\begin{aligned} \bar{f}\left(\hat{\Omega}, E|\mathbf{r}', \hat{\Omega}', E'\right) = & \\ & \frac{\nu_s(\mathbf{r}', E')\Sigma_s(\mathbf{r}', E')}{\bar{v}(\mathbf{r}', E')\Sigma_t(\mathbf{r}', E')} f_{\text{scat}}\left(\hat{\Omega}, E|\mathbf{r}', \hat{\Omega}', E'\right) + \\ & \frac{\nu_f(\mathbf{r}', E')\Sigma_f(\mathbf{r}', E')}{k\bar{v}(\mathbf{r}', E')\Sigma_t(\mathbf{r}', E')} f_{\text{fiss}}\left(\hat{\Omega}, E|\mathbf{r}', \hat{\Omega}', E'\right). \end{aligned} \quad (19)$$

B. Averaging over all Scattering Events

Consider the following particle history. A neutron enters a collision at P_1 , and then leaves that collision at

P_2 . The particle then undergoes a flight and experiences a fission at P_3 . The fission at P_3 then contributes to the fission emission density at P_4 . It is assumed that P_4 is located within the generalized phase space region \mathcal{R} , which will act as our cancellation region.⁴ This partial particle history is depicted in Fig. 1. Despite the fact that $\mathbf{r}_4 = \mathbf{r}_3$, P_3 is not, in general, located in the cancellation region \mathcal{R} , as $\hat{\Omega}_3$ and E_3 may not be within the domain of \mathcal{R} . In order to examine the fission emission density at point P_4 , we must first determine the collision density $\psi(P_3)$, for a given collision at P_1 . From Eq. (8) and Eq. (9), it follows that

$$\psi(P_3) = \mathbb{T}\mathbb{C}\psi(P_3) = \int T(P_2 \rightarrow P_3) \int C(P_1 \rightarrow P_2)\psi(P_1)dP_1dP_2 \quad (20)$$

$$\begin{aligned} = & \iiint d\mathbf{r}_1 d\hat{\Omega}_1 dE_1 \iiint d\mathbf{r}_2 d\hat{\Omega}_2 dE_2 \psi\left(\mathbf{r}_1, \hat{\Omega}_1, E_1\right) \bar{v}(\mathbf{r}_1, E_1) \bar{f}\left(\hat{\Omega}_2, E_2|\mathbf{r}_1, \hat{\Omega}_1, E_1\right) \delta(\mathbf{r}_2 - \mathbf{r}_1) \\ & \Sigma_t(\mathbf{r}_3, E_3) \exp\left(-\int_0^{|\mathbf{r}_3 - \mathbf{r}_2|} \Sigma_t\left(\mathbf{r}_2 + u\hat{\Omega}_2, E_2\right) du\right) \frac{\delta\left(\hat{\Omega}_3 - \frac{\mathbf{r}_3 - \mathbf{r}_2}{|\mathbf{r}_3 - \mathbf{r}_2|}\right) \delta\left(\hat{\Omega}_3 - \hat{\Omega}_2\right) \delta(E_3 - E_2)}{|\mathbf{r}_3 - \mathbf{r}_2|^2} \end{aligned} \quad (21)$$

$$\begin{aligned} = & \iiint d\mathbf{r}_1 d\hat{\Omega}_1 dE_1 \psi\left(\mathbf{r}_1, \hat{\Omega}_1, E_1\right) \bar{v}(\mathbf{r}_1, E_1) \\ & \delta\left(\hat{\Omega}_3 - \frac{\mathbf{r}_3 - \mathbf{r}_1}{|\mathbf{r}_3 - \mathbf{r}_1|}\right) \frac{\bar{f}\left(\hat{\Omega}_3, E_3|\mathbf{r}_1, \hat{\Omega}_1, E_1\right) \Sigma_t(\mathbf{r}_3, E_3)}{|\mathbf{r}_3 - \mathbf{r}_1|^2} \exp\left(-\int_0^{|\mathbf{r}_3 - \mathbf{r}_1|} \Sigma_t\left(\mathbf{r}_1 + u\hat{\Omega}_3, E_3\right) du\right). \end{aligned} \quad (22)$$

The integral over P_1 in Eq. (22) indicates that $\psi(P_3)$ is a sum of contributions from all possible initial phase space points P_1 for which $\hat{\Omega}_3 = \frac{\mathbf{r}_3 - \mathbf{r}_1}{|\mathbf{r}_3 - \mathbf{r}_1|}$.

The fission emission density $\chi_f(P_4)$ is defined as

$$\chi_f(P_4) = \frac{1}{k} \mathbb{C}_f \psi(P_4). \quad (23)$$

Combining Eq. (14) and Eq. (22), we obtain

$$\begin{aligned} \chi_f(P_4) = & \iiint d\mathbf{r}_1 d\hat{\Omega}_1 dE_1 \psi\left(\mathbf{r}_1, \hat{\Omega}_1, E_1\right) \bar{v}(\mathbf{r}_1, E_1) \\ & \int dE_3 \frac{\nu_f(\mathbf{r}_4, E_3)\Sigma_f(\mathbf{r}_4, E_3)f_{\text{fiss}}\left(\hat{\Omega}_4, E_4|\mathbf{r}_4, \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3\right) \bar{f}\left(\frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3|\mathbf{r}_1, \hat{\Omega}_1, E_1\right)}{k|\mathbf{r}_4 - \mathbf{r}_1|^2} \\ & \exp\left(-\int_0^{|\mathbf{r}_4 - \mathbf{r}_1|} \Sigma_t\left(\mathbf{r}_1 + u\frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3\right) du\right) \\ = & \iiint \psi\left(\mathbf{r}_1, \hat{\Omega}_1, E_1\right) \bar{v}(\mathbf{r}_1, E_1) \zeta(P_1 \rightarrow P_4) d\mathbf{r}_1 d\hat{\Omega}_1 dE_1. \end{aligned} \quad (24)$$

⁴ While our previous work in Ref. 22 used cancellations which only spanned space, we now consider cancellation regions spanning all dimensions of phase space. We therefore must consider three spa-

tial dimensions, two dimensions for direction, and one dimension for energy.

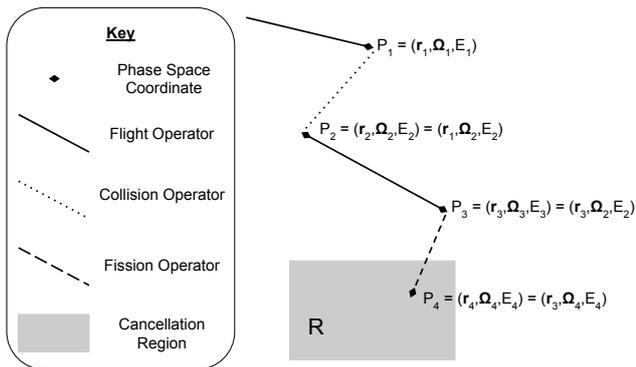


FIG. 1. Depicted here is the relationship between phase space points P_1 , P_2 , P_3 , and P_4 . Points connected by a flight operator (solid line) share the same direction and energy, and are only discontinuous in position. Points connected by a collision or fission operator (dotted and dashed lines respectively) share the same position, but are generally discontinuous in direction and energy. Any branches which might result from the application of the collision operator from P_1 to P_2 are not depicted.

In the last step we have introduced the function ζ :

$$\zeta(P_1 \rightarrow P_4) = \int dE_3 \frac{\nu_f(\mathbf{r}_4, E_3) \Sigma_f(\mathbf{r}_4, E_3) f_{\text{fiss}}(\hat{\Omega}_4, E_4 | \mathbf{r}_4, \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3) \bar{f}(\frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3 | \mathbf{r}_1, \hat{\Omega}_1, E_1)}{k|\mathbf{r}_4 - \mathbf{r}_1|^2} \exp\left(-\int_0^{|\mathbf{r}_4 - \mathbf{r}_1|} \Sigma_t\left(\mathbf{r}_1 + u \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3\right) du\right) \quad (25)$$

Here $\zeta(P_1 \rightarrow P_4)$ is the transition kernel for a particle starting at P_1 , undergoing a collision, then a flight, and then producing fission particles at P_4 .

We now wish to construct an estimator for the expected fission emission density at a point $Q \in \mathcal{R}$. Our estimator operates on events where a fission particle is emitted at $P_4 \in \mathcal{R}$, from a particle originally entering a collision at P_1 . To be unbiased, our estimator $\vartheta(P_1 \rightarrow P_4 | \mathcal{R}, Q)$ for the fission emission density must have the property [4]

$$\int \zeta(P_1 \rightarrow P_4) \vartheta(P_1 \rightarrow P_4 | \mathcal{R}, Q) dP_4 = \zeta(P_1 \rightarrow Q). \quad (26)$$

In order to achieve regional cancellation, we would like to define an estimator ϑ for the fission emission density at Q where a portion of the fission emission density is located exactly at Q , and the remaining portion is uniformly distributed within the phase space region \mathcal{R} . We shall define this estimator to have the form

$$\vartheta_\eta(P_1 \rightarrow P_4 | \mathcal{R}, Q) = (1 - \eta) \delta(Q - P_4) + \frac{\eta}{\mathcal{V}_{\mathcal{R}}}. \quad (27)$$

Here, $\mathcal{V}_{\mathcal{R}}$ is the generalized phase space volume occu-

ried by \mathcal{R} , and η is the portion of the fission emission density that we wish to uniformly distribute within \mathcal{R} .⁵ If η is taken to be a constant with respect to P_4 , then, upon evaluation of the left-hand side of Eq. (26), using Eq. (27), we obtain:

$$\begin{aligned} \int \zeta(P_1 \rightarrow P_4) \vartheta_\eta(P_1 \rightarrow P_4 | \mathcal{R}, Q) dP_4 &= \\ \int \zeta(P_1 \rightarrow P_4) \left[(1 - \eta) \delta(Q - P_4) + \frac{\eta}{\mathcal{V}_{\mathcal{R}}} \right] dP_4 &= \\ (1 - \eta) \zeta(P_1 \rightarrow Q) + \frac{\eta}{\mathcal{V}_{\mathcal{R}}} \int \zeta(P_1 \rightarrow P_4) dP_4. \end{aligned} \quad (28)$$

Comparing Eq. (28) and Eq. (26), it is clear that the only unbiased option is $\eta = 0$, corresponding to no cancellation. It is permissible however to allow $\eta = \eta(P_1, P_4)$ to be both a function of P_1 and P_4 , as ϑ is already a function of these parameters. Using the ansatz

$$\eta(P_1, P_4) = \frac{\beta}{\zeta(P_1 \rightarrow P_4)}, \quad (29)$$

inspired by Eq. (3), we see that

⁵ Note that the parameter η may take any value (real or complex);

in particular, it is not required to lie in the $[0, 1]$ interval.

$$\int \zeta(P_1 \rightarrow P_4) \vartheta_{\eta(P_1, P_4)}(P_1 \rightarrow P_4 | \mathcal{R}, Q) dP_4 = \int \zeta(P_1 \rightarrow P_4) \left[\left(1 - \frac{\beta}{\zeta(P_1 \rightarrow P_4)} \right) \delta(Q - P_4) + \frac{\beta}{\mathcal{V}_{\mathcal{R}} \zeta(P_1 \rightarrow P_4)} \right] dP_4 = \zeta(P_1 \rightarrow Q) - \beta + \beta = \zeta(P_1 \rightarrow Q), \quad (30)$$

which, compared with right-hand side of Eq. (26), shows that this choice leads to an unbiased estimator.

Equation (30) indicates that we are allowed to distribute a factor $\beta/\zeta(P_1 \rightarrow P_4)$ of the fission particle uniformly within \mathcal{R} , so long as β has no functional dependence on P_4 . This requirement on β is essential to ensure that, after integrating over P_4 , both β terms will cancel; note however that β is allowed to depend on P_1 . We are therefore allowed to pick $\beta = 0$ whenever it is convenient, so long as information from P_4 is not used to make this choice.

Additionally, Eq. (30) indicates that we must require $\zeta(P_1 \rightarrow P_4) \neq 0 \forall P_4 \in \mathcal{R}$. If this is not the case, then η is undefined. In particular, this implies that we require $\Sigma_f(\mathbf{r}_4, E_3) > 0$ everywhere within our cancellation region. We must also require $f_{\text{fiss}}(\hat{\mathbf{\Omega}}_4, E_4 | \mathbf{r}_4, \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3) > 0 \forall P_4 \in \mathcal{R}$; as fission is nearly perfectly isotropic, the angular component is not problematic, but the energy component could indeed be zero for very low energies, and care must therefore be taken when selecting the energy bounds for \mathcal{R} . Despite these restrictions, we are given some liberty as to the definition of \mathcal{R} , as it is allowed to be non-convex.

The ansatz of Eq. (29) has the following remarkable property. Suppose that $\zeta(P_1 \rightarrow P_4)$ has the structure

$$\zeta(P_1 \rightarrow P_4) = h_0(P_1) \zeta_0(P_1 \rightarrow P_4). \quad (31)$$

Consider the estimators

$$\vartheta_{\zeta} = \vartheta \left(P_1 \rightarrow P_4 | \mathcal{R}, Q, \eta = \frac{\beta}{\zeta} \right) \quad (32)$$

$$\vartheta_{\zeta_0} = \vartheta \left(P_1 \rightarrow P_4 | \mathcal{R}, Q, \eta = \frac{\beta_0}{\zeta_0} \right). \quad (33)$$

The two estimators are actually identical for $\beta_0 = \beta/h_0$. In other words, any factor in $\zeta(P_1 \rightarrow P_4)$ that is independent of P_4 can be pulled out of the definition of ζ and still yield an unbiased estimator for the fission emission density.

Finally, note that the integral over E_3 in the definition of ζ could be somewhat problematic and/or expensive to compute in a continuous-energy Monte Carlo code. It effectively corresponds to averaging over all possible nuclides, reaction channels, and energies, which could have been sampled in determining P_3 , and leading to a fission particle at P_4 . We therefore would like to determine if there is an alternative, simpler, unbiased option.

C. Expected-Value Estimators for Cancellation

As it has been developed, $\zeta(P_1 \rightarrow P_4)$ can be interpreted as a type of expected-value estimator [24], because it is the expected contribution to the fission emission density at P_4 , for a particle entering a collision at P_1 . For the purpose of carrying out weight cancellation, several kinds of expected-value estimators for the fission emission density at P_4 could potentially be used in place of the form given by Eq. (25). All that is required of $\zeta(P_1 \rightarrow P_4)$ is that it be non-zero for all points P_4 in \mathcal{R} . This is required by Eq. (30), as we can only distribute fission emission density uniformly within \mathcal{R} if $\zeta(P_1 \rightarrow P_4) > 0 \forall P_4 \in \mathcal{R}$. With this in mind, we will now consider what types of expected-value estimators could be used in lieu of ζ .

Next-event estimators are particular forms of expected-value estimators that average the sampled quantity over the following event in the stochastic process. Let us evaluate if a next-event estimator is suitable for the purpose of cancellation. Consider a next-fission estimator for the fission emission density; such an estimator is applied to particles undergoing a collision at $P_3 = (\mathbf{r}_3, \hat{\mathbf{\Omega}}_3, E_3)$ and yields the expected fission emission density at a generic point $Q = (\mathbf{r}, \hat{\mathbf{\Omega}}, E) \in \mathcal{R}$. Since fission does not change the position of particles, the contribution of the next-fission estimator vanishes everywhere except for $\mathbf{r} = \mathbf{r}_3$. Therefore, a next-fission estimator is not able to yield a non-vanishing contribution at all the points in cancellation region \mathcal{R} .

Thus, in order for cancellation to be possible, we need to include more than one event in our expected-value estimator, i.e. we need to use at least a next-next-event estimator, or possibly an estimator of even higher order. It is then crucial to determine the number of events that our estimator needs to look ahead and average over, in order to yield a non-vanishing contribution to all the phase space points in the cancellation region. Indeed, we want to *minimize* the number of look-ahead events, because the evaluation of expected-value estimators becomes more and more cumbersome as the number of look-ahead events increases.

Consider now the possibility of a next-flight-fission estimator. In our notation, such an estimator acts on particles emitted at P_2 and yields the expected fission emission density at Q , averaged over all possible flights from P_2 and all possible fission events. Since the flight operator does not modify the particle direction (see Eq. (11)),

the expected fission density vanishes everywhere except at positions reachable from \mathbf{r}_2 with direction $\hat{\Omega}_2$. In general, this does not cover the whole cancellation region, except in the one-dimensional case [22, 23].

It is now probably clear that a next-collision-flight-fission estimator should *in general* yield a non-vanishing contribution to the fission emission density at all phase space points within \mathcal{R} . In other words, given a particle undergoing a collision at P_1 , the expected fission emission density (averaged over the next collision, flight, and fission) should not vanish anywhere within \mathcal{R} . This corresponds to Eq. (25) above and justifies the construction of the previous section.

Two remarks are in order here. First, there are cases where even a next collision-flight-fission estimator is not sufficient to achieve a non-vanishing expected fission emission density at all points within \mathcal{R} . Indeed, the collision between P_1 and P_2 may be subject to kinematic constraints, and Eq. (25) shows that the expected fission emission density vanishes if $\bar{f}\left(\frac{\mathbf{r}-\mathbf{r}_1}{|\mathbf{r}-\mathbf{r}_1|}, E_3|\mathbf{r}_1, \hat{\Omega}_1, E_1\right) = 0$ for some \mathbf{r} in cancellation region \mathcal{R} . Second, an estimator based on Eq. (25) would require the evaluation of the integral over E_3 at every collision, which is impractical. In fact, regional cancellation attempts to perform the cancellation algorithm a posteriori, after flights have already been sampled, and fission particles have been produced. Since we have already sampled a Monte Carlo history from P_1 through P_2 , P_3 , and P_4 , we would like to reuse as much information as possible from the sampled history to remove part of the fission density from P_4 and redistribute it uniformly within \mathcal{R} .

Thus, our expected-value estimator needs to average over sufficiently many event samplings to be able to “see” the whole region \mathcal{R} ; at the same time, we want our estimator to average over the strict minimum number of samplings. Each additional real variable that we average over introduces an extra integration in the expression of the expected fission emission density and reduces the usefulness of the $P_1 \rightarrow P_4$ history that we have already sampled.

D. Intermediate Collision Points

For the subsequent analysis, it is useful to consider a different form of the collision kernel, more aligned with

how most continuous-energy Monte Carlo codes sample a collision event. While Eq. (17) presents the collision kernel in terms of the averaged macroscopic cross sections and yields, most continuous-energy Monte Carlo codes do not handle collisions in such a manner. In production-level codes, microscopic cross sections are tabulated for different nuclides and different reaction channels (elastic, level inelastic, etc.) [14–17]. Each combination of nuclide and reaction channel has an independent transfer function for each type of non-capture collision. The concentration $N_i(\mathbf{r}')$ of nuclide i is a function of position, and the total microscopic cross section $\sigma_i(\mathbf{r}', E')$ is a function of position and energy.⁶ The total macroscopic cross section is

$$\Sigma_t(\mathbf{r}', E') = \sum_i N_i(\mathbf{r}')\sigma_i(\mathbf{r}', E'). \quad (34)$$

At a collision site, we select the nuclide with which our particle will undergo a collision: nuclide i is chosen with probability $N_i(\mathbf{r}')\sigma_i(\mathbf{r}', E')/\Sigma_t(\mathbf{r}', E')$. With nuclide i having been sampled, a reaction channel m must next be sampled. If we let $\sigma_{i,m}(\mathbf{r}', E')$ be the partial microscopic cross section for channel m , then the total microscopic cross section is

$$\sigma_i(\mathbf{r}', E') = \sum_m \sigma_{i,m}(\mathbf{r}', E'), \quad (35)$$

and channel m will be selected with probability $\sigma_{i,m}(\mathbf{r}', E')/\sigma_i(\mathbf{r}', E')$. This channel has an associated yield of $\nu_{i,m}(E')$, and transfer function $f_{i,m}(\hat{\Omega}, E|\hat{\Omega}', E')$. Continuous-energy nuclear data files typically give $f_{i,m}$ as a product of a marginal PDF in energy and a conditional PDF in direction:

$$f_{i,m}(\hat{\Omega}, E|\hat{\Omega}', E') = f_{i,m}(E|\hat{\Omega}', E') f_{i,m}(\hat{\Omega}|\hat{\Omega}', E', E). \quad (36)$$

When this is the case, the energy E is first sampled from the marginal PDF, and the direction is subsequently sampled from the conditional PDF. With these provisions, it is then possible to write the collision kernels, Eqs. (13) and (14), as

⁶ The microscopic cross section is typically given as a function of temperature and energy. However, since the temperature is a function of position, we have chosen to present the microscopic

cross section as a function of position and energy, to avoid the introduction of a superfluous variable.

$$C_s(P' \rightarrow P) = \frac{\delta(\mathbf{r} - \mathbf{r}')}{\Sigma_t(\mathbf{r}', E')} \sum_i N_i(\mathbf{r}') \sum_{\substack{m \\ m \neq \text{fiss}}} \nu_{i,m}(E') \sigma_{i,m}(\mathbf{r}', E') f_{i,m}(E|\hat{\Omega}', E') f_{i,m}(\hat{\Omega}|\hat{\Omega}', E', E) \quad (37a)$$

$$C_f(P' \rightarrow P) = \frac{\delta(\mathbf{r} - \mathbf{r}')}{\Sigma_t(\mathbf{r}', E')} \sum_i N_i(\mathbf{r}') \nu_{i,\text{fiss}}(E') \sigma_{i,\text{fiss}}(\mathbf{r}', E') f_{i,\text{fiss}}(E|\hat{\Omega}', E') f_{i,\text{fiss}}(\hat{\Omega}|\hat{\Omega}', E', E). \quad (37b)$$

Based on the form of Eqs. (37), we introduce then the concept of an “intermediate collision point”, indicating that the required pieces of information are sampled incrementally when performing a collision. Examples of intermediate collision points would be the state where we have sampled only the nuclide, or the nuclide and the channel, or the nuclide, reaction channel, and energy. With the concept of an intermediate collision, it is then evident that there is an intermediate collision point between P_1 and P_2 , where the particle has selected an isotope to collide with (i), a reaction channel (m), and even an outgoing energy (E_2), but has yet to select a

direction $\hat{\Omega}_2$ out of the collision. In general, this intermediate collision point between P_1 and P_2 is the strict minimum number of steps we must look back in a particle’s history, in order to see a nonzero fission emission density everywhere within \mathcal{R} (assuming that is possible to scatter into all directions subtended by \mathcal{R}). This state is accessible in a Monte Carlo simulation, as the nuclide, reaction channel, and energy E_3 were all sampled when producing the fission particle at P_4 , and this information can be stored with the particle. The transition kernel from P_1 to P_4 , given a collision with nuclide i in reaction channel m and outgoing energy E_3 , is then

$$\zeta(P_1 \rightarrow P_4|i, m, E_3) = \frac{\nu_f(\mathbf{r}_4, E_3) \Sigma_f(\mathbf{r}_4, E_3) f_{\text{fiss}}(\hat{\Omega}_4, E_4|\mathbf{r}_4, \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3) f_{i,m}(\frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1}| \hat{\Omega}_1, E_1, E_3)}{k|\mathbf{r}_4 - \mathbf{r}_1|^2} \times \exp\left(-\int_0^{|\mathbf{r}_4 - \mathbf{r}_1|} \Sigma_t\left(\mathbf{r}_1 + u \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3\right) du\right). \quad (38)$$

This is now quite reminiscent of the fission density function which we used in our previous work [22], as summarized in Sec. II (see Eq. (1)). It is worth stressing that Eq. (38) uses the macroscopic fission cross section Σ_f and the average fission transfer function f_{fiss} , which are averaged over all fissile nuclides at \mathbf{r}_4 . In general, these quantities might vary within \mathcal{R} , due to spatial dependence in the nuclide concentrations and temperature. Examining under what circumstances the estimator $\vartheta(P_1 \rightarrow P_4|\mathcal{R}, Q, \eta)$ is unbiased for the transition kernel presented in Eq. (38), it is straightforward to observe that this condition is met for the choice of $\eta = \beta/\zeta(P_1 \rightarrow P_4|i, m, E_3)$.

Equation (38) (and its associated estimator) are subject to the same constraints as Eq. (25), as discussed in Sec. III C: namely, the expected fission emission density must be non-zero at all the points in the cancellation region. In particular, it is required that $f_{i,m}(\frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1}| \hat{\Omega}_1, E_1, E_3) > 0 \forall \mathbf{r}_4 \in \mathcal{R}$. A special case arises when the reaction channel m uses a delta distribution for either the energy or direction (such as in level inelastic scattering). If such a channel was selected dur-

ing the last collision, then the value of $\zeta(P_1 \rightarrow P_4)$ is infinite at the collision point (as we are evaluating the delta distribution at the singularity), and it vanishes almost everywhere within \mathcal{R} . Thus, such channels do not generally partake in cancellation, as the uniform portion would then necessarily be zero according to Eq. (29). Finally, not all nuclear data facilitates the decomposition provided by Eq. (36). Sometimes the joint PDF might be provided as a marginal PDF in direction, and a conditional PDF in energy. If this is the case, one must go back to the intermediate collision point before having sampled the direction, in order to see the entire cancellation region.

If the region \mathcal{R} contains only one material, which is completely homogeneous in nuclide concentrations and temperature, then ν_f and Σ_f are independent of P_4 . As we have discussed in Sec. III B, these factors may be removed from the definition of ζ , without compromising the unbiasedness of the method. In addition, if fission is assumed to be perfectly isotropic (a frequent assumption), and if the fission energy E_4 is completely independent of the incident energy and direction, then we do not actu-

ally need to perform cancellation on the fission emission density, but only on the collision density at \mathbf{r}_4 . For the multi-group benchmark in Ref. 22 in which we previously demonstrated exact regional cancellation in 3D, we used homogeneous cancellation regions, where both fission and scattering were isotropic, and the fission energy was also assumed to be independent of the incident energy E_3 . This indicates that ζ could be simplified to

$$\zeta(P_1 \rightarrow P_4 | E_3) = \frac{\exp\left(-\int_0^{|\mathbf{r}_4 - \mathbf{r}_1|} \Sigma_t\left(\mathbf{r}_1 + u \frac{\mathbf{r}_4 - \mathbf{r}_1}{|\mathbf{r}_4 - \mathbf{r}_1|}, E_3\right) du\right)}{|\mathbf{r}_4 - \mathbf{r}_1|^2} \quad (39)$$

and still result in an unbiased cancellation method.

E. Non-Uniform Cancellation

Suppose now that, instead of distributing some fission emission density uniformly over \mathcal{R} , we would like to distribute fission emission density according to an arbitrary function, $D(Q)$. In this case, our estimator must be modified as

$$\vartheta_D(P_1 \rightarrow P_4 | \mathcal{R}, Q, \eta) = \frac{\zeta(P_1 \rightarrow Q | i, m, E_3) - D(Q)\beta}{\zeta(P_1 \rightarrow Q | i, m, E_3)} \delta(Q - P_4) + \frac{\beta}{\zeta(P_1 \rightarrow P_4 | i, m, E_3)} \frac{D(Q)}{\mathcal{V}_{\mathcal{R}}}. \quad (40)$$

This can be shown by following the same approach taken in Eq. (30).

Here, we have placed a portion $(\zeta(P_1 \rightarrow Q | i, m, E_3) - D(Q)\beta)/\zeta(P_1 \rightarrow Q | i, m, E_3)$ of the weight at the sampled point P_4 , and a portion $\beta/\zeta(P_1 \rightarrow Q | i, m, E_3)$ of the particle is distributed according to $D(Q)$. It is only assumed here that $D(Q)$ is dimensionless and that it is piece-wise continuous. In theory, there is no reason that $D(Q)$ could not be negative, or even complex valued⁷; if $D(Q)$ is negative or complex, it might not be possible to sample it directly, but such a situation might be treated using e.g. importance sampling [4]. Finally, we note that for the choice of $D(Q) = 1$ the case of uniform cancellation is retrieved.

⁷ While this paper only considers particles with a single real-valued statistical weight, some transport problems require that particles carry a complex weight [7, 9].

F. Fission Emission Density Function with Delta-Tracking Schemes

In the above derivations, we have often made use of the non-homogeneous exponential distribution

$$\Sigma_t(\mathbf{r} + d\hat{\Omega}, E) \exp\left(-\int_0^d \Sigma_t(\mathbf{r} + s\hat{\Omega}, E) ds\right), \quad (41)$$

which occurs in the flight kernel $T(P' \rightarrow P)$ and in the transition kernel $\zeta(P' \rightarrow P)$. This distribution is sampled when trying to determine the distance d a particle will travel from initial position \mathbf{r} along direction $\hat{\Omega}$, before undergoing a collision. For the case of piece-wise constant macroscopic cross sections, this distribution is straightforward to sample for d . However, when the macroscopic cross section is not piece-wise constant, more sophisticated methods than direct sampling are often employed [10]. Delta-tracking and negative-weighted delta-tracking are two such methods that sample the distance to collision using a sampling cross section $\Sigma_{\text{smp}}(E)$, and then sample whether a real or virtual collision has occurred with a specific criterion [19–21]; in the distinct case of negative-weighted delta-tracking, a weight modifier may be additionally applied to the particle's weight, which could potentially be negative [20, 21].⁸ In a real collision, the particle undergoes a reaction as normal. In a virtual collision, the particle's energy and direction do not change; this event is usually known as *delta scattering*. The particle continues to sample new flight distances and to move to the new location, until a real collision is sampled. Coleman [25] and Legrady et al. [21] have previously provided evidence as to why such sampling methods are unbiased.

It is possible to include virtual collisions in the transport equations by modifying the flight kernel to be

$$T_{DT}(P' \rightarrow P) = \frac{\Sigma_{\text{smp}}(E') \exp(-\Sigma_{\text{smp}}(E')|\mathbf{r} - \mathbf{r}'|)}{|\mathbf{r} - \mathbf{r}'|^2} \delta\left(\hat{\Omega} - \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|}\right) \delta(\hat{\Omega}' - \hat{\Omega}) \delta(E - E'), \quad (42)$$

and the collision kernel to be

$$C_{DT}(P' \rightarrow P) = \frac{\Sigma_t(\mathbf{r}', E')}{\Sigma_{\text{smp}}(E')} C(P' \rightarrow P) + \left(1 - \frac{\Sigma_t(\mathbf{r}', E')}{\Sigma_{\text{smp}}(E')}\right) \delta(E - E') \delta(\hat{\Omega} - \hat{\Omega}') \delta(\mathbf{r} - \mathbf{r}'). \quad (43)$$

⁸ Using negative weights can be advantageous in some cases, as it allows $\Sigma_{\text{smp}}(E)$ to be less than $\Sigma_t(\mathbf{r}, E)$. In delta-tracking, it is required that $\Sigma_{\text{smp}}(E) \geq \Sigma_t(\mathbf{r}, E)$ everywhere in the problem domain: because of this requirement, it could be difficult to determine $\Sigma_{\text{smp}}(E)$ for delta-tracking, when considering spatially continuous cross sections [10].

These equations are valid for both delta tracking and negative-weighted delta tracking. From Eq. (42), the PDF for leaving a collision site at \mathbf{r}' and flying to \mathbf{r} and inducing a fission (given that we are flying in the direction of \mathbf{r} , i.e. $\hat{\Omega}' = \frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|}$) is

$$\Sigma_f(\mathbf{r}, E') \exp(-\Sigma_{\text{smp}}(E')|\mathbf{r}-\mathbf{r}'|), \quad (44)$$

which is exactly the form presented in Eq. (1). Thus, delta-tracking-like algorithms provide the large advantage of not requiring the integration of the total cross section along the flight path. This makes them interesting for the purpose of performing exact regional cancellation.

The form of Eq. (42) is valid regardless of whether the collision at \mathbf{r}' was real or virtual. Equation (43) shows that the angular distribution for virtual collisions is singular, because it is described by a delta distribution. As discussed in Sec. III D, channels with singular distributions are not allowed to partake in cancellation, i.e. we need to set $\beta = 0$ for all such channels. For the particular case of virtual collisions, however, another treatment is possible. At the site where the virtual collision took place, there was a probability that the particle could have instead undergone a real collision. We can therefore imagine “splitting” the particle before the collision. A weight $w(1 - \Sigma_t(\mathbf{r}, E')/\Sigma_{\text{smp}}(E'))$ is considered to undergo a virtual collision, and have its next collision at P_3 ; this virtual collision portion cannot be used in cancellation, as the angular distribution was a delta distribution, and the uniform component is then always zero, as explained in Sec. III D. The rest of the particle weight, namely $w\Sigma_t(\mathbf{r}, E')/\Sigma_{\text{smp}}(E')$, is considered to undergo a real collision and have its next collision at P_3 , like the virtual part. However, this part can also partake in cancellation. The point-wise fission particle weight which must remain at the sampled fission particle site is then

$$\begin{aligned} & w \left(1 - \frac{\Sigma_t(\mathbf{r}, E')}{\Sigma_{\text{smp}}(E')} \right) + w \frac{\Sigma_t(\mathbf{r}, E')}{\Sigma_{\text{smp}}(E')} \left(1 - \frac{\beta}{\zeta(P' \rightarrow P)} \right) = \\ & w \left(1 - \frac{\beta}{\zeta(P' \rightarrow P)} \frac{\Sigma_t(\mathbf{r}, E')}{\Sigma_{\text{smp}}(E')} \right) = \\ & w \left(1 - \frac{\beta'}{\zeta(P' \rightarrow P)} \right), \end{aligned} \quad (45)$$

where we have set $\beta' = \beta\Sigma_t/\Sigma_{\text{smp}}$. Thus, splitting shows that *all* collisions can be assumed to partake in cancellation as if they were real, because the presence of virtual collisions only affects the choice of β . Since the estimator is unbiased for any β , the factor $\Sigma_t(\mathbf{r}, E')/\Sigma_{\text{smp}}(E')$ is not necessary. However, note that this approach is only unbiased so long as at the virtual collision site the real component of the scattering kernel for forward scattering with no energy change is not zero (i.e. $C(P \rightarrow P) \neq 0$). Otherwise, the real collision component could not reach P_3 , as it would be impossible to have a real collision with forward scattering and no change in energy. This was possible in our previous multi-group example, because

in-group scattering was always allowed and all scattering was assumed to be isotropic; however, this might not be as trivial in a continuous-energy setting.

IV. OPTIMIZATION OF CANCELLATION EFFICIENCY

We now turn our attention to the optimal choice of the free parameter β of the cancellation estimator, used to calculate the weight that can be uniformly distributed over the cancellation region. In Booth and Gubernatis’s seminal paper [23] and in our previous work [22], β was chosen to be the minimum of the expected fission density over the cancellation region. This choice has the advantage of being relatively easy to evaluate, but it is not necessarily the most efficient one. In this section we attempt to introduce a better strategy to determine the cancellation parameter β for each particle partaking in cancellation.

In order to optimize for the cancellation efficiency, one must first properly define the quantity to be optimized. As we mentioned in Sec. II, the maximum amount of cancellation will occur when the sum of the absolute value of all the weights remaining after cancellation in the region has been minimized. For N particles which initially land in a cancellation region, we define the absolute value of all weight in a region after cancellation as

$$\Gamma_1 = \sum_{k=1}^N |w_{k,p}| + \left| \sum_{k=1}^N w_{k,u} \right|, \quad (46)$$

with $w_{k,p}$ being the point-wise weight of particle k , and $w_{k,u}$ the uniform weight portion of particle k .⁹ Equation (46) is the total post-cancellation weight discussed in Sec. II, Eq. (7). As each particle has a different value for the cancellation parameter β , we then may substitute to obtain

$$\Gamma_1 = \sum_{k=1}^N \left| \frac{\zeta_k - \beta_k}{\zeta_k} w_k \right| + \left| \sum_{k=1}^N \frac{\beta_k}{\zeta_k} w_k \right|, \quad (47)$$

where w_k and β_k are respectively the pre-cancellation weight and the cancellation parameter of the k -th particle, and $\zeta_k = \zeta(P'_k \rightarrow P_k)$ is the expected fission density of the k -th particle, which is assumed to have had its previous collision in P'_k and its fission event in P_k (note that we have simplified the notation here compared to Sec. III; for a given particle, P'_k and P_k respectively correspond to P_1 and P_4).

Due to the presence of the absolute values, it is quite difficult to optimize the expression of Γ_1 analytically with

⁹ While there are N particles in the cancellation region before the cancellation operations have been carried out, there will be more than N particles after cancellation, due to the new uniform particles which are created during the cancellation process.

respect to β_k . We instead define a modified quantity Γ_2 , which shares a minimum with Γ_1 :

$$\Gamma_2 = \sum_{k=1}^N \left(\frac{\zeta_k - \beta_k}{\zeta_k} w_k \right)^2 + \left(\sum_{k=1}^N \frac{\beta_k}{\zeta_k} w_k \right)^2. \quad (48)$$

We now wish to obtain the set of optimal values β_k that minimize Γ_2 . To remain unbiased, we are not allowed to calculate β_k based on the phase space coordinates P_k where the particle k landed in the cancellation region (this was made evident in Eq. (30)). As $\zeta_k = \zeta(P'_k \rightarrow P_k)$, we cannot directly minimize Eq. (48). In the two subsequent sections, we will present two reasonable options to avoid this problem, both possibly giving way to a method of optimizing the regional cancellation algorithm. We go through the optimization for each case, obtaining two different formulations for calculating the set of optimal values for β_k .

A. Replacing ζ_k with $\langle \zeta_k \rangle$

The first approach consists in averaging ζ_k over the entire phase space of the region \mathcal{R} , such that

$$\langle \zeta_k \rangle = \frac{\int_{\mathcal{R}} \zeta(P'_k \rightarrow P_k) dP_k}{\int_{\mathcal{R}} dP_k}. \quad (49)$$

We may then replace ζ_k with $\langle \zeta_k \rangle$ in Eq. (48), and optimize the new approximate form

$$\Gamma_2^* = \sum_{k=1}^N \left(\frac{\langle \zeta_k \rangle - \beta_k}{\langle \zeta_k \rangle} w_k \right)^2 + \left(\sum_{k=1}^N \frac{\beta_k}{\langle \zeta_k \rangle} w_k \right)^2, \quad (50)$$

which is now independent of P_k . The detailed derivation for this approach is presented in Appendix A, and the resulting equation for the cancellation parameter β_k is found to be

$$\beta_k = \langle \zeta_k \rangle \left(1 - \frac{S^*}{w_k} \right), \quad (51)$$

where we make use of the definition

$$S^* = \frac{W}{N+1}, \quad (52)$$

and

$$W = \sum_{k=1}^N w_k, \quad (53)$$

W being the net weight in the region \mathcal{R} before cancellation. It should also be mentioned that Eq. (51) would also be obtained if we first minimized Eq. (48) with respect to β_k and then averaged over P_k .

B. Optimization of $\langle \Gamma_2 \rangle$

The second approach consists in averaging Γ_2 over the phase space of the region \mathcal{R} , obtaining

$$\langle \Gamma_2 \rangle = \frac{\int_{\mathcal{R}} \Gamma_2 \prod_{k=1}^N \zeta_k dP_k}{\int_{\mathcal{R}} \prod_{k=1}^N \zeta_k dP_k}. \quad (54)$$

We may then optimize $\langle \Gamma_2 \rangle$ instead of Γ_2 . The complete derivation is provided in Appendix B, and leads to a different equation for β_k :

$$\beta_k = \langle \zeta_k \rangle c_k \left(1 - \frac{S}{w_k} \right). \quad (55)$$

Here, we have made use of the two following definitions:

$$c_k = \left(2 \langle \zeta_k \rangle \left\langle \frac{1}{\zeta_k} \right\rangle - 1 \right)^{-1}, \quad (56)$$

where the angle brackets have the same meaning as in Eq. (49), and

$$S = \frac{\sum_{k=1}^N c_k w_k}{1 + \sum_{k=1}^N c_k}. \quad (57)$$

C. Small Region Limit

For the two possible methods that we have outlined to minimize the weight after cancellation, we are left with two different possibilities for the value of β_k . At first glance, these two choices of β_k look quite different. Upon closer inspection of the definition of c_k in Eq. (56), we notice that $c_k = 1$ only if $1/\langle \zeta_k \rangle = \langle 1/\zeta_k \rangle$. When this is the case, it then follows from Eq. (57) that $S = S^*$. This then indicates that the two definitions of β_k given by Eq. (51) and Eq. (55) are equivalent only when $1/\langle \zeta_k \rangle = \langle 1/\zeta_k \rangle$. In general, however, Jensen's inequality implies $1/\langle \zeta_k \rangle \leq \langle 1/\zeta_k \rangle$ [26]. If a cancellation region were defined such that it is small enough that one could reasonably assume that $\zeta(P'_k \rightarrow P_k)$ is nearly constant within the region, then $c_k \approx 1$, leading to the two methods being equivalent. However, this would likely require such a small region that it is very unlikely that any other particles would be located within the region, which will make cancellation very ineffective.

V. MONTE CARLO IMPLEMENTATION

A. Cancellation with Distributed Memory Simulations

Most production Monte Carlo codes make use of distributed-memory parallel computing techniques such as Message Passing Interface (MPI), although the exact algorithm used varies from code to code [14–17]. Generally speaking, distributed-memory parallelization can pose a problem for cancellation, which is by construction more efficient when there are more particles in each cancellation region. With distributed-memory parallelization, the fission particles within a given cancellation region will be distributed amongst several nodes. To ensure the highest possible efficiency, cancellation must be performed on the entire fission source. One method to do this is to send all of the fission particles to the master node between power iteration generations, and then perform cancellation only on the master node. Another option would be to use a method inspired by domain decomposition [27], where certain nodes perform cancellation for certain regions, and fission particles would need to be sent to the node which corresponds to their cancellation region. This method is certainly possible, but likely much more difficult to implement in production Monte Carlo codes. For the proof of concept presented in this paper, we have chosen to use the former method, sending all fission particles to the master node for cancellation.

B. Calculation of $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$

In order to evaluate β_k according to Eq. (51), we must have knowledge of $\langle \zeta_k \rangle$. For Eq. (55), we additionally need knowledge of $\langle 1/\zeta_k \rangle$. In general, it is not possible to analytically calculate either of these quantities for particle k , born with phase space coordinates P_k located within cancellation region \mathcal{R} . However, it is possible to estimate both of these quantities with a Monte Carlo sampling approach. For each particle k , we know the phase space coordinates P'_k of its previous collision, and we know the bounds of the outgoing phase space coordinates P_k which define the cancellation region \mathcal{R} .

Assume that a set of non-overlapping, hypercuboid cancellation regions are imposed on top of the problem domain. Then, between each generation of power iteration, the fission particles (having stored their parent's previous phase space coordinates P'_k) may be sorted into the cancellation regions, based on their phase space coordinates P_k . Once this is accomplished for a given cancellation region \mathcal{R} , we may iterate over all particles in \mathcal{R} , and estimate their values of $\langle \zeta_k \rangle$ and additionally $\langle 1/\zeta_k \rangle$, depending on which optimisation algorithm is chosen.

The estimates for these quantities may be obtained using

$$\langle \zeta_k \rangle \approx \frac{1}{N_s} \sum_{i=1}^{N_s} \zeta(P'_k \rightarrow \tilde{P}_i) \quad (58)$$

and

$$\left\langle \frac{1}{\zeta_k} \right\rangle \approx \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{1}{\zeta(P'_k \rightarrow \tilde{P}_i)}, \quad (59)$$

respectively, where N_s is the number of samples to be used in the estimation, and the outgoing phase space coordinates \tilde{P}_i are pseudo-randomly sampled so that $\tilde{P}_i \in \mathcal{R}$ and $\tilde{P}_i \sim \mathcal{U}(\mathcal{R})$. This is straightforward to accomplish with cuboid regions.

With this approach, a better estimate of $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$ may be obtained by augmenting the number of samples. As N_s is increased, the error on the estimate of the two expectation values will decrease according to $\mathcal{O}(1/\sqrt{N_s})$ [28]. This indicates that a large N_s may be required to obtain a suitable estimate of $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$. Even more problematic is the fact that evaluating $\zeta(P'_k \rightarrow \tilde{P}_i)$ could be quite costly; this is especially true in the case of continuous-energy neutron transport problems, where many evaluations of scattering distributions would be necessary. In order to reduce N_s , while still obtaining adequate estimates for $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$, we propose the use of a quasi-random technique, using a Sobol' sequence [28] to sample the outgoing phase space coordinates \tilde{P}_i . This approach generally has a better convergence rate than using a pseudo-random number generator to sample \tilde{P}_i , as it leads to a more uniform exploration of the phase space [28]. At any rate, we stress that statistical uncertainties on the estimation of $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$ only affect the efficiency of the cancellation method, as we have proved that the method is unbiased for any values of the free parameters β_k .

C. Heterogeneous Cancellation Regions

In this work, we have proposed two possible approaches to selecting an optimal value of β , to optimize the amount of weight which is cancelled. Neither of these approaches requires the minimum value of the fission emission density within the cancellation region \mathcal{R} . Hence, it is no longer necessary to restrict the cancellation regions to be cuboids, as we were required to do in Ref. 22. In light of our proposed sampling methods to estimate $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$ to obtain the optimized cancellation parameter β_k , it is evident that a rejection technique may be applied to isolate different material regions within a given cuboid cancellation region. If we have a cancellation region with fuel and water, then all of the fission particles are of course only born inside the fuel portion, and the fission density everywhere in the water should be zero. When sampling the random phase space coordinates \tilde{P}_i ,

we must now add the requirement that \tilde{P}_i have spatial coordinates that are located inside of the fuel material.

Other special cases can be handled using this approach. For example, cancellation can also be performed when there are two non-connected fuel regions within the same cancellation mesh region. We may also have cancellation regions which contain multiple different fuel regions. Using rejection sampling to determine cancellation regions makes it easy to apply regional cancellation to complex geometries encountered in realistic reactor physics problems. Of course, the rejection procedure must also be applied to the sampling of the phase space coordinates of the uniformly distributed particles.

D. Monte Carlo Implementation in the open-source code MGMC

For our previous work on regional cancellation, a multi-group Monte Carlo mini-app called MGMC was used to test cancellation on a well-known reactor physics benchmark. MGMC has been developed to facilitate the fast and easy implementation and testing of new transport algorithms. Being only ≈ 13 k lines of code, it is much faster to test new ideas in MGMC than it would be in a large industrial code. General 3D geometries are supported using a standard constructive solid geometry formalism based on surfaces, universes, and lattices, familiar to any user of other well-known Monte Carlo codes [14–17]. Different mesh tallies are available for flux or reaction rates, with track-length or collision estimators. MGMC can solve fixed-source, k -eigenvalue, and neutron noise problems, using both shared and distributed memory parallelism. Shared memory parallelism is implemented with OpenMP, while the distributed memory parallelism is implemented using MPI. Different transport methods such as surface-tracking, delta-tracking, and negative-weighted delta-tracking are also available. All of the outlined cancellation algorithms have been implemented in MGMC, which was used to run the simulations presented in the next section. MGMC has been made publicly available as free software under the CeCILL v2.1 license [29].

VI. SIMULATION RESULTS

For our numerical simulations, in this Section we will make use of the modified C5G7 benchmark which we introduced in our previous work [22]. The C5G7 is a multi-group neutron transport benchmark which comes from the nuclear reactor physics community, for the purpose of validating different codes [30]. Our modified version makes use of square profile fuel pins with side lengths of 0.756 cm, in lieu of cylindrical pins of radius 0.54 cm as proposed in the original specifications. This modification allows a regular $170 \times 170 \times 765$ mesh to be imposed on top of the geometry over the fuel assemblies to act as can-

cellation regions, and guaranteed that each cancellation region contained a unique material. For continuity, we make use of the same cancellation mesh. For transport, we again use the negative-weighted delta-tracking variant proposed by Carter et al. [20], and identical sampling cross sections to the previous study: the sampling cross section for the first group is 90% of the majorant cross section, while all other sampling cross sections were taken to be the majorant. This means that the sampling cross section underestimates the total cross section in the first energy group for all fuel pins in the problem. Whenever a virtual collision occurs for a particle in the first energy group, inside a fuel pin, its weight will then change sign. Once a particle leaves the first group, it is impossible for the sign to change at a collision (although signs can possibly change during cancellation). Virtual collisions lead to the presence of negative weights in the system, and we have shown that weight cancellation is mandatory for k -eigenvalue power iteration problem to converge when using negative-weighted delta-tracking [22]. All simulations were initiated with 10^6 particles, and ran for 2500 generations, with the first 500 being discarded to allow for source convergence.

As we have shown in Ref. 22, the total weight of all the fission particles between two generations increases without bound if weight cancellation is not applied. This increase in total weight is accompanied by an increase in the number of particles and large statistical fluctuations in estimated quantities, making it nearly impossible to estimate the multiplication factor and static flux for the system. The effect of cancellation is to limit the growth of W_{tot} to a saturation value; the more efficient cancellation is, the lower the saturation value will be. Thus, we have chosen to assess the efficiency of cancellation by using the saturation value of W_{tot} , which is calculated immediately after applying the cancellation procedure. Note that W_{tot} has a lower theoretical limit of W_{net} , which is kept constant by normalizing all particle weights between generations [22].

A. Comparison of Optimization Strategies

To determine which method of choosing β_k leads to the most efficient cancellation of positive and negative weights, the optimization techniques described in Sec. IV A and Sec. IV B were compared against the original implementation using the minimum value of the fission density within the region. Both optimization options utilized $N_s = 100$ samples for estimating $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$. The values of W_{tot} are plotted against the number of generations in Figure 2. For comparison, curves corresponding to no cancellation and approximate cancellation have also been presented. Approximate cancellation imposes a mesh on top of the geometry, and sorts fission particle into this mesh. The average weight of all particles in each mesh element can then be calculated and assigned to the particles [31, 32]. This method is quite efficient, but is

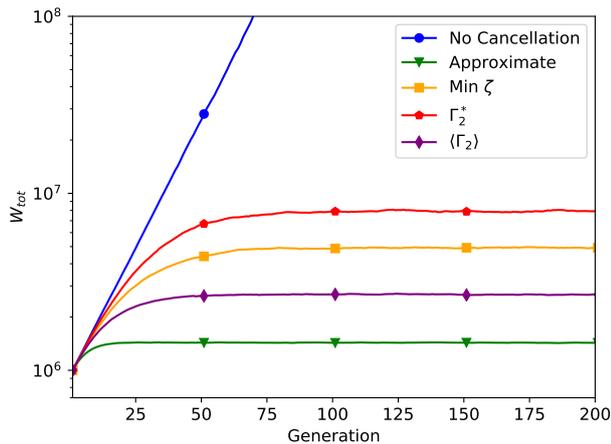


FIG. 2. Behavior of W_{tot} as a function of generation, for different cancellation methods, including no cancellation and approximate cancellation.

not exact, and imposes a bias on the fission source and on the eigenvalue (though the bias can be made arbitrarily small by using a sufficiently fine mesh).

If no cancellation technique is used, the total weight increases exponentially, without bound. This phenomenon is expected when using negative-weighted delta-tracking with k -eigenvalue power iteration, as described previously [22]. When taking $\beta_k = \min_{\mathcal{R}}(\zeta_k)$, an asymptotic value of $W_{\text{tot}} \approx 4.9 \cdot 10^6$ was seen. The most efficient method of determining β_k was that obtained from optimizing $\langle \Gamma_2 \rangle$ in Eq. (55), resulting in $W_{\text{tot}} \approx 2.7 \cdot 10^6$, almost half the amount of total weight obtained with the minimum strategy. Calculating β_k from Eq. (51) for the case of replacing ζ_k with $\langle \zeta_k \rangle$ is less efficient than using the minimum value of ζ_k within the cancellation region, resulting in $W_{\text{tot}} \approx 8.0 \cdot 10^6$. It is not known why this approximation does not perform as well as using the minimum of ζ_k , and this intriguing question calls for future investigations.

Approximate cancellation yielded the lowest total weight (and therefore the highest cancellation efficiency), with $W_{\text{tot}} \approx 1.4 \cdot 10^6$, but is not an exact approach. Currently, we do not know of any way to estimate, or to put a limit on the bias imposed by this method without running several realizations, each with a different mesh size.

B. Strategies for Evaluating the Average Fission Emission Densities

The analysis in Sec. VIA shows that the optimal choice for determining β_k for the C5G7 benchmark is Eq. (55), from the optimization of $\langle \Gamma_2 \rangle$. We now consider the optimal strategy for estimating the requisite values of $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$ for each particle. Figure 3 depicts the behav-

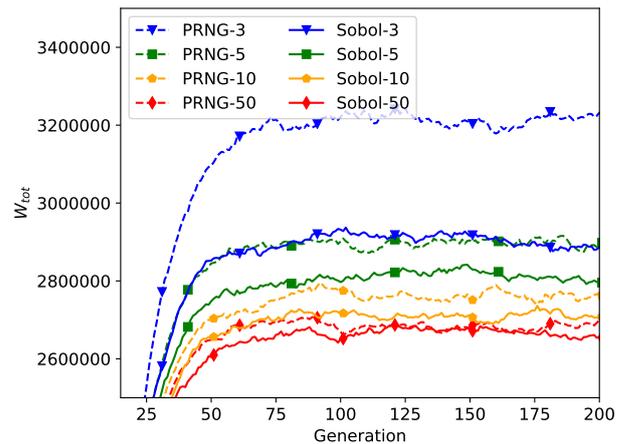


FIG. 3. Behaviour of W_{tot} as a function of generation, for different cancellation parameters. For all curves, β_k is determined from the optimization of $\langle \Gamma_2 \rangle$, Eq. (55). The estimated values of $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$ are determined with a varying number of points, using either a pseudo-random number generator (PRNG) or a Sobol' sequence.

ior of W_{tot} where points \tilde{P}_i (from Eqs. (58) and (59)) are sampled with either a pseudo-random number generator (PRNG) or a Sobol' sequence.

First of all, the spread among the different strategies for estimating $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$ is much smaller than the spread among the different minimization strategies of Fig. 2. It is observed that in general, when $N_s < 50$, using Sobol' points leads to more efficient weight cancellation. This effect is most apparent for $N_s = 3$, where the Sobol' sequence leads to approximately 9.2% less total weight being transported, compared to the PRNG estimation strategy. The increased efficiency observed in the Sobol' points diminishes however with increasing N_s . Sobol' estimation gives a 2.8% improvement for $N_s = 5$, 1.8% improvement for $N_s = 10$, and only a 0.5% improvement for $N_s = 50$. This would indicate that the estimated values for $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$ start to become independent of the evaluation strategy at around $N_s = 50$.

In addition to achieving more weight cancellation, the Sobol' points also have the added benefit of being slightly easier to compute, as the quasi-random numbers used to compute the points can be tabulated in advance, and written in the code. All that is then needed is a table lookup to get a Sobol' value, whereas several mathematical operations must be performed to calculate each value generated from a PRNG. However, a drawback with the use of Sobol' points is that one does not necessarily know in advance how many points will be needed, when considering heterogeneous cancellation regions.

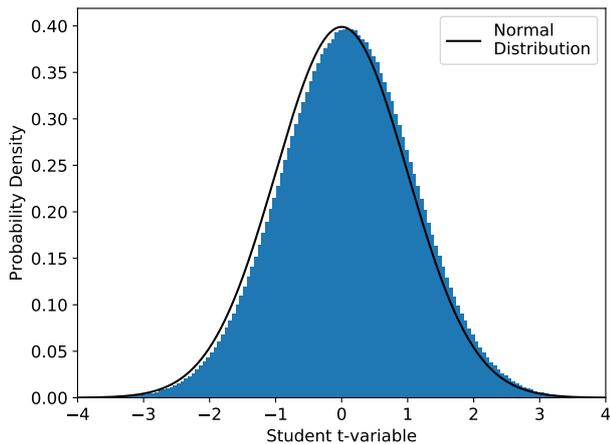


FIG. 4. Histogram distribution of the Student t -variable, comparing the flux computed from delta-tracking to the flux computed with negative weighted delta-tracking. A normal distribution is plotted on top of the histogram for a reference.

C. Demonstration of Heterogeneous Cancellation Regions on the C5G7 Benchmark

We also tested the rejection-based sampling technique described in Section V C, for performing regional cancellation in cuboid regions which contain multiple materials. Instead of using our modified version of the C5G7 benchmark, we have opted to use the original version with cylindrical fuel pins [30], in combination with the same $170 \times 170 \times 765$ mesh as used in our previous simulations. A reference calculation was performed using standard delta-tracking and obtained a multiplication factor of $k_{\text{eff}} = 1.18383 \pm 0.00003$, which is in agreement with the reference solution for the 3D version of the benchmark [30]. Cancellation used the method for calculating β proposed in Sec. IV B, with $N_s = 10$ samples being used to estimate $\langle \zeta_k \rangle$ and $\langle 1/\zeta_k \rangle$.

When running the same simulation with negative-weighted delta-tracking, using the same sampling cross sections as before, and exact cancellation, a multiplication factor of $k_{\text{eff}} = 1.18382 \pm 0.00009$ was obtained, which is in agreement with the delta-tracking value. A comparison of the two estimations of the flux were also made, looking at the Student t -variable, which is defined as

$$t_i = \frac{\varphi_{i,A} - \varphi_{i,B}}{\sqrt{\sigma_{i,A}^2 + \sigma_{i,B}^2}}, \quad (60)$$

where $\varphi_{i,x}$ is the average value of the flux in the i -th bin for calculation x , and $\sigma_{i,x}$ is its standard error. A corresponds to the results from the delta-tracking simulation without weight cancellation, and B corresponds to the results from the negative-weighted delta-tracking simulation with cancellation. For independent, normally distributed variables with a large number of degrees of free-

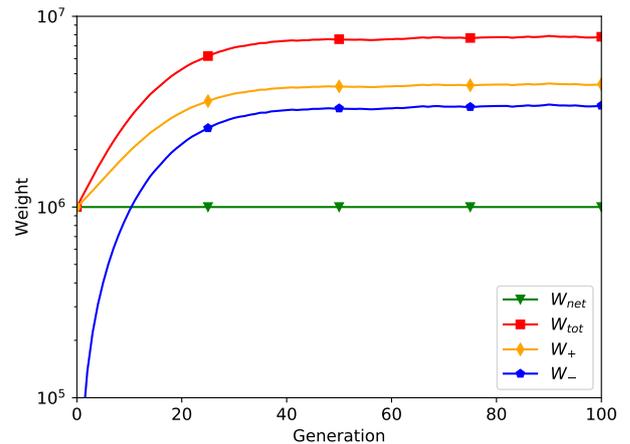


FIG. 5. Plots of the positive, negative, net, and total weights as a function of generation, for the original C5G7 benchmark. Exact regional cancellation with heterogeneous regions was used to perform the simulation.

dom, the distribution of the t -variable should approach a normal distribution. A plot of the empirical distribution of the t -variable is provided in Figure 4. We excluded from the comparison all the bins where the flux was estimated to be zero, or where the relative standard error was greater than 20%. This was done in an effort to ensure that each bin was approximately normally distributed, for the Student t -variable distribution assumptions to be reasonable. It is clear from Figure 4 that the two flux estimates are in good agreement. The t -variable distribution is not perfectly normal, which is to be expected, as there are correlations between the scores in different flux bins. In general, this is a very strong indication that our cancellation method has not imposed any bias on the fission source, and that the method is still exact when applied to heterogeneous cancellation regions.

The behavior of the total weight is shown in Fig. 5. While W_{tot} is larger than in the case of square fuel pins with homogeneous cancellation region presented in Fig. 2, the behavior is in general similar. A large W_{tot} indicates that there are more negative particles, which will increase the variance in scores. This is indeed the case, as our previous work obtained an uncertainty for k_{eff} of only 5×10^{-5} (albeit for a slightly different problem) [22], while an uncertainty of 9×10^{-5} was obtained for this problem.

VII. CONCLUSIONS

This work has leveraged the integral form of the Boltzmann transport equation to provide a more in-depth mathematical analysis of the exact regional weight cancellation technique, considerably expanding on previous works on the subject [22, 23]. Not only has this formal

approach given a much better understanding as to the mechanics of regional cancellation in simplified isotropic multi-group problems, but it has illuminated the non-trivial path to performing exact cancellation in more complex problems, where scattering is anisotropic, and the fission spectrum may depend on the incident energy of a particle. The analysis highlights the fact that the implementation of exact regional cancellation is rather straightforward in a multi-group Monte Carlo code, but will be more difficult in a continuous-energy code, as one will need access to conditional scattering distributions in the laboratory frame, which are not always available for all reactions. The implementation of exact regional cancellation in a continuous-energy code is thus a subject which will require further research.

Additionally, a strategy to determine an optimal value of the cancellation parameter β for each particle undergoing cancellation has been conceived. For each particle k within a cancellation region, its optimal cancellation parameter β_k can be computed if both the average value of the fission emission density in the region and the average value of the inverse of the fission emission density in the region are known. Our previous implementation required that cancellation regions be homogeneous and cubical, which restricted its applicability to simple problems. Thanks to the improvements in the optimization technique proposed in this work, both requirements have been relaxed.

On a modified version of the C5G7 benchmark, our technique to optimize weight cancellation was demonstrated to reduce the total weight in the simulation by approximately 45%, when compared to using the minimum value of the fission emission density in the region for β , as previously suggested in the literature. In order to estimate the average fission emission density and the average inverse of the fission emission density, a sampling approach has been proposed, where the averages are estimated using the values of the fission emission density at pseudo-random or quasi-random points. It was demonstrated that the quasi-random Sobol' sequence requires slightly fewer points than the pseudo-random sequence to reach the asymptotic limit of the optimized cancellation algorithm. As a comparison, the use of 3 Sobol' points had very similar performance to the use of 5 pseudo-random points for the modified version of the C5G7 benchmark examined here. However, such results are likely to be highly problem-dependent, and more systems should be analyzed to ascertain what sort of performance improvements could be expected in general. We

also tested the use of heterogeneous cancellation regions on the original C5G7 benchmark, with cylindrical fuel pins. No bias was observed in the resulting fundamental eigenvalue, or flux tally.

Appendix A: Optimization of Γ_2^*

We remind the reader of the definition of Γ_2^* :

$$\Gamma_2^* = \sum_{k=1}^N \left(\frac{\langle \zeta_k \rangle - \beta_k}{\langle \zeta_k \rangle} w_k \right)^2 + \left(\sum_{k=1}^N \frac{\beta_k}{\langle \zeta_k \rangle} w_k \right)^2. \quad (\text{A1})$$

We optimize Γ_2^* simultaneously for all particles by differentiating with respect to β_j , and setting the partial derivative equal to zero:

$$\frac{\partial \Gamma_2^*}{\partial \beta_j} = -2 \frac{\langle \zeta_j \rangle - \beta_j}{\langle \zeta_j \rangle^2} w_j^2 + 2 \frac{w_j}{\langle \zeta_j \rangle} \sum_{k=1}^N \frac{\beta_k w_k}{\langle \zeta_k \rangle} = 0. \quad (\text{A2})$$

This may be simplified to

$$-w_j + \frac{\beta_j w_j}{\langle \zeta_j \rangle} + \sum_{k=1}^N \frac{\beta_k w_k}{\langle \zeta_k \rangle} = 0. \quad (\text{A3})$$

On the left-hand-side, the second term matches the argument of the sum in the third term. Summing over index j , we see that

$$-W + \sum_{j=1}^N \frac{\beta_j w_j}{\langle \zeta_j \rangle} + N \sum_{k=1}^N \frac{\beta_k w_k}{\langle \zeta_k \rangle} = 0. \quad (\text{A4})$$

Here, we used the definition provided in Eq. (53). This allows us to isolate the sum

$$\sum_{k=1}^N \frac{\beta_k w_k}{\langle \zeta_k \rangle} = S^* = \frac{W}{N+1}. \quad (\text{A5})$$

Applying this substitution to Eq. (A3) while also using Eqs. (52) and (53), we find that the optimized value of β_j is

$$\beta_j = \langle \zeta_j \rangle \left(1 - \frac{W}{(N+1)w_j} \right) = \langle \zeta_j \rangle \left(1 - \frac{S^*}{w_j} \right). \quad (\text{A6})$$

Appendix B: Optimization of $\langle \Gamma_2 \rangle$

Substituting Eq. (48) into Eq. (54), and partially expanding the squared terms, we see that

$$\langle \Gamma_2 \rangle = \frac{\int_{\mathcal{R}} \left[\sum_{k=1}^N \left(1 - \frac{2\beta_k}{\zeta_k} + \frac{\beta_k^2}{\zeta_k^2} \right) w_k^2 + \sum_{k=1}^N \sum_{\substack{l=1 \\ l \neq k}}^N \frac{\beta_k}{\zeta_k} \frac{\beta_l}{\zeta_l} w_k w_l \right] \prod_{m=1}^N \zeta_m dP_m}{\int_{\mathcal{R}} \prod_{n=1}^N \zeta_n dP_n} \quad (\text{B1})$$

$$= \frac{\int_{\mathcal{R}} \left[\sum_{k=1}^N \left(1 - \frac{2\beta_k}{\zeta_k} + \frac{\beta_k^2}{\zeta_k^2} \right) w_k^2 + \sum_{k=1}^N \sum_{\substack{l=1 \\ l \neq k}}^N \frac{\beta_k}{\zeta_k} \frac{\beta_l}{\zeta_l} w_k w_l + \sum_{k=1}^N \frac{\beta_k^2}{\zeta_k^2} w_k^2 \right] \prod_{m=1}^N \zeta_m dP_m}{\prod_{n=1}^N \langle \zeta_n \rangle} \quad (\text{B2})$$

$$= \sum_{k=1}^N \left(1 - \frac{2\beta_k}{\langle \zeta_k \rangle} + \frac{\beta_k^2}{\langle \zeta_k \rangle} \left\langle \frac{1}{\zeta_k} \right\rangle \right) w_k^2 + \sum_{k=1}^N \sum_{\substack{l=1 \\ l \neq k}}^N \frac{\beta_k}{\langle \zeta_k \rangle} \frac{\beta_l}{\langle \zeta_l \rangle} w_k w_l + \sum_{k=1}^N \frac{\beta_k^2}{\langle \zeta_k \rangle} \left\langle \frac{1}{\zeta_k} \right\rangle w_k^2 \quad (\text{B3})$$

$$= \sum_{k=1}^N \left(w_k^2 - \frac{2\beta_k w_k^2}{\langle \zeta_k \rangle} \right) + \sum_{k=1}^N \frac{\beta_k^2 w_k^2}{\langle \zeta_k \rangle} \left(2 \left\langle \frac{1}{\zeta_k} \right\rangle - \frac{1}{\langle \zeta_k \rangle} \right) + \sum_{k=1}^N \sum_{l=1}^N \frac{\beta_k}{\langle \zeta_k \rangle} \frac{\beta_l}{\langle \zeta_l \rangle} w_k w_l. \quad (\text{B4})$$

It is convenient to use the constant c_k , defined by Eq. (56), which may be substituted into Eq. (B4) to produce

$$\langle \Gamma_2 \rangle = \sum_{k=1}^N \left(w_k^2 - \frac{2\beta_k w_k^2}{\langle \zeta_k \rangle} \right) + \sum_{k=1}^N \frac{\beta_k^2 w_k^2}{c_k \langle \zeta_k \rangle^2} + \sum_{k=1}^N \sum_{l=1}^N \frac{\beta_k}{\langle \zeta_k \rangle} \frac{\beta_l}{\langle \zeta_l \rangle} w_k w_l. \quad (\text{B5})$$

Now that all of the integrals have been simplified, we are left with $\langle \Gamma_2 \rangle$ as a function of β_k , $\langle \zeta_k \rangle$, and $\left\langle \frac{1}{\zeta_k} \right\rangle$ $\forall k = 1, \dots, N$. We now optimize $\langle \Gamma_2 \rangle$ with respect to the cancellation parameter β_j by solving for

$$\frac{\partial \langle \Gamma_2 \rangle}{\partial \beta_j} = 0. \quad (\text{B6})$$

From Eq. (B5), one may then proceed by solving

$$\frac{\partial \langle \Gamma_2 \rangle}{\partial \beta_j} = -\frac{2w_j^2}{\langle \zeta_j \rangle} + \frac{2\beta_j w_j^2}{c_j \langle \zeta_j \rangle^2} + \frac{2w_j}{\langle \zeta_j \rangle} \sum_{k=1}^N \frac{\beta_k w_k}{\langle \zeta_k \rangle} = 0. \quad (\text{B7})$$

Upon a division by $2w_j / \langle \zeta_j \rangle$ on both sides, we are left with

$$-w_j + \frac{\beta_j w_j}{c_j \langle \zeta_j \rangle} + \sum_{k=1}^N \frac{\beta_k w_k}{\langle \zeta_k \rangle} = 0. \quad (\text{B8})$$

It is possible to isolate the sum in the third term on the

left-hand-side by multiplying by c_j , and then summing over j :

$$-\sum_{j=1}^N c_j w_j + \sum_{j=1}^N \frac{\beta_j w_j}{\langle \zeta_j \rangle} + \sum_{j=1}^N c_j \sum_{k=1}^N \frac{\beta_k w_k}{\langle \zeta_k \rangle} = 0. \quad (\text{B9})$$

We will now define

$$S = \sum_{k=1}^N \frac{\beta_k w_k}{\langle \zeta_k \rangle}, \quad (\text{B10})$$

and substitute Eq. (B10) into Eq. (B9), allowing one to solve for S . Doing so, one may obtain the result provided by Eq. (57). Now that the summation term, S , can be computed without knowledge of β_j , we may substitute Eq. (57) and Eq. (B10) into Eq. (B8), and solve for β_j , producing

$$\beta_j = \langle \zeta_j \rangle c_j \left(1 - \frac{S}{w_j} \right). \quad (\text{B11})$$

- [1] D. Schneider, F. Dolci, F. Gabriel, J. Palau, M. Guillo, B. Pothet, P. Archier, K. Ammar, F. Auffret, R. Baron, *et al.*, in *PHYSOR 2016* (2016).
 [2] B. Collins, S. Stimpson, B. W. Kelley, M. T. Young, B. Kochunas, A. Graham, E. W. Larsen, T. Downar,

- and A. Godfrey, *J. Comput. Phys.* **326**, 612–628 (2016).
 [3] J. Rhodes, K. Smith, and D. Lee, in *PHYSOR 2006*, Vol. 144 (2006).
 [4] I. Lux and L. Koblinger, *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations* (CRC Press,

- 1991).
- [5] P. K. Romano, S. P. Hamilton, R. O. Rahaman, A. Novak, E. Merzari, S. M. Harper, P. C. Shriwise, and T. M. Evans, *Nucl. Sci. Eng.* **195**, 1–21 (2020).
- [6] D. Mancusi, M. Faucher, and A. Zoia, *Eur. Phys. J. Plus* **137**, 127 (2022).
- [7] T. Yamamoto, *Ann. Nucl. Energy* **47**, 14–20 (2012).
- [8] T. E. Booth, *Nucl. Sci. Eng.* **143**, 291–300 (2003).
- [9] A. Rouchon, A. Zoia, and R. Sanchez, *Ann. Nucl. Energy* **102**, 465–475 (2017).
- [10] H. Belanger, D. Mancusi, and A. Zoia, *Eur. Phys. J. Plus* **135**, 877 (2020).
- [11] D. M. Arnow, M. H. Kalos, M. A. Lee, and K. E. Schmidt, *J. Chem. Phys.* **77**, 5562–5572 (1982).
- [12] J. Sellier, M. Nedjalkov, and I. Dimov, *Phys. Rep.* **577**, 1–34 (2015).
- [13] R. Assaraf, M. Caffarel, and A. Khelif, *J. of Phys. A: Math. Theor.* **40**, 1181 (2007).
- [14] E. Brun, F. Damian, C. Diop, E. Dumonteil, F. Hugot, C. Jouanne, Y. Lee, F. Malvagi, A. Mazzolo, O. Petit, J. Trama, T. Visonneau, and A. Zoia, *Ann. Nucl. Energy* **82**, 151–160 (2015).
- [15] P. K. Romano, N. E. Horelik, B. R. Herman, A. G. Nelson, B. Forget, and K. Smith, *Ann. Nucl. Energy* **82**, 90–97 (2015).
- [16] J. Leppänen, M. Pusa, T. Viitanen, V. Valtavirta, and T. Kaltiaisenaho, *Ann. Nucl. Energy* **82**, 142–150 (2015).
- [17] T. Goorley, M. James, T. Booth, F. Brown, J. Bull, L. J. Cox, J. Durkee, J. Elson, M. Fensin, R. A. Forster, J. Hendricks, H. G. Hughes, R. Johns, B. Kiedrowski, R. Martz, S. Mashnik, G. McKinney, D. Pelowitz, R. Prael, J. Sweezy, L. Waters, T. Wilcox, and T. Zukaitis, *Nucl. Technol.* **180**, 298–315 (2012).
- [18] E. R. Woodcock, T. Murphy, P. J. Hemmings, and T. C. Longworth, *Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry*, Tech. Rep. (Argonne National Laboratory, 1965) ANL-7050.
- [19] J. Leppänen, *Ann. Nucl. Energy* **105**, 161–167 (2017).
- [20] L. L. Carter, E. D. Cashwell, and W. M. Taylor, *Nucl. Sci. Eng.* **48**, 403–411 (1972).
- [21] D. Legrady, B. Molnar, M. Klausz, and T. Major, *Ann. Nucl. Energy* **102**, 116–123 (2017).
- [22] H. Belanger, D. Mancusi, and A. Zoia, *Phys. Rev. E* **104**, 015306 (2021).
- [23] T. E. Booth and J. E. Gubernatis, *Nucl. Sci. Eng.* **165**, 283–291 (2010).
- [24] J. Spanier and E. M. Gelbard, *Monte Carlo Principles and Neutron Transport Problems* (Dover Publications, 2008).
- [25] W. A. Coleman, *Nucl. Sci. Eng.* **32**, 76–81 (1968).
- [26] W. Feller, *An Introduction to Probability Theory and its Applications*, Vol. 1 (Wiley, New York, NY, 1968) p. 229.
- [27] T. A. Brunner and P. S. Brantley, *J. Comput. Phys.* **228**, 3882–3890 (2009).
- [28] H. Niederreiter, *Bull. Am. Math. Soc.* **84**, 957–1041 (1978).
- [29] H. Belanger, 10.5281/zenodo.6546715 (2022), <https://github.com/HunterBelanger/mgmc>.
- [30] E. E. Lewis, M. A. Smith, N. Tsoufanidis, G. Palmiotti, T. A. Taiwo, and R. N. Blomquist, *Benchmark specification for Deterministic 2-D/3-D MOX fuel assembly transport calculations without spatial homogenisation (C5G7 MOX)*, Tech. Rep. JT00105087 (NEA/NSC, 2001).
- [31] P. Zhang, H. Lee, and D. Lee, *J. Comput. Phys.* **305**, 387–402 (2016).
- [32] H. Belanger, D. Mancusi, and A. Zoia, in *M&C 2021* (2021).

10 - Conclusions for Weight Cancellation

In Part II, we have investigated the use of negative-weighted delta tracking in k -eigenvalue power iteration simulations, and explored different weight cancellation techniques. Our main findings are summarized below.

10.1 . Negative-Weighted delta tracking with Power Iteration

The results of Part I indicated that spatially continuous cross sections can be successfully treated using negative-weighted delta tracking. However, Chapter 4 only compared the behavior of tracking methods in fixed-source problems, while most reactor physics problems make use of k -eigenvalue power iteration. Therefore, in Chapters 7 and 8, we attempted to use negative-weighted delta tracking to solve a simple power iteration problem. Unfortunately, large instabilities with the method were encountered, and it was impossible to complete a simulation, as the number of particles stored in memory would grow exponentially with each generation.

We were able to understand the observed explosion in the particle population by writing a set of coupled transport equations where the flux of the positive and negative particles have been considered separately. From this analysis, conducted in Chapter 8, it was observed that the addition of negative weights in the algorithm also introduces a second set of eigenstates which do not correspond to the true Boltzmann equation for the system, but a fictitious system with less sterile capture. One of these new fictitious eigenstates will always have the largest eigenvalue; power iteration will therefore converge to this fictitious state instead of converging to the fundamental mode for the physical Boltzmann equation. Further analysis indicated that the application of a weight cancellation procedure could potentially remove the fictitious eigenstates from the system, permitting power iteration to converge on the physical eigenstate. This hypothesis was confirmed on several simplified transport problems. In addition, we also examined the three-dimensional C5G7 multi-group benchmark, where we showed that weight cancellation was indeed a viable option to enable the convergence of power iteration.

10.2 . Weight Cancellation

Having determined that weight cancellation allowed the convergence of power iteration simulations performed using negative-weighted delta tracking, we began to examine what algorithms were at our disposal to accomplish such an operation in Monte Carlo simulations. A few approximate cancellation methods were identified in the literature; we chose to favor the method proposed by Zhang et al., which imposes a regular mesh on top of the geometry and assumes that, between fission generations, the fission source particles are sorted into the elements of this cancellation mesh based on their phase space coordinates. The average weight of all the particles in each cancellation mesh element is computed, and this average weight is then assigned to all of the fission source particles in that mesh element. This method is very fast and easy to implement in an existing Monte Carlo code with minimal modifications. Approximate regional cancellation was demonstrated on the C5G7 multi-group benchmark in Chapter 7, and allowed the simulation to complete normally, despite the use of negative-weighted delta tracking. The main limitation of approximate regional cancellation is not necessarily that it is approximate, but that there is no way to estimate the bias that it introduces in the simulation results. Currently, the only way to determine if the cancellation mesh is “fine enough” is to refine the mesh, run the simulation, and see how the results have changed. Finding a better method to determine the appropriate refinement of the cancellation mesh in advance should be the subject of future research.

Exact cancellation methods were also considered, which do not impose a bias on the fission source. An exact regional cancellation method proposed by Booth and Gubernatis was considered, as its linear computational complexity with the number of particles increases its potential for use in a production-

level Monte Carlo code. The initial description and implementation of this technique was done for a one-dimensional single-speed problem, leaving many questions unanswered as to how one might go about implementing this method in higher spatial dimensions, or in continuous energy. We have further developed exact regional cancellation for use in three dimensional multi-group benchmarks, demonstrating the method on two versions of the C5G7 benchmark in Chapters 8 and 9. We have also developed the theoretical framework describing under what conditions exact regional cancellation is unbiased, and suggested how it might be performed in continuous-energy simulations, in Chapter 9. Future work will need to focus on the treatment of regional cancellation in a continuous-energy setting: it is likely that the main difficulty in this process will be gaining access to a representation of the nuclear data that is compatible with the requirements of cancellation, which is not necessarily available in standard Monte Carlo codes.

In Chapter 9, we have also developed a technique to choose the parameter used in exact regional cancellation such that the amount of cancellation performed is maximized. Using the optimal approach on the C5G7 benchmark, a higher cancellation efficiency was achieved than when using the minimum value of the expected fission emission density in the cancellation region, as was proposed by Booth and Gubernatis. The implementation of this method has also made it possible to perform cancellation in non-cuboid cancellation regions (e.g. cylindrical fuel pins). Despite this improvement in the efficiency of the algorithm, it was still observed that the efficiency of exact regional cancellation is extremely sensitive to the refinement of the cancellation mesh used to define cancellation regions. Currently, there is no way of knowing if a cancellation mesh has the appropriate refinement to allow enough weight cancellation to occur, which is in turn mandatory for power iteration simulations to converge. The meshes used in this work were mostly determined by trial and error; determining an adequate cancellation mesh before running a simulation should be a topic of future investigations to improve the usability of exact regional cancellation.

In Part II, we have only considered the application of weight cancellation to the convergence of k -eigenvalue power iteration simulations when using negative-weighted delta tracking. There are many other types of Monte Carlo problems in the field of nuclear reactor physics which necessitate the use of negative particle weights. Examples of these are the computation of higher harmonics of the flux, the search for critical buckling, and the solution of the neutron noise equation. All of these types of simulations could potentially benefit from the new weight cancellation techniques which have been presented in this work. In Part III, we examine the impact of weight cancellation methods on solving the neutron noise equation by Monte Carlo simulation.

Part III

Neutron Noise

11 - Application of Weight Cancellation to Neutron Noise

Our exploration of weight cancellation techniques up until this point has been solely motivated by the newly discovered fact that k -eigenvalue power iteration problems using negative-weighted delta tracking do not converge on the fundamental eigenstate of the Boltzmann equation in the absence of weight cancellation. The desire to use negative-weighted delta tracking for power iteration problems is rooted in the results gathered in Part I, where the tracking method appeared to be a very suitable option for treating spatially continuous cross sections as initially tested on fixed-source problems. There are, however, many other types of Monte Carlo problems in the domain of nuclear reactor physics where mixtures of positive and negative weights are used. We will briefly overview these applications, as the cancellation methods discussed in this thesis are also likely to be of interest in these cases.

Both the approximate regional cancellation method and the exact regional cancellation method were initially proposed in papers about the calculation of the second harmonic (or even higher harmonics) of the Boltzmann k -eigenvalue equation [1–5]. While the first harmonic (the static flux) is positive throughout the problem domain, higher harmonics necessarily have “nodes”, where the solution changes sign from positive to negative [6]. Knowing these higher eigenstates is useful for higher-order perturbation theory; furthermore, the second eigenstate enables evaluating the dominance ratio, defined as k_1/k_0 , where k_0 is the eigenvalue associated with the first harmonic (i.e. k_{eff}), and k_1 is the eigenvalue for the second harmonic [6]. The dominance ratio determines how fast the fission source in a Monte Carlo criticality simulation will converge [7]. These higher modes can be of interest to reactor physicists for a variety of calculations, and their investigation has fostered the development of the approximate fission matrix method, which has been implemented in a variety of general-purpose Monte Carlo codes [8]. The fission matrix method is only able to approximate the higher eigenstates of the fission source [8]. Booth was one of the first to propose an exact method to estimate higher eigenstates of the flux [1]. His method poses the advantage of not needing to store the large fission matrix, and being able to solve for only the first few harmonics (as opposed to obtaining the entire spectrum with the fission matrix method). The pioneering methods developed by Booth to calculate higher harmonics have been expanded and improved by several authors: the common point of all these techniques is that they all require that particles to be allowed to carry positive and negative weights, and that a weight cancellation technique be applied to these particles [3, 4].

Another kind of problem involving particles with complex weights, where each component can be positive or negative, is that of critical buckling search [9, 10]. Several formulations of this problem have been proposed; in the full variant where particles are required to carry complex weights, it is assumed that the angular flux can be factored into the following form:

$$\varphi(\mathbf{r}, \hat{\Omega}, E) = \phi(\mathbf{r}, \hat{\Omega}, E) \exp(i\mathbf{B} \cdot \mathbf{r}), \quad (11.1)$$

where $\phi(\mathbf{r}, \hat{\Omega}, E)$ is a spatial fine structure, \mathbf{B} is the geometric buckling vector, and $\exp(i\mathbf{B} \cdot \mathbf{r})$ is a macroscopic energy-independent factor [10]. Such an algorithm can be used to estimate the diffusion coefficient by means of Monte Carlo methods, or to perform leakage-corrected calculations of pin cells or fuel assemblies, such as those done for the generation of multi-group cross sections [10]. Yamamoto’s proposed method can either be used to determine the magnitude of the geometric buckling vector for a given value of k_{eff} , or conversely determine k_{eff} for a known geometric buckling. In order to ensure convergence, Yamamoto makes use of a cancellation procedure in his implementation, making critical buckling a domain of interest for the application of exact or approximate weight cancellation techniques as well.

A phenomenon referred to as “neutron noise” is known to affect power reactors when fluid-structure interactions or fluctuations in the moderator density occur in the core, causing a time dependent perturbation around the stationary neutron flux. Although noise is generally an unwanted phenomenon, the information carried in the neutron flux fluctuations can be usefully extracted for a variety of purposes, ranging from core monitoring and diagnostics to cross section measurements. Perhaps the oldest work addressing this topic comes from Oak Ridge National Laboratory in the late 1940’s [11]. Since then, many occurrences of

neutron noise have been encountered, and many analytical and computational methods have been developed to model and predict the presence of such perturbations in power reactor cores [12–19]. A proper summary of the history of this vast field is unfortunately not possible here, but the interested reader is encouraged to read the first chapter of Rouchon's PhD thesis, which gives a thorough overview of the subject [20]. Neutron noise can be an indication of potentially hazardous behavior in the core, such as fuel rod or assembly vibrations which could lead to fretting. Since it is impossible to look inside the core to examine the cause of these vibrations, the EU H2020 CORTEX project (2017-2021) has recently promoted the development of simulation tools aimed at understanding what types of core phenomena could generate specific noise perturbations, in view of solving the inverse problem of locating the origin of the detected noise [21]. From this international effort, two Monte Carlo solvers have been independently developed to simulate the perturbation of the static neutron flux for a variety of different types of core perturbations [20, 22–24]. Both of these methods solve the neutron noise equations in the frequency domain, where particles must carry complex statistical weights with real and imaginary components being allowed to be positive or negative. As illustrated in this Chapter, the (complex) noise field can be estimated by solving a fixed-source problem, where the noise source particles are sampled from a complex source and are then propagated by means of a Boltzmann-like operator which is also complex. For reasons that will be discussed subsequently, the simulation method proposed by Rouchon et al. did not make use of any cancellation method [24], whereas Yamamoto's initial implementation did make use of weight cancellation [22]. It is quite possible that the advancements to weight cancellation methods developed in Part II could be applied to the problem of neutron noise, and Part III of this manuscript is dedicated to this question. In Sec. 11.1, we shall summarize the derivation of the canonical linearized neutron noise equations in the frequency domain. Section 11.2 covers the methodology for sampling the noise source particles, which are then transported according to the rules presented in Sec. 11.3. A branchless noise sampling technique is examined in Chapter 12 as a possible variance reduction technique, the results of which are summarized in Sec. 11.4. Finally, we will summarize the observed effects of applying weight cancellation to neutron noise simulations in Sec. 11.5. A thorough analysis will be provided in Chapter 12 for the case of neutron noise induced by oscillations, and in Chapter 13 for the case of neutron noise induced by mechanical vibrations.

11.1 . Derivation of the Canonical Noise Equations

In deriving the linearized neutron noise equations, we first start with the time-dependent Boltzmann transport equation:

$$\begin{aligned} \mathcal{B}_k(\mathbf{r}, \hat{\Omega}, E, t)\varphi(\mathbf{r}, \hat{\Omega}, E, t) = 0 = & \left[\frac{1}{v} \frac{\partial}{\partial t} + \hat{\Omega} \cdot \nabla + \Sigma_t(\mathbf{r}, E, t) \right] \varphi(\mathbf{r}, \hat{\Omega}, E, t) - \\ & \iint [\nu\Sigma f]_s(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, t) \varphi(\mathbf{r}, \hat{\Omega}', E', t) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \iint [\nu\Sigma f]_{f,p}(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, t) \varphi(\mathbf{r}, \hat{\Omega}', E', t) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \sum_j \iiint \lambda_j e^{-\lambda_j(t-t')} [\nu\Sigma f]_{f,d}^j(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, t') \varphi(\mathbf{r}, \hat{\Omega}', E', t) dE' d\hat{\Omega}' dt', \quad (11.2) \end{aligned}$$

where the shorthand

$$[\nu\Sigma f]_{\alpha}(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, t') = \nu_{\alpha}(\mathbf{r}, E) \Sigma_{\alpha}(\mathbf{r}, E) f_{\alpha}(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) \quad (11.3)$$

has been employed. In Eq. (11.2), one will notice a factor of $1/k_{\text{eff}}$ scaling the fission production terms. Neutron noise occurs in power reactors which are critical, and dividing by the multiplication factor enforces this.¹ In general, we consider that all of the material terms (energy-angle distributions, cross sections, and

¹While a reactor may be exactly critical in real life, our simulations of the same reactor often are not, due to uncertainties in technological specifications of the core and in nuclear data.

yields) for a reaction α can be decomposed into a stationary and a perturbed component as²

$$[\nu\Sigma f]_{\alpha}(\mathbf{r}, E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}', t) = [\nu\Sigma f]_{\alpha}(\mathbf{r}, E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}') + \delta[\nu\Sigma f]_{\alpha}(\mathbf{r}, E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}', t). \quad (11.4)$$

Such a decomposition permits us to also decompose the transport operator as

$$\mathcal{B}_k(\mathbf{r}, \hat{\Omega}, E, t) = \mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) + \delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) \quad (11.5)$$

with the operator \mathcal{B} taking the form of

$$\begin{aligned} \mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) = & \frac{1}{v} \frac{\partial}{\partial t} + \hat{\Omega} \cdot \nabla + \Sigma_t(\mathbf{r}, E) - \iint [\nu\Sigma f]_s(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \iint [\nu\Sigma f]_{f,p}(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \sum_j \iiint \lambda_j e^{-\lambda_j(t-t')} [\nu\Sigma f]_{f,d}^j(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) dE' d\hat{\Omega}' dt' \end{aligned} \quad (11.6)$$

and the perturbation operator $\delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t)$ taking the form of

$$\begin{aligned} \delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) = & \hat{\Omega} \cdot \nabla + \delta\Sigma_t(\mathbf{r}, E, t) - \iint \delta[\nu\Sigma f]_s(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, t) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \iint \delta[\nu\Sigma f]_{f,p}(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, t) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \sum_j \iiint \lambda_j e^{-\lambda_j(t-t')} \delta[\nu\Sigma f]_{f,d}^j(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, t) dE' d\hat{\Omega}' dt' \end{aligned} \quad (11.7)$$

We also assume that the flux can be decomposed in a similar manner:

$$\varphi(\mathbf{r}, \hat{\Omega}, E, t) = \varphi_c(\mathbf{r}, \hat{\Omega}, E) + \delta\varphi(\mathbf{r}, \hat{\Omega}, E, t), \quad (11.8)$$

where $\varphi_c(\mathbf{r}, \hat{\Omega}, E)$ is the critical flux, and the small perturbation $\delta\varphi(\mathbf{r}, \hat{\Omega}, E, t)$ is the *neutron noise*. Substituting Eqs. (11.5) and (11.8) into Eq. (11.2) yields

$$\begin{aligned} \mathcal{B}_k(\mathbf{r}, \hat{\Omega}, E, t) \varphi(\mathbf{r}, \hat{\Omega}, E, t) = & \left[\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) + \delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) \right] \left[\varphi_c(\mathbf{r}, \hat{\Omega}, E) + \delta\varphi(\mathbf{r}, \hat{\Omega}, E, t) \right] = \\ & \mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) \varphi_c(\mathbf{r}, \hat{\Omega}, E) + \mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) \delta\varphi(\mathbf{r}, \hat{\Omega}, E, t) + \\ & \delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) \varphi_c(\mathbf{r}, \hat{\Omega}, E) + \delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) \delta\varphi(\mathbf{r}, \hat{\Omega}, E, t) = 0. \end{aligned} \quad (11.9)$$

Disregarding the second-order time-dependent term, and realizing that $\mathcal{B}\varphi_c = 0$, the two remaining terms may be rearranged as

$$\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) \delta\varphi(\mathbf{r}, \hat{\Omega}, E, t) = -\delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, t) \varphi_c(\mathbf{r}, \hat{\Omega}, E). \quad (11.10)$$

This is the linearized neutron noise equation in the time domain. Typically we are interested in periodic solutions corresponding to times long after the transient regime. It is therefore beneficial to move to the frequency domain using a Fourier transform

$$g(\omega) = \mathcal{F}[g(t)] = \int_{-\infty}^{+\infty} g(t) e^{-i\omega t} dt. \quad (11.11)$$

Applying Eq. (11.11) to Eq. (11.10) finally results in the ‘‘orthodox’’ linearized neutron noise equation in the frequency domain [14]:

$$\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, \omega) \delta\varphi(\mathbf{r}, \hat{\Omega}, E, \omega) = -\delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, \omega) \varphi_c(\mathbf{r}, \hat{\Omega}, E) \quad (11.12)$$

²In this work, we do not consider the possibility of precursor decay constants being time-dependent.

with

$$\begin{aligned} \mathcal{B}(\mathbf{r}, \hat{\Omega}, E, \omega) = & i\frac{\omega}{v} + \hat{\Omega} \cdot \nabla + \Sigma_t(\mathbf{r}, E) - \iint [\nu\Sigma f]_s(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \iint [\nu\Sigma f]_{f,p}(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \sum_j \iint \frac{\lambda_j}{\lambda_j + i\omega} [\nu\Sigma f]_{f,d}^j(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) dE' d\hat{\Omega}' \end{aligned} \quad (11.13)$$

and

$$\begin{aligned} \delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, \omega) = & \hat{\Omega} \cdot \nabla + \delta\Sigma_t(\mathbf{r}, E, \omega) - \iint \delta[\nu\Sigma f]_s(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, \omega) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \iint \delta[\nu\Sigma f]_{f,p}(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, \omega) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \sum_j \iint \frac{\lambda_j}{\lambda_j + i\omega} \delta[\nu\Sigma f]_{f,d}^j(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, \omega) dE' d\hat{\Omega}'. \end{aligned} \quad (11.14)$$

Eq. (11.12) has several interesting features which should be noted. First of all, there is no eigenvalue associated with the neutron noise ($\delta\varphi$), indicating that this is actually a fixed-source problem. The noise source is $-\delta\mathcal{B}\varphi_c$, and depends on the critical flux. Finally, since both $\delta\mathcal{B}$ and \mathcal{B} are complex, the neutron noise is also complex and we deduce that our noise particles must correspondingly have complex statistical weights.

11.2 . Sampling of the Noise Source

As the noise equation is effectively a fixed-source problem, we start our treatment of how to solve the noise equation with the sampling of the noise source particles. A key point is that the noise source requires the critical flux φ_c . A noise simulation therefore starts with normal k -eigenvalue power iteration, where a large enough number of batches are performed so that the fission source has converged to the fundamental eigenstate. Once the fission source has converged, one or more additional power iteration generations are performed, during which the noise source particles for a single noise batch are sampled based on the term $-\delta\mathcal{B}\varphi_c$. The exact methodology for sampling the noise particles depends on the type of perturbation which is being considered, which in turn determines the perturbation operator $-\delta\mathcal{B}$. In this thesis, we will consider two different types of perturbations: oscillations in the macroscopic cross sections, and mechanical vibrations.

11.2.1 . Cross Section Oscillations

The problem of cross section oscillations makes the assumption that the perturbation in the core takes the form of a small sinusoidal oscillation on top of the static value of the macroscopic cross sections:

$$\Sigma_\alpha(\mathbf{r}, E, t) = \Sigma_\alpha(\mathbf{r}, E) [1 + \varepsilon_\alpha \mathbb{1}_{\mathcal{P}}(\mathbf{r}) \sin(\omega_0 t)] = \Sigma_\alpha(\mathbf{r}, E) + \delta\Sigma_\alpha(\mathbf{r}, E, t), \quad (11.15)$$

and $\mathbb{1}_{\mathcal{P}}(\mathbf{r})$ is an indicator function for the perturbed region \mathcal{P} , so that

$$\mathbb{1}_{\mathcal{P}}(\mathbf{r}) = \begin{cases} 1 & \mathbf{r} \in \mathcal{P} \\ 0 & \mathbf{r} \notin \mathcal{P} \end{cases}. \quad (11.16)$$

In the frequency domain, inside the perturbed region \mathcal{P} , this perturbation term is then

$$\delta\Sigma_\alpha(\mathbf{r}, E, \omega) = -i\pi\varepsilon_\alpha\Sigma_\alpha(\mathbf{r}, E) [\delta(\omega - \omega_0) + \delta(\omega + \omega_0)]. \quad (11.17)$$

Such a problem is admittedly academic, as it is rather unlikely that a phenomenon exists which could impose a perturbation on the macroscopic cross section, without affecting the macroscopic energy-angle

distributions. Nonetheless, this remains an excellent problem for benchmarking codes, and comparing different noise solvers [25]. Familiarity with the treatment of cross section oscillations is also integral to understanding the sampling strategy which we will develop for mechanical vibrations in Chapter 13. The sampling of the noise source for cross section oscillations is briefly outlined here and thoroughly examined in Chapters 12 and 13.

Time-dependent cross sections in Eq. (11.15) lead to the integrands in Eq. (11.14) having the form

$$\delta [\nu\Sigma f]_{\alpha}(\mathbf{r}, E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}', \omega) = \nu_{\alpha}(\mathbf{r}, E) f_{\alpha}(\mathbf{r}, E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}') \delta\Sigma_{\alpha}(\mathbf{r}, E, \omega) \quad (11.18)$$

inside the perturbed region of the geometry, where only the macroscopic cross sections are assumed to be time-dependent. The integrals for scattering and fission appearing in Eq. (11.14) can then be rewritten as

$$\iint \underbrace{\frac{\delta\Sigma_{\alpha}(\mathbf{r}, E', \omega)}{\Sigma_{\alpha}(\mathbf{r}, E')}}_{\text{complex importance factor}} \underbrace{\nu_{\alpha}(\mathbf{r}, E') \Sigma_{\alpha}(\mathbf{r}, E') f_{\alpha}(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega})}_{\text{standard production rate}} dE' d\hat{\Omega}'. \quad (11.19)$$

From this perspective, the perturbation operator in Eq. (11.14) can be sampled as a standard production rate, but with an additional complex yield. During the last power iteration generation, neutrons are propagated normally, using implicit capture and forced fission. Additionally, whenever a neutron undergoes a collision in the perturbed volume \mathcal{P} , the noise source sampling procedure is applied. A nuclide is sampled in the standard manner, which will be used for sampling noise source particles. Should the sampled nuclide be fissile, a random variable $\xi \sim \mathcal{U}(0, 1)$ is used to determine how many fission noise particles will be sampled:

$$n_f = \left\lfloor \frac{\nu_f(E) \sigma_f(E)}{\sigma_t(E) k_{\text{eff}}} + \xi \right\rfloor. \quad (11.20)$$

These n_f fission noise neutrons are born with a weight $w \delta\Sigma_f(\mathbf{r}, E, \omega) / \Sigma_f(\mathbf{r}, E)$ (w being the weight of the parent particle), and have their energies and directions sampled in the standard manner. Any delayed neutrons have their weights multiplied by an additional factor of $\lambda_j / (\lambda_j + i\omega)$. A scatter noise source particle is also sampled, born with a weight of

$$w \frac{\delta\Sigma_s(\mathbf{r}, E, \omega) \sigma_s(E)}{\Sigma_s(\mathbf{r}, E) \sigma_t(E)}, \quad (11.21)$$

where $\sigma_{i,t}(E)$ is the microscopic total cross section of nuclide i , and $\sigma_{i,s}(E)$ is the microscopic scattering cross section. The scattering channel, energy, and direction are sampled as usual. Finally, a “copy” of the incident particle is created, with the same direction and energy, and a weight of

$$w \frac{\delta\Sigma_t(\mathbf{r}, E, \omega)}{\Sigma_t(\mathbf{r}, E)}. \quad (11.22)$$

All of these noise source particles are placed into a special bank, for use after the last power iteration cycle.

11.2.2 . Mechanical Vibrations

Mechanical vibrations typically refer to either the vibrations of individual fuel pins, control rods, or entire fuel assemblies in the core. The initial methodology to sample the noise source from vibrations in the development version of TRIPOLI-4[®] was an approximate technique, and is covered in Sec. 13.3.2. In this section we summarize a novel exact methodology for sampling the noise source from vibrations, which will then be extensively detailed in Sec. 13.3.2.

Mechanical vibrations can be modeled in terms of the interfaces between material regions being time-dependent: the neutron noise is induced by the fact that the material present at a given location becomes a function of time, where “invading” materials are moving into “host” materials. By linear superposition, the effects of multiple interfaces can be decomposed into the sum of the effects induced by each interface. We will then consider the case of a material interface in one spatial dimension, where material M_L can be found at positions $x < x_0$ and material M_R can be found at positions $x > x_0$ in nominal conditions, before

the perturbation is introduced in the system. If this interface at x_0 begins vibrating periodically along the x -axis, Zoia et al. have demonstrated that the material at position x and time t can be written as [26]

$$M(x, t) = \Delta M [H(x - x_0) - H(x - x_0 - \varepsilon \sin(\omega_0 t))], \quad (11.23)$$

$H(x)$ being the Heaviside step function. The noise source will thus stem from $M(x, t)$ being a function of time, which in turns makes cross sections, energy-angle distributions and yields also functions of time. The difficulty of sampling the noise source for this type of perturbation is not immediately evident at this point. When treating the case of cross section oscillations, only one material formally “partakes” in the perturbation, as the geometry is not being perturbed: only the properties of the materials at a given position become time-dependent. Conversely, in mechanical vibrations, the geometric configuration of these materials is changing over time. Taking the example of a fuel pin vibrating in water, when a neutron undergoes a collision in the water, just outside of the fuel pin, it should be possible to produce fission noise source particles, despite the fact that no fissile isotopes exist at that position in the static geometry. Thus, for cross section oscillations, only the material in the static geometry at the collision site was required, while in mechanical vibrations we must also consider the material on the other side of the interface, which is not located at our current particle position in the static geometry. From this insight, we then deduce that all of the time dependence for the production rate of a given reaction is better represented as a time dependence of the isotope concentrations:

$$\nu_\alpha(\mathbf{r}, E, t) \Sigma_\alpha(\mathbf{r}, E, t) f_\alpha(\mathbf{r}, E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}', t) = \sum_k N_k(\mathbf{r}, t) \nu_{k,\alpha}(E) \sigma_{k,\alpha}(E) f_{k,\alpha}(E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}'), \quad (11.24)$$

where N_k is the concentration of isotope k . The perturbation term in the frequency domain is then written as

$$\delta[\nu \Sigma f]_\alpha(\mathbf{r}, E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}', \omega) = \sum_k \delta N_k(\mathbf{r}, \omega) \nu_{k,\alpha}(E) \sigma_{k,\alpha}(E) f_{k,\alpha}(E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}'). \quad (11.25)$$

Section 13.3.2 will show that for a vibration along the x -axis about the interface at x_0 the Fourier-transformed isotope concentrations stemming from Eq. (11.23) is

$$\delta N_k(\mathbf{r}, \omega) = \Delta N_k \left\{ c_0(x, x_0) \delta(\omega) + \sum_n c_n(x, x_0) [\delta(\omega - n\omega_0) + \delta(\omega + n\omega_0) (-1)^n] \right\}, \quad (11.26)$$

where $\Delta N_k = N_k(x < x_0) - N_k(x > x_0)$ is the difference of the two isotope concentrations across the interface, and

$$c_n(x, x_0) = \frac{2}{n} \sin\left(n \arcsin\left(\frac{x - x_0}{\varepsilon}\right)\right) e^{-in\pi/2} \quad \forall n \geq 1, \quad (11.27)$$

and

$$c_0(x, x_0) = \begin{cases} \pi - 2 \arcsin\left(\frac{x - x_0}{\varepsilon}\right) & \text{for } x \geq x_0 \\ -\pi - 2 \arcsin\left(\frac{x - x_0}{\varepsilon}\right) & \text{for } x < x_0 \end{cases}, \quad (11.28)$$

are the Fourier coefficients.

Sampling Eq. (11.25) is non-trivial, as $\delta N_k(\mathbf{r}, \omega) \neq 0$ when $N_k(\mathbf{r}) = 0$. We therefore cannot use the same strategy as in Eq. (11.19), multiplying and dividing by $N_k(\mathbf{r})$ in order to obtain a production rate. To get around this problem, we define a fictitious material M^* which is composed of the union of all isotopes in materials M_L and M_R . With this new fictitious material, we can now rewrite the integrals for sampling the noise source particles as

$$\iint \sum_{k \in \hat{M}} \frac{\delta N_k(\mathbf{r}, \omega)}{N_k^*} N_k^* \nu_{k,\alpha}(E) \sigma_{k,\alpha}(E) f_{k,\alpha}(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) dE' d\hat{\Omega}'. \quad (11.29)$$

In this form, it becomes clear that we can now sample noise source particles in a manner similar to the method used for cross section oscillations. Nuclide k is sampled from the fictitious material M^* as would

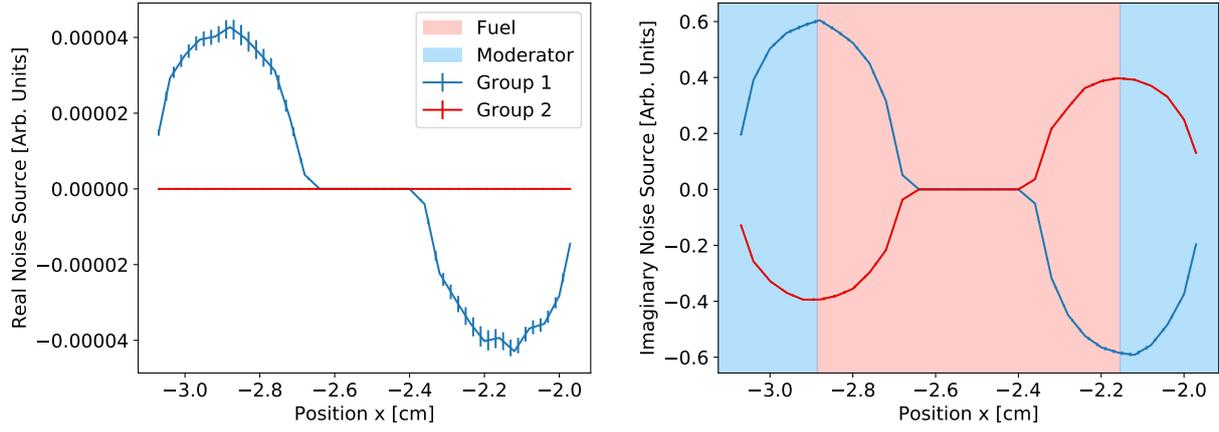


Figure 11.1: Example of the noise source from a vibrating square fuel pin, in a 2-group simulation.

be done for a normal collision. Fission, scatter, and copy noise source particles are sampled from nuclide k in a manner nearly identical to that used for cross section oscillations. Equation (11.20) is used to determine how many fission noise particles will be sampled, each being born with a weight of

$$w \frac{\delta N_k(\mathbf{r}, \omega)}{N_k^*} \frac{\Sigma_t^*(E)}{\Sigma_t(\mathbf{r}, E)}, \quad (11.30)$$

Σ_t^* being the macroscopic total cross section of the fictitious material M^* . Delayed fission noise particles have their weight multiplied by $\lambda_j/(\lambda_j + i\omega)$, as before. A scatter noise source particle is also sampled, having a weight of

$$w \frac{\delta N_k(\mathbf{r}, \omega)}{N_k^*} \frac{\Sigma_t^*(E)}{\Sigma_t(\mathbf{r}, E)} \frac{\sigma_s(E)}{\sigma_t(E)}. \quad (11.31)$$

The copy noise source particle does not actually need the fictitious material to be sampled; it will be created with a weight of

$$w \frac{\delta \Sigma_t(\mathbf{r}, E)}{\Sigma_t(\mathbf{r}, E)}, \quad (11.32)$$

$\delta \Sigma_t(\mathbf{r}, E)$ being calculated in a manner identical to Eq. (11.26).

An example of the noise source for a single vibrating square fuel pin in a UOX assembly is presented in Fig. 11.1. This is the same problem that is examined in Chapter 13. Comparisons of this new exact source sampling technique are made with the approximate method in TRIPOLI-4[®], in Sec. 13.6.1. For a realistic problem such as the vibration of a fuel pin along a given axis, the vibrating region actually has two interfaces, one on each side. The real component of the noise source in Fig. 11.1 is much smaller than the imaginary component. Upon examination of Eqs. (11.27) and (11.30), we see that δN_k is purely imaginary for the first harmonic (i.e. $n = 1$). The real component therefore only stems from the imaginary component of the complex factor

$$\frac{\lambda_j}{\lambda_j + i\omega} = \frac{\lambda_j^2 - i\lambda_j\omega}{\lambda_j^2 + \omega^2} \quad (11.33)$$

multiplying the yield of delayed neutrons. For realistic problems from reactor physics, frequencies of interest are generally of the order of $\omega \approx 2\pi \text{ rad s}^{-1}$, while the average precursor decay constant is approximately $\lambda \approx 0.08 \text{ s}^{-1}$ [20, 25]. This leads to a real component which is several orders of magnitude smaller than the imaginary component for the first harmonic ($n = 1$).

Looking at the left and right side of Fig. 11.1, it is noticed that the source is positive on one side of the pin, and negative on the other (which side is positive depends on the energy group in question). In Sec. 13.6.2 we will show that having a positive source very close to a negative source in phase space leads to very large statistical uncertainties in the noise field, as intuitively expected, due to the linear superposition of positive and negative contributions.

11.2.3 . Frequency Dependence of the Noise Equation

Our initial derivation of the linearized noise equation treated the angular frequency ω as a continuous variable. However, in Secs. 11.2.1 and 11.2.2 we have shown that the noise source for oscillations and mechanical vibrations is actually composed of discrete harmonics which are multiples of the perturbation frequency ω_0 :

$$-\delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, \omega)\varphi_c(\mathbf{r}, \hat{\Omega}, E) = -\sum_n \delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, n\omega_0)\varphi_c(\mathbf{r}, \hat{\Omega}, E)\delta(\omega - n\omega_0). \quad (11.34)$$

Consequently, the noise field must be decomposed into a sum of discrete harmonics:

$$\delta\varphi(\mathbf{r}, \hat{\Omega}, E, \omega) = \sum_n \delta\varphi(\mathbf{r}, \hat{\Omega}, E, n\omega_0)\delta(\omega - n\omega_0). \quad (11.35)$$

Equation (11.12) can therefore be described as a set of fully decoupled equations, with one equation per harmonic:

$$\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, n\omega_0)\delta\varphi(\mathbf{r}, \hat{\Omega}, E, n\omega_0) = -\delta\mathcal{B}(\mathbf{r}, \hat{\Omega}, E, n\omega_0)\varphi_c(\mathbf{r}, \hat{\Omega}, E). \quad (11.36)$$

For the case of oscillating cross sections, there are only two possible frequencies ($n = \pm 1$), while there is an infinite number of positive and negative discrete frequencies for the case of vibrations. The noise fields resulting from positive and negative harmonics are related through

$$\delta\varphi(\mathbf{r}, \hat{\Omega}, E, -n\omega_0) = \delta\varphi^\dagger(\mathbf{r}, \hat{\Omega}, E, n\omega_0), \quad (11.37)$$

where $(\cdot)^\dagger$ indicates the complex conjugate, which means that we only need to solve for the positive frequencies. Cross section oscillations are therefore monochromatic problems, and there is only one frequency which needs to be solved for ($\omega = \omega_0$). When solving the noise equation for vibration problems by Monte Carlo, each simulation considers only a single (positive) harmonic $\delta\varphi(\mathbf{r}, \hat{\Omega}, E, n\omega_0)$ at a time. In practice, on physical grounds it is only necessary to obtain the first few harmonics $n = 1$ and $n = 2$. Furthermore, Zoia et al. have shown that the higher harmonics ($n \geq 2$) might substantially deviate from the exact Fourier-transformed solutions due to the approximations made in the orthodox linearization [26]. The noise field for vibrations also has a component at $n = 0$, which represents the time-averaged effect on the critical flux, which is imposed by the vibration. Since adding a vibration will in general change the reactivity and fundamental eigenstate, using the critical flux φ_c to sample the noise source without taking into account the $n = 0$ correction introduces a further approximation. Zoia et al. mention this limit of the linearized noise equations, and discuss possible approaches to improve on this approximation [26].

11.3 . Transport of Noise Particles

Now that noise source particles have been sampled, they must be transported, according to operator \mathcal{B} from Eq. (11.13). Upon inspection, Eq. (11.13) is actually quite similar to the standard Boltzmann operator used to sample particle flights and collisions in standard Monte Carlo codes, the main differences being the addition of a complex absorption term, $i\omega/v$, and a complex factor in front of the delayed fission operator. The complex factor for delayed neutrons can be dealt with using a weight multiplier, as detailed earlier in the context of the noise source sampling. Two different methodologies have been presented in the literature to handle the complex absorption term, and will be outlined here.

Yamamoto, who wrote the first Monte Carlo noise solver in the frequency domain, decided to handle the imaginary absorption term with a complex exponential transform [22]. In this approach, the distance to a collision site is sampled with the total macroscopic cross section, and the complex weight of the particle changes continuously over the length of the flight, as [22]

$$w(\mathbf{r}_0 + d\hat{\Omega}) = w(\mathbf{r}_0) \left[\cos\left(\frac{\omega}{v}d\right) - i \sin\left(\frac{\omega}{v}d\right) \right]. \quad (11.38)$$

The application of a complex exponential transform complicates the scoring of track-length tallies for the noise field as well. A noise particle which will fly distance d_k through the tally mesh element k will contribute

$$\int_0^{d_k} w(\mathbf{r}_0 + s\hat{\Omega}) ds = w(\mathbf{r}_0) \frac{\omega}{v} \left[\sin\left(\frac{\omega}{v} d_k\right) + i \left(\cos\left(\frac{\omega}{v} d_k\right) - 1 \right) \right] \quad (11.39)$$

to the noise field [22]. Two additional modifications are required in the transport algorithm, the first being with the sampling of fission particles. As the weight of noise particles is complex, the number of fission particles produced at a collision site should be determined as

$$n = \left\lfloor \frac{\nu(E)\sigma_f(E)}{\sigma_t(E)k_{\text{eff}}} + \xi \right\rfloor, \quad (11.40)$$

each fission noise particle inheriting its parent's weight.³ All delayed neutrons must be multiplied by the same complex factor in Eq. (11.33), as used when sampling fission noise source particles. The second modification is in relation to Russian roulette, which is typically only defined for particles with a single positive weight. Yamamoto chose to perform roulette on the real and imaginary components of the weight separately (using their absolute value), only killing the particle once both the real and imaginary components have been sent to zero [22]. Apart from these modifications, transport proceeds as normal.

A second Monte Carlo noise solver has been implemented in the development version of TRIPOLI-4[®] by Rouchon et al., using a slightly different methodology [24]. The main difference in the transport of noise particles between the two implementations is the handling of the complex absorption term. Rouchon et al. have modified Eq. (11.13) by adding and subtracting a factor of $\eta\omega\delta\varphi/v$, where η is a real positive quantity, resulting in a transport equation which takes the form

$$\left[\hat{\Omega} \cdot \nabla + \underbrace{\Sigma_t(\mathbf{r}, E) + \eta \frac{\omega}{v}}_{\tilde{\Sigma}_t} \right] \delta\varphi(\mathbf{r}, \hat{\Omega}, E, \omega) = \underbrace{\frac{\eta - i}{\eta}}_{\nu_\omega} \underbrace{\eta \frac{\omega}{v}}_{\Sigma_\omega} \delta\varphi(\mathbf{r}, \hat{\Omega}, E, \omega) + \dots \quad (11.41)$$

We now have an effective real and positive total cross section of $\tilde{\Sigma}_t$, which can be used to sample flight distances. The problem of complex absorption has now been transformed into a complex copy term appearing at the right-hand side, with corresponding copy cross section Σ_ω . Once at the collision site, a copy of the incident particle is made with a probability of $\Sigma_\omega/\tilde{\Sigma}_t$; should the copy particle be created, its weight is multiplied by the complex yield ν_ω [24]. With this methodology, no modifications are required for tallies, as the particle weight remains constant over the flight. Apart from this new complex copy term in the collision operator, transport is performed as normal, where we simply use $\tilde{\Sigma}_t$ in place of Σ_t . Rouchon et al. make use of the same roulette algorithm as Yamamoto. In general, it was determined that for most frequencies of interest for reactor physics $\eta = 1$ is a suitable choice, although different values of η were required for convergence at frequencies outside the “plateau region” [24]

$$\lambda < \omega < \lambda + \frac{\beta_{\text{eff}}}{\Lambda_{\text{eff}}}. \quad (11.42)$$

Both Yamamoto and Rouchon et al. decided to let their particles carry complex statistical weights [22, 24]. This is certainly a valid option, but it is worth noting that there is another option. One could choose to let each particle be either real **or** imaginary, but not both. It is unclear whether such a “split species” approach is beneficial for noise simulations. Intuitively, splitting the real and imaginary components could potentially reduce correlations between the two quantities, but further investigations would be required to quantify this statement. Performing split species transport would not require any specific modifications to the transport methodology, other than splitting noise particles after any complex yield is applied to the particle (at the copy operation, and at delayed fission).

³This approach for determining the number of fission neutrons is standard in several Monte Carlo codes, but differs from the approach taken by MCNP (see Eq. (6.2)).

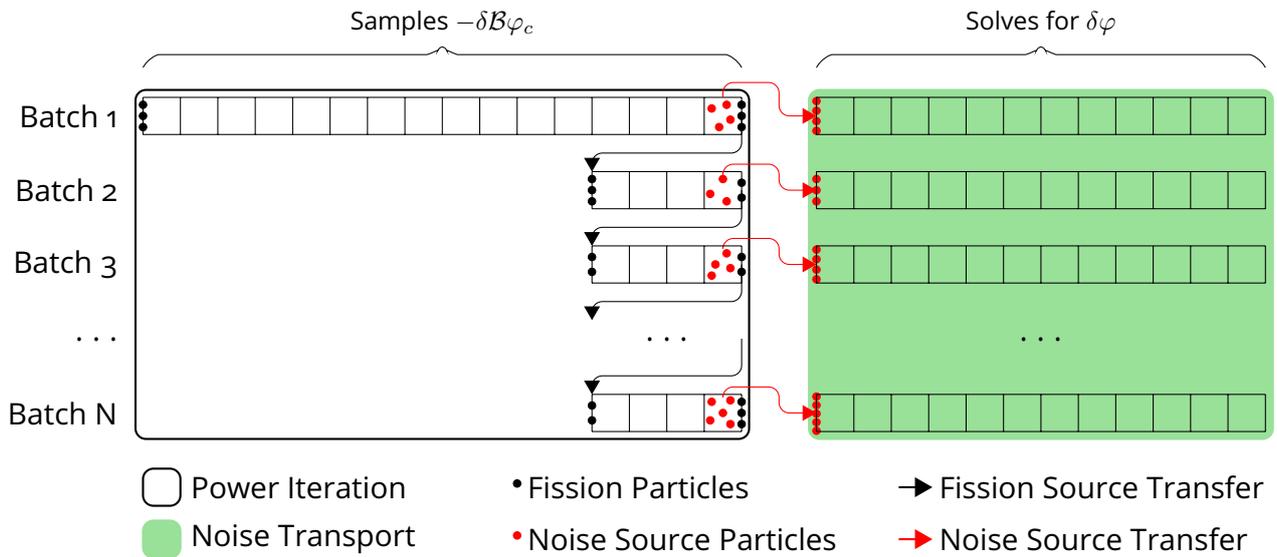


Figure 11.2: Depiction of the solution strategy for neutron noise simulations. Each noise batch begins with a series of power iteration generations which are required to sample the noise source. Sampled noise particles are then sent to noise transport until they have all died. A new noise source can be sampled from the fission particles which remain by performing more power iteration generations.

In Sec. 11.3 we have detailed how the noise source particles are sampled during the final power iteration generation. In the present Section we have subsequently outlined how those particles are transported, until they have all either leaked or been killed by roulette. Once this occurs, one noise batch has finally been completed. In order to estimate the variance for our noise field, we must obtain a collection of (ideally independent) replicas. Yamamoto's method was not able to do this, due to implementation limitations [22, 23]. Rouchon et al.'s method made use of the newly developed iteration scheme for dynamic Monte Carlo simulations in TRIPOLI-4[®] to perform independent replicas [27, 28]. After the noise batch has completed, the program will resume with power iteration. Several generations of power iteration can be performed to decorrelate the noise source between batches, before performing one generation where a new noise source is sampled. After this generation, power iteration is paused, and noise transport may be resumed again, performing a second noise batch. This can be repeated indefinitely, to generate statistics for the noise field as required. Figure 11.2 depicts this solution strategy.

11.4 . Branchless Sampling for the Noise Source

The noise source for vibrations has nearly equal positive and negative components which are located adjacent to one another, as depicted in Fig. 11.1. Transport of this source poses a number of computational challenges. From Sec. 11.2, we have seen that at each collision site within the perturbed region at least two noise source particles will be sampled: one for the copy operator, and one for the scattering operator. Additionally, should the material be fissile, it is also possible that a number of fission noise particles will be sampled. Each of these sampled noise source particles will be born with a different statistical weight; depending on $\delta\Sigma_\alpha$, some of these might be dominantly real, while others dominantly imaginary. The sign of each weight component can also vary. Therefore, at the same position, we are sampling positively and negatively-weighted source particles, which could potentially be increasing the variance of the tally for the resulting noise field. Each noise particle's contribution to the noise field will have the same sign as its weight: positive and negative contributions then are summed up to obtain the noise field tallies, leading to an increase in the associated variance. In Chapter 12, we will hypothesize that the variance of the noise field estimation may be reduced by only sampling one type of noise source particle at a collision site, using

a “branchless noise source sampling” scheme. Branchless collision sampling is a class of variance reduction methods [7]: when a particle undergoes a collision, it either undergoes a scatter, or creates a single fission particle (in a manner which is not analog). The underlying idea is that suppressing the branches of the neutron histories induced by fission events should lead to a smaller dispersion in the tallies (and in a reduction of the simulation time). While seldom used in general-purpose Monte Carlo codes, these methods have been recently shown to be useful for time-dependent (kinetic) simulations in reactor physics [27, 29]. Sjenitzer and Hoogenboom have developed a variant where, for standard neutron transport (without any complex weights), the probability of sampling reaction channel j is

$$P_j = \frac{\nu_j \Sigma_j}{\sum_k \nu_k \Sigma_k}, \quad (11.43)$$

and a weight correction factor of

$$m = \frac{\sum_k \nu_k \Sigma_k}{\sum_k \Sigma_k} \quad (11.44)$$

is always applied to the weights (regardless of the type of reaction that was sampled). We see that this approach conserves the expected production per reaction channel, as

$$m P_j = \frac{\nu_j \Sigma_j}{\Sigma_t}. \quad (11.45)$$

Unfortunately, this formulation is impossible to use with the sampling of noise source particles, as it would lead to complex-valued probabilities. In Sec. 12.3, we derive a method for performing branchless collisions with complex yields, which is summarized here.

The objective is to determine the probability P_j of sampling a noise source channel, where P_j must be a real positive quantity, with $0 \leq P_j \leq 1$. We must also determine a complex weight multiplier $m_j = m_{j,R} + im_{j,I}$, which ensures that

$$m_j P_j = \frac{(\nu \delta \Sigma)_j}{\Sigma_t}. \quad (11.46)$$

We want to minimize the average squared magnitude of the outgoing particle’s weight, which is then equivalent to minimizing

$$\sum_j (m_{j,R}^2 + m_{j,I}^2) P_j. \quad (11.47)$$

Through the method of Lagrange multipliers, we must minimize

$$\Gamma = \sum_j (m_{j,R}^2 + m_{j,I}^2) P_j - \gamma \left(\sum_j P_j - 1 \right) - \sum_j \gamma_{j,R} \left(m_{j,R} P_j - \frac{(\nu \delta \Sigma)_{j,R}}{\Sigma_t} \right) - \sum_j \gamma_{j,I} \left(m_{j,I} P_j - \frac{(\nu \delta \Sigma)_{j,I}}{\Sigma_t} \right), \quad (11.48)$$

where γ , $\gamma_{j,R}$, and $\gamma_{j,I}$ are the Lagrange multipliers associated with the constraints for our problem. Following the derivation in Sec. 12.3, we obtain

$$P_j = \frac{|(\nu \delta \Sigma)_j|}{\sum_k |(\nu \delta \Sigma)_k|}, \quad (11.49)$$

and

$$m_j = \frac{\sum_k |(\nu \delta \Sigma)_k|}{\Sigma_t} \frac{(\nu \delta \Sigma)_j}{|(\nu \delta \Sigma)_j|}. \quad (11.50)$$

From Eq. (11.50), we notice that the weight multiplier m_j now has a constant magnitude, but a different complex phase for each reaction channel j . Due to the sum of complex magnitudes in Eqs. (11.49) and (11.50), it is actually not possible to consider delayed fission as a single reaction channel. Each precursor family has a different complex yield, and must therefore be considered as an individual reaction channel.

In Sec. 12.5, we implement and test this technique in a two-group reflected assembly benchmark problem for cross section oscillations, which has been previously used to compare different noise solvers [25]. From the results presented in Sec. 12.5.1, we observe that using the branchless noise sampling method unfortunately (and quite disappointingly) does not improve the efficiency of the calculation, and may even lead to a slight degradation in efficiency. The reason for this slight degradation in performance is not yet been elucidated, and deserves further investigation. One reason might be that the weight modifier has a different complex phase for each reaction channel, so there are still positive and negative weight components being produced in the perturbation region, while we have also reduced the number of noise particles which will contribute to the generation of statistics. It might therefore be more important (at least for the benchmark problem examined in Sec. 12.5.1) to have more noise particles to generate statistics than to reduce the variance in the weight magnitude of noise source particles. Future work should be devoted to a more thorough analysis of the behavior of this sampling method; in particular, branchless noise source sampling should also be tested on vibration problems, to see if similar results are obtained.

11.5 . Weight Cancellation for Variance Reduction in Noise Problems

The presence of positive and negative weight components in the neutron noise equation leads us quite naturally to propose the weight cancellation methods examined in Part II as variance reduction technique for noise simulations. For the simulations conducted in Part II, however, weight cancellation was always used on k -eigenvalue problems, and applied between subsequent fission generations of the power iteration algorithm. Performing cancellation at fission sites is the most efficient option as its isotropic nature effectively reduces the dimensionality of the cancellation operation. The propagation of neutron noise resembles instead a fixed-source problem, and the structure of the calculation does not inherently lead to the use of fission generations, where we could immediately apply weight cancellation. Thus, the first nontrivial question that we have to address is how to incorporate weight cancellation in the fixed-source transport scheme used for neutron noise problems. In the following we briefly detail our implementation strategy; a thorough discussion is presented in Chapters 12 and 13.

11.5.1 . How to Apply Weight Cancellation to Fixed-Source Problems

In Part II, it has been outlined that applying weight cancellation to fission particles is an efficient choice for power iteration, since it allows having the entire collection of particles be present synchronously.⁴ For noise simulations, when fission noise neutrons are produced we typically do not have access to all of the fission noise particles synchronously: fission noise particles are added to a special secondary particle bank, and will be subsequently transported by the same computer node, without pausing to perform any communications with other nodes partaking in the simulation. This MPI communication scheme is depicted in Fig. 11.3. With this approach, there is never a step in the transport algorithm where all of the fission noise particles are on the same node, for weight cancellation to be performed.

To gain access to the fission noise banks, we have chosen to break each noise batch into “inner fission generations”. All of the initial noise source particles will be distributed to the computer nodes to be transported, with fission noise particles being stored in a separate bank. Once there are no more noise particles to be transported (apart from those in the fission noise banks), all of the fission noise particles can be sent to the master node. Once there, the weight cancellation method of choice can be applied to all of

⁴Cancellation is most efficient when all particles can participate. The easiest way to accomplish this is to send all particles to the master node. Alternatively, one could also use a domain decomposition approach for cancellation, where all particles within the same geometric region are sent to the same node for cancellation. Either approach is possible, though the domain decomposition version is likely more difficult to implement.

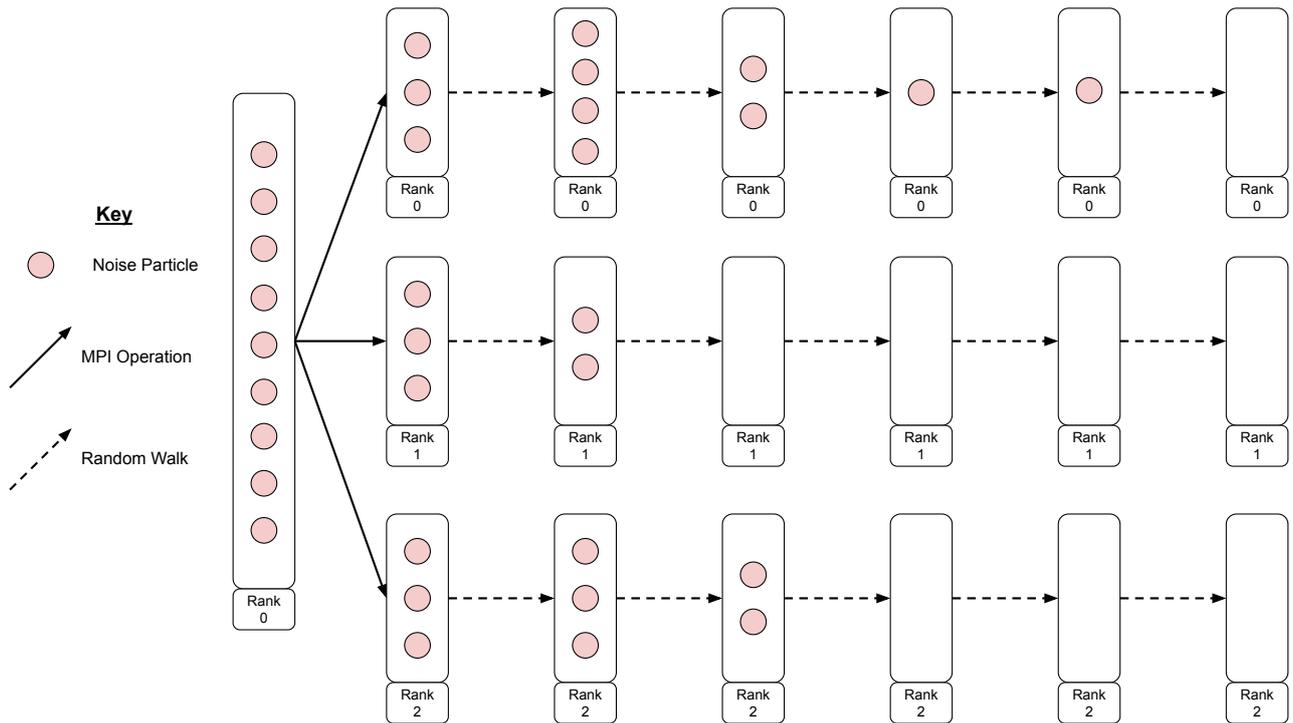


Figure 11.3: Naïve MPI communication scheme for a single noise batch where computer nodes do not communicate.

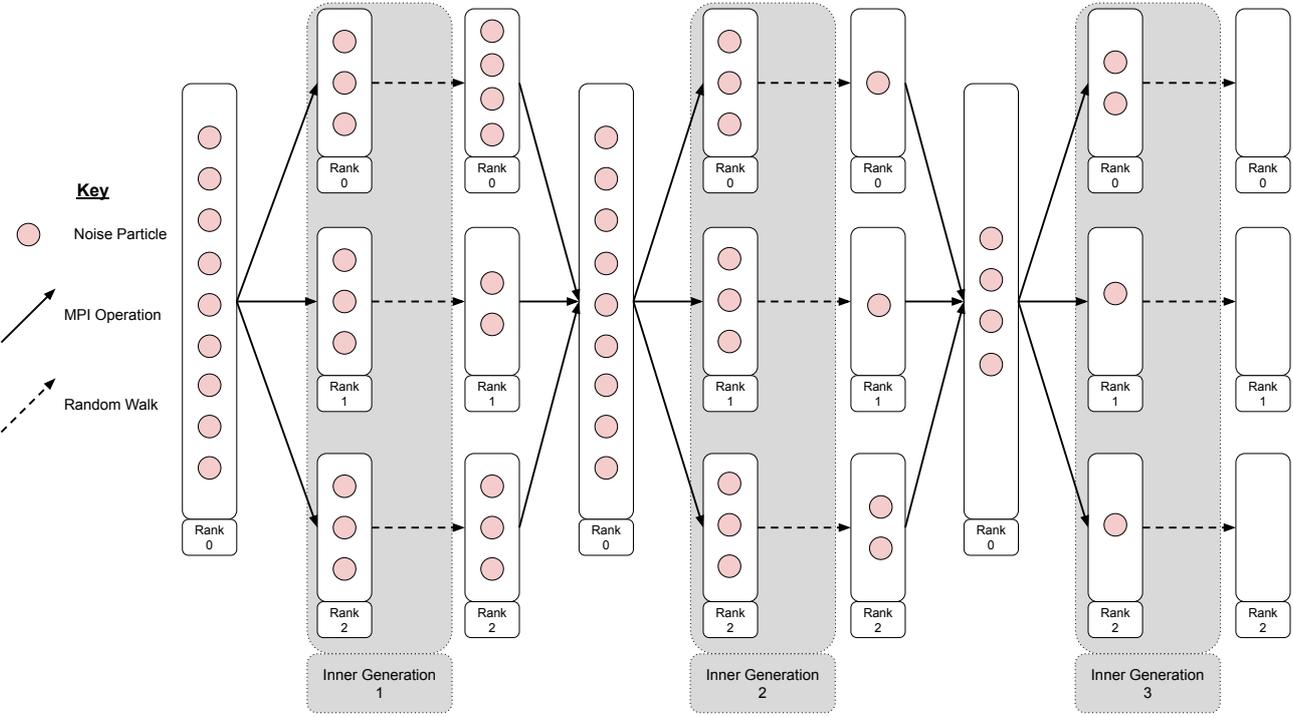


Figure 11.4: MPI communication scheme for a single noise batch which is decomposed into inner fission generations.

the fission noise particles. The particles resulting from cancellation are then redistributed to the computer nodes to be transported. Their fission progeny will again be kept in the separate fission noise bank, so that weight cancellation can be performed at the end of the second inner generation. This process will continue until eventually no fission noise particles are produced during an inner fission generation, at which point the noise batch has completed.⁵ Figure 11.4 depicts this new MPI communication scheme, with the application of inner generations inside of each noise batch. Readers familiar with similar schemes will perhaps note that this is nearly identical to the MPI communication strategy used in many Monte Carlo transport codes for power iteration problems [30].

While it is certainly possible to apply weight cancellation between every inner generation with this scheme, it is also easy to implement a variant where cancellation will only be performed for a user-defined number of inner generations at the beginning of each noise batch, and then the decomposition into inner generations options is turned off. The impact of this variant is explored in Sec. 12.5.2. One could also choose to use inner generations while not using any weight cancellation algorithm. At first, this choice would appear to be counterintuitive, as it would increase the amount of MPI communications required to perform the simulation, without even being able to gain any potential benefits in efficiency from weight cancellation. Nonetheless, while verifying the MPI implementation for this algorithm, we noticed that quite surprisingly, the simulation time can be reduced by a non negligible factor by simply breaking up a noise batch into inner generations (without applying cancellation). By redistributing fission noise particles amongst the computer nodes at each inner generation, the computational load is more evenly balanced, so nodes spend less time waiting for other nodes to finish. This question of load balancing has yet to be properly studied, and will hopefully be the subject of future research.

11.5.2 . Observed Performance Improvements

In Chapters 12 and 13 we carefully assess the impact of applying weight cancellation to neutron noise problems, the former considering cross section oscillations, and the latter considering mechanical vibrations. The observed results on the calculation efficiency are summarized in this section, based on the FOM as a metric. For our noise simulations, it is instructive to consider two different times in the FOM formula (see Eq. (2.19)): the wall-clock time spent to complete the entire simulation (power iteration to sample the noise source and noise transport), and the wall-clock time spent performing just noise transport. Applying weight cancellation to the noise particle transport will have no effect on the power iteration component of the simulation, which must always be performed to obtain the critical flux, so it is certainly a valid option to only consider the effects on the time spent transporting noise particles when looking at the FOM.

Cross Section Oscillations

The test problem for testing cross section oscillations is the same problem proposed in Ref. 25, the parameters of which are summarized in Sec. 12.5. It is a two-dimensional two-group reflected 17×17 fuel assembly, with square profile fuel pins. It is assumed that one of these fuel pins exhibits a cross section oscillation. In Sec. 12.5.2 we show that the average improvement in the FOM, when comparing inner generations with approximate weight cancellation to inner generations without cancellation, is approximately a factor of 24 – 29, when considering the total run time. If one instead considers the time only spent transporting noise particles, this factor raises then to 46 – 54.

If we compare the performance of using inner generations with approximate cancellation to no inner generations (and no cancellation by consequence), these factors are even larger. When using the total run time, the FOM is improved by a factor of 35 – 40; using the time spent transporting noise particles, this is actually an improvement of 88 – 100 in the FOM. From these values, it is evident that a large improvement in the FOM can be achieved by simply decomposing a noise batch into inner generations and performing load balancing.

⁵Currently, we take for granted the fact that a noise batch will finish. There are difficulties in convergence that arise for frequencies which lie outside of the plateau regime [20, 24].

		Real		Imaginary	
		Group 1	Group 2	Group 1	Group 2
Approximate Coarse	Total Run Time	304	304	234	235
	Noise Run Time	1623	1624	1252	1254
Approximate Fine	Total Run Time	129	129	122	122
	Noise Run Time	487	487	461	461
Exact	Total Run Time	8	8	8	8
	Noise Run Time	14	14	14	14

Table 11.1: FOM gain factors for the calculation of the first harmonic, for given cancellation methods when compared to the baseline solution strategy which does not use weight cancellation or inner generations.

Mechanical Vibrations

In Chapter 13 we consider weight cancellation for the case of mechanical vibrations. A variation to the same benchmark used for oscillations is considered, where the fuel pin is now vibrating periodically, at an angular frequency of $2\pi \text{ rad s}^{-1}$, with an amplitude of 0.2 cm. Three different cancellation methods are applied to this problem in Sec. 13.6.2: approximate cancellation on a “coarse” 170×170 mesh, approximate cancellation on a “fine” 340×340 mesh, and exact cancellation on a 170×170 mesh. The resulting average improvements in the FOM, when compared to not using inner generations or weight cancellation, are provided in Table 11.1. Considering the noise transport time, using approximate cancellation with the coarse mesh, an improvement by more than a factor of 1200 is observed in the FOM, for both the real and imaginary component. Using approximate cancellation with the finer mesh will reduce the efficiency of cancellation, but lead to large improvements in the FOM which are still factors of 487 for the real component, and 461 for the imaginary component. Unfortunately, exact regional cancellation is much less efficient, and only yields an improvement by a factor of 14 in the FOM.

As is depicted in Sec. 13.6.2, the estimation for the resulting noise field is essentially unusable in the absence of weight cancellation techniques, because the statistical uncertainty is overwhelmingly large. The application of weight cancellation reduces the variance in the estimated noise field to the point that it becomes feasible to perform sensitivity analysis in Monte Carlo neutron noise simulations, i.e. investigate the effects of small changes in the physical parameters on the resulting neutron noise field. For this purpose, in Sec. 13.6.3 we compute the noise field for the same benchmark geometry, but with a slightly larger vibration amplitude, or with a slightly higher frequency. Thanks to the application of weight cancellation, we are able to observe the small changes to the neutron noise field, and examine that changing the vibration amplitude and changing the vibration frequency does result in slightly different shapes of the noise amplitude and phase.

References

- [1] T. E. Booth, “Computing the Higher k-Eigenfunctions by Monte Carlo Power Iteration: A Conjecture,” *Nuclear Science and Engineering*, vol. 143, no. 3, p. 291–300, 2003.
- [2] T. E. Booth and J. E. Gubernatis, “Exact Regional Monte Carlo Weight Cancellation for Second Eigenfunction Calculations,” *Nuclear Science and Engineering*, vol. 165, no. 3, p. 283–291, 2010.
- [3] T. Yamamoto, “Convergence of the second eigenfunction in Monte Carlo power iteration,” *Annals of Nuclear Energy*, vol. 36, no. 1, p. 7–14, 2009.
- [4] P. Zhang, H. Lee, and D. Lee, “A general solution strategy of modified power method for higher mode solutions,” *Journal of Computational Physics*, vol. 305, p. 387–402, 2016.

- [5] P. Zhang, H. Lee, M. Lemaire, C. Kong, J. Choe, J. Yu, F. Khoshahval, and D. Lee, "Practical Monte Carlo simulation using modified power method with preconditioning," *Annals of Nuclear Energy*, vol. 127, p. 372–384, 2019.
- [6] G. Bell and S. Glasstone, *Nuclear Reactor Theory*. Van Nostrand Reinhold Company, 1970.
- [7] I. Lux and L. Koblinger, *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations*. CRC Press, 1991.
- [8] S. Carney, F. Brown, B. Kiedrowski, and W. Martin, "Theory and applications of the fission matrix method for continuous-energy Monte Carlo," *Annals of Nuclear Energy*, vol. 73, p. 423–431, 2014.
- [9] T. Yamamoto, "Monte Carlo algorithm for buckling search and neutron leakage-corrected calculations," *Annals of Nuclear Energy*, vol. 47, p. 14–20, 2012.
- [10] —, "Monte Carlo method with complex weights for neutron leakage-corrected calculations and anisotropic diffusion coefficient generations," *Annals of Nuclear Energy*, vol. 50, p. 141–149, 2012.
- [11] A. M. Weinberg and H. C. Schweinler, "Theory of Oscillating Absorber in a Chain Reactor," *Physical Review*, vol. 74, no. 8, p. 851–863, 1948.
- [12] M. M. R. Williams, *Random Processes in Nuclear Reactors*. Pergamon Press, Oxford, 1974.
- [13] J. A. Thie, *Power Reactor Noise*. American Nuclear Society, 1981.
- [14] I. Pázsit and C. Demazière, "Noise techniques in nuclear systems," in *Handbook of nuclear engineering*, D. Cacuci, Ed. Springer Verlag, 2010, ch. 14, pp. 1629–1737.
- [15] D. N. Fry, "Experience in Reactor Malfunction Diagnosis Using On-Line Noise Analysis," *Nuclear Technology*, vol. 10, no. 3, p. 273–282, 1971.
- [16] M. Antonopoulos-Domis, "Reactivity and neutron density noise excited by random rod vibration," *Annals of Nuclear Energy*, vol. 3, no. 9–10, p. 451–459, 1976.
- [17] M. Antonopoulos-Domis and A. Colebrooke, "Neutron fluctuations induced by absorber vibrations: the QMC reactor experiment—I," *Annals of Nuclear Energy*, vol. 6, no. 11–12, p. 579–584, 1979.
- [18] I. Pázsit and O. Glöckler, "On the Neutron Noise Diagnostics of Pressurized Water Reactor Control Rod Vibrations. I. Periodic Vibrations," *Nuclear Science and Engineering*, vol. 85, no. 2, p. 167–177, 1983.
- [19] M. Seidl, K. Kosowski, U. Schüler, and L. Belblidia, "Review of the historic neutron noise behavior in German KWU built PWRs," *Progress in Nuclear Energy*, vol. 85, p. 668–675, 2015.
- [20] A. Rouchon, "Analyse et développement d'outils numériques déterministes et stochastiques résolvant les équations du bruit neutronique et applications aux réacteurs thermiques et rapides," Ph.D. dissertation, Université Paris-Saclay, 2016.
- [21] C. Demazière, P. Vinai, M. Hursin, S. Kollias, and J. Herb, "Overview of the CORTEX project," in *Proceedings of Physor2018*, Cancun, Mexico, 2018.
- [22] T. Yamamoto, "Monte Carlo method with complex-valued weights for frequency domain analyses of neutron noise," *Annals of Nuclear Energy*, vol. 58, p. 72–79, 2013.
- [23] —, "Implementation of a frequency-domain neutron noise analysis method in a production-level continuous energy Monte Carlo code: Verification and application in a BWR," *Annals of Nuclear Energy*, vol. 115, p. 494–501, 2018.

- [24] A. Rouchon, A. Zoia, and R. Sanchez, "A new Monte Carlo method for neutron noise calculations in the frequency domain," *Annals of Nuclear Energy*, vol. 102, p. 465–475, 2017.
- [25] P. Vinai, H. Yi, A. Mylonakis, C. Demazière, B. Gasse, A. Rouchon, A. Zoia, A. Vidal-Ferràndiz, D. Ginestar, G. Verdú, and T. Yamamoto, "Comparison of Neutron Noise Solvers Based on Numerical Benchmarks in a 2-D Simplified UOX Fuel Assembly," in *M&C 2021*, 2021.
- [26] A. Zoia, A. Rouchon, B. Gasse, C. Demazière, and P. Vinai, "Analysis of the neutron noise induced by fuel assembly vibrations," *Annals of Nuclear Energy*, vol. 154, p. 108061, 2021.
- [27] M. Faucher, D. Mancusi, and A. Zoia, "New kinetic simulation capabilities for Tripoli-4[®]: Methods and applications," *Annals of Nuclear Energy*, vol. 120, p. 74–88, 2018.
- [28] A. Rouchon, W. Jarrah, and A. Zoia, "The new neutron noise solver of the Monte Carlo code TRIPOLI-4," in *Proceedings of M&C2019*, Portland USA, 2019.
- [29] B. L. Sjenitzer and J. E. Hoogenboom, "Dynamic Monte Carlo Method for Nuclear Reactor Kinetics Calculations," *Nuclear Science and Engineering*, vol. 175, no. 1, p. 94–107, 2013.
- [30] T. Goorley, M. James, T. Booth, F. Brown, J. Bull, L. J. Cox, J. Durkee, J. Elson, M. Fensin, R. A. Forster, J. Hendricks, H. G. Hughes, R. Johns, B. Kiedrowski, R. Martz, S. Mashnik, G. McKinney, D. Pelowitz, R. Prael, J. Sweezy, L. Waters, T. Wilcox, and T. Zukaitis, "Initial MCNP6 Release Overview," *Nuclear Technology*, vol. 180, no. 3, p. 298–315, 2012.

12 - Variance Reduction Techniques for Monte Carlo Neutron Noise Simulations

This chapter has previously appeared as:

H. Belanger, D. Mancusi, and A. Zoia, "Variance Reduction Techniques for Monte Carlo Neutron Noise Simulations," In *Proceedings of the International Conference on Physics of Reactors 2022 (PHYSOR 2022)*, May 2022, Pittsburgh, PA, p. 544-553.

Variance Reduction Techniques for Monte Carlo Neutron Noise Simulations

Hunter Belanger, Davide Mancusi, and Andrea Zoia

Université Paris-Saclay, CEA, Service d'Études des Réacteurs et de Mathématiques Appliquées,
91191, Gif-sur-Yvette, France
hunter.belanger@cea.fr, davide.mancusi@cea.fr, andrea.zoia@cea.fr

ABSTRACT

Advanced Monte Carlo methods have been recently proposed in order to solve the neutron noise equations in the frequency domain, which allows establishing reference solutions to validate faster but approximate deterministic solvers. Due to the presence of particles carrying two statistical weights (for the real and imaginary components of the noise field), both of which may be positive or negative, Monte Carlo simulations of neutron noise pose distinct challenges in terms of variance reduction. In this work we investigate two variance-reduction techniques, namely branchless collisions and weight cancellation, and probe their effectiveness for a benchmark configuration concerning the noise field induced by a pin with oscillating cross sections in a fuel assembly.

KEYWORDS: Neutron noise, Monte Carlo, Weight cancellation, Branchless

1. INTRODUCTION

The small fluctuations of the neutron flux caused by small perturbations of the macroscopic cross sections take the name of neutron noise. The physical origin of this phenomenon is related to variations introduced in the core, such as vibrations of fuel or control rods. Having tools to aid in analyzing and locating the source of such fluctuations would allow taming them without needing to shutdown the reactor or forcing it to operate at lower power. Neutron noise has been of great interest as of late, with the recent CORTEX H2020 project [1]. Current neutron noise analysis tools are mostly based on state-of-the-art numerical codes, accounting for arbitrary cross section variations in the time or frequency domain [2–8]. Recently, advanced deterministic [9, 10] and Monte Carlo solvers [3, 4] have been developed to perform neutron noise analysis in the frequency domain. These methods allow one to address realistic applications at the scale of fuel assemblies or full cores, with unprecedented accuracy [2, 9, 11–13]. Each simulation code has distinct advantages and disadvantages, due to implementation choices and implicit assumptions.

In this paper, we focus on the Monte Carlo method for noise analysis in the frequency domain. Such a tool is absolutely necessary, as Monte Carlo is the “gold standard” in reactor physics, as it requires very few (if any) approximations, and no discretization in space or energy. However, performing noise calculations with the Monte Carlo method poses particular challenges, due to the fact that the neutron noise is a complex quantity and thus noise particles must carry two statistical weights (for the real and imaginary components), both of which may be positive or negative [3, 4]. The use of negative weights in Monte Carlo codes is known to cause problems with convergence, or lead to large variances [14, 15], and the neutron noise problem is no exception [3, 4]. In order to tackle these issues, Yamamoto implemented a weight cancellation method that was applied to noise particles at fission events [3]. Unfortunately, the analysis of the necessity of weight cancellation was not attempted, and no comparisons were made with calculations that did not use cancellation. Additionally, implementation constraints inhibited the estimation of the variance of scores, making it impossible to quantitatively assess the performance effects of weight cancellation. Weight cancellation was abandoned in a later implementation, as it was felt that adding

such a technique to a production-level code would make it less versatile [11]. On the other hand, the methods implemented in TRIPOLI-4[®] by Rouchon et al. for the study of neutron noise did not use weight cancellation [4].

The question as to the effects and necessity of weight cancellation in Monte Carlo methods for neutron noise remains unanswered. Recently, we have shown that weight cancellation is necessary when performing power iteration with negative weights and certain transport methods [15], and new algorithms for exact regional cancellation in three dimensions were proposed [15, 16]. The goal of the present paper is to explore the potential of using weight cancellation as a variance reduction method in neutron noise problems. Additionally, we consider the use of branchless collision methods, where a single particle is forced out of each collision event [17], to improve the sampling of the neutron noise source.

2. MONTE CARLO SIMULATIONS OF NEUTRON NOISE

Noise theory is based on the assumption that the material cross sections have time-dependent perturbations around their stationary value, namely, $\Sigma_r(\mathbf{r}, E, t) = \Sigma_r(\mathbf{r}, E) + \delta\Sigma_r(\mathbf{r}, E, t)$. Plugging these expressions in the time-dependent Boltzmann equation, linearizing and moving to the frequency domain by taking the Fourier transform yields the noise equation

$$\mathcal{B}(\omega)\delta\varphi(\omega) = -\delta\mathcal{B}(\omega)\varphi_c(\mathbf{r}, E, \boldsymbol{\Omega}) \quad (1)$$

for the frequency-domain neutron noise $\delta\varphi(\omega) = \delta\varphi(\mathbf{r}, E, \boldsymbol{\Omega}, \omega)$, where ω denotes the angular frequency. Here we have introduced the Fourier-transformed Boltzmann operator

$$\begin{aligned} \mathcal{B}(\omega) = & i\frac{\omega}{v} + \Sigma_t(\mathbf{r}, E) + \boldsymbol{\Omega} \cdot \nabla - \iint f_s(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}, E' \rightarrow E)\Sigma_s(\mathbf{r}, E')dE' d\boldsymbol{\Omega}' \\ & - \frac{\chi_p(E)}{4\pi k_{\text{eff}}} \iint \nu_p(E')\Sigma_f(\mathbf{r}, E')dE' d\boldsymbol{\Omega}' - \sum_j \frac{\lambda_j}{\lambda_j + i\omega} \frac{\chi_d^j(E)}{4\pi k_{\text{eff}}} \iint \nu_d^j(E')\Sigma_f(\mathbf{r}, E')dE' d\boldsymbol{\Omega}', \quad (2) \end{aligned}$$

and $\varphi_c(\mathbf{r}, E, \boldsymbol{\Omega})$ is the critical flux satisfying the k -eigenvalue Boltzmann equation (having associated fundamental eigenvalue k_{eff}). We have also defined the Fourier-domain perturbation operator

$$\begin{aligned} \delta\mathcal{B}(\omega) = & \delta\Sigma_t(\mathbf{r}, E, \omega) - \iint f_s(\boldsymbol{\Omega}' \cdot \boldsymbol{\Omega}, E' \rightarrow E)\delta\Sigma_s(\mathbf{r}, E', \omega)dE' d\boldsymbol{\Omega}' \\ & - \frac{\chi_p(E)}{4\pi k_{\text{eff}}} \iint \nu_p(E')\delta\Sigma_f(\mathbf{r}, E', \omega)dE' d\boldsymbol{\Omega}' - \sum_j \frac{\lambda_j}{\lambda_j + i\omega} \frac{\chi_d^j(E)}{4\pi k_{\text{eff}}} \iint \nu_d^j(E')\delta\Sigma_f(\mathbf{r}, E', \omega)dE' d\boldsymbol{\Omega}', \quad (3) \end{aligned}$$

where $\delta\Sigma_r(\omega) = \delta\Sigma_r(\mathbf{r}, E, \omega)$ are the Fourier-transformed cross section perturbations.

In order to estimate the noise field $\delta\varphi(\omega)$ that satisfies Eq. (1), in this work we have written an independent implementation of the Monte Carlo sampling strategy used by TRIPOLI-4[®], which has been outlined by Rouchon et al. [4]. This implementation has been added to the MGMC multi-group Monte Carlo code, which is freely available and distributed under the CeCILL-v2.1 open-source license [18]. The first step consists of determining the critical flux φ_c , achieved by running a regular power iteration with a sufficient number of inactive generations, starting from an arbitrary neutron distribution. Once the neutron population is distributed according to φ_c , we sample the noise source $-\delta\mathcal{B}(\omega)\varphi_c$. The operator $-\delta\mathcal{B}(\omega)$ must be prepared before running the simulation: it contains the Fourier-transformed perturbed cross sections $\delta\Sigma_r(\omega)$, which are problem-dependent. Since $\delta\Sigma_r(\omega)$ are in general complex quantities, the noise source will be also complex: the sampled noise particles carry thus complex statistical weights, where the signs of the real and imaginary parts of the weights can be positive or negative [4, 13]. The noise source population must

then be propagated according to the operator $\mathcal{B}(\omega)$, in order to determine the noise field $\delta\varphi(\omega)$. Due to the close resemblance between $\mathcal{B}(\omega)$ and the regular Boltzmann operator, it is possible to solve the transport problem associated with $\mathcal{B}(\omega)$ by a set of stochastic rules obtained by suitably modifying those associated to the regular Boltzmann operator [4, 13]. Finally, M independent noise replicas (essentially fixed-source batches) are performed, ensuring that error bars on the noise field $\delta\varphi(\omega)$ can be estimated.

In Ref. 3, the complex cross section $\Sigma_t + i\omega/v$ on the left-hand-side of Eq. (2) is taken into account explicitly by modifying the particle weights during flights. As outlined by Rouchon et al., we choose instead to modify the collision kernel [4]: we add a term $(\eta\omega/v)\delta\varphi$, η being a real positive constant, to both sides of Eq. (2), and we move the term $(i\omega/v)\delta\varphi$ to the right-hand-side. Then, the noise equation becomes:

$$\left(\boldsymbol{\Omega} \cdot \nabla + \Sigma_t(\mathbf{r}, E) + \eta \frac{\omega}{v}\right) \delta\varphi(\mathbf{r}, E, \boldsymbol{\Omega}, \omega) = \frac{\eta - i}{\eta} \eta \frac{\omega}{v} \delta\varphi(\mathbf{r}, E, \boldsymbol{\Omega}, \omega) + \left(\dots\right). \quad (4)$$

In this case, we have a real modified total cross section $\tilde{\Sigma}_t(\mathbf{r}, E, \omega) = \Sigma_t(\mathbf{r}, E) + \eta\omega/v > 0$. Hence, flight lengths are sampled as in standard Monte Carlo calculations, provided that $\tilde{\Sigma}_t$ is used instead of Σ_t . Because of the structure of Eq. (4), the collision operator is now different from that of the regular Boltzmann equation, and we have to treat a new type of particle production associated to the special copy operator appearing at the right hand side of Eq. (4). More details about the methodology may be found in Refs. 9 and 4. Due to the presence of complex yields, the copy and delayed fission operators lead to a coupling between the real and imaginary components of $\delta\varphi(\omega)$. Implicit capture (with forced fission) and Russian roulette can be used as is customary. Similarly as in Ref. 3, Russian roulette is applied separately to the absolute value of the real and imaginary parts of the particle weight: the particle is killed only if the real and the imaginary parts are both killed.

Contrary to the original TRIPOLI-4[®] implementation, where noise particles (and their descendants) were followed from birth to death, in this work we break each independent noise replica into inner generations, where the noise particles generated at fission events are stored in a bank until the other particles have died. By virtue of this strategy, the stored fission noise particles may thus undergo cancellation before being transported (in turn also storing their fission daughters in a bank). This is very similar to the method proposed by Yamamoto [3].

3. BRANCHLESS NOISE SOURCE SAMPLING

When sampling the neutron noise source $-\delta\mathcal{B}(\omega)\varphi_c$ during power iteration, every time a particle has a collision at a location where $\delta\Sigma_t(\mathbf{r}, E, \omega) \neq 0$, noise particles can be produced at that location through several different channels. In the strategy used by TRIPOLI-4[®], each channel is forced, and the resulting particle is multiplied by a corrective weight factor so that there is no bias [4]. The first channel is a copy of the incident particle, which arises from the $\delta\Sigma_t$ operator in Eq. (3). The second is the scattering operator when $\delta\Sigma_s(\mathbf{r}, E, \omega) \neq 0$. In addition to these two, should the perturbation occur in a fissile medium, then fission noise particles related to $\delta\Sigma_f(\mathbf{r}, E, \omega)$ may be produced as well. Since the copy, scattering and fission contributions are in general complex, producing multiple noise particles at every collision site means that there are several particles at the same location, some with positive weight components, and some with negative weight components. As this could potentially lead to large increases in the variance, we have investigated the possibility of sampling just one noise particle at each collision site, using branchless collision methods. Branchless methods in Monte Carlo simulation are well known [17], and have been used in many applications, such as for reactor kinetics [19–22]. In branchless schemes, regardless of the reaction channel selected at the collision, only one particle leaves the collision site and the particle weight is adjusted accordingly. While using branchless collisions precludes the possibility of using forced fission, it still may provide variance reduction benefits in certain applications [19–22]. For standard branchless collisions, the probability of selecting reaction channel j , which has a cross section Σ_j and a yield of ν_j , is

given as

$$P_j = \frac{\nu_j \Sigma_j}{\sum_i \nu_i \Sigma_i}. \quad (5)$$

After the event has been sampled, the particle weight w' becomes

$$w' = \frac{\sum_i \nu_i \Sigma_i}{\sum_i \Sigma_i} w. \quad (6)$$

For this to be unbiased, all yields and cross sections must be non-negative.

The utility of this method is that only one particle is ever emitted from a collision (even in the case of fission) and that the magnitude of the weight of the outgoing particle is independent of the selected channel. As noise particles may have positive and negative real and imaginary weight components, it is possible that producing only one source particle at a collision could reduce the variance in the noise field induced by the source, as there could be fewer positive and negative components being added at the same source location. The traditional formulation of the branchless collision, however, must be modified to sample the noise source, as in this case the yields and cross sections are complex. We must determine the probability P_j of sampling reaction j , which must of course be a positive real number. We also search for a complex weight multiplier $m_j = m_{j,R} + im_{j,I}$. We must also require that

$$P_j m_{j,R} = \frac{(\nu \delta \Sigma)_{j,R}}{\Sigma_t} \quad \text{and} \quad P_j m_{j,I} = \frac{(\nu \delta \Sigma)_{j,I}}{\Sigma_t} \quad (7)$$

to remain unbiased. The complex quantity $(\nu \delta \Sigma)_j$ is unique for each channel, and can be determined from examination of Eq. (2). As an example, for the case of scattering $(\nu \delta \Sigma)_s = \delta \Sigma_s$. In the case of delayed fission (with one precursor family), $(\nu \delta \Sigma)_{f,d} = \nu_d \frac{\lambda}{\lambda + i\omega} \delta \Sigma_f$. From these conditions, we may use the method of Lagrange multipliers to find the choice of $m_{j,R}$, $m_{j,I}$, and P_j which minimize the average magnitude squared of the outgoing particle's weight $\sum_j (m_{j,R}^2 + m_{j,I}^2) P_j$. The quantity to minimize is thus

$$\Gamma = \sum_j (m_{j,R}^2 + m_{j,I}^2) P_j - \gamma \left(\sum_j P_j - 1 \right) - \sum_i \gamma_{j,R} \left(P_j m_{j,R} - \frac{(\nu \delta \Sigma)_{j,R}}{\Sigma_t} \right) - \sum_j \gamma_{j,I} \left(P_j m_{j,I} - \frac{(\nu \delta \Sigma)_{j,I}}{\Sigma_t} \right), \quad (8)$$

where γ , $\gamma_{j,R}$, and $\gamma_{j,I}$ are the real valued Lagrange multipliers. Looking at a specific channel j , we may proceed with

$$\frac{\partial \Gamma}{\partial m_{j,R}} = 2m_{j,R} P_j - \gamma_{j,R} P_j = 0 \quad \Rightarrow \quad \gamma_{j,R} = 2m_{j,R} \quad \text{and, by symmetry, } \gamma_{j,I} = 2m_{j,I} \quad (9)$$

$$\frac{\partial \Gamma}{\partial P_j} = m_{j,R}^2 + m_{j,I}^2 - \gamma - \gamma_{j,R} m_{j,R} - \gamma_{j,I} m_{j,I} = 0 \quad \Rightarrow \quad \gamma = -|m_j|^2 \quad \Rightarrow \quad m_j = m e^{i\varphi_j}, \quad (10)$$

where φ_j is real. From these equations, we are able to see that the probability P_j of selecting reaction channel j , and its complex weight multiplier m_j are respectively

$$P_j = \frac{|(\nu \delta \Sigma)_j|}{\sum_j |(\nu \delta \Sigma)_j|}, \quad m_j = \frac{\sum_j |(\nu \delta \Sigma)_j|}{\underbrace{\Sigma_t}_m} \frac{(\nu \delta \Sigma)_j}{|(\nu \delta \Sigma)_j|}. \quad (11)$$

Regardless of the reaction channel chosen, the magnitude of the complex weight multiplier is always the same, and only the complex phase φ_j changes with the reaction channel. This approach allows us to produce only one noise particle at each collision site when sampling the noise source.

4. WEIGHT CANCELLATION

For the preliminary examination of the effects of cancellation examined in this work, we have chosen to use an approximate cancellation technique, with a mesh imposed on top of the geometry [23]. In between each inner generation of a noise replica, the fission particles that have been generated and placed in the bank are sorted into the appropriate bin in the mesh. Within each bin, the average weight of the real component, and the average weight of the imaginary component are calculated. The real and imaginary weight portion of every particle in the bin is then set equal to the average value estimated for the bin. The smaller the bins, the better this approximation becomes, but also the less efficient, as there will be fewer particles per bin.

At the beginning of a noise replica, all of the noise particles are localized within the regions where $\delta\mathcal{B}(\omega)$ is non-vanishing, and they will then propagate throughout the rest of the geometry during the noise transport simulation. Because of this behavior, it is reasonable to conjecture that it is more efficient to perform weight cancellation early on in the noise propagation, before the particles are able to move very far away from the noise source. At the early stages of noise transport (i.e., during the first few inner fission generations), positive and negative noise particles are more likely to be close to each other spatially, and will therefore lead to more particles per bin, and more weight cancellation. We therefore test two different applications of weight cancellation: one where cancellation is performed after every inner generation of the noise transport, and another where cancellation is performed only after each of the first 10 inner generations, and then no cancellation is performed after that.

5. SIMULATION RESULTS

All tests were performed on a two-group, two-dimensional fuel assembly problem, which was presented as *exercise 2* in the benchmark proposed in Ref. 24. One fuel pin in a 17x17 UOX assembly is perturbed by an oscillation of the cross sections. A depiction of this system is provided in Fig. 1a. The perturbed cross sections are defined as

$$\Sigma_{t,g}(t) = \Sigma_{t,g}[1 + 0.041 \cos(\omega_0 t)], \quad (12)$$

$$\Sigma_{s,g \rightarrow g'}(t) = \Sigma_{s,g \rightarrow g'}[1 + 0.034 \cos(\omega_0 t)], \quad (13)$$

$$\Sigma_{f,g}(t) = \Sigma_{f,g}[1 + 0.021 \cos(\omega_0 t)], \quad (14)$$

with $\omega_0 = 2\pi Hz$. All simulations were performed with 10^6 particles per power-iteration generation, and 1000 noise replicas were scored. Using the entropy metric, it was determined that only 10 power-iteration generations needed be discarded to allow for convergence of the critical source. After each noise replica, three power-iteration generations were performed, to reduce correlations between the sampled noise sources (for more details, see [13]); noise sources were only sampled in the third power-iteration generation between noise replicas. The weight magnitude cutoff for roulette was $|w| = 0.8$, with a survival weight of $|w| = 1$ (respecting the initial sign of the particle). When cancellation was used, it was performed on a 68x68 mesh, cutting each fuel pin into four equal parts. While this is a somewhat coarse mesh, no bias was observed when compared to results obtained from APOLLO3[®] [24]. An example of the resulting noise field $\delta\varphi(\omega)$ is provided in Fig. 1b.

The efficiency of different variance reduction methods has been examined by use of the Figure of Merit $FOM = 1/(\sigma^2 T)$, where σ is the estimated error of the score, and T is the wall-clock time required to obtain that error. The time used when calculating the FOM can be either taken as the total runtime of the simulation, or the time that the code spent transporting noise particles (i.e., after the noise source has been sampled). The latter option stems from the power-iteration portion of the code being always present, and rather independent of the variance reduction techniques which are applied to noise transport. We have chosen to provide both values here, as we think both are of interest. The FOM was calculated at all points within the 170x170 regular mesh used to tally the real and imaginary portions of the noise field, each having its own FOM.

Figure 1: Depictions of the test system, and the resulting noise field.

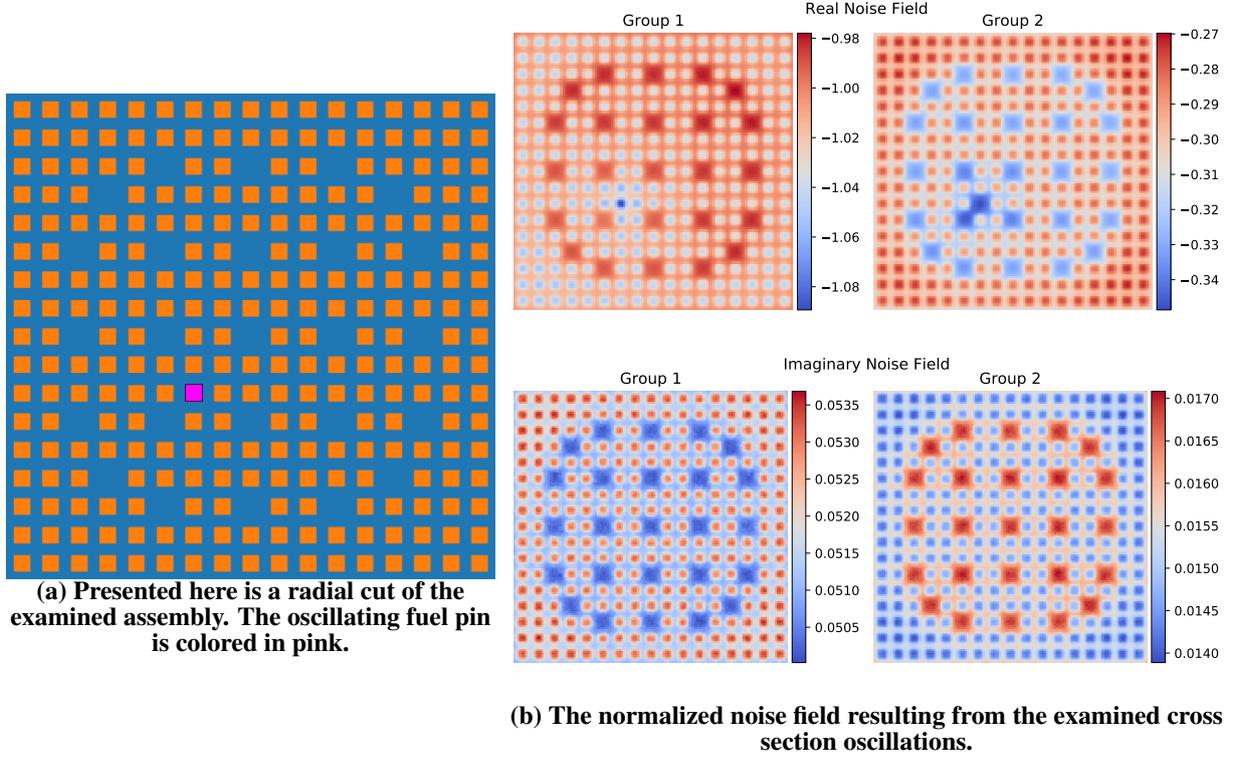


Table I: Average ratio of the FOMs between branchless noise source sampling, and standard noise source sampling. Values larger than 1 indicate that branchless sampling was more efficient than standard sampling.

(a) Average real noise field FOM ratios.			(b) Average imaginary noise field FOM ratios.		
	Group 1	Group 2		Group 1	Group 2
Runtime FOM	0.89	0.89	Runtime FOM	0.68	0.68
Noise FOM	1.06	1.06	Noise FOM	0.80	0.80

5.1. Effects of Branchless Noise Source Sampling on the FOM

The comparison of FOMs between using the branchless noise sampling method and using the standard noise sampling method is summarized in Table I. These tests *did not* use any weight cancellation. Looking at only the real portion for the noise field in Table Ia, there does not appear to be a large difference in performance. When using the total runtime of the simulation, the branchless sampling is slightly less efficient, while using the noise time indicates that the branchless method is not significantly different from the standard sampling scheme.

Examining the imaginary portion in Table Ib, both the runtime and noise FOMs demonstrate a slight decrease in performance when using branchless sampling. However, the observed differences in the FOM between branchless and standard sampling are not overwhelming. There is always less than a factor of 2 between the two methods, so it is unreasonable to state that the standard sampling technique is truly advantageous over the branchless method.

The exact reason as to the slight decrease in performance with branchless sampling is not entirely understood. Fewer noise particles need to be transported in this method, leading to a speed increase, and it was hoped that creating only one particle at a point could also reduce the variance in scores. Nonetheless, these effects may not be large enough, as the physical parameters for the examined benchmark exercise allow very little coupling between the real and imaginary noise field. The only way to sample an imaginary noise source particle is through sampling a delayed fission, which is rare; it is also rare for real noise particles to become imaginary (or gain an imaginary component). Generating imaginary noise particles in the standard sampling scheme was already a rare event, and the branchless method leads to this occurring even less frequently. It might therefore be more important to sample more imaginary noise particles that can go generate statistics, than it is to try and reduce the variance of the imaginary noise source.

5.2. Effects of Weight Cancellation on the FOM

The effects of cancellation on the FOM are much more pronounced than those of branchless sampling. In Table II, all calculations used the standard noise sampling method. Weight cancellation was performed between every inner generation of the noise replicas. For the real noise field, an increase in performance by a factor of 23 was observed when using the runtime FOM; this factor increases to 54 when using the noise FOM. The results are even better when looking at the imaginary noise field, where increases by factors of 29, and 66 were observed in the runtime FOM and noise FOM respectively. Such improvements are quite astonishing, and far from being trivial.

The reason for such large increases in performance is two-fold. While the simulation without cancellation took approximately 14.5 hours to complete, the simulation which performed cancellation between each inner noise generation took only 5.3 hours to complete. At first glance, this result is quite counter-intuitive. Performing weight cancellation between every inner noise generation requires performing many more computations, and also requires time to be spent performing MPI operations, as all particles must first be sent to the same node before cancellation can be performed. After cancellation, particles must then be redistributed back to all worker nodes. Despite having to perform these additional operations, less time is required as weight cancellation reduces the total weight $\sum_j |w_j|$ of the system, allowing particle histories to die much faster than if cancellation were not used. In addition to shortening particle histories, weight cancellation also reduces the variance in the scores, as instead of scoring many positive and negative contributions, the contributions are closer in value to what the average should be. Such large improvements are made possible through the combination of these two factors. The increase in the FOM for the imaginary portion is likely larger than the gain for the real portion, due to the fact that the complex magnitude of this system is heavily dominated by the real noise field, and the average value of the imaginary field has a much lower magnitude, meaning it is more likely to have positive and negative score contributions in the imaginary field.

Table II: Average ratio of the FOMs between performing weight cancellation between all inner generations, and not performing weight cancellation. Values larger than 1 indicate cancellation was more efficient than no cancellation.

(a) Average real noise field FOM ratios.

	Group 1	Group 2
Runtime FOM	23.64	23.64
Noise FOM	54.40	54.41

(b) Average imaginary noise field FOM ratios.

	Group 1	Group 2
Runtime FOM	28.60	28.61
Noise FOM	65.83	65.83

Table III: Average ratio of the FOMs between performing weight cancellation between the first 10 inner generations, and not performing weight cancellation.

(a) Average real noise field FOM ratios.

	Group 1	Group 2
Runtime FOM	25.44	25.46
Noise FOM	53.81	53.85

(b) Average imaginary noise field FOM ratios.

	Group 1	Group 2
Runtime FOM	21.95	21.95
Noise FOM	46.43	46.41

Given the increases in performance occurred while performing cancellation between each inner noise generation, it was hoped that even better increases in FOM might be observed when only using cancellation between the first 10 generations. Table III shows that this is not the case, however. While there were still drastic improvements compared to not using cancellation at all, the gains for the imaginary noise field reported in Table IIIb were slightly lower, with observed factors of 21 for the runtime FOM and 46 for the noise FOM. The FOMs for the real noise field were more similar to the results obtained when performing cancellation between all inner generations: a slight increase was observed in the runtime FOM, while a slight decrease was observed in the noise FOM. These results are somewhat discouraging: it was hoped that reducing the number of MPI operations would decrease the wall-clock time, while still reducing the variance in scores with the weight cancellation being performed when it is most effective. It is entirely possible, however, that the optimal number of inner generations for which cancellation should be performed is more than 10. In general, the number of inner generations per noise replica was around 700. These preliminary investigations should be extended by performing cancellation for a varying number of generations, to see if there might be an optimal value (as opposed to simply performing cancellation at the end of each inner generation). This being said, always performing weight cancellation is certainly an excellent approach for the benchmark configuration examined in this work, given the observed performance increases it provides, even if a slightly more effective approach might exist.

6. CONCLUSIONS

In this paper, we have investigated the effectiveness of two variance-reduction techniques for noise calculations. First, we examined the use of branchless methods to sample the noise source. It was observed that such techniques seem to pose no real benefit, and may even be detrimental to performance. While this puzzling behaviour should be investigated further, it seems unlikely that the branchless technique for complex weights (at least in the form proposed in this paper) has much to offer in terms of improving the sampling of the noise source.

Subsequently, the effects of weight cancellation were considered. There are currently two main algorithms in use for performing state-of-the-art neutron noise calculations in Monte Carlo simulations, which have been implemented in the production-level codes TRIPOLI-4[®] [4] and MCNP [11]. The method developed by Rouchon et al. for TRIPOLI-4[®] does not use any weight cancellation; while Yamamoto used a weight cancellation technique in an initial implementation [3], it was later abandoned, as it was determined to not be strictly necessary for the typical domains of interest, and would be difficult to implement in a large production-level code [11]. As Yamamoto's initial implementation could not estimate the variance of the noise field, no performance studies were conducted. This has left an open question as to the necessity and effects of weight cancellation in Monte Carlo noise simulations. In this work, we implemented the noise method used in TRIPOLI-4[®] [4], but added a weight cancellation scheme similar to that used by Yamamoto [3]. It was demonstrated that the use of weight cancellation during noise transport can produce large increases in the Figure of Merit (FOM). When using the total runtime to calculate the FOM, increases by a factor of more than 28.6 were sometimes observed. If considering the FOM calculated with only the time spent transporting noise particles, this factor increases to 65.8. Such large improvements are far from trivial: while these results must be confirmed on a larger set of benchmark configurations, they indicate that cancellation should be made available in any production-level Monte Carlo code capable of performing neutron noise calculations. Further examinations should also look at the case of neutron noise induced by vibrations: this is typically a more difficult problem for Monte Carlo codes than oscillations, and is an area where weight cancellation may play an even more important role [25].

ACKNOWLEDGEMENTS

We would like to express our gratitude to Dr. Amélie Rouchon for helpful discussions, and providing APOLLO3[®] reference solutions.

REFERENCES

- [1] C. Demazière, P. Vinai, M. Hursin, S. Kollias, and J. Herb. "Overview of the CORTEX project." In *Proceedings of PHYSOR2018*. Cancun, Mexico (2018).
- [2] C. Demazière. "CORE SIM: A multi-purpose neutronic tool for research and education." *Annals of Nuclear Energy*, **volume 38**(12), p. 2698–2718 (2011).
- [3] T. Yamamoto. "Monte Carlo method with complex-valued weights for frequency domain analyses of neutron noise." *Annals of Nuclear Energy*, **volume 58**, p. 72–79 (2013).
- [4] A. Rouchon, A. Zoia, and R. Sanchez. "A new Monte Carlo method for neutron noise calculations in the frequency domain." *Annals of Nuclear Energy*, **volume 102**, p. 465–475 (2017).
- [5] U. Rohde, M. Seidl, S. Kliem, and Y. Bilodid. "Neutron noise observations in German KWU built PWRs and analyses with the reactor dynamics code DYN3D." *Annals of Nuclear Energy*, **volume 112**(Proc. R. Soc. Lond. A 458 2002), p. 715–734 (2018).
- [6] N. Olmo-Juan, C. Demazière, T. Barrachina, R. Miró, and G. Verdú. "PARCS vs CORE SIM neutron noise simulations." *Progress in Nuclear Energy*, **volume 115**, p. 169–180 (2019).
- [7] A. Vidal-Ferrándiz, A. Carreño, D. Ginestar, C. Demazière, and G. Verdú. "A time and frequency domain analysis of the effect of vibrating fuel assemblies on the neutron noise." *Annals of Nuclear Energy*, **volume 137**, p. 107076 (2020).
- [8] D. Chionis, A. Dokhane, L. Belblidia, H. Ferroukhi, G. Girardin, and A. Pautz. "Development and verification of a methodology for neutron noise response to fuel assembly vibrations." *Annals of Nuclear Energy*, **volume 147**, p. 107669 (2020).

- [9] A. Rouchon. *Analyse et développement d'outils numériques déterministes et stochastiques résolvant les équations du bruit neutronique et applications aux réacteurs thermiques et rapides*. Ph.D. thesis, Université Paris-Saclay (2016).
- [10] H. Yi, P. Vinai, and C. Demazière. “Discrete ordinates solver with diffusion synthetic acceleration for simulations of 2-d and 2-energy group neutron noise problems.” In *Proceedings of M&C2019*. Portland USA (2019).
- [11] T. Yamamoto. “Implementation of a frequency-domain neutron noise analysis method in a production-level continuous energy Monte Carlo code: Verification and application in a BWR.” *Annals of Nuclear Energy*, **volume 115**, p. 494–501 (2018).
- [12] A. Mylonakis, H. Yi, P. Vinai, and C. Demazière. “Neutron noise simulations in a heterogeneous system: A comparison between a diffusion-based and a discrete ordinates solver.” In *Proceedings of M&C2019*. Portland USA (2019).
- [13] A. Rouchon, W. Jarrah, and A. Zoia. “The new neutron noise solver of the Monte Carlo code TRIPOLI-4.” In *Proceedings of M&C2019*. Portland USA (2019).
- [14] H. Belanger, D. Mancusi, and A. Zoia. “Review of Monte Carlo methods for particle transport in continuously-varying media.” *The European Physical Journal Plus*, **volume 135**(11), p. 877 (2020).
- [15] H. Belanger, D. Mancusi, and A. Zoia. “Exact weight cancellation in Monte Carlo eigenvalue transport problems.” *Physical Review E*, **volume 104**(1), p. 015306 (2021).
- [16] T. E. Booth and J. E. Gubernatis. “Exact Regional Monte Carlo Weight Cancellation for Second Eigenfunction Calculations.” *Nuclear Science and Engineering*, **volume 165**(3), p. 283–291 (2010).
- [17] I. Lux and L. Koblinger. *Monte Carlo particle transport methods: neutron and photon calculations*. CRC press (2018).
- [18] H. Belanger. “MGMC.” (2021). URL <https://github.com/HunterBelanger/mgmc>. v0.2.0-alpha.
- [19] B. L. Sjenitzer and J. E. Hoogenboom. “Dynamic Monte Carlo Method for Nuclear Reactor Kinetics Calculations.” *Nuclear Science and Engineering*, **volume 175**(1), p. 94–107 (2013).
- [20] M. Faucher, D. Mancusi, and A. Zoia. “New kinetic simulation capabilities for Tripoli-4@: Methods and applications.” *Annals of Nuclear Energy*, **volume 120**, p. 74–88 (2018).
- [21] B. Molnar, G. Tolnai, and D. Legrady. “A GPU-based direct Monte Carlo simulation of time dependence in nuclear reactors.” *Annals of Nuclear Energy*, **volume 132**, p. 46–63 (2019).
- [22] D. Mancusi and A. Zoia. “Zero-Variance Schemes for Kinetic Monte Carlo Simulations.” *Eur Phys J Plus*, **volume 135**(5), p. 401 (2020).
- [23] H. Belanger, D. Mancusi, and A. Zoia. “Solving Eigenvalue Transport Problems With Negative Weights and Regional Cancellation.” In *Proceedings of M&C2021*. Raleigh USA (2021).
- [24] P. Vinai, C. Demazière, B. Gasse, A. Rouchon, A. Zoia, A. Vidal-Ferràndiz, D. Ginestar, G. Verdú, and T. Yamamoto. “Comparison of Neutron Noise Solvers Based on Numerical Benchmarks in a 2-D Simplified UOX Fuel Assembly.” In *Proceedings of M&C2021*. Raleigh, NC, USA (2021).
- [25] A. Zoia, A. Rouchon, B. Gasse, C. Demazière, and P. Vinai. “Analysis of the neutron noise induced by fuel assembly vibrations.” *Annals of Nuclear Energy*, **volume 154**, p. 108061 (2021).

13 - Variance Reduction and Noise Source Sampling Techniques for Monte Carlo Simulations of Neutron Noise Induced by Mechanical Vibrations

This chapter has been accepted for publication in the journal *Nuclear Science and Engineering*.

The preprint version appearing in this manuscript can be found at <https://arxiv.org/abs/2207.09822>.

Variance Reduction and Noise Source Sampling Techniques for Monte Carlo Simulations of Neutron Noise Induced by Mechanical Vibrations

Hunter Belanger,^{*} Davide Mancusi,[†] Amélie Rouchon,[‡] and Andrea Zoia[§]

Université Paris-Saclay, CEA, Service d'Études des Réacteurs et de Mathématiques Appliquées, 91191, Gif-sur-Yvette, France

Neutron noise in nuclear power reactors refers to the small fluctuations around the average neutron flux at steady state resulting from time-dependent perturbations inside the core. The neutron noise equations in the frequency domain can be solved using Monte Carlo simulation codes, which are capable of obtaining reference solutions involving almost no approximations, but are hindered by severe issues affecting the statistical convergence: the simultaneous presence of positive and negative particles, which is required by the nature of the complex noise equations, leads to catastrophically large variance in the tallies. In this work, we consider the important case of neutron noise problems induced by mechanical vibrations. First, we derive a new exact sampling strategy for the noise source. Then, building upon our previous findings in other contexts, we show that weight cancellation methods can be highly beneficial in dealing with the presence of negative weights, enabling extremely large gains in the figure of merit. We successfully demonstrate our results on a benchmark configuration consisting of a fuel assembly with a vibrating pin and we discuss possible pathways for further improvements.

I. INTRODUCTION

Neutron noise in nuclear power reactors is defined as the small fluctuations occurring around the average neutron flux at steady state due to perturbations typically induced by fluid-structure interactions or moderator density fluctuations inside the core [1–3]. These perturbations may affect different reactor components, at the scale of single or multiple fuel rods, entire assemblies, or the full core vessel, depending on the specific physical origin of the disturbances. Generally speaking, neutron noise is an unwanted phenomenon, which in some extreme cases might lead to a significant decrease in reactor power in order to ensure safe operation, or even to the shut-down of the reactor [4]. Notwithstanding, such disturbances carry an information content that can be usefully exploited to improve reactor diagnostics through the application of inverse problem techniques, for instance by locating anomalous control rod or fuel assembly vibrations, or monitoring moderator speed and void fraction [5–10].

The neutron noise equations are established by considering the effect of temperature and density changes and/or material displacements on the operators occurring in the Boltzmann equation and in the precursor equations: the noise is considered as a small time-dependent perturbation with respect to the stationary flux, and the time-dependent operators are similarly decomposed into a stationary part and a residual time-dependent perturbation [3]. Then, assuming that the products of perturbed quantities can be neglected (the so-called ‘orthodox’ linearization [11]), a Fourier transform finally leads to a complex-valued equation for the neutron noise in the frequency domain [3].

A number of analytical and semi-analytical results concerning the solutions of the noise equations have been obtained using diffusion theory, simplified few-group transport equations, and Green’s function methods [12–17]: although these approaches shed light on the behaviour of the neutron noise field in a variety of physical configurations, their application to real-world systems is somewhat limited. Recently, a renewed interest in neutron noise stimulated by industrial problems related to core diagnostics has fostered the development of novel numerical codes capable of solving the noise equations in the time or frequency domain using state-of-the-art methods [18–24]. In this respect, a remarkable contribution has been provided by the EU H2020 CORTEX project (2017-2021) [25], during which several deterministic and Monte Carlo solvers for the noise equation have been conceived [26–28] and validated against benchmark problems [29–31] and experimental data stemming from dedicated measurement campaigns in research reactors [32, 33].

Since the number of available experimental datasets for the qualification of faster, albeit approximate, deterministic solvers for the noise equations is still rather small, one would like to rely on Monte Carlo simulation as a gold-standard tool to compute reference solutions involving almost no approximations, similarly to what is customarily done for the regular Boltzmann equation in reactor physics and radiation shielding problems [30, 31]. Monte Carlo sampling methods devoted to the noise equations have been devised and successfully tested in several applications, but numerical investigations have shown that their statistical convergence may be extremely poor in some cases: this is basically due to the fact that solving the noise equations requires sampling particles carrying complex statistical weights with positive and negative real and imaginary components [20, 21]. When positive and negative contributions are tallied in order to estimate the neutron noise, the variance might become overwhelmingly large and a huge

^{*} hunter.belanger@cea.fr

[†] davide.mancusi@cea.fr

[‡] amelie.rouchon@cea.fr

[§] andrea.zoia@cea.fr

number of sampled particles is mandatory to achieve a reasonable statistical uncertainty, which makes such simulations unwieldy even for relatively simple configurations [30, 31].

The possibility of using Monte Carlo simulations as a routine tool for industrial calculations in the field of neutron noise therefore crucially depends on the availability of ad hoc variance reduction and population control techniques capable of handling the simultaneous presence of positive and negative particles. Recent insights have shown that weight cancellation methods might be the key to successfully dealing with such issues [20, 34, 35]. Preliminary investigations involving cross section oscillation problems in a fuel assembly suggest that the introduction of weight cancellation actually leads to astounding improvements in the figure of merit of the noise simulations by Monte Carlo codes, at the expense of extensive restructuring of the sampling routines [36]. The benefits induced by the vastly reduced statistical uncertainty establish Monte Carlo simulations as a practical tool to obtain reference solutions and outweighs the efforts involved in code rewriting. In this work, we extend and considerably generalize our previous findings to the case of neutron noise induced by mechanical vibrations, which poses distinct challenges in both sampling the noise source particles and handling the variance of the resulting noise field.

This work is organized as follows: In Sec. II we recall the derivation of the neutron noise equations in the frequency domain, and we show that they can be interpreted as a complex-valued fixed-source transport problem, starting from a complex-valued source. Subse-

quently, in Sec. III we review existing sampling methods for the noise source, and introduce a new exact sampling strategy for the case of mechanical vibrations. After briefly recalling in Sec. IV the Monte Carlo methods for solving the transport of noise particles, in Sec. V we discuss the application of weight cancellation to reduce the variance of noise problems. The effectiveness of these methods is illustrated in Sec. VI on a benchmark configuration consisting of a fuel assembly where a pin is subject to periodic mechanical vibrations. Conclusions are finally drawn in Sec. VII.

II. THE NEUTRON NOISE EQUATIONS

We start by considering a neutron multiplying system which is initially critical. To allow for possible model bias due to technological uncertainties and/or nuclear data, we assume that the fission production is normalized by the fundamental eigenvalue k_{eff} , which enforces the stationary state. Now, if a perturbation is introduced into the system, so that the macroscopic quantities such as cross sections, yields, and scattering laws become time-dependent, then the resulting neutron flux $\varphi(\mathbf{r}, \hat{\Omega}, E, t)$ will also be time-dependent, and will be therefore described by the time-dependent Boltzmann equation, coupled to the evolution equations for the delayed neutron precursors. After injecting the precursor equations into the Boltzmann equation, we concisely denote the time-dependent transport equation as $\mathcal{B}_k(t)\varphi(\mathbf{r}, \hat{\Omega}, E, t) = 0$, where

$$\begin{aligned} \mathcal{B}_k(t) = & \frac{1}{v} \frac{\partial}{\partial t} + \hat{\Omega} \cdot \nabla + \Sigma_t(\mathbf{r}, E, t) - \int_{4\pi} \int_0^\infty \nu_s(\mathbf{r}, E', t) \Sigma_s(\mathbf{r}, E', t) f_s(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, t) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \int_{4\pi} \int_0^\infty \nu_{f,p}(\mathbf{r}, E', t) \Sigma_f(\mathbf{r}, E', t) f_{f,p}(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, t) dE' d\hat{\Omega}' - \\ & \frac{1}{k_{\text{eff}}} \sum_j \int_{-\infty}^t \int_{4\pi} \int_0^\infty \lambda_j e^{-\lambda_j(t-t')} \nu_{f,d}^j(\mathbf{r}, E', t') \Sigma_f(\mathbf{r}, E', t') f_{f,d}^j(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, t') dE' d\hat{\Omega}' dt'. \quad (1) \end{aligned}$$

Due to the effect of the perturbation, it is assumed that all of the time-dependent terms in Eq. (1) can be decomposed into a stationary part and an additional small, periodic, time-dependent contribution¹. For example, the total macroscopic cross section is decomposed as

$$\Sigma_t(\mathbf{r}, E, t) = \Sigma_t(\mathbf{r}, E) + \delta\Sigma_t(\mathbf{r}, E, t). \quad (2)$$

This can similarly be done for reaction yields, scattering laws and fission spectra. As it would become slightly cumbersome to write each integral explicitly, we use the following shorthand for the decomposition of the term associated with the generic reaction channel α :

$$\begin{aligned} \nu_\alpha(\mathbf{r}, E, t) \Sigma_\alpha(\mathbf{r}, E, t) f_\alpha(E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}', t) = \\ \nu_\alpha(\mathbf{r}, E) \Sigma_\alpha(\mathbf{r}, E) f_\alpha(E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}') \\ + \delta[\nu_\alpha \Sigma_\alpha f_\alpha](\mathbf{r}, E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}', t), \quad (3) \end{aligned}$$

where all of the time dependence has been placed in the single $\delta[\nu_\alpha \Sigma_\alpha f_\alpha]$ term. A thorough discussion on this de-

¹ In this work, we do not consider the possibility of perturbations in the precursor decay constants.

composition will be provided in Sec. III. In this formalism, we may break the operator $\mathcal{B}_k(t)$ into two different

components

$$\mathcal{B}_k(t) = \mathcal{B}(t) + \delta\mathcal{B}(t), \quad (4)$$

with the operator

$$\begin{aligned} \mathcal{B}(t) = & \frac{1}{v} \frac{\partial}{\partial t} + \hat{\mathbf{\Omega}} \cdot \nabla + \Sigma_t(\mathbf{r}, E) - \int_{4\pi} \int_0^\infty \nu_s(\mathbf{r}, E') \Sigma_s(\mathbf{r}, E') f_s(E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}) dE' d\hat{\mathbf{\Omega}}' - \\ & \frac{1}{k_{\text{eff}}} \int_{4\pi} \int_0^\infty \nu_{f,p}(\mathbf{r}, E') \Sigma_f(\mathbf{r}, E') f_{f,p}(E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}) dE' d\hat{\mathbf{\Omega}}' - \\ & \frac{1}{k_{\text{eff}}} \sum_j \int_{-\infty}^t \int_{4\pi} \int_0^\infty \lambda_j e^{-\lambda_j(t-t')} \nu_{f,d}^j(\mathbf{r}, E') \Sigma_f(\mathbf{r}, E') f_{f,d}^j(E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}) dE' d\hat{\mathbf{\Omega}}' dt', \end{aligned} \quad (5)$$

and the perturbation operator

$$\begin{aligned} \delta\mathcal{B}(t) = & \delta\Sigma_t(\mathbf{r}, E, t) - \int_{4\pi} \int_0^\infty \delta[\nu_s \Sigma_s f_s](\mathbf{r}, E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}, t) dE' d\hat{\mathbf{\Omega}}' - \\ & \frac{1}{k_{\text{eff}}} \int_{4\pi} \int_0^\infty \delta[\nu_{f,p} \Sigma_f f_{f,p}](\mathbf{r}, E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}, t) dE' d\hat{\mathbf{\Omega}}' - \\ & \frac{1}{k_{\text{eff}}} \sum_j \int_{-\infty}^t \int_{4\pi} \int_0^\infty \lambda_j e^{-\lambda_j(t-t')} \delta[\nu_{f,d}^j \Sigma_f f_{f,d}^j](\mathbf{r}, E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}, t') dE' d\hat{\mathbf{\Omega}}' dt'. \end{aligned} \quad (6)$$

Consequently, we postulate that the time-dependent flux which solves Eq. (1) can be similarly decomposed as

$$\varphi(\mathbf{r}, \hat{\mathbf{\Omega}}, E, t) = \varphi_c(\mathbf{r}, \hat{\mathbf{\Omega}}, E) + \delta\varphi(\mathbf{r}, \hat{\mathbf{\Omega}}, E, t), \quad (7)$$

where φ_c is the fundamental eigenmode of the Boltzmann equation associated with the fundamental eigenvalue k_{eff} , before the time-dependent disturbance was introduced, and the flux perturbation $\delta\varphi$ is the neutron noise. From these definitions, we see that

$$\begin{aligned} \mathcal{B}_k(t)\varphi(\mathbf{r}, \hat{\mathbf{\Omega}}, E, t) &= 0 \\ &= \left[\mathcal{B}(t) + \delta\mathcal{B}(t) \right] \left[\varphi_c(\mathbf{r}, \hat{\mathbf{\Omega}}, E) + \delta\varphi(\mathbf{r}, \hat{\mathbf{\Omega}}, E, t) \right] \\ &= \mathcal{B}(t)\delta\varphi(\mathbf{r}, \hat{\mathbf{\Omega}}, E, t) + \delta\mathcal{B}(t)\varphi_c(\mathbf{r}, \hat{\mathbf{\Omega}}, E) \\ &+ \mathcal{B}(t)\varphi_c(\mathbf{r}, \hat{\mathbf{\Omega}}, E) + \delta\mathcal{B}(t)\delta\varphi(\mathbf{r}, \hat{\mathbf{\Omega}}, E, t). \end{aligned} \quad (8)$$

The term $\mathcal{B}(t)\varphi_c(\mathbf{r}, \hat{\mathbf{\Omega}}, E)$ vanishes, since φ_c is the critical (stationary) flux. Next, we apply the ‘‘orthodox’’ linearization, where we assume that for small perturbations the terms involving products of perturbed quan-

tities can be neglected, namely, $\delta\mathcal{B}\delta\varphi \approx 0$. The validity of this assumption has been discussed by several authors [11, 12, 37], and has been recently revisited for the case of mechanical vibrations [16, 17]. Upon rearranging the two remaining terms, we obtain a Boltzmann-like transport equation for the neutron noise which resembles a fixed-source problem:

$$\mathcal{B}(t)\delta\varphi(\mathbf{r}, \hat{\mathbf{\Omega}}, E, t) = -\delta\mathcal{B}(t)\varphi_c(\mathbf{r}, \hat{\mathbf{\Omega}}, E), \quad (9)$$

the term $-\delta\mathcal{B}\varphi_c$ representing the ‘‘noise source’’.

Since we are typically interested in the periodic solution to the noise equation, we apply then a Fourier transform

$$g(\omega) = \mathcal{F}[g(t)](\omega) = \int_{-\infty}^{+\infty} e^{-i\omega t} g(t) dt \quad (10)$$

to Eq. (9), ω denoting the angular frequency, which yields the linearized noise equation in the frequency domain:

$$\mathcal{B}(\omega)\delta\varphi(\mathbf{r}, \hat{\mathbf{\Omega}}, E, \omega) = -\delta\mathcal{B}(\omega)\varphi_c(\mathbf{r}, \hat{\mathbf{\Omega}}, E). \quad (11)$$

The Fourier-transformed operators appearing in Eq. (11) are complex-valued and read

$$\begin{aligned} \mathcal{B}(\omega) = & i\frac{\omega}{v} + \hat{\mathbf{\Omega}} \cdot \nabla + \Sigma_t(\mathbf{r}, E) - \int_{4\pi} \int_0^\infty \nu_s(\mathbf{r}, E') \Sigma_s(\mathbf{r}, E') f_s(E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}) dE' d\hat{\mathbf{\Omega}}' - \\ & \frac{1}{k_{\text{eff}}} \int_{4\pi} \int_0^\infty \nu_{f,p}(\mathbf{r}, E') \Sigma_f(\mathbf{r}, E') f_{f,p}(E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}) dE' d\hat{\mathbf{\Omega}}' - \\ & \frac{1}{k_{\text{eff}}} \sum_j \frac{\lambda_j}{\lambda_j + i\omega} \int_{4\pi} \int_0^\infty \nu_{f,d}^j(\mathbf{r}, E') \Sigma_f(\mathbf{r}, E') f_{f,d}^j(E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}) dE' d\hat{\mathbf{\Omega}}', \quad (12) \end{aligned}$$

and

$$\begin{aligned} \delta\mathcal{B}(\omega) = & \delta\Sigma_t(\mathbf{r}, E, \omega) - \int_{4\pi} \int_0^\infty \delta[\nu_s \Sigma_s f_s](\mathbf{r}, E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}, \omega) dE' d\hat{\mathbf{\Omega}}' - \\ & \frac{1}{k_{\text{eff}}} \int_{4\pi} \int_0^\infty \delta[\nu_{f,p} \Sigma_f f_{f,p}](\mathbf{r}, E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}, \omega) dE' d\hat{\mathbf{\Omega}}' - \\ & \frac{1}{k_{\text{eff}}} \sum_j \frac{\lambda_j}{\lambda_j + i\omega} \int_{4\pi} \int_0^\infty \delta[\nu_{f,d}^j \Sigma_f f_{f,d}^j](\mathbf{r}, E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}, \omega) dE' d\hat{\mathbf{\Omega}}'. \quad (13) \end{aligned}$$

For symmetry, it is sometimes convenient to write the $\delta\Sigma_t(\mathbf{r}, E, \omega)$ term in Eq. (13) in the same form as the other terms:

$$\begin{aligned} \delta\Sigma_t(\mathbf{r}, E, \omega) = & \int_{4\pi} \int_0^\infty \delta\Sigma_t(\mathbf{r}, E', \omega) \delta(E' - E) \delta(\hat{\mathbf{\Omega}}' - \hat{\mathbf{\Omega}}) dE' d\hat{\mathbf{\Omega}}', \quad (14) \end{aligned}$$

which shows that $\delta\Sigma_t(\mathbf{r}, E, \omega)$ is associated to a unit yield and a “delta-copy” spectrum $\delta(E' - E)\delta(\hat{\mathbf{\Omega}}' - \hat{\mathbf{\Omega}})$. The terms $\delta\Sigma_t(\mathbf{r}, E, \omega)$ and $\delta[\nu_\alpha \Sigma_\alpha f_\alpha](\mathbf{r}, E' \rightarrow E, \hat{\mathbf{\Omega}}' \rightarrow \hat{\mathbf{\Omega}}, \omega)$ appearing in the noise source involve the Fourier-transformed perturbation of the transport operators, and as such their precise functional form is determined by the specific type of perturbation of the noise model (e.g., mechanical vibrations or oscillations). Regardless of the nature of the solver (deterministic or Monte Carlo), these quantities must thus be prepared in advance, in the source code, before solving the noise equation (11). In the next section we will discuss the exact form of the noise source for two different types of perturbations, and the Monte Carlo strategies that we have implemented to sample them.

III. SAMPLING THE NEUTRON NOISE SOURCE

Sampling the neutron noise source requires the knowledge of the Fourier-transformed perturbation operator $-\delta\mathcal{B}(\omega)$ and the fundamental eigenmode $\varphi_c(\mathbf{r}, \hat{\mathbf{\Omega}}, E)$. The fundamental eigenmode can be obtained by running the well-known power iteration algorithm, which leads to a collection of neutrons whose flux is precisely $\varphi_c(\mathbf{r}, \hat{\mathbf{\Omega}}, E)$, after a sufficiently large numbers of inactive cycles. When convergence has been attained,

$-\delta\mathcal{B}(\omega)\varphi_c(\mathbf{r}, \hat{\mathbf{\Omega}}, E)$ can be estimated by running one or more additional power iteration cycles, where the noise source particles are sampled from the histories of the neutrons, and are placed in a dedicated bank [28]. This sampling strategy is inspired from the algorithm used to implement Monte Carlo kinetic simulations [38]. As previously mentioned, the explicit form of $-\delta\mathcal{B}(\omega)$ depends on the type of noise source to be examined.

A. Cross Section Oscillations

First, we shall consider the simple case of cross section oscillations, which, although somewhat artificial, is nonetheless important in that it will play a key role in addressing the more involved case of mechanical vibrations, covered in Sec. III B. Cross section oscillations are relatively straightforward, since they do not involve any spatial movement of the materials. Suppose there is a spatial region \mathcal{P} where the macroscopic cross sections exhibit a periodic time dependence. We write the cross sections as [30]

$$\Sigma_\alpha(\mathbf{r}, E, t) = \Sigma_\alpha(\mathbf{r}, E) [1 + \varepsilon_\alpha \sin(\omega_0 t) \mathbb{1}_{\mathcal{P}}(\mathbf{r})], \quad (15)$$

where ε_α is the amplitude of the perturbation, and

$$\mathbb{1}_{\mathcal{P}}(\mathbf{r}) = \begin{cases} 1 & \text{for } \mathbf{r} \in \mathcal{P} \\ 0 & \text{for } \mathbf{r} \notin \mathcal{P} \end{cases}. \quad (16)$$

Upon taking the Fourier transform, the cross section perturbation in the frequency domain is then

$$\delta\Sigma_\alpha(\mathbf{r}, E, \omega) = \Sigma_\alpha(\mathbf{r}, E) \mathbb{1}_{\mathcal{P}}(\mathbf{r}) h_\alpha(\omega, \omega_0, \varepsilon_\alpha), \quad (17)$$

where

$$h_\alpha(\omega, \omega_0, \varepsilon_\alpha) = -i\pi\varepsilon_\alpha [\delta(\omega - \omega_0) + \delta(\omega + \omega_0)]. \quad (18)$$

We assume that no other material properties display any time dependence. Such a problem is admittedly academic, but provides an excellent benchmark for comparison of codes and verification of algorithms. The noise source terms in Eq. (13) can be thus written as

$$\int_{4\pi} \int_0^\infty \nu_\alpha(\mathbf{r}, E') \delta\Sigma_\alpha(\mathbf{r}, E', \omega) \times f_\alpha(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) \varphi_c(\mathbf{r}, \hat{\Omega}', E') dE' d\hat{\Omega}'. \quad (19)$$

By multiplying and dividing by $\Sigma_\alpha(\mathbf{r}, E')$ in the integrand, and rearranging, we then have

$$\int_{4\pi} \int_0^\infty \underbrace{\frac{\delta\Sigma_\alpha(\mathbf{r}, E', \omega)}{\Sigma_\alpha(\mathbf{r}, E')}}_{\text{complex importance factor}} \times \underbrace{\nu_\alpha(\mathbf{r}, E') \Sigma_\alpha(\mathbf{r}, E') f_\alpha(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega})}_{\text{standard production rate}} \times \varphi_c(\mathbf{r}, \hat{\Omega}', E') dE' d\hat{\Omega}'. \quad (20)$$

Eq. (20) can be handled by Monte Carlo codes with importance sampling, where we use the standard sampling methods for the production rate, and multiply the weight of the sampled noise particles corresponding to reaction α by the complex factor $\delta\Sigma_\alpha(\mathbf{r}, E, \omega)/\Sigma_\alpha(\mathbf{r}, E) = h_\alpha(\omega, \omega_0, \varepsilon_\alpha) \mathbb{1}_{\mathcal{P}}(\mathbf{r})$. To sample the noise source, action only needs to be taken when a neutron has a collision within the region \mathcal{P} exhibiting the cross section oscillations. At this point, we need to separately sample the three components of the noise source, that is, the copy term $\delta\Sigma_t \varphi_c$, the fission term, and the scattering term. The noise source contribution from the term $\delta\Sigma_t \varphi_c$ is sampled by creating an exact copy of the particle and multiplying its weight by $h_t(\omega, \omega_0, \varepsilon_t)$. For fission and scattering, a nuclide is sampled from the material at \mathbf{r} , in the standard manner, and we apply forced fission and implicit absorption. If the nuclide is fissile, then a random variable $\xi \sim \mathcal{U}(0, 1)$ is used to calculate the number of fission noise particles to produce:

$$n = \left\lfloor \frac{\nu_f(E) \sigma_f(E)}{\sigma_t(E) k_{\text{eff}}} + \xi \right\rfloor, \quad (21)$$

where $\sigma_t(E)$ and $\sigma_s(E)$ are respectively the microscopic total and scattering cross sections for the selected nuclide. Each fission noise particle will inherit the weight of its parent, and have its energy and direction sampled normally. The weights of these particles are multiplied by $h_f(\omega, \omega_0, \varepsilon_f)$; furthermore, any delayed neutrons will also have their weight multiplied by the complex yield $\lambda_j/(\lambda_j + i\omega)$. Finally, noise particles from scattering must be sampled; first we sample a scattering channel, and then we sample the energy and direction of the scattered noise particle. In addition to multiplying the noise particle's weight by $h_s(\omega, \omega_0, \varepsilon_s)$, it is also necessary to multiply by the probability of scatter, $\sigma_s(E)/\sigma_t(E)$, where $\sigma_s(E)$ is the microscopic scattering cross sections for the selected nuclide.

B. Mechanical Vibrations

Mechanical vibrations are inherently different from the previous case of cross section oscillations, due to the fact that they involve material displacements as a function of time. Assuming that the displacement occurs along one spatial dimension, we will have material M_L at position $x < x_0$ and material M_R at position $x > x_0$ before the perturbation is applied. In stationary conditions, the material as a function of position is thus described by the function

$$M(x) = \begin{cases} M_L & \text{for } x < x_0 \\ M_R & \text{for } x > x_0 \end{cases}. \quad (22)$$

Now, if a perturbation is introduced into this system in the form of a sinusoidal vibration of the interface between the two materials, with angular frequency ω_0 and amplitude $\varepsilon > 0$, then the material found at a given position in the perturbed region is also a function of time. For positions where $|x - x_0| < \varepsilon$, the function which determines the material at position x and time t is written as

$$M(x, t) = M(x) + \Delta M [H(x - x_0) - H(x - x_0 - \varepsilon \sin(\omega_0 t))], \quad (23)$$

where $\Delta M = M_L - M_R$, and $H(x)$ is the Heaviside function [16]. This function always returns either M_L or M_R . Based on the material perturbation $\delta M = M(x, t) - M(x)$ across the interface, the corresponding perturbation of the operator kernels $\delta[\nu_\alpha \Sigma_\alpha f_\alpha]$ must then be

$$\delta[\nu_\alpha \Sigma_\alpha f_\alpha](x, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, t) = \Delta[\nu_\alpha \Sigma_\alpha f_\alpha] \times [H(x - x_0) - H(x - x_0 - \varepsilon \sin(\omega_0 t))], \quad (24)$$

where

$$\Delta[\nu_\alpha \Sigma_\alpha f_\alpha] = \nu_\alpha^L(E) \Sigma_\alpha^L(E) f_\alpha^L(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) - \nu_\alpha^R(E) \Sigma_\alpha^R(E) f_\alpha^R(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}). \quad (25)$$

The Fourier transform of Eq. (24) has been shown to be [16]

$$\delta[\nu_\alpha \Sigma_\alpha f_\alpha](x, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}, \omega) = \Delta[\nu_\alpha \Sigma_\alpha f_\alpha] \left\{ c_0(x, x_0) \delta(\omega) + \sum_{n=1}^{\infty} c_n(x, x_0) [\delta(\omega - n\omega_0) + \delta(\omega + n\omega_0) (-1)^n] \right\}, \quad (26)$$

with

$$c_n(x, x_0) = \frac{2}{n} \sin\left(n \arcsin\left(\frac{x - x_0}{\varepsilon}\right)\right) e^{-in\pi/2} \quad (27)$$

for $n \geq 1$, and

$$c_0(x, x_0) = \begin{cases} \pi - 2 \arcsin\left(\frac{x-x_0}{\varepsilon}\right) & \text{for } x \geq x_0 \\ -\pi - 2 \arcsin\left(\frac{x-x_0}{\varepsilon}\right) & \text{for } x < x_0 \end{cases}. \quad (28)$$

Immediately apparent from Eq. (26) is that, while the noise source for cross section oscillations is monochromatic, the noise source for mechanical vibrations contains an infinite number of discrete harmonics at $n\omega_0$, all multiples of the perturbation frequency ω_0 . Additionally, there is a non-trivial spatial dependence indicated by Eq. (27). All even harmonics are purely real and have even spatial parity about x_0 , whereas all odd harmonics are purely imaginary and have odd parity. We will now consider possible methods to sample the noise source for mechanical vibrations in a Monte Carlo code.

1. Approximate Noise Source Sampling

From Eq. (24), we see that the noise source is caused by a difference of the product of the cross section, reaction yield, and scattering law. In the initial approach implemented in the noise solver of the TRIPOLI-4[®] Monte Carlo code, developed at CEA [39], it was assumed that

$$\Delta[\nu_\alpha \Sigma_\alpha f_\alpha] \approx \nu_\alpha f_\alpha \Delta\Sigma_\alpha(x, x_0, \omega), \quad (29)$$

where only the difference in the macroscopic cross sections was considered. It must be noted that, while $\Delta\Sigma_\alpha$ requires evaluating the cross section on both sides of the interface, ν_α and f_α were both evaluated only for the material at the location where the collision occurred. This was partially done for simplicity, as evaluating f_α on both sides of the interface to obtain $\Delta[\nu_\alpha \Sigma_\alpha f_\alpha]$ could pose particular challenges².

At first glance, it would appear that Eq. (29) could be handled with importance sampling, as was done for cross section oscillations, where one only needs information from the material at which the collision occurred, and this is the main advantage of the proposed approximation. Sadly, further assumptions are required. As a tangible example, let us consider the case of a fuel pin vibrating in water moderator. Unlike for the case of oscillating cross sections, where all of the nuclides participating in the perturbation are present at the sampling site in the unperturbed system, in Eq. (29) isotopes from the fuel participate in the perturbation in the water and vice versa. It should therefore be possible to sample fission noise particles from collisions occurring in the water surrounding the vibrating pin, which would not be allowed

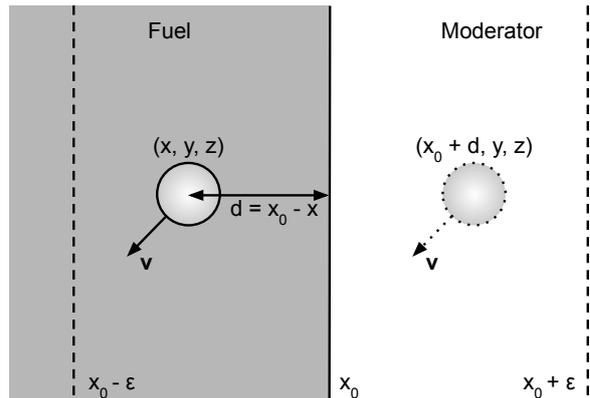


FIG. 1. Depicted is the operation of copying noise particles from fission in the fuel, and placing them in the moderator. The line at x_0 is the boundary between the fuel and the moderator for the static problem. Dashed lines at $x_0 - \varepsilon$ and $x_0 + \varepsilon$ represent the minimum and maximum positions of the interface with the vibration. The particle in the fuel is the sampled fission noise particle. An exact copy of this fission noise particle (identical direction, energy, and weight) is placed at the symmetric position within the moderator, corresponding to an x -coordinate of $x_0 + d$. The weight of the copied particle in the moderator must be multiplied by $(-1)^{n+1}$, to account for the parity of the coefficients $c_n(x, x_0)$.

with the importance sampling scheme from oscillations. A clever method was devised to circumvent this issue in TRIPOLI-4[®]: when a fission noise particle is sampled in the fuel region, a copy is made and then translated across the boundary of the vibration, to the symmetric location relative to the vibrating interface (taking into account the parity of the given harmonic). So long as the flux is approximately constant over the distance of the translation, this is likely a reasonable approximation, allowing for the placement of fission noise source particles in the water. This approximation of the placement of fission noise particles in the moderator is depicted in Fig. 1. Otherwise, the scheme implemented in TRIPOLI-4[®] samples noise source particles for vibrations in the same manner as for cross section oscillations. TRIPOLI-4[®] therefore also ignores the possibility of creating scatter noise particles from isotopes present on the other side of the interface (i.e. a noise particle born from a scattering event with a fuel isotope, when having a collision in the water).

2. Exact Noise Source Sampling

The noise equation is typically written using macroscopic material cross sections, yields, and energy-angle distributions. Each of the factors in the source term is actually a weighted average of the cross sections, yields and energy-angle distributions of each individual isotope present in the material. It is therefore possible to rewrite

² While the data representations used in Monte Carlo codes allow for easy sampling from the distribution f_α , they often do not facilitate the evaluation of the distribution f_α corresponding to a given argument.

Eq. (3) for mechanical vibrations as

$$\nu_\alpha(\mathbf{r}, E, t) \Sigma_\alpha(\mathbf{r}, E, t) f_\alpha(\mathbf{r}, E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}', t) = \sum_k N_k(\mathbf{r}, t) \nu_{k,\alpha}(\mathbf{r}, E) \sigma_{k,\alpha}(\mathbf{r}, E) f_{k,\alpha}(\mathbf{r}, E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}'). \quad (30)$$

Here, N_k is the concentration of nuclide k , and $\sigma_{k,\alpha}$ is the microscopic cross section of nuclide k for reaction channel α . For an individual nuclide, the microscopic cross sections, reaction yields and energy-angle distributions are time-independent. However, the macroscopic material-averaged quantities become time-dependent through the concentration $N_k(\mathbf{r}, t)$ of nuclides in the perturbed region, which is clearly a time-dependent quantity. In Monte Carlo codes, each individual nuclide is modelled explicitly, and the ensemble material averages naturally arise from neutrons undergoing collisions with all of the nuclides present in a given material. This suggests a new sampling strategy for the noise source due to mechanical vibrations. The concentration of nuclides in the perturbed region can be decomposed as

$$N_k(\mathbf{r}, t) = N_k(\mathbf{r}) + \delta N_k(\mathbf{r}, t), \quad (31)$$

allowing us to write each term of the noise source as

$$\int_{4\pi} \int_0^\infty \sum_k \delta N_k(\mathbf{r}, t) \nu_{k,\alpha}(\mathbf{r}, E') \sigma_{k,\alpha}(\mathbf{r}, E') \times f_{k,\alpha}(\mathbf{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) \varphi_c(\mathbf{r}, \hat{\Omega}', E') dE' d\hat{\Omega}'. \quad (32)$$

The form of $\delta N_k(\mathbf{r}, t)$ depends again on the material interface behavior given in Eq. (23), and therefore leads to a similar form for the Fourier transform $\delta N_k(\mathbf{r}, \omega)$.

To sample Eq. (32), one might be inclined to multiply and divide by $N_k(\mathbf{r})$, similar to the approach taken for oscillations, so to convert Eq. (32) into a complex-weighted particle production rate. Sadly, such a strategy is not possible for the case of vibrations. Consider the case of a UO_2 fuel pin vibrating in water. When inside the perturbed region in the water surrounding the pin, $\delta N_{235\text{U}} \neq 0$, but $N_{235\text{U}} = 0$, and it is therefore impossible to divide by $N_{235\text{U}}(\mathbf{r})$. Our proposed solution to this problem is to define a new fictitious material which is only used for the sampling of the noise source. This material must contain all nuclides which are present in the two materials involved in the vibration, and all of their concentrations must be nonzero. The exact choice of the concentration for each isotope is arbitrary. In this work, we decided that if a nuclide is only present in the left or the right material (but not both), then its concentration in the material where it is present is taken for use in the fictitious material. If an isotope exists in both the left and right material, then the average concentration was used. The question of the optimal choice of these concentrations is left for future work. We shall denote the isotope concentrations of this fictitious material as $N_k^*(\mathbf{r})$; correspondingly, the fictitious material has

a total cross section of

$$\Sigma_t^*(\mathbf{r}, E) = \sum_k N_k^*(\mathbf{r}) \sigma_{k,t}(E). \quad (33)$$

It is then possible to multiply and divide Eq. (32) by this fictitious concentration, and perform importance sampling. The collision kernel which should be used to sample the noise source is therefore

$$\frac{\Sigma_t^*(\mathbf{r}, E)}{\Sigma_t(\mathbf{r}, E)} \sum_k \frac{\delta N_k(\mathbf{r}, \omega)}{N_k^*(\mathbf{r})} \frac{N_k^*(\mathbf{r}) \sigma_{k,t}(E)}{\Sigma_t^*(\mathbf{r}, E)} \times \sum_m \frac{\sigma_{k,m}(E)}{\sigma_{k,t}(E)} \nu_{k,m}(E) f_{k,m}(E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}'), \quad (34)$$

m being a dummy index for the reaction channel. Now, the noise source sampling strategy is as follows: During power iteration, when a neutron undergoes a collision in the region of a vibration, a new fictitious material must be constructed. Nuclide k is sampled with probability $N_k^*(\mathbf{r}) \sigma_{k,t}(E) / \Sigma_t^*(\mathbf{r}, E)$, and will be used only for the sampling of the noise source particles. At this point, the sampling of the noise source is congruent to the approach used for cross section oscillations. We have effectively reduced the problem of sampling the vibration source to the problem of sampling an oscillating (space-dependent) source. If nuclide k is fissile, then fission noise particles must be sampled. Next, the noise particles born from scattering are sampled. A scattering channel for the nuclide can then be sampled nominally, along with the outgoing energy and direction. All the sampled noise source particles must have their weights multiplied by the complex factor

$$\frac{\Sigma_t^*(\mathbf{r}, E) \delta N_k(\mathbf{r}, \omega)}{\Sigma_t(\mathbf{r}, E) N_k^*(\mathbf{r})}. \quad (35)$$

Finally, a copy of the incident particle is also created and added to the noise source, with its weight multiplied by $\delta \Sigma_t(\mathbf{r}, E, \omega) / \Sigma_t(\mathbf{r}, E)$. The fictitious material is not actually needed for producing the copy noise particle, which can be handled in an identical manner to the approach taken earlier in Eq. (14).

It should be noted that for realistic problems, vibration regions could overlap. For the case of a fuel pin with cladding, there would effectively be two material interfaces (fuel/cladding and cladding/water): if the amplitude of the vibration is larger than the thickness of the cladding, the two vibration regions will overlap. This will have the effect of creating three effective regions: one where a fuel-cladding mixture is visible, one where a fuel-cladding-water mixture is visible, and another where a cladding-water mixture is visible. Since the effects of vibrations can be combined linearly, treatment of this situation is straightforward. One just needs to ensure that all nuclides of all vibrations at a given position are included in the fictitious material, and then sum all the contributions to $\delta N_k(\mathbf{r}, \omega)$ coming from the vibration of

each interface. In our implementation we chose to construct the fictitious material only when it is known what vibration regions are influencing the noise sampling at the given collision site. This avoids determining all possible region overlaps and material combinations in advance, which might be rather involved depending on the geometric form of each vibration.

IV. TRANSPORT OF NOISE PARTICLES

Once the noise source has been sampled as shown in the previous section, noise particles must be transported according to the stochastic rules defined by the operator $\mathcal{B}(\omega)$, in order to solve the fixed-source problem in Eq. (11). For both oscillations and mechanical vibrations, the structure of the noise source term is

$$-\delta\mathcal{B}(\omega)\varphi_c = -\sum_n \delta\mathcal{B}_n\varphi_c\delta(\omega - n\omega_0), \quad (36)$$

where $\delta\mathcal{B}_n = \delta\mathcal{B}(n\omega_0)$ is the perturbation operator evaluated at frequency $n\omega_0$. (The case of oscillation is special in that the source contains only a single frequency.) Correspondingly, we therefore expect the noise field to be of the form

$$\delta\varphi(\mathbf{r}, \boldsymbol{\Omega}, E, \omega) = \sum_n \delta\varphi_n(\mathbf{r}, \boldsymbol{\Omega}, E)\delta(\omega - n\omega_0), \quad (37)$$

where $\delta\varphi_n(\mathbf{r}, \boldsymbol{\Omega}, E)$ is the solution of the Fourier-transformed noise equation corresponding to the noise source component at frequency $n\omega_0$, namely

$$\mathcal{B}_n\delta\varphi_n = -\delta\mathcal{B}_n\varphi_c, \quad (38)$$

with $\mathcal{B}_n = \mathcal{B}(n\omega_0)$ the noise operator evaluated at the discrete frequencies $n\omega_0$ of the source. This yields an infinite system of fully decoupled linear equations for the noise components $\delta\varphi_n$. For negative frequencies where $n < 0$, we make use of the fact that $\delta\varphi_{-n} = \delta\varphi_n^\dagger$, the symbol \dagger indicating the complex conjugate. The null frequency at $n = 0$ represents the time-averaged effect on the static flux, as the perturbation will introduce a change of reactivity into the system, which is typically neglected in the orthodox linearization approach [16].

The operator $\delta\mathcal{B}$ is complex-valued, indicating that the noise source particles must therefore carry complex statistical weights, and each component can be positive or negative. The noise transport operator \mathcal{B} is also complex-valued, and differs from the standard Boltzmann operator in three places, which therefore alter the regular flight and collision sampling procedures. We first notice that the fission production must be divided by k_{eff} , which must be known before starting the noise simulation. The second modification is that delayed neutrons must have their weights multiplied by the complex yield factor $\lambda_j/(\lambda_j + i\omega)$, depending on the angular frequency ω and on the precursor family j . Both of these changes are minor, and straightforward to implement in most Monte

Carlo codes. More problematic is the fact that \mathcal{B} involves an effective total cross section which is complex:

$$\Sigma_t(\mathbf{r}, E) + i\frac{\omega}{v}. \quad (39)$$

Two different methods have been proposed to address this peculiarity. Yamamoto proposed to change the particle's weight continuously along the flight, using a complex exponential transform involving also the modification of the track-length estimator [20]. In this work, we use an alternative implementation proposed by Rouchon et al., which adds the real term

$$\eta\frac{\omega}{v}\delta\varphi \quad (40)$$

to both sides of Eq. (11), where η is real and has the same sign as ω [21]. Doing so effectively adds a new copy reaction with cross section $\Sigma_\omega(E) = \eta\omega/v$, having a complex yield $\nu_\omega = (\eta - i)/\eta$, and the effective total cross section becomes

$$\tilde{\Sigma}(\mathbf{r}, E) = \Sigma_t(\mathbf{r}, E) + \Sigma_\omega(E), \quad (41)$$

which is a real positive quantity. Flight distances of noise particles are now sampled using the effective total cross section $\tilde{\Sigma}$. At each collision, a copy of the noise particle is made with probability $\Sigma_\omega/\tilde{\Sigma}$, which has its complex weight multiplied by ν_ω . This is consistent with treatment of $\delta\Sigma_t$ given in Eq. (14). In general, Σ_t must be replaced by $\tilde{\Sigma}$ throughout the transport algorithms. Otherwise, transport algorithms are left unchanged. Variance reduction techniques such as implicit capture and roulette can still be used. In the method proposed by Rouchon et al., implicit capture is performed by multiplying the complex weight by $\Sigma_s/\tilde{\Sigma}$. For roulette, we follow the algorithm proposed by Yamamoto, where the real and imaginary weight components undergo roulette independently [20, 21]. The choice of $\eta = 1$ was used, as this has been previously demonstrated to yield good performance for most frequencies of interest for reactor physics [21].

Each component of the complex weight of noise particles can be positive or negative; correspondingly, the noise field will be estimated through the sum of these positive and negative contributions, which causes a very large variance in the scores. This phenomenon has been observed in cross section oscillation problems [30], but is exacerbated by the case of vibrating fuel pins [31]. On the one hand, as the noise source for cross section oscillations is constant within the vibration region, the particles of differing sign are only produced through delayed fission, which is a weak effect in most realistic systems. During noise transport, sign changes to particle weights can only occur through the copy channel, and delayed fission. Both of these reaction channels are typically weak, and this therefore leads to minimal appearances of particles of differing sign components. On the other hand, as is depicted in Sec. VI A, and previously shown by Zoia et al.

[16], the noise source of a vibrating fuel rod has a positive contribution on one side, and a negative contribution on the other, leading to nearly equal quantities of positive and negative noise source particles at the beginning of the problem. The resulting noise field is effectively the difference of two nearly equal stochastic quantities, which leads to catastrophic convergence issues for the statistical uncertainty. From these preliminary investigations, it is evident that the development of specialized variance reduction techniques is required if we want to obtain usable neutron noise results via Monte Carlo methods.

V. WEIGHT CANCELLATION FOR VARIANCE REDUCTION

Monte Carlo transport problems involving a mixture of positive and negative particles are not unique to neutron noise, and have appeared repeatedly in the literature for e.g. performing delta tracking without a majorant cross section [40, 41], for the calculation of the higher harmonics of k -eigenvalue problems [42, 43], and for the computation of critical buckling [44, 45]. Based on his pioneering work on higher harmonics and critical buckling, Yamamoto’s initial implementation for solving the noise equation by Monte Carlo methods utilized an approximate weight cancellation technique, as it was determined to be necessary when trying to solve the noise equation for frequencies outside of the “plateau region” $\lambda < \omega < \lambda + \beta_{\text{eff}}/\Lambda_{\text{eff}}$ [20]. When solving the noise equation for frequencies within the plateau region, weight cancellation was determined to be unnecessary, and was abandoned in Yamamoto’s later implementation done in MCNP4C, due to the complexity of incorporating such changes in a production-level Monte Carlo code [27]. Furthermore, Yamamoto’s initial implementation using weight cancellation was not able to estimate the variance of scores, and it was therefore impossible to determine what effect, if any, weight cancellation had on the estimation of the noise field, beside enabling the calculations to run in finite time [20]. The method developed by Rouchon et al. is able to estimate the variance of the noise field; previous results from simpler case-studies showed that convergence within the plateau region could be achieved without weight cancellation, which was therefore not implemented in the code [21]. Nonetheless, the convergence difficulties mentioned above for the case of vibrations has led us to begin an examination of the possible benefits of approximate or exact weight cancellation. Our recent work has demonstrated that the use of weight cancellation can provide a very large improvement in computational efficiency for the case of neutron noise induced by cross section oscillations, leading to an improvement by a factor of 90-100 on the figure of merit (FOM) when compared to the implementation in TRIPOLI-4[®] [36]. In this section, we show how weight cancellation can be performed on neutron noise problems for the case of vibrations, and illustrate the computational gains which

can be achieved through the use of the algorithm.

A. When to Perform Cancellation

Previous investigations in the context of weight cancellation for delta tracking methods have shown that it is most efficient to perform cancellation on the fission source [34]: Fission particles are unique in that their angular distribution is almost always considered to be perfectly isotropic, and their energy is almost independent of the incident energy. This effectively reduces the dimensionality of weight cancellation, which makes exact regional cancellation methods more efficient (in terms of the amount of weight being cancelled), and reduces the bias inherent to approximate cancellation methods [34, 35].

Most of our previous work on cancellation has been applied to eigenvalue problems solved by using power iteration, where one naturally has access to the fission source between subsequent fission generations, providing the ideal opportunity to apply the weight cancellation algorithm [34]. However, neutron noise equations are effectively fixed-source problems, where source particles are sampled and then distributed to all of the MPI processes participating in the simulation. Since fixed-source problems must always be subcritical, all simulated particles eventually die; once an MPI process receives its packet of particles to transport, it never needs to communicate with the other processes again, until it receives the next packet of particles for the subsequent batch (independent replica). Any fission particles born during a batch are typically added to the secondary particle bank, so that they can be transported later on. The MPI process will eventually transport all of the generated fission noise particles (because of sub-criticality), completing the batch. When using MPI, each process is identified by its rank, which is an index that is unique to that process. A depiction of the MPI communication scheme for this standard fixed-source algorithm is provided in Figure 2. Since all of the fission particles are just added to the bank of particles to be transported during the current batch, there is never a step in the algorithm where we have access to the entire fission source during the computation, contrary to what happens in power iteration schemes.

However, having access to the entire fission source is essential to maximizing the efficiency of weight cancellation; if not all of the particles participate, less weight will be canceled. We have therefore implemented a different fixed-source iteration algorithm for noise problems, in order to gain access to the entire fission source. The scheme presented here was inspired by the decomposition proposed by Yamamoto, although differing in implementation [20, 27]. For a single batch, the initial source particles are sampled, and then sent to all the available MPI processes. Each process will transport all of the assigned initial particles, and will store all of the produced fission particles in a separate bank. Once it has transported the

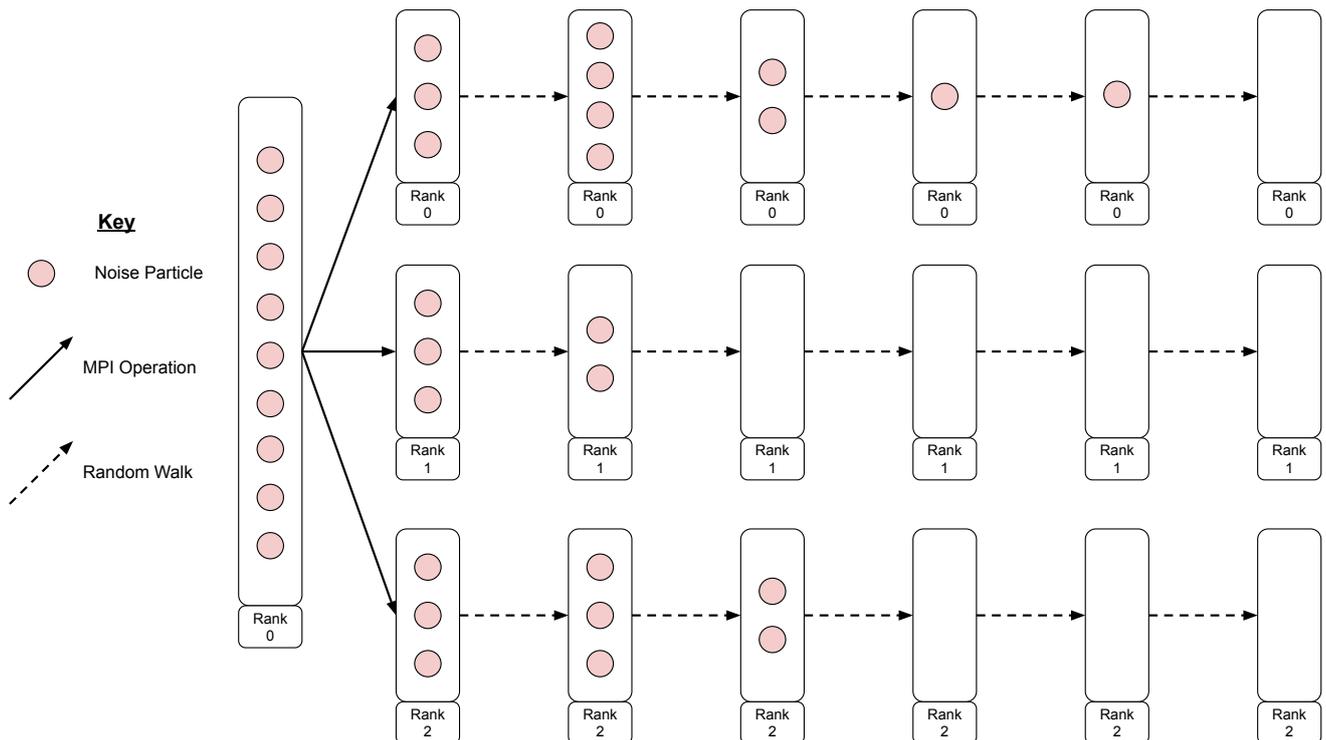


FIG. 2. Standard MPI communication scheme for a single fixed-source batch.

initial assigned packet of particles, each process will send its bank of fission particles back to the master process. Now, with all of the fission particles on the master process, it is possible to perform weight cancellation on the entire fission source. Once the cancellation operation is complete, the new fission particles are redistributed to all of the MPI processes, where they will be transported. Any resulting fission particles will again be stored in a separate bank; once each process has finished, the fission particles are again sent to the master process, where cancellation can occur again. This practically breaks the fixed-source batch into “inner generations”, as depicted in Fig. 3. This MPI communication scheme is nearly identical to that used in standard power iteration implementations, with the exception that all of the particles will eventually die out, signaling the end of the fixed-source batch.

B. Approximate Cancellation

In this work, we will consider both exact and approximate weight cancellation methods. We will first examine approximate cancellation, based on the technique initially proposed by Zhang et al., which is conceptually simple, easy to implement in a Monte Carlo code, and results in very good cancellation efficiency [46]. For this purpose, a rectilinear mesh is imposed on top of the prob-

lem domain, and the particles which will undergo cancellation are then sorted into this mesh, based on their position (and possibly energy, if desired). The average weight of all particles in each mesh element is calculated, and this average weight is then assigned to each of the particles in the element. Summing the particle weights to compute the average is what effectively leads to cancellation, as the total weight will be the sum of positive and negative components. While this approach imposes a bias on the resulting fission source, the bias can be made minimal by taking a sufficiently fine mesh. However, refining the mesh is a double-edged sword: the bias will be reduced, but the efficiency of cancellation will be reduced as well, as there will be fewer particles present in each mesh element. Despite this, our previous work has shown approximate cancellation to be much more efficient at cancelling weight than the exact version (treated in Sec. V C), and we have not observed any measurable bias for a reasonably refined mesh [35].

C. Exact Cancellation

Exact regional cancellation was originally proposed by Booth and Gubernatis, for use in calculating the higher harmonics of k -eigenvalue problems [47]. The initial formulation was limited to one-dimensional configurations, but we have previously expanded this formalism to three-

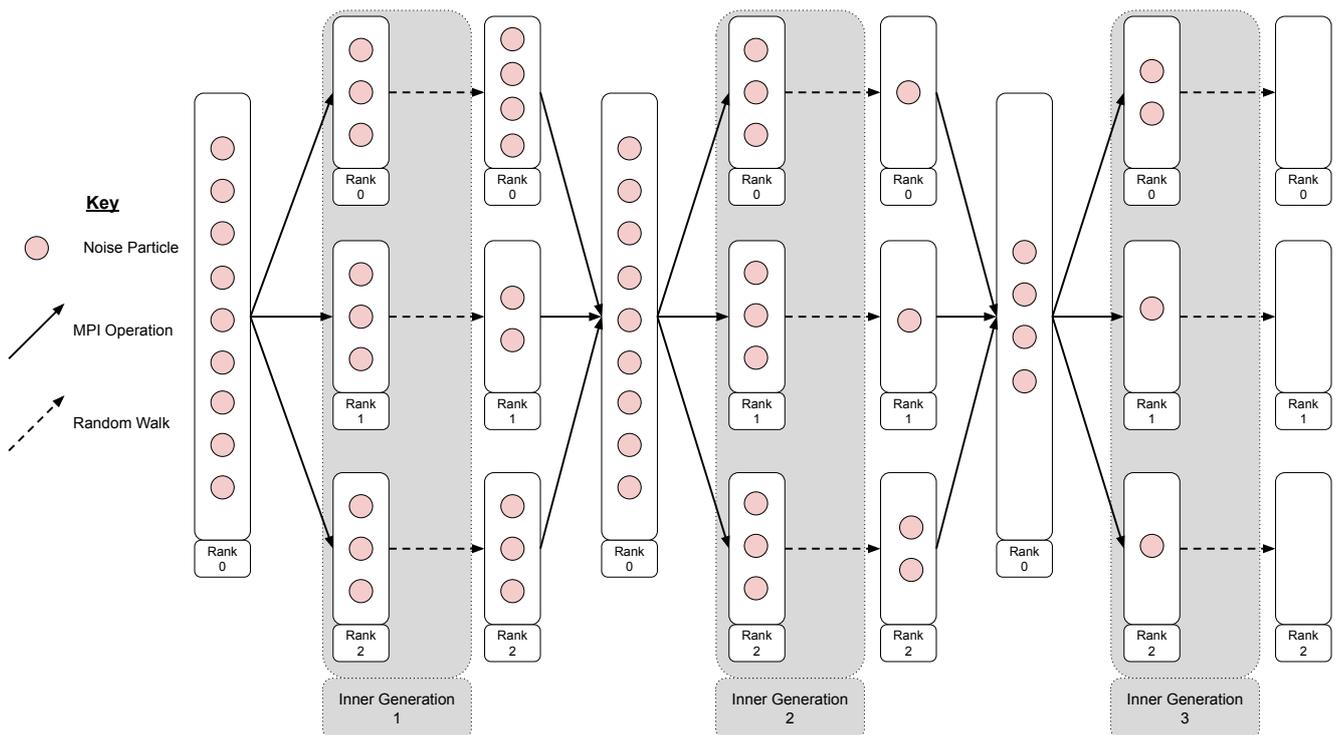


FIG. 3. Proposed MPI communication scheme where a single fixed-source batch is broken into inner fission generations. While only three inner generations are depicted here to demonstrate the concept, there will generally be hundreds or thousands of inner generations per noise batch.

dimensional multi-group settings. This cancellation technique is applied on the fission source, and in practice can be used in combination with delta tracking [48] or negative-weighted delta tracking [40, 41] transport methods: we have provided a successful numerical demonstration of exact regional cancellation for a modified version of the C5G7 reactor physics benchmark [34]. Fission particles are sorted into geometric volumes, referred to as cancellation regions. Our previous work used a regular rectilinear mesh, with each mesh element acting as a cancellation region. Fission particles must store the position of their parent's previous collision (\mathbf{r}'), their parent's energy (E'), and their parent's penultimate direction ($\hat{\Omega}''$), in addition to their own phase space coordinates ($\mathbf{r}, \hat{\Omega}, E$). From this information, the fission density function can be calculated, which for the case of delta tracking reads

$$\zeta(\mathbf{r}|\mathbf{r}', \hat{\Omega}', E') = \frac{\mathcal{P}\left(\frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|} \cdot \hat{\Omega}'\right) \Sigma_f(\mathbf{r}, E')}{2\pi|\mathbf{r}-\mathbf{r}'|^2} e^{-\Sigma_{\text{maj}}(E')|\mathbf{r}-\mathbf{r}'|}, \quad (42)$$

where \mathcal{P} is the probability density function for the cosine of the scattering angle between directions $\hat{\Omega}'$ and $\frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|}$, and $\Sigma_{\text{maj}}(E')$ is the majorant cross section used in the delta tracking algorithm [34, 48, 49]. Using this fission

density function, each fission particle i (having a weight w_i) is split into two portions: a point-wise part with weight

$$w_{p,i} = w_i \frac{\zeta(\mathbf{r}|\mathbf{r}', \hat{\Omega}'', E') - \beta_i}{\zeta(\mathbf{r}|\mathbf{r}', \hat{\Omega}'', E')}, \quad (43)$$

and a uniform part with weight

$$w_{u,i} = w_i \frac{\beta_i}{\zeta(\mathbf{r}|\mathbf{r}', \hat{\Omega}'', E')}. \quad (44)$$

Here β_i is a cancellation parameter, which can take any value, so long as it does not depend on $(\mathbf{r}, \hat{\Omega}, E)$ [34, 47, 50]. The point-wise portion represents the piece of the fission particle which must be placed exactly at \mathbf{r} , and the uniform portion represents the piece which can be uniformly distributed across the entire cancellation region. A thorough discussion on how to determine an appropriate value of β_i , which ultimately affects the cancellation efficiency, is provided in Refs. 34 and 50. For each cancellation region \mathcal{R} , we iterate over all of the particles in the region, summing their uniform parts

$$U_{\mathcal{R}} = \sum_{i \in \mathcal{R}} w_{u,i}. \quad (45)$$

We then set all of the fission particle weights to be equal to their point-wise portion ($w_i := w_{p,i}$). Lastly, we must

sample fission particles with a total weight equal to the remaining weight $U_{\mathcal{R}}$, which will be uniformly distributed within \mathcal{R} . The exact cancellation algorithm can be applied as is to neutron noise problems; the only required modification concerns the number of uniform particles to be added to each region, which for particles carrying real and imaginary statistical weights reads

$$n = \lceil \max(|\operatorname{Re}\{U_{\mathcal{R}}\}|, |\operatorname{Im}\{U_{\mathcal{R}}\}|) \rceil, \quad (46)$$

each with a weight of $w = U_{\mathcal{R}}/n$.

Compared to approximate cancellation, exact regional cancellation is much more difficult to implement in a Monte Carlo code. Additionally, it results in much less total weight cancellation as well, which can lead to only minimal performance improvements in problems which require cancellation [34, 35, 50]. However, the exact regional method aligns with the Monte Carlo philosophy, in that no approximation nor bias is introduced in the sampling scheme.

VI. SIMULATION RESULTS

The Monte Carlo noise equation solution method devised by Rouchon et al. was implemented in the development version of TRIPOLI-4[®] [21, 39]. Since TRIPOLI-4[®] is a mature general-purpose code with about 400 kSLOC (source lines of code), implementing all of the changes necessary to perform the exact noise source sampling technique outlined in Sec. III B 2, or the cancellation methods described in Sec. V, would require an exceptionally large rewriting effort. Therefore, all of the methods described in the paper have been implemented in a multi-group Monte Carlo mini-app called MGMC, with approximately 13 kSLOC and thus allowing for quick implementation and testing of new transport algorithms [51]. MGMC is able to perform transport in general 3D geometries, composed of surface-based volumes, universes, and lattices. Shared-memory parallelism is implemented via OpenMP, and distributed-memory parallelism is implemented via MPI. Basic quantities such as flux and reaction rates can be scored using either track-length or collision estimators, across regular rectilinear meshes, and are written to binary NumPy files, for easy data processing in Python [52]. These features make MGMC entirely representative of a larger production-level code; therefore, it can be assumed that any improvement in calculation efficiency observed in MGMC would be similar to the improvements one could expect from implementing the same methods in a production-level code, such as TRIPOLI-4[®]. Delta tracking and weight cancellation have previously been added to MGMC, along with neutron noise transport [34, 36]. Newly implemented for this work was the sampling the noise source corresponding to mechanical vibrations. MGMC has been released as free software, and is available under the CeCILL-v2.1 license [51].

In order to illustrate the novel noise source sampling and the weight cancellation methods implemented in MGMC, in this work we have selected a benchmark problem that has been previously used to compare the results of several different neutron noise solvers [30, 31]. It is a 2D, 2 group problem, in the form of a reflected 17x17 fuel assembly, with square pins. A diagram of this system is presented in Fig. 4. The assembly pitch is 1.26 cm, with fuel pins having a side length of 0.7314 cm. A 0.08 cm water blade surrounds the entire assembly. Reflective boundary conditions are assumed for the four outer edges of the problem domain. The neutron noise is induced by a vibrating fuel pin (marked in pink in Fig. 4), subject to a sinusoidal displacement along the x-axis, with an angular frequency of $\omega_0 = 2\pi \text{ rad s}^{-1}$ and an amplitude of $\varepsilon = 0.2 \text{ cm}$, as per benchmark specifications. The two vibrating interfaces are located at $x = -2.8857 \text{ cm}$ and $x = -2.1543 \text{ cm}$.

Since exact regional weight cancellation can only be performed with delta tracking, all simulations were run with this transport method, to ensure fairness in the comparison of simulation runtimes. All simulations began with 10^6 particles in power iteration, distributed uniformly within the assembly, all in the first group. The first 13 generations were discarded to allow for source convergence. Between each noise batch, three extra power iteration inactive generations were performed, to ensure proper decorrelation of the noise source between replicas. Simulations were set to run for either 10^4 noise batches, or a maximum run time of 2 days (whichever came first). Simulations were run on a computing cluster at CEA, each run using 16 MPI processes, and 32 OpenMP threads per MPI process. The noise field was estimated with a track-length estimator in all simulations. The real and imaginary components of the field are scored in separate tallies. Breaking the scoring of complex quantities into a real and imaginary component facilitates the addition of neutron noise simulations to existing codes.

For the purpose of code-to-code comparison and verification, our Monte Carlo simulations were compared to the results generated by the deterministic noise solver previously implemented in APOLLO3[®] [53, 54]. The noise equation solver has been added to the Integro-Differential Transport (IDT) lattice solver of APOLLO3[®], which uses the method of short characteristics in conjunction with the discrete ordinates method [54]. To accomplish this, the standard iteration loops are applied to the complex fission source and scattering source, with the addition of an iteration loop between the real and imaginary components. Thus, the standard one-group transport solver methods can be used, and one can consequently benefit from all numerical methods already implemented in APOLLO3[®] [55]. To compare the noise source and noise field between codes, these quantities were normalized by the estimated value of the static flux at the center of the assembly in group 2, for the respective code.

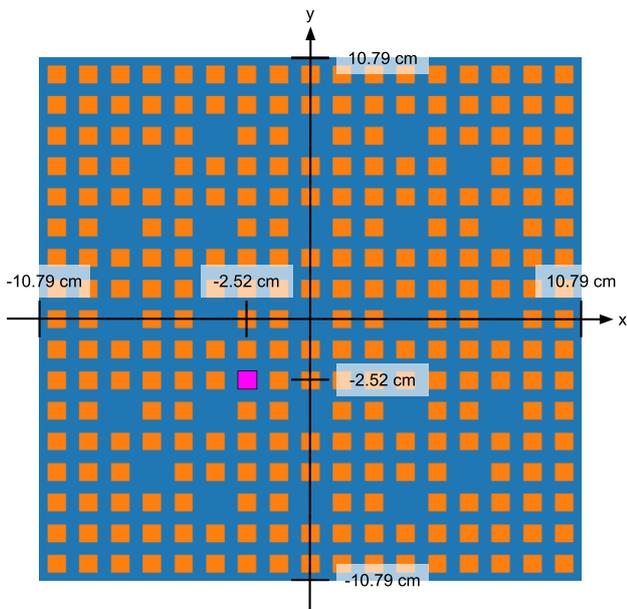


FIG. 4. Depiction of the 2D reflected assembly for the neutron noise benchmark problem proposed in [30]. The pink fuel pin (located two cells down and two cells left from the center) experiences a small sinusoidal vibration to the left and right. All other fuel pins are unperturbed.

A. Sampling of the Noise Source

We first compare the first harmonic of the noise source, sampled with the approximate method and the exact method, against the noise source obtained from APOLLO3[®]. By default, the deterministic noise solver of APOLLO3[®] computes the harmonics of the exact noise source (Sec. III B 2). The real and imaginary components of the source are shown as a function of the x coordinate at $y = -2.52$ cm, in Fig. 5. The real component of group 2 is zero, as this portion of the source only comes from delayed neutrons, which according to the benchmark model are all born in group 1. Immediately apparent is the fact that the amplitude of the source from the approximate method is much smaller than the true source, for the imaginary component. The approximate source sampling also has an asymmetry, where the source has a slightly larger amplitude in the fuel pin than in the water. Both of these effects are primarily due to the approximate source sampling method ignoring the difference in the scattering law in the noise source. This is evident upon examination of the real component in group 1, which is purely due to delayed fission: very good agreement is observed between the approximate method and APOLLO3[®]. Since the real component (which is only due to fission) appears to have a shape and amplitude which agree with the exact source from APOLLO3[®], it can be inferred that the discrepancy in the imaginary component is primarily due to the inadequate treat-

ment of the scattering laws in the approximate sampling method. The agreement observed in the real component also indicates that making a copy of the fission noise source particles in the fuel and moving them into the water is an adequate approximation for this system, where the flux is relatively constant over the perturbed region. Other numerical investigations concerning systems with larger flux gradients (such as the one-dimensional rod model investigated in [16]) show that this approximation might become inappropriate and lead to large differences in the resulting noise source.

The exact noise source sampling method has excellent agreement with the APOLLO3[®] results, for all groups and components. In the imaginary components shown in Fig. 5, the exact source has the correct amplitude and shape when compared to APOLLO3[®]. By using the same fictitious material to sample the noise source on both sides of the vibration interface, it is possible to eliminate the discontinuity which was observed on in the approximate method. This demonstrates that our proposed exact source sampling method is not just correctly treating the perturbation in the macroscopic cross sections, but also in the scattering laws.

Next, we look at the second harmonic of the noise source. Here, group 2 of the imaginary component is zero, and group 1 of the imaginary component comes from delayed fission, as shown in Fig. 6. Most of the noise source is now in groups 1 and 2 of the real component. Again, we observe that the proposed exact noise source sampling technique is in excellent agreement with the source calculated by APOLLO3[®]. The noise source sampled using the approximate method agrees quite well with the APOLLO3[®] source and the exact sampling in the imaginary component, which is only due to delayed fission. Examining the real component, it is again observed that the amplitude of the source using the approximate method is smaller than the one obtained with the exact method. The asymmetry which was observed in the first harmonic is less visible in the second harmonic. In the first harmonic this asymmetry was visible exactly at the material interface. For the second harmonic, the amplitude of the real component of the first group is slightly smaller in the water than in the fuel.

As the exact noise source sampling technique has now been verified, and demonstrated to have far superior agreement with APOLLO3[®], only the exact noise source sampling method is used for the remainder of the paper.

B. Application of Weight Cancellation

Next, the effects of weight cancellation on the performance of noise simulations is examined. Three different cancellation methods were used: approximate cancellation with a coarse mesh, approximate cancellation with a fine mesh, and exact cancellation. A 170×170 regular rectilinear mesh was used for both the approximate

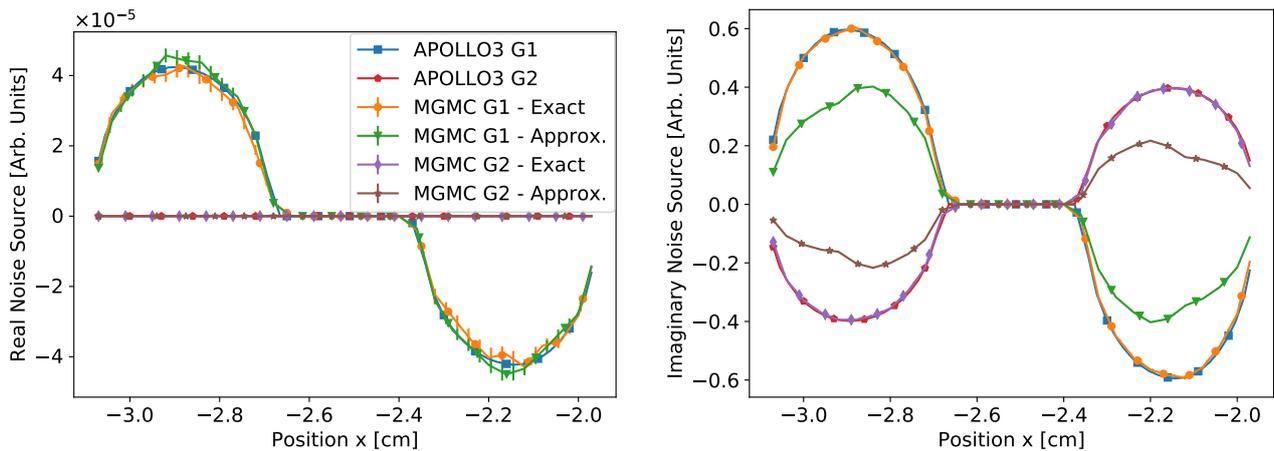


FIG. 5. Real and imaginary components of the noise source through the center of the vibrating fuel pin at $y = -2.52$ cm, for the first harmonic.

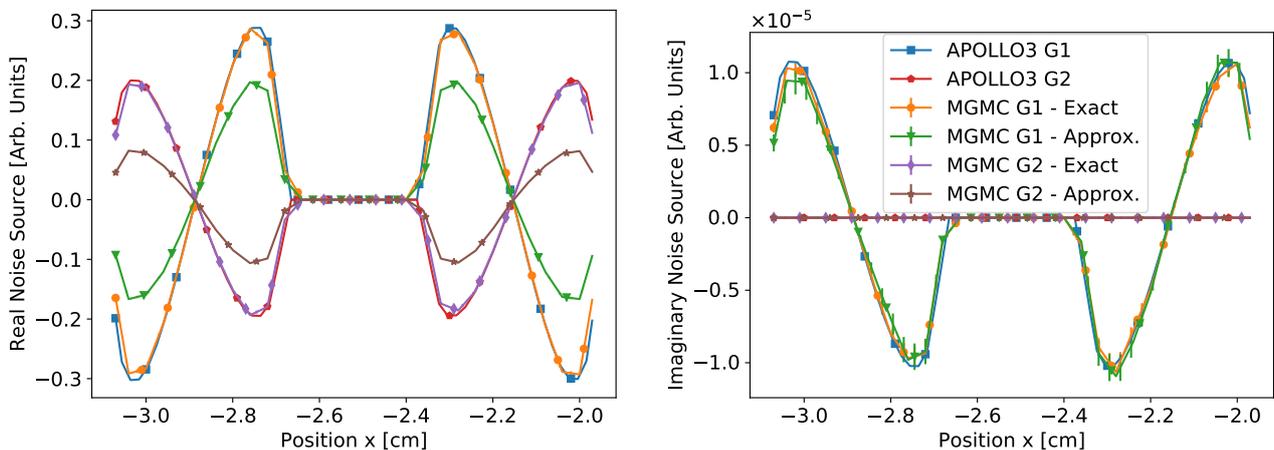


FIG. 6. Real and imaginary components of the noise source through the center of the vibrating fuel pin at $y = -2.52$ cm, for the second harmonic.

coarse and exact cancellation methods. For approximate cancellation with a fine mesh, a 340×340 rectilinear mesh was used instead. A simulation was also run without cancellation, and without inner generations (essentially the algorithm used in TRIPOLI-4[®]), which served as a baseline for all comparisons. The resulting noise field was scored using the track-length estimator over a 138×138 regular rectilinear mesh. To compare the efficiency of each simulation method, we have used the average ratio of the figure of merit (FOM) for results obtained using cancellation to results obtained using the baseline algorithm. The FOM is defined as

$$\text{FOM} = \frac{1}{T\sigma^2}, \quad (47)$$

where T is the wall-clock run time, and σ^2 is the variance of the quantity being examined. Cancellation is only ap-

plied to the noise particles, and can therefore only change the amount of time spent transporting noise particles. It will have no effect on the time spent performing power iteration to sample the noise particles to begin each noise replica. Because of this, it can be argued that the FOM should only be calculated using the time spent transporting noise particles. We have chosen to look at the FOM for both the total run time (time spent during power iteration, noise transport, and cancellation), and the noise run time (time spent during noise transport and cancellation), as we think both quantities are of interest. For each run time, the ratio for the FOM was computed in each element of the noise field scoring mesh, and then the average of that ratio was calculated independently for each component (real or imaginary) of the noise field and each energy group. As a result, for each cancellation method, 8 FOM ratios are provided, 4 calculated with

the total run time, and 4 calculated with the noise run time.

1. Analysis of the First Harmonic

For the first harmonic of the original benchmark problem, a slice of the noise field at the axis through the middle of the perturbed fuel pin is provided in Fig. 7. These plots compare all four Monte Carlo simulations to the deterministic results obtained with APOLLO3[®]. Immediately, one is drawn to the results from the baseline algorithm where no cancellation was used. Not only are the error bars quite large for both components and both groups, but it also appears as though there was a bias in the result, since this noise field systematically has a higher value than in all other simulations. This is most noticeable in the real component, and at regions far away from the perturbed pin in the imaginary component. Running the same simulation parameters with a different seed for the random number generation indicated that this behavior is not systematic, and that sometimes a similar phenomenon is observed, but with a systematically lower value than the deterministic results. This seems to suggest that the Monte Carlo results from the baseline algorithm presented here are far from converged, and would require a much larger number of noise batches to generate better statistics. Unfortunately, this was not feasible given the poor performance of this algorithm. It could also be the case that the error bars are under-estimated because of the correlations induced by power iteration; to test this hypothesis, further investigations with an increased number of decorrelation cycles would be needed. Looking at the noise field estimated with exact regional cancellation, there is certainly an observable improvement in the resulting variance, and the systematic bias observed without cancellation has been reduced. For the imaginary component, the error bars are now overlapping the deterministic results at some positions. Despite the disagreements observed far from the pin, very decent agreement is observed in the vicinity of the perturbation, where the amplitude of the noise field is largest.

The two approximate cancellation methods had the best agreement with the deterministic results. While there were always visible differences in the real component, there was always agreement within the error bars. For the imaginary component, both approximate methods had nearly perfect agreement with APOLLO3[®], although the fine mesh variant did have slightly better agreement. These small differences could be due to the approximate nature of the cancellation technique, as refining the cancellation mesh should reduce the bias in the results. Based on the error bars and the level of agreement observed between the two approximate methods and the deterministic results, it is difficult to conclude if the slight disagreement in the coarse mesh results are actually due to the bias induced by approximate cancel-

lation. At the observed level of agreement, one must also call into question the accuracy of the deterministic results, which have a bias induced by the geometric and angular discretization.

The improvements in the FOM for the simulations using cancellation, when compared to the baseline algorithm without cancellation, are presented in Tab. I. The approximate cancellation with the coarse mesh had the best improvement in performance. Looking at the total run time, the real component was improved by a factor of about 300 and the imaginary component by a factor of about 230. If the noise run time is used, these become factors of 1620 and 1250 respectively. Approximate cancellation with the fine mesh gave an improvement of approximately 120 when looking at the total run time, and approximately 480 when looking at the noise run time. The performance gains when using the fine mesh might have been smaller due to less weight being cancelled with a finer mesh, as a finer mesh leads to fewer noise particles per mesh element. This performance penalty leads to a theoretical reduction in any bias in the results, although we were unfortunately not able to measure this phenomenon. Exact regional cancellation leads to the lowest FOM improvements, with a factor of 8 observed when using the total run time, and a factor of 14 for the noise run time.

A word of caution must be added to the FOM ratios provided in Tab. I. All of these ratios used the baseline algorithm results as a denominator. Since the FOM only considers the variance of the examined score, these reported factors do not consider the fact that the average values obtained with the baseline algorithm seemed to have a systematic bias. There is no good way to quantitatively measure the improvements in the average that cancellation provided, as demonstrated in Fig. 7. As previously mentioned, this apparent bias could also indicate that the error bars from the results obtained with the baseline algorithm were underestimated, and the performance improvements might actually be even larger; to support this statement, one would need to perform independent replicas and estimate the “true” error bars. Additionally, the observed improvements in the FOM are not solely due to the application of weight cancellation. It was observed that breaking the fixed-source problem into inner generations, without applying any weight cancellation between inner generations, reduced the simulation time when compared to the baseline method. This phenomenon will be the subject of future work.

2. Analysis of the Second Harmonic

For the sake of completeness, we have then examined how our proposed approach behaves when applied to the calculation of the second harmonic of the noise field. We chose to use the two different approximate cancellation methods and the baseline method to calculate the second harmonic, to evaluate the performance gains pro-

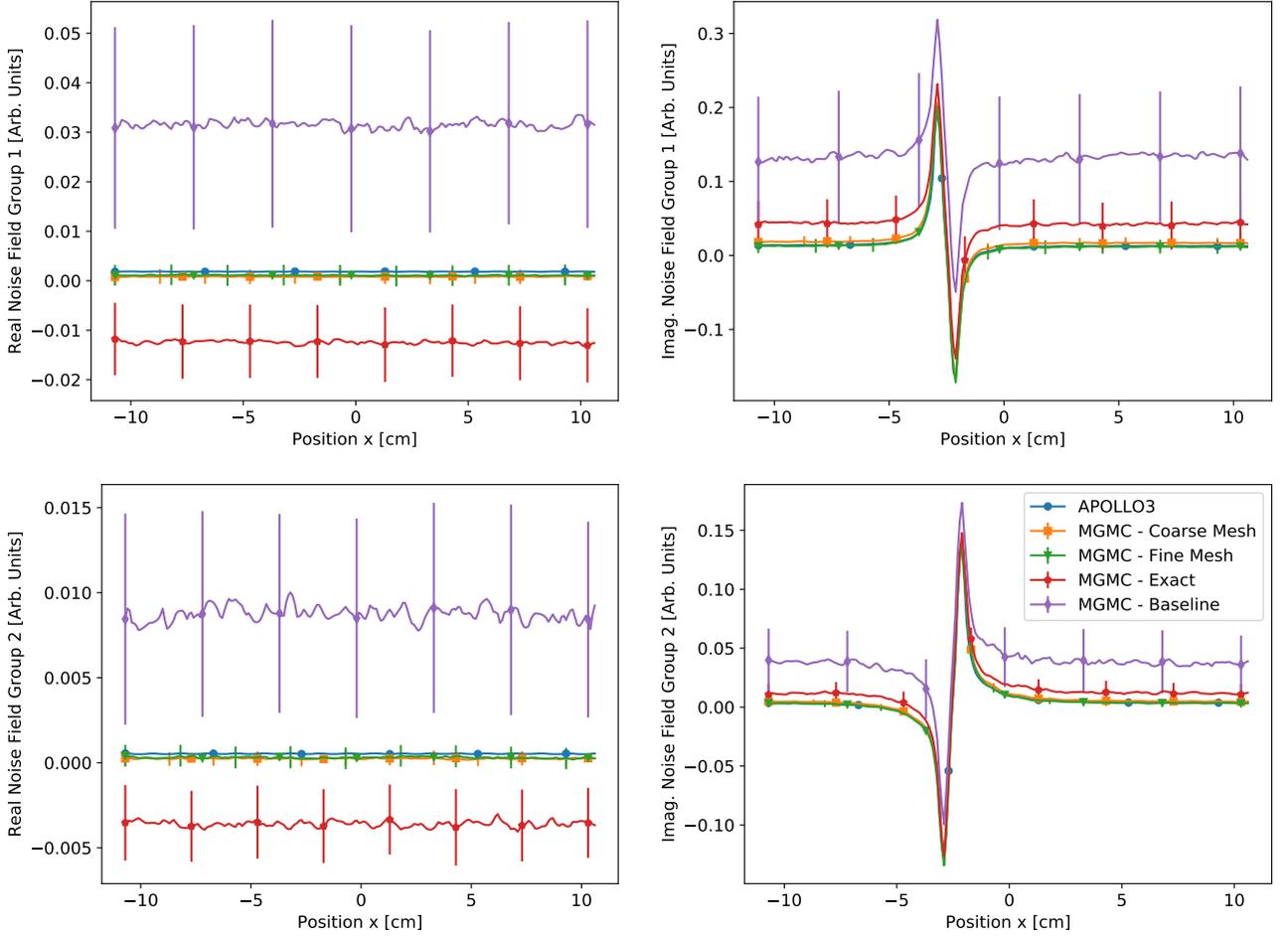


FIG. 7. First harmonic of the noise field running through the center of the vibrating fuel pin at $y = -2.52$ cm, obtained with APOLLO3[®] and MGMC using different weight cancellation methods (or no weight cancellation).

		Real		Imaginary	
		Group 1	Group 2	Group 1	Group 2
Approximate Coarse	Total Run Time	304	304	234	235
	Noise Run Time	1623	1624	1252	1254
Approximate Fine	Total Run Time	129	129	122	122
	Noise Run Time	487	487	461	461
Exact	Total Run Time	8	8	8	8
	Noise Run Time	14	14	14	14

TABLE I. First harmonic improvement factors for the FOM of given weight cancellation methods when compared to the baseline solution strategy which does not use weight cancellation.

vided by weight cancellation. It was chosen to not use exact weight cancellation here, as its performance was significantly poorer than the two approximate methods. The resulting noise fields are depicted in Fig. 8. While the amplitude of the second harmonic in the vicinity of the vibration appears to be smaller than the amplitude observed in the first harmonic, this is not the case at

positions farther from the perturbation. Far to the left or right of the vibration, the amplitude of the second harmonic is much larger than that of the first, which is coherent with previous findings by Zoia et al. [16]. This might happen because of two possibly concurrent reasons: specific symmetries of the system could suppress the first harmonic and thus promote the second; more-

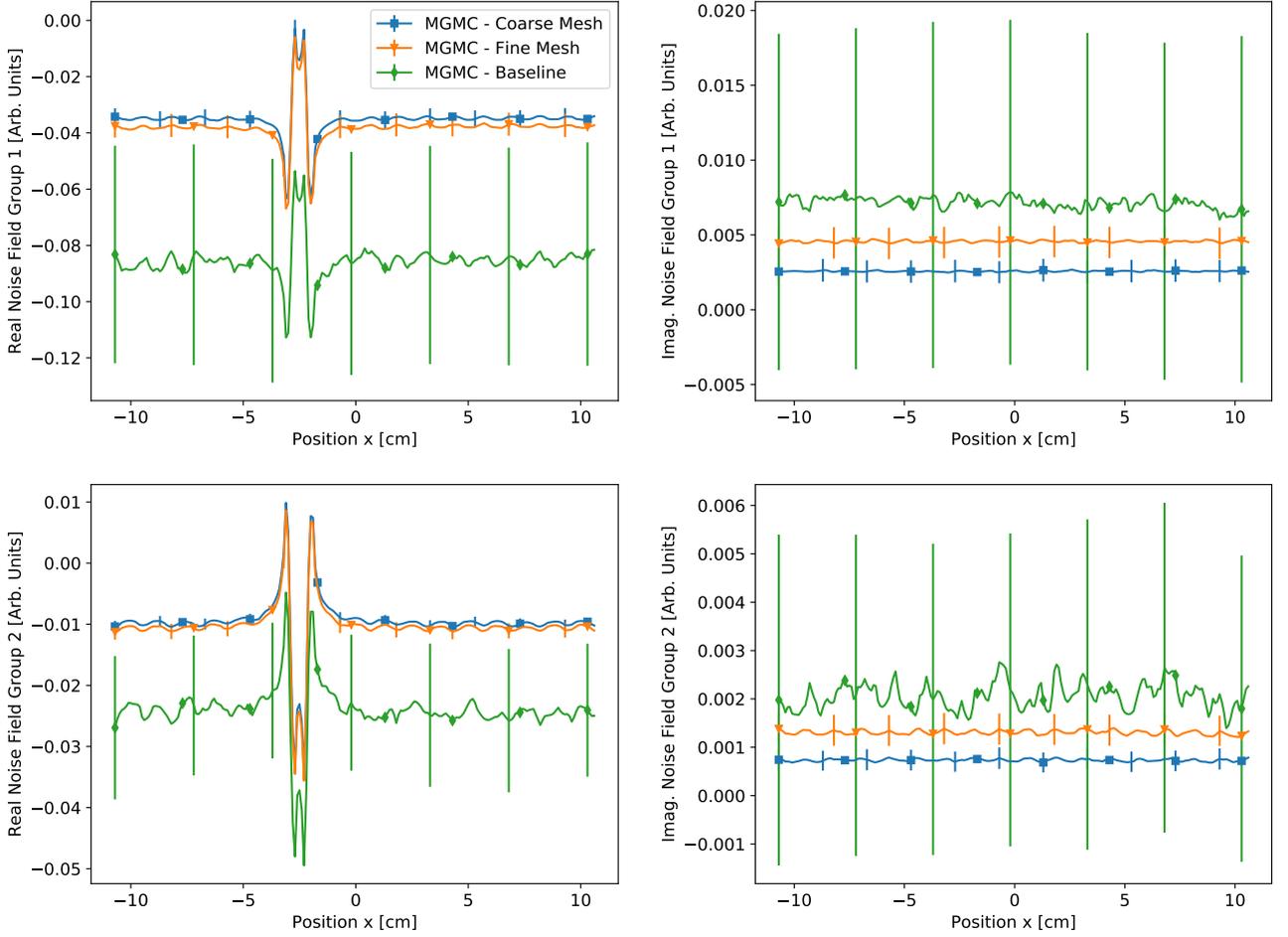


FIG. 8. Second harmonic of the noise field running through the center of the vibrating fuel pin at $y = -2.52$ cm, obtained with MGMC using different approximate weight cancellation meshes, or without weight cancellation.

		Real		Imaginary	
		Group 1	Group 2	Group 1	Group 2
Approximate Coarse	Total Run Time	244	244	344	344
	Noise Run Time	1322	1323	1864	1864
Approximate Fine	Total Run Time	119	119	152	151
	Noise Run Time	446	446	570	569

TABLE II. Second harmonic improvement factors for the FOM of given weight cancellation methods when compared to the baseline solution strategy which does not use weight cancellation.

over, it has been shown that the orthodox linearization of the noise equations can nonphysically amplify the second harmonic (with respect to the exact solution of the exact noise equations) while leaving the first harmonic almost unaffected. Examining the real component, it then appears as though the results from the baseline method demonstrate a large bias, and do not agree with the two other results. The estimated value of the imaginary component has much better agreement with the values of the

solutions which used approximate cancellation, but there is more than 100% relative error.

Focusing on the two approximate weight cancellation methods, both solutions are in excellent agreement in the immediate vicinity of the vibrating pin in the real component. Moving farther away from the pin, the results obtained with the fine cancellation mesh have a slightly lower value than those obtained with the coarse mesh. It is possible that this difference is due to the finer mesh im-

posing less of a bias when compared to the coarse mesh. Both solutions agree within one standard deviation however, and the results would require further convergence to confirm this hypothesis. In both the real and imaginary components, the structure of the static flux is visible, which was not the case for the first harmonic results, which is coherent with the expected behaviour of local and global components of the noise field (see e.g. [3] and references therein).

Table II displays the FOM ratios for the two approximate cancellation methods, when compared to the baseline method. In general, the performance improvements are very similar to the values seen with the first harmonic. In the first harmonic, the imaginary component of the noise field was dominant, while the real component is dominant for the second harmonic. Comparing the FOM ratio of the coarse mesh for the imaginary component of the first harmonic with the real component of the first harmonic, we see that the coarse mesh cancellation method was actually slightly more effective at estimating the noise field for the dominant component of the second harmonic than of the first. This is also true for the non-dominant component with the coarse mesh as well. Strangely, this is not true for the fine cancellation mesh, as this method had slightly poorer performance on the dominant component of the second harmonic, than on the first.

C. The Effects of Vibration Frequency and Amplitude

Given the remarkable improvements in performance that weight cancellation has enabled, it is now possible to envision the use of Monte Carlo noise simulations to perform reactor analysis and assess the impact of specific parameters on the noise field. In this section, examine the first harmonic for two altered versions of the benchmark: one where the angular frequency of the vibration of the fuel pin has been increased from $\omega_0 = 2\pi \text{ rad s}^{-1}$ to $\omega_0 = 4\pi \text{ rad s}^{-1}$ (without changing the amplitude), and a second one where the vibration amplitude has been increased from $\varepsilon = 0.2 \text{ cm}$ to $\varepsilon = 0.4 \text{ cm}$ (without changing the frequency). The resulting noise amplitude and phase for these two problems (in addition to modulus and phase of the original benchmark parameters) are presented in Fig. 9. Approximate cancellation with the fine mesh was used to obtain these results. In the previous sections, we examined the complex noise field $\delta\varphi$ because this is the quantity which is estimated in the Monte Carlo simulation, and the associated error bars are available, permitting a comparison in performance with the FOM. Reactor physicists, however, are typically more interested in the modulus ($|\delta\varphi|$) and phase ($\text{atan2}(\text{Im}\{\delta\varphi\}, \text{Re}\{\delta\varphi\})$), which can be calculated from the complex field. Despite having error bars for the complex field, it was not possible to exhibit error bars for the modulus and phase, as the covariance between the real and imaginary part was

not scored. Due to this limitation, no error bars are given in Fig. 9.

First, we will consider the effects of the proposed variations to the vibration parameters on the noise modulus. From Fig. 9, we first note that in the immediate vicinity of the perturbation, changing only the frequency seems to have little impact on the modulus. Increasing the amplitude of the vibration however, leads to an increase in magnitude and width of the peaks. One would certainly expect a widening of the peak in this case, as the area of the perturbation has been increased. With a larger perturbation region, more noise source particles will be sampled in the vicinity of each interface, contributing to the larger magnitude. Farther away from the perturbed pin, we see that both the $\omega_0 = 4\pi \text{ rad s}^{-1}$ and the $\varepsilon = 0.4 \text{ cm}$ cases have a slightly larger amplitude, although this effect is slightly less apparent in group 2. These results could indicate that for this system the global component of the noise is more sensitive to the frequency of the vibration, while the local component is more sensitive to the amplitude of the vibration, in agreement with the spectral analysis previously performed by Rouchon [53].

Examining the noise phase, it appears as though augmenting the perturbation (either in frequency or amplitude) has little effect on the depth of the observed well. Changing the vibration parameters did seem to change the width of the phase well in both cases, and this effect is apparent in both energy groups. The increased frequency reduced the well width, while the increased amplitude enlarged it. Curiously, the observed changes in the well width only ever occur on $+x$ side of the well in the first group, and on the $-x$ side of the well for the second group. This asymmetric behavior might be due to the water hole located near the bottom left corner of the vibrating pin. Examining a problem with a more symmetric configuration could therefore be of interest in future investigations.

VII. CONCLUSIONS

In this work, we have considerably extended our previous findings concerning the use of Monte Carlo methods for neutron noise simulations. The contributions presented in this paper are twofold. First, we have proposed a novel method to sample in an exact manner the noise source particles produced from mechanical vibrations within a system. Contrary to the previous implementation in TRIPOLI-4[®], which for the sake of simplicity utilized many approximations, we have shown that the noise source due to mechanical vibrations can be represented without any approximation by mapping it to a perturbation of the isotopic concentrations. For this purpose, a fictitious material must be defined, containing the union of the nuclides on either side of the vibrating interface. A nuclide is then randomly chosen from this fictitious material, which allows sampling the noise source particles in a manner which is nearly identi-

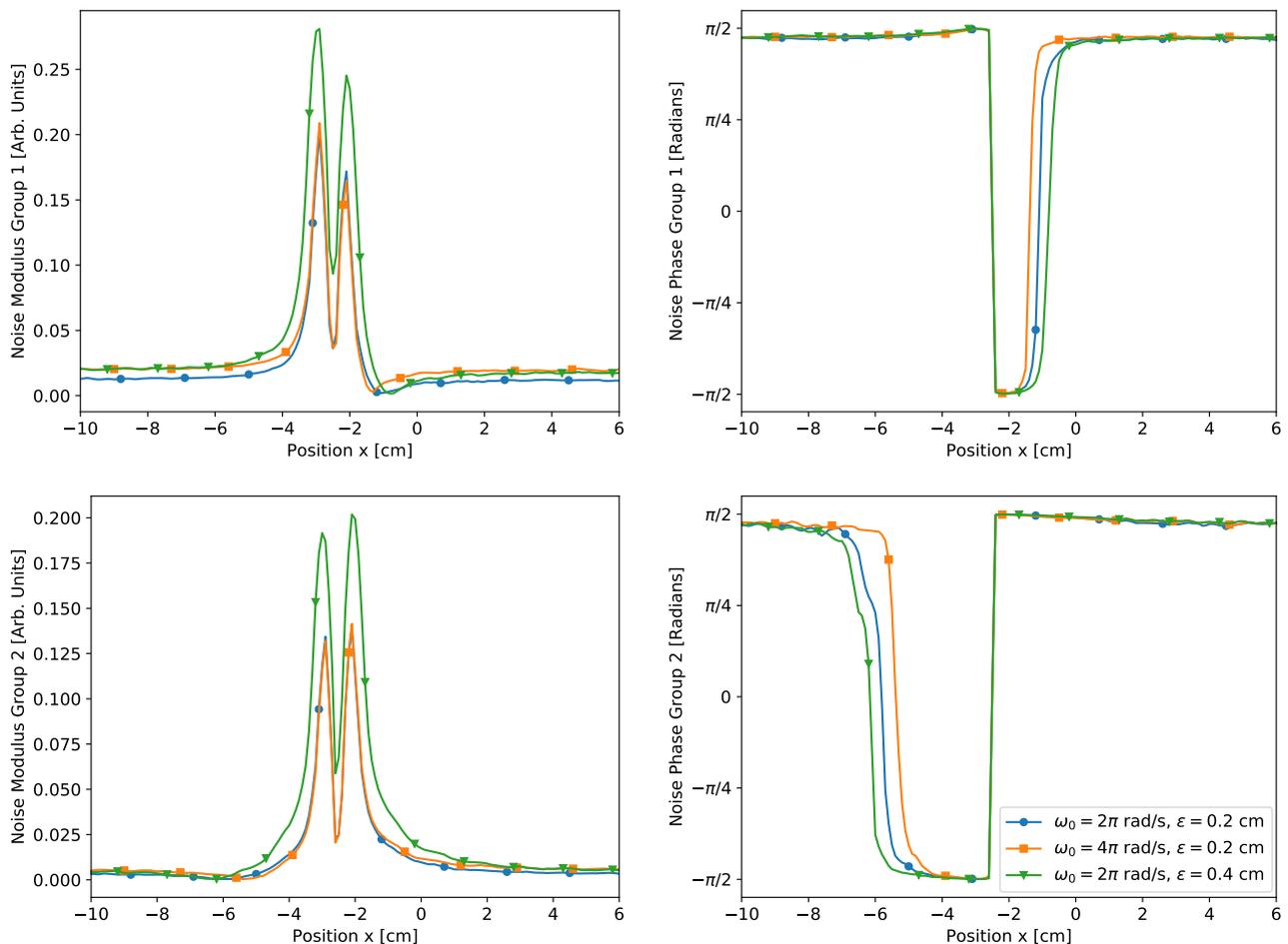


FIG. 9. Modulus and phase of the noise field through the center of the vibrating fuel pin at $y = -2.52$ cm, for different vibration parameters. All results were obtained using MGMC, with approximate cancellation and the fine cancellation mesh.

cal to the method used in the much simpler case of cross section oscillations. This new method was implemented in the multi-group Monte Carlo mini-app MGMC and verified against the noise source calculated with the deterministic noise solver in APOLLO3[®]: the two noise sources were in excellent agreement.

Second, we have applied weight cancellation methods as a variance reduction technique for neutron noise simulations. For the case of mechanical vibrations, the noise source contains nearly equal positive and negative contributions in both the real and imaginary components, which makes it very difficult to estimate the resulting noise field, because of very large variances in scores. Preliminary investigations on the case of cross section oscillations have shown that these issues can be overcome using weight cancellation methods [36]. In view of probing the effects of weight cancellation for the more involved case of mechanical vibrations, we have implemented a new Monte Carlo noise solver in MGMC. We have started from the Monte Carlo neutron noise method which was previously demonstrated in TRIPOLI-4[®], as

it is able to produce error bars for the noise field. This solution scheme was then modified, breaking the fixed-source noise batches into inner fission generations, in order to accommodate weight cancellation. To test our new solution strategy with weight cancellation, we have selected a recent benchmark for neutron noise problems, consisting in a two-dimensional, two-group reflected fuel assembly with a single vibrating fuel pin. Approximate weight cancellation and exact regional weight cancellation were tested on this benchmark problem: use of a relatively coarse mesh with approximate cancellation leads to improvements in the FOM by factors as high as 1624 when compared to the original neutron noise scheme implemented in TRIPOLI-4[®], at the expense of a slight but undetected bias in the noise field. Use of a finer mesh for approximate cancellation, which is required in order to quench the bias inherent to this method, yielded an improvement in the FOM by factors as high as 487. By comparison, the exact weight cancellation technique only improved the FOM by a factor of 14 and showed less potential for implementation in production Monte Carlo

codes, due to its algorithmic complexity.

The application of approximate weight cancellation improves the quality of the resulting noise field due to mechanical vibrations to the point that it becomes feasible to use Monte Carlo methods to analyse changes in the noise field stemming from small changes to the vibration frequency and amplitude; previously, without the use of weight cancellation, such a comparison would have been extremely challenging, if not impossible, for the case of a fuel pin vibration [31]. Our results would indicate that the application of weight cancellation to Monte Carlo neutron noise simulations is essentially mandatory if one is to obtain usable results for the study of neutron noise in power reactors. Although further investigations are required, our analysis shows that the bias imposed on the resulting noise field by the approximate weight cancellation is typically small, and more than acceptable since it

can be reduced by refining the cancellation mesh. While refining the mesh will reduce the efficiency of cancellation (and also the efficiency of the noise calculation), the observed improvements in the FOM for the examined benchmark problem suggest that this is likely not a problem. In the future, we hope to demonstrate this new Monte Carlo neutron noise solution scheme on larger problems, such as vibrating fuel assemblies, or vibrating clusters of fuel pins, to further demonstrate the potential for Monte Carlo method to be applied to neutron noise analysis.

ACKNOWLEDGEMENTS

TRIPOLI-4[®] and APOLLO3[®] are registered trademarks of CEA. The authors thank EDF and Framatome for partial financial support.

-
- [1] M. M. R. Williams, *Random Processes in Nuclear Reactors*. Pergamon Press, Oxford, 1974.
- [2] J. A. Thie, *Power Reactor Noise*. American Nuclear Society, 1981.
- [3] I. Pázsit and C. Demazière, “Noise techniques in nuclear systems,” in *Handbook of nuclear engineering*, D. Cacuci, Ed. Springer Verlag, 2010, ch. 14, pp. 1629–1737.
- [4] M. Seidl, K. Kosowski, U. Schüler, and L. Belblidia, “Review of the historic neutron noise behavior in German KWU built PWRs,” *Progress in Nuclear Energy*, vol. 85, pp. 668–675, 2015.
- [5] K. Behriguer, G. Kosály, and L. Kostic, “Reactivity and neutron density noise excited by random rod vibration,” *Annals of Nuclear Energy*, vol. 63, pp. 306–318, 1977.
- [6] D. N. Fry, J. March-Leuba, and F. J. Sweeney, “Use of neutron noise of diagnosis of in-vessel anomalies in light-water reactors,” Oak Ridge National Laboratory, Tech. Rep. ORNL/TM-8774, 1986.
- [7] I. Pázsit and G. T. Analytis, “Theoretical investigation of the neutron noise diagnostics of two-dimensional control rod vibrations in a PWR,” *Annals of Nuclear Energy*, vol. 7, pp. 171–183, 1980.
- [8] G. Kosály, “Noise investigations in boiling-water and pressurized-water reactors,” *Progress in Nuclear Energy*, vol. 5, pp. 145–199, 1980.
- [9] I. Pázsit, L. A. Torres, M. Hursin, H. Nylén, V. Dykin, and C. Montalvo, “Development of a new method to determine the axial void velocity profile in BWRs from measurements of the in-core neutron noise,” *Progress in Nuclear Energy*, vol. 138, p. 103805, 2021.
- [10] V. Arzhanov and I. Pázsit, “Diagnostics of core barrel vibrations by in-core and ex-core neutron noise,” *Progress in Nuclear Energy*, vol. 43, no. 1, pp. 151–158, 2003.
- [11] I. Pázsit, “The linearization of vibration-induced noise,” *Annals of Nuclear Energy*, vol. 11, pp. 441–454, 1984.
- [12] I. Pázsit and J. Karlsson, “On the perturbative calculation of the vibration noise by strong absorber,” *Annals of Nuclear Energy*, vol. 24, pp. 449–466, 1996.
- [13] A. Jonsson, H. N. Tran, V. Dykin, and I. Pázsit, “Analytical investigation of the properties of the neutron noise induced by vibrating absorber and fuel rods,” *Kerntechnik*, vol. 77, pp. 371–380, 2012.
- [14] T. Yamamoto and H. Sakamoto, “Decomposition of neutron noise in a reactor into higher-order mode components and investigation of the space and frequency dependence,” *Progress in Nuclear Energy*, vol. 117, p. 103098, 2019.
- [15] I. Pázsit and V. Dykin, “The role of the eigenvalue separation in reactor dynamics and neutron noise theory,” *J. Nucl. Sci. Technol.*, vol. 55, pp. 484–495, 2018.
- [16] A. Zoia, A. Rouchon, B. Gasse, C. Demazière, and P. Vinai, “Analysis of the neutron noise induced by fuel assembly vibrations,” *Annals of Nuclear Energy*, vol. 154, p. 108061, 2021.
- [17] C. Demazière, A. Rouchon, and A. Zoia, “Understanding the neutron noise induced by fuel assembly vibrations in linear theory,” *Annals of Nuclear Energy*, vol. 175, p. 109169, 2022.
- [18] U. Rohde, M. Seidl, S. Kliem, and Y. Bilodid, “Neutron noise observations in German KWU built PWRs and analyses with the reactor dynamics code DYN3D,” *Annals of Nuclear Energy*, vol. 112, pp. 715 – 734, 2018.
- [19] C. Demazière, “CORESIM: a multi-purpose neutronic tool for research and education,” *Annals of Nuclear Energy*, vol. 38, p. 2698–2718, 2011.
- [20] T. Yamamoto, “Monte carlo method with complex-valued weights for frequency domain analyses of neutron noise,” *Annals of Nuclear Energy*, vol. 58, p. 72–79, 2013.
- [21] A. Rouchon, A. Zoia, and R. Sanchez, “A new Monte Carlo method for neutron noise calculations in the frequency domain,” *Annals of Nuclear Energy*, vol. 102, p. 465–475, 2017.
- [22] N. Olmo-Juan, C. Demazière, T. Barrachina, R. Miró, and G. Verdú, “PARCS vs CORE SIM neutron noise simulations,” *Progress in Nuclear Energy*, vol. 115, pp. 169 – 180, 2019.
- [23] A. Vidal-Ferrándiz, A. Carreno, D. Ginestar, C. Demazière, and G. Verdú, “A time and frequency domain analysis of the effect of vibrating fuel assemblies on the neutron noise,” *Annals of Nuclear Energy*, vol. 137, p.

- 107076, 2020.
- [24] D. Chionis, A. Dokhane, L. Belblidia, H. Ferroukhi, G. Girardin, and A. Pautz, “Development and verification of a methodology for neutron noise response to fuel assembly vibrations,” *Annals of Nuclear Energy*, vol. 147, p. 107669, 2020.
- [25] C. Demazière, P. Vinai, M. Hursin, S. Kollias, and J. Herb, “Overview of the CORTEX project,” in *Proceedings of Physor2018*, Cancun, Mexico, 2018.
- [26] H. Yi, P. Vinai, and C. Demazière, “On the simulation of neutron noise using a discrete ordinates method,” *Annals of Nuclear Energy*, vol. 164, p. 108570, 2021.
- [27] T. Yamamoto, “Implementation of a frequency-domain neutron noise analysis method in a production-level continuous energy Monte Carlo code: verification and application to a BWR,” *Annals of Nuclear Energy*, vol. 115, pp. 494–501, 2018.
- [28] A. Rouchon, W. Jarrah, and A. Zoia, “The new neutron noise solver of the Monte Carlo code TRIPOLI-4,” in *Proceedings of M&C2019*, Portland USA, 2019.
- [29] A. Mylonakis, H. Yi, P. Vinai, and C. Demazière, “Neutron noise simulations in a heterogeneous system: A comparison between a diffusion-based and a discrete ordinates solver,” in *Proceedings of M&C2019*, Portland USA, 2019.
- [30] P. Vinai, H. Yi, A. Mylonakis, C. Demazière, B. Gasse, A. Rouchon, A. Zoia, A. Vidal-Ferrándiz, D. Ginestar, G. Verdú, and T. Yamamoto, “Comparison of Neutron Noise Solvers Based on Numerical Benchmarks in a 2-D Simplified UOX Fuel Assembly,” in *M&C 2021*, 2021.
- [31] P. Vinai, H. Yi, C. Demazière, A. Rouchon, A. Zoia, A. Vidal-Ferrándiz, A. Carreño, D. Ginestar, and G. Verdú, “On the simulation of neutron noise induced by vibrations of fuel pins in a simplified fuel assembly,” *Annals of Nuclear Energy*, Submitted.
- [32] Lamirand, Vincent *et al.*, “Neutron noise experiments in the akr-2 and crocus reactors for the european project cortex,” *EPJ Web Conf.*, vol. 225, p. 04023, 2020.
- [33] Mylonakis, A. G. *et al.*, “CORE SIM+ simulations of COLIBRI fuel rods oscillation experiments and comparison with measurements,” *EPJ Web Conf.*, vol. 247, p. 21006, 2021.
- [34] H. Belanger, D. Mancusi, and A. Zoia, “Exact weight cancellation in Monte Carlo eigenvalue transport problems,” *Physical Review E*, vol. 104, no. 1, p. 015306, 2021.
- [35] —, “Solving Eigenvalue Transport Problems with Negative Weights and Regional Cancellation,” in *M&C 2021*, Oct 2021.
- [36] —, “Variance Reduction Techniques for Monte Carlo Neutron Noise Simulations,” in *PHYSOR 2022*, 2022.
- [37] I. Pázsit, “Investigation of the space-dependent noise induced by a vibrating absorber,” *Atomkernenergie*, vol. 30, pp. 29–35, 1977.
- [38] M. Faucher, D. Mancusi, and A. Zoia, “New kinetic simulation capabilities for Tripoli-4®: Methods and applications,” *Annals of Nuclear Energy*, vol. 120, p. 74–88, 2018.
- [39] E. Brun, F. Damian, C. Diop, E. Dumonteil, F. Hugot, C. Jouanne, Y. Lee, F. Malvagi, A. Mazzolo, O. Petit, J. Trama, T. Visonneau, and A. Zoia, “TRIPOLI-4®, CEA, EDF and AREVA reference Monte Carlo code,” *Annals of Nuclear Energy*, vol. 82, p. 151–160, 2015.
- [40] L. L. Carter, E. D. Cashwell, and W. M. Taylor, “Monte Carlo Sampling with Continuously Varying Cross Sections Along Flight Paths,” *Nuclear Science and Engineering*, vol. 48, no. 4, p. 403–411, 1972.
- [41] D. Legrady, B. Molnar, M. Klausz, and T. Major, “Woodcock tracking with arbitrary sampling cross section using negative weights,” *Annals of Nuclear Energy*, vol. 102, p. 116–123, 2017.
- [42] T. E. Booth, “Computing the Higher k-Eigenfunctions by Monte Carlo Power Iteration: A Conjecture,” *Nuclear Science and Engineering*, vol. 143, no. 3, p. 291–300, 2003.
- [43] T. Yamamoto, “Convergence of the second eigenfunction in Monte Carlo power iteration,” *Annals of Nuclear Energy*, vol. 36, no. 1, p. 7–14, 2009.
- [44] —, “Monte Carlo algorithm for buckling search and neutron leakage-corrected calculations,” *Annals of Nuclear Energy*, vol. 47, p. 14–20, 2012.
- [45] —, “Monte Carlo method with complex weights for neutron leakage-corrected calculations and anisotropic diffusion coefficient generations,” *Annals of Nuclear Energy*, vol. 50, p. 141–149, 2012.
- [46] P. Zhang, H. Lee, and D. Lee, “A general solution strategy of modified power method for higher mode solutions,” *Journal of Computational Physics*, vol. 305, p. 387–402, 2016.
- [47] T. E. Booth and J. E. Gubernatis, “Exact Regional Monte Carlo Weight Cancellation for Second Eigenfunction Calculations,” *Nuclear Science and Engineering*, vol. 165, no. 3, p. 283–291, 2010.
- [48] E. R. Woodcock, T. Murphy, P. J. Hemmings, and T. C. Longworth, “Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry,” Argonne National Laboratory, Tech. Rep., 1965, ANL-7050.
- [49] J. Leppänen, “On the use of delta-tracking and the collision flux estimator in the Serpent 2 Monte Carlo particle transport code,” *Annals of Nuclear Energy*, vol. 105, p. 161–167, 2017.
- [50] H. Belanger, D. Mancusi, and A. Zoia, “Unbiasedness and optimization of regional weight cancellation,” *Physical Review E*, vol. 106, no. 2, p. 025302, 2022.
- [51] H. Belanger, “MGMC,” 2022, v0.2.0. [Online]. Available: <https://github.com/HunterBelanger/mgmc>
- [52] C. R. Harris *et al.*, “Array programming with NumPy,” *Nature*, vol. 585, no. 7825, p. 357–362, 2020.
- [53] A. Rouchon, “Analyse et développement d’outils numériques déterministes et stochastiques résolvant les équations du bruit neutronique et applications aux réacteurs thermiques et rapides,” Ph.D. dissertation, Université Paris-Saclay, 2016.
- [54] D. Schneider, F. Dolci, F. Gabriel, J. Palau, M. Guillo, B. Pothet, P. Archier, K. Ammar, F. Auffret, R. Baron *et al.*, “APOLLO3: CEA/DEN deterministic multi-purpose code for reactor physics analysis,” in *PHYSOR 2016*, 2016.
- [55] A. Rouchon, M. V. Le Brun, and A. Zoia, “Analysis and Comparison of APOLLO3 and TRIPOLI-4 Neutron Noise Solvers,” in *PHYSOR 2020*, vol. 247, 2021, p. 21002. [Online]. Available: <https://doi.org/10.1051/epjconf/202124721002>

14 - Conclusions for Neutron Noise

In Part III, we have focused on the development of possible variance reduction techniques which could be applied to neutron noise problems. For noise simulations, particles carry complex statistical weights, and the real and imaginary parts are allowed to be either positive or negative. The estimation of the noise field is then a sum of positive and negative contributions, which greatly increases the variance of the noise tally. Stimulated by the findings of Part II, where we have shown that weight cancellation was capable of solving the convergence issues of eigenvalue problems involving positive and negative weights, we have applied similar strategies to noise simulations. In the following we recall our key findings and original contributions.

For the problem of noise induced by cross section oscillations, we considered a branchless sampling approach to sample the noise source particles, the underlying idea being to quench the statistical dispersion due to the creation of multiple noise particles at the same collision site. A new formalism for performing branchless sampling with complex yields was developed for this purpose in Chapter 12. Unfortunately, using this technique on cross section oscillations seemed to have very little effect on the FOM for the estimated noise field, or may have even led to a slight degradation in performance. The exact cause of this observation is not yet understood. Future work should revisit this method, to see if similar results are obtained for mechanical vibration problems, and also examine possible alternative formulations which might improve the method's efficiency.

The application of weight cancellation to noise problems yielded more promising results. Using approximate weight cancellation for the cross section oscillation problem in Chapter 12, a typical improvement by a factor of 100 in the FOM was observed (using the time spent transporting neutrons) when comparing the new calculation scheme with inner generations and weight cancellation to the original TRIPOLI-4[®] baseline implementation, that does not use inner fission generations and does not apply weight cancellation. In Chapter 13, the problem of mechanical vibrations was considered. Using approximate cancellation on a coarse mesh, for a two-group benchmark with a single vibrating fuel pin, improved the FOM by a factor of more than 1200; using a fine cancellation mesh led to a factor of more than 460. Exact regional cancellation was unfortunately not as effective, and only improved the FOM by a factor of 14. Weight cancellation reduced the variance in the noise field tally to a great enough extent that it was possible to perform sensitivity analysis in noise simulations, i.e., observing the small deviations in the noise field resulting from small changes in the physical parameters of the benchmark. Without the use of cancellation, the large statistical uncertainties resulting from the baseline algorithm would have made the noise field for mechanical vibration problems essentially unusable, and sensitivity analysis impossible. The results presented in Chapter 13 strongly suggest that weight cancellation should be mandatory for Monte Carlo codes that solve the noise equation. We hope to test weight cancellation on larger noise problems in the future, such as vibrating assemblies in a full core, in continuous-energy.

In addition to exploring the effects of weight cancellation, we have also made another original contribution to the solution of noise problems. The initial noise sampling scheme for vibrations in TRIPOLI-4[®] made many simplifying approximations, which resulted in a different noise source than that predicted by APOLLO3[®]. To remedy this situation, we have developed a novel algorithm to sample the noise source from mechanical vibrations in an exact manner, making it possible to more accurately represent this phenomenon in our Monte Carlo simulations. The complete derivation of this method was performed in Chapter 13, and was checked against the noise source obtained from APOLLO3[®] as a reference. The new approach requires the construction of a fictitious material which contains the union of isotopes from the materials on either side of the vibrating interface. A nuclide can be sampled from this fictitious material in the standard manner, and be used to sample the noise source particles with a methodology which is very similar to that of the simpler case of cross section oscillations. Our initial implementation of this exact noise sampling scheme has been added to Chenille for flat vibrating surfaces. In the near future, we hope to extend this functionality to cylindrical interfaces as well, facilitating the treatment of standard fuel pins and cladding.

15 - Conclusions and Future Work

Continual advances in high performance computing resources allow Monte Carlo code developers to consider performing simulations which were once impossible. To accompany these new capabilities, the initial goal of this thesis was to improve the fidelity of material representations in Monte Carlo transport codes. Currently, most production-level codes assume that material properties are piece-wise constant over the elementary cells of the geometric model. In most real-world applications, such an assumption is not a truly faithful representation of reality, as temperature and density fields are generally continuous functions of position: this is particularly relevant in the domain of reactor physics, where the neutron field is coupled with the multi-physics feedback due to temperature and density effects in the fuel and in the moderator. With new coupled multi-physics simulations, one can therefore envision that during neutron transport the temperature and density at any given position can be queried from the thermal-hydraulic and thermo-mechanic solvers coupled with the Monte Carlo code. Allowing material properties to be spatially continuous functions gives rise to macroscopic cross sections which are also spatially continuous. Sampling the distance that a particle will fly before undergoing a collision becomes much more difficult under these circumstances.

In Part I of this manuscript, we performed an analysis of the existing particle tracking algorithms that could potentially be used to deal with spatially continuous cross sections. We found that the two best candidates were delta tracking [1] and a variant of negative-weighted delta tracking proposed by Carter et al. [2]. Delta tracking requires a majorant cross section which is used to sample the distance to tentative collision sites. Negative-weighted delta tracking only requires a sampling cross section which need not be a majorant; this additional flexibility comes at the cost of introducing negative particle weights. When used on fixed-source transport problems, both methods have comparable performance, so long as the sampling cross section for negative-weighted delta tracking does not underestimate the total cross section by a large factor. These results also indicate that negative-weighted delta tracking might be the preferred method for application in a general-purpose Monte Carlo code. Determining the majorant cross section across a spatially continuous temperature and density field for use in delta tracking could potentially be quite challenging. Delta tracking is known to demonstrate poor performance in reactor physics problems at certain energies, due to the localized heavy absorbed problem [3], and Carter et al.'s negative-weighted delta tracking would have similar difficulties. While Serpent is able to counter this inefficiency by switching to surface tracking at these energies, this would not be possible when using spatially continuous cross sections.

We instead considered regionalization as a possible solution, where the problem domain could be divided into regions via a simple mesh, and each region would contain a unique majorant (or sampling) cross section. Both delta tracking and negative-weighted delta tracking demonstrated better performance when the number of unique regions was increased, although with diminishing returns. It is likely that a regionalization approach might be effective to counter the localized heavy absorbed problem, but more work is needed to develop a methodology for best determining these regions.

In Part I we also assessed the efficiency of tracking methods in stochastic media, for the important class of Poisson tessellations, which might be used to model fuel fragmentation following severe accidents. Surface tracking and delta tracking were compared for different tessellation densities, where the average chord length, Λ , was either smaller than, greater than, or approximately equal to the mean free path of particles, λ . Three different acceleration techniques were also considered: neighbor maps, cell search meshes, and storing the geometry kernel with banked particles. In general, if a cell search mesh is used, then surface tracking with a neighbor map is better than delta tracking when $\Lambda \gg \lambda$, while delta tracking is superior in the regime where $\Lambda \ll \lambda$. When a cell search mesh is not utilized, surface tracking with a neighbor map was always more efficient than delta tracking.

This analysis was carried out exclusively for the case of Poisson tessellations, however, and likely does not hold for the case of stochastic spherical inclusions in a background matrix (which would model TRISO fuel particles for example): the spheres only have one neighbor (the background matrix), and the background matrix is a neighbor to all of the spheres. Future research should therefore address this class of stochastic

media, to determine which methods might be best for this use case.

For our comparison of delta tracking and negative-weighted delta tracking with spatially continuous cross sections, we initially focused on fixed-source problems. In view of their key role for most reactor physics applications, we then turned our attention to k -eigenvalue problems, and we assessed the performance of negative-weighted delta tracking when used in combination with power iteration. In Part II of the manuscript, we have performed power iteration simulations using delta tracking and negative-weighted delta tracking. While the delta tracking algorithm demonstrated no difficulties, as expected from the literature, the negative-weighted delta tracking simulations led to catastrophic termination. The number of particles in the simulation would increase without bound, eventually overwhelming the computer memory, leading the program to be killed by the operating system. Eventually, by modeling the system as a set of coupled transport equations for the positive and negative flux, we were able to determine that the presence of negative weights introduces a new family of nonphysical eigenstates, and that the dominant eigenstate of the set of coupled equations will always correspond to one of these nonphysical eigenstates. Moreover, we were able to prove that weight cancellation (where particles with positive and negative weights annihilate with one another) can allow power iteration to converge to the fundamental mode of the physical transport equation. Our analysis also showed that there is a minimum amount of weight cancellation that is required for the system to converge to the physical flux.

If a general Monte Carlo code implements the negative-weighted delta tracking method (and does not use a majorant cross section), then a weight cancellation technique must also be implemented, and having as many options for this as possible is of benefit to both code developers and users. To perform weight cancellation, two different techniques were taken from the literature. The first method was an approximate regional weight cancellation method [4]. This method is very easy to implement, and is quite fast. Using this algorithm on the C5G7 benchmark with negative-weighted delta tracking, the particle population was stabilized, and the simulation was able to complete normally. Although the method does impart a bias on the results, this bias was not visible for the cancellation mesh which we utilized in our simulation. An exact regional cancellation method had previously been proposed in the literature [5], but had only been demonstrated in a one-dimensional single-speed transport problem. In Part II we extended this technique for use in three-dimensional multi-group transport problems. Using this new algorithm on the C5G7 benchmark with negative-weighted delta tracking, it was possible to complete a power iteration simulation nominally, albeit in a longer time frame and with a larger variance than if delta tracking had been used. We subsequently proposed a method to optimize the selection of the cancellation parameters to increase the amount of weight which is cancelled by the algorithm. This optimal choice of cancellation parameters was shown to improve the efficiency of exact regional cancellation, although the technique is still much less efficient in terms of figure of merit than the approximate regional cancellation method.

The theory for how one might go about performing exact weight cancellation in continuous-energy simulations has been sketched, but we have also stressed that gaining access to the required nuclear data in a Monte Carlo code may be difficult. Future work will need to attempt a full three-dimensional and continuous-energy implementation of exact weight cancellation. Even if it were possible to perform exact cancellation in continuous-energy, the efficiency of the method might not be high enough to be effective at permitting the convergence of power iteration for certain simulations. In such cases, we might be forced to rely exclusively on approximate weight cancellation: a challenging topic of research would thus be the quantification of the error imposed by this method. Currently, the only way to determine if a cancellation mesh is refined enough so that the bias in the results is not visible is to perform several simulations with different cancellation meshes. Being able to quantify the error of approximate cancellation and predict the required cancellation mesh refinement in advance would greatly contribute to the confidence in the method.

Although initially of interest in this work for performing power iteration with negative-weighted delta tracking, weight cancellation techniques are useful for other key applications of Monte Carlo simulations in the domain of reactor physics. In the literature, the two weight cancellation techniques that we considered in this thesis were previously used for estimating the second harmonic of the flux [4, 5], determining the critical buckling [6, 7], and solving the neutron noise equations in the frequency domain [6–8]. In Part

III of the manuscript, we have applied exact and approximate weight cancellation methods to the particles carrying complex statistical weights involved in neutron noise simulations, which are notoriously hindered by serious convergence issues. When applied to the academic problem of neutron noise induced by cross section oscillations, approximate weight cancellation could improve the figure of merit by a factor of 100, when considering only the time spent transporting noise particles. For the case of neutron noise induced by mechanical vibrations, we found an improvement of more than 1200 with a coarse approximate cancellation mesh, and a factor of approximately 470 for a fine approximate cancellation mesh. Exact regional cancellation was not as effective, however, and only resulted in an improvement by a factor of 14 for mechanical vibrations.

Our results indicate that weight cancellation is absolutely essential to obtain results from neutron noise simulations. Without weight cancellation, the variance in the noise field is too large for the obtained results to be of any use. For the examined noise problems, most of the amplitude of the noise field is either in the imaginary or the real component, but never equally shared between the two. While the relative error of the dominant component can be reduced to very acceptable levels, the relative error of the weaker component is always somewhat large: a remaining question is thus whether variance reduction techniques exist which could improve the relative error of the weaker component of the noise field.

We have also proposed and tested another possible variance reduction technique, branchless noise source sampling: for unknown reasons, this technique seemed to have little effect, and may have even led to a slight degradation in performance. Further investigations will be needed in order to understand the reasons of this failure and to propose possible improvements.

In the benchmark problems considered in Part III, the noise detectors were somewhat close to the noise source: in the general case where the detectors are located far from the source, it would be interesting to see whether adjoint-driven variance reduction methods could be developed for the noise equation, possibly leading to effective zero-variance schemes such as those proposed for other types of reactor physics simulations [9–11].

Finally, to perform the neutron noise simulations of a vibrating fuel pin, we have developed a new exact algorithm for sampling the noise source from a vibrating interface. This method will improve the fidelity of future Monte Carlo noise simulations, as only approximate techniques were available before. In the future, we hope to demonstrate these newly developed techniques on a noise problem in continuous-energy, and on larger problems, such as vibrating fuel assemblies.

References

- [1] E. R. Woodcock, T. Murphy, P. J. Hemmings, and T. C. Longworth, “Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry,” Argonne National Laboratory, Tech. Rep., 1965, ANL-7050.
- [2] L. L. Carter, E. D. Cashwell, and W. M. Taylor, “Monte Carlo Sampling with Continuously Varying Cross Sections Along Flight Paths,” *Nuclear Science and Engineering*, vol. 48, no. 4, p. 403–411, 1972.
- [3] J. Leppänen, “Performance of Woodcock delta-tracking in lattice physics applications using the Serpent Monte Carlo reactor physics burnup calculation code,” *Annals of Nuclear Energy*, vol. 37, no. 5, p. 715–722, 2010.
- [4] P. Zhang, H. Lee, and D. Lee, “A general solution strategy of modified power method for higher mode solutions,” *Journal of Computational Physics*, vol. 305, p. 387–402, 2016.
- [5] T. E. Booth and J. E. Gubernatis, “Exact Regional Monte Carlo Weight Cancellation for Second Eigenfunction Calculations,” *Nuclear Science and Engineering*, vol. 165, no. 3, p. 283–291, 2010.
- [6] T. Yamamoto, “Monte Carlo algorithm for buckling search and neutron leakage-corrected calculations,” *Annals of Nuclear Energy*, vol. 47, p. 14–20, 2012.

- [7] —, “Monte Carlo method with complex weights for neutron leakage-corrected calculations and anisotropic diffusion coefficient generations,” *Annals of Nuclear Energy*, vol. 50, p. 141–149, 2012.
- [8] —, “Monte Carlo method with complex-valued weights for frequency domain analyses of neutron noise,” *Annals of Nuclear Energy*, vol. 58, p. 72–79, 2013.
- [9] I. Lux and L. Koblinger, *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations*. CRC Press, 1991.
- [10] J. E. Hoogenboom, “Zero-Variance Monte Carlo Schemes Revisited,” *Nuclear Science and Engineering*, vol. 160, no. 1, p. 1–22, 2008.
- [11] D. Mancusi and A. Zoia, “Zero-variance schemes for kinetic Monte Carlo simulations,” *The European Physical Journal Plus*, vol. 135, no. 6, p. 401, 2020.

A - Résumé Détaillé en Français

A.1 . Introduction

La distribution des neutrons et des photons dans l'espace des phases au sein des systèmes d'intérêt dans les applications de radioprotection et de physique des réacteurs nucléaires peut être décrite par l'équation de Boltzmann. Deux familles de méthodes numériques existent pour résoudre l'équation de Boltzmann pour les neutrons. Les méthodes déterministes sont la catégorie la plus connue et englobent une variété de techniques qui reposent sur la discrétisation de l'espace des phases. Tous les solveurs déterministes utilisent l'approximation multi-groupes, où les sections efficaces des nucléides sont approximées comme étant constantes dans un intervalle d'énergie [1]. La composante de direction des solutions est souvent discrétisée le long de plusieurs ordonnées discrètes (ce qui conduit aux méthodes S_n), ou peut être approchée comme une somme finie d'harmoniques sphériques (ce qui conduit aux méthodes P_n) [2]. Les coordonnées spatiales sont généralement discrétisées au moyen d'un maillage géométrique, bien que certaines techniques telles que la méthode des caractéristiques ne nécessitent pas de maillage spatial mais supposent que la source de neutrons est une fonction constante ou linéaire par morceaux des coordonnées de l'espace de phase [2]. Les codes déterministes tels que APOLLO3[®], développé au CEA, sont capables de résoudre l'équation de transport relativement rapidement,¹ et sont donc les solveurs les plus couramment utilisés dans l'industrie nucléaire. Afin de discrétiser finement les six dimensions de l'espace des phases (ou sept, pour les problèmes non stationnaires où il faut ajouter le temps aux variables de l'espace des phases), le nombre de degrés de liberté pour le problème d'un cœur complet peut atteindre 10^{21} , ce qui est hors de portée de la génération actuelle d'ordinateurs [3]. Pour pallier ce problème, des modèles d'ordre réduit et des approximations sont introduits dans les solveurs déterministes, ce qui ajoute un biais de modélisation, en plus des inévitables erreurs de discrétisation, dans les résultats finaux. Ceci est particulièrement délicat pour le traitement de la variable énergie : l'utilisation de sections efficaces multi-groupes, qui nécessite des modèles d'autoprotection très sophistiqués, peut entraîner des erreurs dont l'amplitude est très difficile à prédire [2].

La méthode Monte Carlo a été initialement développée dans les années 1940, pour résoudre l'équation de Boltzmann par une approche probabiliste [4]. Dans cette technique, une population de neutrons est suivie à travers une séquence de vols libres et de collisions avec les matériaux traversés. L'évolution de chaque particule est intrinsèquement stochastique et doit être échantillonnée selon les lois physiques (sections efficaces, distributions énergie-angle et rendements) fournies dans les bibliothèques de données nucléaires. Des nombres aléatoires sont utilisés en conjonction avec des probabilités connues pour différents événements afin de décider le destin de la particule, de sa naissance jusqu'à sa mort due à la capture ou à la fuite. Chaque fois qu'un événement d'intérêt est échantillonné, la particule correspondante contribue à la valeur estimée de l'observable associée à l'événement, appelée un score ou un tally. Une fois que toute la population a été traitée², la moyenne d'ensemble sur toutes les contributions des particules fournit une estimation non biaisée de l'observable recherchée.

La connaissance des opérateurs de vol et de collision de l'équation de Boltzmann (en plus de la source) est suffisante pour échantillonner les trajectoires des particules : un avantage clé de la méthode Monte Carlo est qu'elle ne nécessite pas la discrétisation de l'équation de Boltzmann (bien que l'espace des phases doive être décomposé en régions pour enregistrer les événements échantillonnés pour chaque quantité d'intérêt). Différentes quantités telles que le flux, le courant et les taux de réaction peuvent être estimées en faisant

¹Il est généralement admis que les solveurs déterministes sont beaucoup plus rapides par rapport aux solveurs Monte Carlo. Ceci est à nuancer, car le temps de calcul des codes déterministes n'inclut généralement pas le temps passé à effectuer des calculs d'autoprotection et d'homogénéisation pour obtenir des sections efficaces multi-groupes.

²L'utilisation de nombres aléatoires avec des grains différents pour échantillonner des processus stochastiques garantit que les résultats de chaque simulation sont différents et indépendants. Un code déterministe donnera toujours le même résultat pour un problème donné.

la moyenne des contributions des particules aux scores correspondants dans la partie donnée de l'espace des phases. En simulant directement le processus physique pour les particules individuelles, il est possible d'obtenir une solution non biaisée et sans approximations, et d'estimer l'incertitude statistique associée (c'est-à-dire le niveau de confiance que nous pouvons avoir dans le résultat obtenu).³ Grâce à l'absence de biais, la méthode Monte Carlo est la méthode étalon pour la vérification et la validation de codes déterministes. Le principal inconvénient des simulations Monte Carlo est qu'elles sont plus lentes que les solveurs déterministes, notamment parce que l'incertitude statistique décroît comme $1/\sqrt{N}$ en fonction du nombre de particules simulées, N . Pour cette raison, traditionnellement, l'utilisation des codes Monte Carlo dans l'industrie nucléaire est relativement limitée par rapport aux solveurs déterministes, et généralement restreinte aux problèmes stationnaires.

A.1.1 . Nouveaux Défis pour les Simulations Monte Carlo en Physique des Réacteurs

Au cours des dernières décennies, les ressources informatiques disponibles pour la plupart des utilisateurs ont considérablement augmenté. Les simulations Monte Carlo qui prenaient auparavant des heures ou des jours sur des super-calculateurs, autrefois à la pointe de la technologie, peuvent désormais être exécutées sur un ordinateur portable personnel en quelques minutes. Alors que les codes Monte Carlo étaient autrefois presque exclusivement utilisés pour l'étude des réacteurs de recherche ou des cœurs en conditions stationnaires, plusieurs projets en Europe (HPMC⁴ et McSAFE⁵) et aux États-Unis (CESAR⁶ et CASL⁷) ont stimulé le développement de nouvelles méthodes de simulation et de techniques de réduction de variance permettant des réalisations sans précédent. Les codes Monte Carlo sont maintenant utilisés pour effectuer une analyse complète du cœur pour les calculs d'évolution avec des solveurs couplés pour la thermohydraulique et la thermomécanique pour prendre en compte la multi-physique [5–8]. Récemment, des simulations Monte Carlo dynamiques ont été effectuées sur des transitoires complets d'un cœur, y compris à l'échelle de temps des précurseurs de neutrons retardés, avec une multi-physique couplée [7, 9, 10]. Avec les améliorations continues apportées aux infrastructures de calcul à haute performance (HPC), le nombre de possibilités pour les simulations Monte Carlo en physique des réacteurs ne cesse de s'étendre. De nouveaux types de simulations qui n'avaient jamais été envisagés auparavant sont actuellement à l'étude. Certaines recherches se sont concentrées sur la résolution des harmoniques supérieures de l'équation de transport, afin d'améliorer la convergence de la source et de calculer le rapport de dominance, qui est essentiel à la sûreté du réacteur [11, 12]. Un autre nouveau sujet d'intérêt, développé dans le cadre du projet EU H2020 CORTEX (2017-2021), a été la résolution de l'équation du bruit neutronique dans le domaine de fréquences, afin de comprendre les effets des vibrations des crayons ou des assemblages dans un cœur [13, 14].

Les progrès du HPC et l'étude de nouveaux types de simulations obligent les développeurs de code Monte Carlo à reconsidérer complètement la manière dont les routines de transport sont mises en œuvre. Le premier exemple de ce type est peut-être le développement de techniques permettant d'effectuer à la volée l'élargissement Doppler des sections efficaces [15, 16]. Un autre exemple est la réécriture d'un algorithme « history-based » vers un algorithme « event-based » pour une meilleure utilisation des GPUs [17]. Comme les GPUs ont été développés pour être plus efficaces lorsqu'ils effectuent la même opération sur plusieurs éléments de données, exploiter efficacement leur potentiel de calcul nécessite de trier les particules qui subissent le même type de réaction. Il ressort clairement de ces exemples que l'avancement de l'analyse Monte Carlo pour la physique des réacteurs nucléaires obligera les développeurs de code à reconsidérer les méthodes traditionnelles, en inventant de nouvelles techniques qui correspondent mieux aux contraintes des architectures informatiques modernes.

³La solution obtenue par la méthode Monte Carlo est exacte pour les données nucléaires, la géométrie et les compositions de matériaux données. S'il existe des incertitudes dans la configuration géométrique et la composition des matériaux, ou si les données nucléaires fournies sont inadéquates, les résultats d'un système peuvent ne pas s'aligner avec les résultats expérimentaux.

⁴High performance Monte Carlo reactor core analysis. <https://www.fp7-hpmc.eu>

⁵High-Performance Monte Carlo Methods for SAFETY Demonstration. <https://cordis.europa.eu/project/id/755097>

⁶Center for Exascale Simulation of Advanced Reactors.

⁷Consortium for Advanced Simulation of Light Water Reactors. <https://casl.gov>

A.1.2 . Des Milieux Continus au Bruit Neutronique

L'objectif principal de cette thèse est de rechercher une approche possible pour améliorer la fidélité de la modélisation des propriétés des matériaux en relation avec le transport de particules par des simulations Monte Carlo. Dans la plupart des codes Monte Carlo actuels, les propriétés des matériaux sont supposées être des fonctions constantes par morceaux de la position spatiale. A chaque volume élémentaire (ou cellule) du modèle géométrique sont typiquement associées une température, une densité et une composition isotopique : au sein de cette région spatiale, ces propriétés physiques sont supposées spatialement homogènes. Sauf dans quelques cas, cette hypothèse n'est généralement pas satisfaite dans la réalité, car toutes ces propriétés peuvent être des fonctions continues de la position spatiale. Ceci est particulièrement pertinent dans les applications de la physique des réacteurs, lorsque l'on considère les simulations de transport neutronique couplées aux champs thermohydraulique, thermomécanique et d'évolution : pour prendre en compte les effets multi-physiques, il est nécessaire de rechercher la température, la densité et les concentrations isotopiques à n'importe quelle position dans le réacteur, ce qui rend les propriétés du matériau vues par les neutrons le long de leurs déplacements également dépendantes de manière continue de la position. L'utilisation de ces informations permettrait une représentation plus précise des systèmes du monde réel. Des bibliothèques sont déjà en cours de développement pour permettre aux codes neutroniques de Monte Carlo d'interroger les températures et densités calculées par les solveurs multi-physiques en mémoire [18]. Les sections efficaces spatialement continues pourraient également être appliquées aux problèmes d'évolution, où les concentrations isotopiques varient dans l'espace. Dans les codes actuels, les crayons de combustible doivent être discrétisés en plusieurs couronnes (≈ 10), afin de tenir compte des effets de peau. Des concentrations isotopiques spatialement continues pourraient potentiellement rendre cette discrétisation en couronnes inutile.

La température, la densité et les concentrations isotopiques sont toutes utilisées pour déterminer la section efficace macroscopique à une position et une énergie données. Si ces propriétés matérielles sont des fonctions continues de la position, alors la section macroscopique l'est aussi. L'échantillonnage de la distance parcourue par une particule avant de subir une collision est une partie intégrante de l'algorithme Monte Carlo, comme cela est décrit dans la section 2.3. Cette tâche devient cependant laborieuse une fois que les sections efficaces ne sont plus constantes par morceaux. Notre objectif était donc d'examiner de manière critique les méthodes de suivi des particules existantes et de les adapter pour échantillonner des distances de vol à partir de sections efficaces spatialement continues.

Dans la partie I, nous considérons plusieurs algorithmes de suivi de particules différents et examinons leurs performances lorsqu'ils sont utilisés dans des problèmes à source fixe avec des sections efficaces spatialement continues. Parmi les algorithmes considérés, notre analyse montre qu'une méthode que nous qualifions de *negative-weighted delta tracking* est la plus prometteuse [19, 20]. La particularité de la méthode *negative-weighted delta tracking* est que les particules peuvent avoir un poids statistique positif ou négatif. Après cette investigation initiale dans le chapitre 4, nous tentons d'examiner les performances du *negative-weighted delta tracking* dans les simulations d'itération de puissance. Bien que l'algorithme fonctionne dans le cas des problèmes à source fixe, toutes les simulations d'itération de puissance échouent, à notre grande surprise. Dans la partie II, nous analysons les raisons de cet échec et proposons une stratégie basée sur l'annulation de poids comme solution. L'annulation de poids est une opération qui « combine » une particule de poids positif et une particule de poids négatif de manière à ce que leurs poids s'annulent l'un avec l'autre. Accomplir cela de manière non biaisée dans une simulation Monte Carlo n'est pas trivial, car les particules sont chacune situées à des coordonnées d'espace de phase différentes, et il est donc impossible qu'une particule positive et négative se retrouve exactement aux mêmes coordonnées. Nous développons deux méthodes d'annulation de poids différentes dans la partie II, et les appliquons à l'itération de puissance avec *negative-weighted delta tracking* : nous montrons que cette approche permet à ces simulations de se terminer normalement.

Alors que le développement des techniques d'annulation de poids apparaît initialement indépendant du problème des milieux continus dans l'espace, il est primordial lors de l'utilisation du *negative-weighted delta tracking*. D'autres problèmes dans le domaine de la physique des réacteurs nucléaires existent qui pourraient

également bénéficier de l'application de l'annulation de poids. Un de ces cas est la résolution de l'équation du bruit neutronique dans le domaine de fréquences. Une précédente thèse au CEA s'est concentrée sur le développement de méthodes Monte Carlo pour résoudre l'équation du bruit neutronique [21], et nous avons décidé de tester les techniques d'annulation de poids développées dans la partie II dans ce domaine. La partie III de cette thèse applique les techniques d'annulation de poids nouvellement développées au problème du bruit neutronique en tant que technique de réduction de la variance.

A.1.3 . Chenille

TRIPOLI-4[®] est le code Monte Carlo de référence pour le transport de particules qui est en développement au CEA depuis le milieu des années 90 [22]. En tant que code utilisé dans l'industrie, TRIPOLI-4[®] possède une pléthore de fonctionnalités (y compris la possibilité de propager des neutrons, des photons et des électrons/positons dans la gerbe électromagnétique) pour des applications allant de la physique des réacteurs à la radioprotection et à l'instrumentation nucléaire : actuellement la base de code se situe à environ 400 kSLOC (Source Lines Of Code). Tenter d'apporter même de petits changements à un programme aussi vaste peut s'avérer difficile, en particulier pour les étudiants qui ne se sont pas encore familiarisés avec le code et ses particularités. Le travail présenté dans cette thèse a nécessité la mise en œuvre de nouveaux algorithmes de transport de particules, la restructuration de la décomposition des problèmes d'itération de puissance et de transport à source fixe, et l'ajout de maillages d'annulation de poids. Toutes ces tâches sont très invasives vis-à-vis de l'architecture d'un code Monte Carlo ; le travail qu'exigeaient les objectifs scientifiques fixés pour le déroulement de cette thèse aurait été difficilement réalisable dans TRIPOLI-4[®], sous les contraintes temporelles d'une thèse. Ainsi, bien que les implémentations existantes dans TRIPOLI-4[®] aient souvent été prises comme point de départ, dans la pratique tous les nouveaux algorithmes ont été conçus et testés à partir de *Chenille*, une nouvelle mini-application Monte Carlo développée à partir de zéro au fil de cette thèse.

Chenille est une mini-application de simulation Monte Carlo capable de simuler le transport de neutrons dans des géométries tridimensionnelles, en utilisant des données nucléaires multi-groupes ou en énergie continue. Avec seulement ≈ 20 kSLOC et écrit de manière très modulaire, la taille et le style de Chenille facilitent la mise en œuvre et le test rapides de nouveaux algorithmes, même ceux nécessitant une réécriture approfondie de l'architecture existante. Comme l'une des principales investigations de cette thèse était l'examen de différentes méthodes de suivi des particules, Chenille a plusieurs options. Le surface tracking standard est la méthode de suivi par défaut. Le delta tracking est également disponible ; actuellement, il n'est pas possible d'utiliser une combinaison hybride de surface tracking et de delta tracking, comme c'est le cas dans Serpent [23]. Une variante du negative-weighted delta tracking, telle que proposée par Carter et al. a également été implémentée dans Chenille [19]. Trois modes de simulation différents sont disponibles : source fixe, valeur propre k et bruit neutronique. Les trois modes sont disponibles en transport de particules multi-groupes et en énergie continue. Dans ce dernier cas, Chenille utilise la bibliothèque Papillon Nuclear Data Library (PapillonNDL), qui est développée et maintenue par moi-même [24]. PapillonNDL est une bibliothèque C++20 qui lit les fichiers ACE pour les données de neutrons en énergie continue, et est publiée en tant que logiciel libre et open-source sous la licence GPLv3. Il fournit les classes qui sont responsables de l'évaluation des sections efficaces microscopiques et de l'échantillonnage des lois de diffusion et de fission. Les lois de diffusion thermique sont également prises en charge et peuvent être utilisées avec PapillonNDL et Chenille. Actuellement, Chenille n'utilise pas de tables de probabilité pour les régions de résonances non résolues. Une vérification de base a été effectuée à l'aide d'une suite de problèmes analytiques multi-groupes [25] et d'une comparaison des résultats avec OpenMC pour un ensemble de problèmes de référence de criticité en énergie continue. Chenille a été écrit avec à la fois un parallélisme de mémoire partagée et un parallélisme de mémoire distribuée, en utilisant respectivement les normes OpenMP et MPI. Les fichiers d'entrée sont écrits en YAML, ce qui les rend très faciles à lire et à écrire. Une version multi-groupes de Chenille, appelée MGMC, a été mise à la disposition du public en tant que logiciel libre sous la licence CeCILL-v2.1 [26]. MGMC possède toutes les fonctionnalités principales de Chenille, mais ne dispose pas des classes et fonctions nécessaires pour lire les fichiers ACE et interagir avec l'API de PapillonNDL.

A.1.4 . Structure de la Thèse

Cette thèse est organisée en trois parties principales, chacune portant sur un thème spécifique :

1. La partie I est consacrée à l'évaluation des performances d'une variété de différentes méthodes de suivi des particules. Deux cas d'utilisation différents sont considérés : les milieux stochastiques et les sections efficaces spatialement continues.
2. La partie II examine l'application du negative-weighted delta tracking aux problèmes d'itération de puissance k , montrant que les algorithmes d'annulation de poids sont obligatoires pour que ces simulations convergent.
3. La partie III applique les nouvelles méthodes d'annulation de poids aux simulations de bruit neutronique, en tant que technique de réduction de la variance.

Chaque partie du manuscrit commence par un chapitre d'introduction, suivi de plusieurs publications de recherche (chacune dans un chapitre distinct), puis d'un dernier chapitre de conclusion. Le chapitre d'introduction est destiné à fournir au lecteur les connaissances préalables nécessaires et les outils qui seront ensuite utilisés dans les problèmes décrits dans les publications ultérieures. Certaines des principales méthodes utilisées dans les articles sont décrites et les résultats importants sont résumés. Toutes les analyses techniques détaillées et les résultats des simulations numériques sont fournis dans ces articles. Ils sont présentés dans un ordre qui doit permettre une transition en douceur entre les différentes parties du manuscrit et rendre le fil de la pensée clair. Une enquête, tout en répondant à certaines questions, en amène d'autres dans un domaine légèrement différent ; finalement, après avoir posé suffisamment de questions puis répondu, il est tout à fait possible de se retrouver très loin de son point de départ (ce qui est la définition d'une marche aléatoire, telle que simulée par Monte Carlo !). Enfin, chaque partie thématique se termine par un chapitre de conclusion, qui résume les principaux résultats, et mentionne les questions ouvertes qui devraient être abordées par les recherches futures. La liste des chapitres qui ont déjà été publiés ou qui ont été soumis pour publication est fournie ci-dessous. Avec chaque chapitre, on trouvera la référence appropriée pour la publication associée.

Chapitre 3: H. Belanger, C. Larmier, D. Mancusi, and A. Zoia, "Optimization of Particle Tracking Methods for Stochastic Media," In *Proceedings of the International Conference on Physics of Reactors 2022 (PHYSOR 2022)*, May 2022, Pittsburgh, PA, p. 294-303.

Chapitre 4: H. Belanger, D. Mancusi, and A. Zoia, "Review of Monte Carlo methods for particle transport in continuously-varying media," *European Physical Journal Plus*, vol. 135, no. 11, p. 877, 2020, doi: 10.1140/epjp/s13360-020-00731-y.

Chapitre 7: H. Belanger, D. Mancusi, and A. Zoia, "Solving Eigenvalue Transport Problems with Negative Weights and Regional Cancellation," In *Proceedings of the The International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering 2021 (M&C 2021)*, October 2021, p. 46-55.

Chapitre 8: H. Belanger, D. Mancusi, and A. Zoia, "Exact weight cancellation in Monte Carlo eigenvalue transport problems," *Physical Review E*, vol. 104, no. 1, p. 015306, 2021, doi: 10.1103/physreve.104.015306.

Chapitre 9: H. Belanger, D. Mancusi, and A. Zoia, "Unbiasedness and optimization of regional weight cancellation," *Physical Review E*, vol. 106, no. 2, p. 025302, 2022, doi: 10.1103/physreve.106.025302.

Chapitre 12: H. Belanger, D. Mancusi, and A. Zoia, "Variance Reduction Techniques for Monte Carlo Neutron Noise Simulations," In *Proceedings of the International Conference on Physics of Reactors 2022 (PHYSOR 2022)*, May 2022, Pittsburgh, PA, p. 544-553.

Chapitre 13: H. Belanger, D. Mancusi, A. Rouchon, and A. Zoia, "Variance Reduction and Noise Source Sampling Techniques for Monte Carlo Simulations of Neutron Noise Induced by Mechanical Vibrations," *Nuclear Science and Engineering*, accepted for publication.

A.2 . Conclusions et Travaux Futurs

Les progrès continus des ressources de calcul haute performance permettent aux développeurs de code Monte Carlo d'envisager d'effectuer des simulations qui étaient autrefois impossibles. Pour accompagner ces nouvelles capacités, l'objectif initial de cette thèse était d'améliorer la fidélité des représentations matérielles dans les codes de transport Monte Carlo. Actuellement, la plupart des codes supposent que les propriétés des matériaux sont constantes par morceaux sur les cellules élémentaires du modèle géométrique. Dans la plupart des applications du monde réel, une telle hypothèse n'est pas une représentation fidèle de la réalité, car les champs de température et de densité sont généralement des fonctions continues de la position : ceci est particulièrement pertinent dans le domaine de la physique des réacteurs, où le champ neutronique est couplé aux contre réactions multi-physique dues aux effets de température et de densité dans le combustible et dans le modérateur. Avec de nouvelles simulations multi-physiques couplées, on peut donc envisager que lors du transport des neutrons, la température et la densité à une position donnée puissent être interrogées à partir des solveurs thermohydrauliques et thermomécaniques couplés au code Monte Carlo. Permettre aux propriétés des matériaux d'être des fonctions spatialement continues donne lieu à des sections efficaces macroscopiques qui sont également spatialement continues. L'échantillonnage de la distance parcourue par une particule avant de subir une collision devient beaucoup plus difficile dans ces circonstances.

Dans la partie I de ce manuscrit, nous avons effectué une analyse des algorithmes de suivi de particules existants qui pourraient potentiellement être utilisés pour traiter des sections efficaces spatialement continues. Nous avons constaté que les deux meilleurs candidats étaient le delta tracking [27] et une variante du negative-weighted delta tracking proposée par Carter et al. [19]. Le delta tracking nécessite une section efficace majorante qui est utilisée pour échantillonner la distance jusqu'aux sites de collision provisoires. Le negative-weighted delta tracking ne nécessite qu'une section efficace d'échantillonnage qui n'a pas besoin d'être majorante; cette flexibilité supplémentaire se fait au prix de l'introduction de poids de particules négatifs. Lorsqu'elles sont utilisées sur des problèmes de transport à source fixe, les deux méthodes ont des performances comparables, tant que la section efficace d'échantillonnage pour le negative-weighted delta tracking ne sous-estime pas la section efficace totale par un facteur important. Ces résultats indiquent également que le negative-weighted delta tracking pourrait être la méthode préférable pour l'application dans un code Monte Carlo à usage général. La détermination de la section efficace majorante à travers un champ de température et de densité spatialement continu pour une utilisation dans le delta tracking pourrait potentiellement être compliquée. Le delta tracking est connu pour avoir de mauvaises performances dans les problèmes de physique des réacteurs à certaines énergies, en raison du problème localisé de forte absorption [28], et le negative-weighted delta tracking de Carter et al. aurait des difficultés similaires. Alors que Serpent est capable de contrer cette inefficacité en passant au surface tracking à ces énergies, cela ne serait pas possible en utilisant des sections efficace spatialement continues.

Nous avons plutôt considéré la régionalisation comme une solution possible, où le domaine du problème pourrait être divisé en régions via un maillage simple, et chaque région contiendrait une section efficace majorante (ou d'échantillonnage) unique. Le delta tracking et le negative-weighted delta tracking ont de meilleures performances lorsque le nombre de régions uniques a augmenté, mais avec des rendements décroissants. Il est probable qu'une approche de régionalisation pourrait être efficace pour contrer le problème localisé de forte absorption, mais des travaux supplémentaires sont nécessaires pour développer une méthodologie permettant de déterminer au mieux ces régions.

Dans la partie I, nous avons également évalué l'efficacité des méthodes de suivi dans les milieux stochastiques, pour la classe importante des mosaïques de Poisson, qui pourraient être utilisées pour modéliser la fragmentation du combustible suite à des accidents graves. Le surface tracking et le delta tracking ont été comparés pour différentes densités de pavage, où la longueur moyenne de la corde, Λ , était inférieure, supérieure ou approximativement égale au libre parcours moyen des particules, λ . Trois techniques d'accélération différentes ont également été envisagées : les cartes de voisins, les maillages de recherche de cellules et le stockage du noyau géométrique avec des particules. En général, si un maillage de recherche de cellules est utilisé, le surface tracking avec une carte de voisins est meilleur que le delta tracking lorsque $\Lambda \gg \lambda$, tandis que le delta tracking est supérieur dans le régime où $\Lambda \ll \lambda$. Lorsqu'un maillage de recherche de cellules n'est pas utilisé, le surface tracking avec une carte de voisins était toujours plus efficace que le delta tracking.

Cependant, cette analyse a été effectuée exclusivement pour le cas des pavages de Poisson et ne serait probablement plus valable pour le cas des inclusions sphériques stochastiques dans une matrice de fond (qui modéliserait les particules de combustible TRISO par exemple) : les sphères n'ont qu'un seul voisin (la matrice de fond), et la matrice de fond est voisine de toutes les sphères. Les recherches futures devraient donc aborder cette classe de milieux stochastiques, afin de déterminer quelles méthodes pourraient être les meilleures pour ce cas d'utilisation.

Pour notre comparaison du delta tracking et du negative-weighted delta tracking avec des sections efficaces spatialement continues, nous nous sommes d'abord concentrés sur les problèmes de source fixe. Compte tenu de leur rôle clé pour la plupart des applications de physique des réacteurs, nous avons ensuite porté notre attention sur les problèmes de valeurs propres k , et nous avons évalué les performances du negative-weighted delta tracking lorsqu'il est utilisé en combinaison avec l'itération de puissance. Dans la partie II du manuscrit, nous avons donc effectué des simulations d'itération de puissance en utilisant le delta tracking et le negative-weighted delta tracking. Alors que l'algorithme de delta tracking n'a démontré aucune difficulté, comme prévu dans la littérature, les simulations de negative-weighted delta tracking ont conduit à une terminaison catastrophique. Le nombre de particules dans la simulation augmente sans limite, submergeant finalement la mémoire de l'ordinateur, conduisant le programme à être tué par le système d'exploitation. Finalement, en modélisant le système comme un ensemble d'équations de transport couplées pour le flux positif et négatif, nous avons pu déterminer que la présence de poids négatifs introduit une nouvelle famille d'états propres non physiques, et que l'état propre dominant de l'ensemble d'équations couplées correspond toujours à l'un de ces états propres non physiques. De plus, nous avons pu prouver que l'annulation de poids (où les particules avec des poids positifs et négatifs s'annihilent les unes avec les autres) peut permettre à l'itération de puissance de converger vers le mode fondamental de l'équation de transport physique. Notre analyse a également montré qu'il existe un minimum d'annulation de poids nécessaire pour que le système converge vers le flux physique.

Si un code Monte Carlo général implémente la méthode de negative-weighted delta tracking (et n'utilise pas de section efficace majorante), une technique d'annulation de poids doit également être implémentée, et avoir autant d'options que possible pour cela soit avantageux pour à la fois les développeurs de code et les utilisateurs. Pour effectuer une annulation de poids, deux techniques différentes ont été tirées de la littérature. La première méthode était une méthode d'annulation de poids régionale approchée [29]. Cette méthode est très facile à mettre en œuvre, et assez rapide. En utilisant cet algorithme sur le benchmark C5G7 avec le negative-weighted delta tracking, la population de particules a été stabilisée et la simulation a pu se terminer normalement. Bien que la méthode confère un biais aux résultats, ce biais n'était pas visible pour le maillage d'annulation que nous avons utilisé dans notre simulation. Une méthode d'annulation régionale exacte avait déjà été proposée dans la littérature [30], mais n'avait été démontrée que dans un problème de transport unidimensionnel à vitesse unique. Dans la partie II, nous avons étendu cette technique pour l'utiliser dans des problèmes de transport multi-groupes tridimensionnels. En utilisant ce nouvel algorithme sur le benchmark C5G7 avec negative-weighted delta tracking il a été possible de compléter une simulation d'itération de puissance nominalement, bien que dans un laps de temps plus long et avec une variance plus grande que si le delta tracking avait été utilisé. Nous avons ensuite proposé une méthode pour optimiser la

sélection des paramètres d'annulation afin d'augmenter la quantité de poids annulée par l'algorithme. Il a été démontré que ce choix optimal des paramètres d'annulation améliore l'efficacité de l'annulation régionale exacte, bien que la technique soit encore beaucoup moins efficace en termes de facteur de mérite que la méthode d'annulation régionale approchée.

La théorie sur la façon dont on pourrait procéder pour effectuer une annulation de poids exacte dans des simulations en énergie continue a été esquissée, mais nous avons également souligné qu'il peut être difficile d'accéder aux données nucléaires requises dans un code Monte Carlo. Les travaux futurs devront tenter une mise en œuvre complète en trois dimensions et en énergie continue de l'annulation de poids exacte. Même s'il était possible d'effectuer une annulation exacte en énergie continue, l'efficacité de la méthode pourrait ne pas être suffisamment élevée pour permettre la convergence de l'itération de puissance pour certaines simulations. Dans de tels cas, nous pourrions être contraints de nous fier exclusivement à l'annulation du poids approchée : un sujet de recherche difficile serait donc la quantification de l'erreur imposée par cette méthode. Actuellement, la seule façon de déterminer si un maillage d'annulation est suffisamment raffiné pour que le biais dans les résultats ne soit pas visible est d'effectuer plusieurs simulations avec différents maillages d'annulation. Être capable de quantifier l'erreur d'annulation approchée et de prédire à l'avance le raffinement du maillage d'annulation requis contribuerait grandement à la confiance dans la méthode.

Bien qu'initialement intéressantes dans ce travail pour effectuer une itération de puissance avec le *negative-weighted delta tracking*, les techniques d'annulation de poids sont utiles pour d'autres applications clés des simulations Monte Carlo dans le domaine de la physique des réacteurs. Dans la littérature, les deux techniques d'annulation de poids que nous avons considérées dans cette thèse étaient auparavant utilisées pour estimer la deuxième harmonique du flux [29, 30], déterminer le laplacien critique [31, 32], et résoudre l'équation du bruit neutronique dans le domaine fréquentiel [31–33]. Dans la partie III du manuscrit, nous avons appliqué des méthodes d'annulation de poids exactes et approchées aux particules portant des poids statistiques complexes impliquées dans les simulations de bruit neutronique, qui sont notoirement entravées par de graves problèmes de convergence. Lorsqu'elle est appliquée au problème académique du bruit neutronique induit par les oscillations de la section efficace, l'annulation approchée du poids a amélioré le facteur de mérite d'un facteur 100, en ne considérant que le temps passé à transporter des particules de bruit. Pour le cas du bruit neutronique induit par les vibrations mécaniques, nous avons trouvé une amélioration de plus de 1200 avec un maillage d'annulation approché grossier, et un facteur d'environ 470 pour un maillage d'annulation approché fin. L'annulation régionale exacte n'a cependant pas été aussi efficace et n'a entraîné qu'une amélioration d'un facteur 14 pour les vibrations mécaniques.

Nos résultats indiquent que l'annulation de poids est absolument essentielle pour obtenir des résultats à partir de simulations de bruit neutronique. Sans annulation de poids, la variance dans le champ de bruit est trop grande pour que les résultats obtenus soient d'une quelconque utilité. Pour les problèmes de bruit examinés, la plus grande partie de l'amplitude du champ de bruit est soit dans la composante imaginaire soit dans la composante réelle, mais jamais également partagée entre les deux. Tandis que l'erreur relative de la composante dominante peut être réduite à des niveaux acceptables, l'erreur relative de la composante la plus faible est toujours quelque peu importante : il reste donc à savoir s'il existe des techniques de réduction de la variance qui pourraient améliorer l'erreur relative de la composante la plus faible du champ de bruit. Nous avons également proposé et testé une autre technique de réduction de variance possible, l'échantillonnage de source de bruit de façon « *branchless* » : pour des raisons inconnues, cette technique semble avoir peu d'effet, et peut même conduire à une légère dégradation des performances. Des investigations complémentaires seront nécessaires pour comprendre les raisons de cet échec et proposer d'éventuelles améliorations. Dans les problèmes de référence considérés dans la partie III, les détecteurs de bruit étaient proches de la source de bruit : dans le cas général où les détecteurs sont situés loin de la source, il serait intéressant de voir si des méthodes de réduction de la variance utilisant le flux adjoint pourraient être développées pour l'équation de bruit, conduisant éventuellement à des schémas de variance nulle tels que ceux proposés pour d'autres types de simulations neutroniques [34–36]. Enfin, pour effectuer les simulations de bruit neutronique d'un crayon de combustible vibrant, nous avons développé un nouvel algorithme exact pour échantillonner la source de bruit à partir d'une interface vibrante. Cette méthode améliorera la fidélité des futures simulations de bruit

Monte Carlo, car seules des techniques approchées étaient disponibles auparavant. Dans le futur, nous espérons démontrer ces techniques nouvellement développées sur un problème de bruit en énergie continue, et sur des problèmes plus vastes, tels que des assemblages de combustibles vibrants dans un cœur complet.

References

- [1] G. Bell and S. Glasstone, *Nuclear Reactor Theory*. Van Nostrand Reinhold Company, 1970.
- [2] A. Hébert, *Applied Reactor Physics*, 3rd ed. Presses Internationales Polytechnique, 2020.
- [3] M. Coste-Delclaux, C. Diop, A. Nicolas, and B. Bonin, “La neutronique,” 2013.
- [4] N. Metropolis and S. Ulam, “The Monte Carlo Method,” *Journal of the American Statistical Association*, vol. 44, no. 247, pp. 335–341, 1949.
- [5] D. J. Kelly, A. E. Kelly, B. N. Aviles, A. T. Godfrey, R. K. Salko, and B. S. Collins, “MC21/CTF and VERA multiphysics solutions to VERA core physics benchmark progression problems 6 and 7,” *Nuclear Engineering and Technology*, vol. 49, no. 6, p. 1326–1338, 2017.
- [6] J. Yu, H. Lee, M. Lemaire, H. Kim, P. Zhang, and D. Lee, “MCS based neutronics/thermal-hydraulics/fuel-performance coupling with CTF and FRAPCON,” *Computer Physics Communications*, vol. 238, p. 1–18, 2019.
- [7] A. Levinsky, V. Valtavirta, F. P. Adams, and V. N. Anghel, “Modeling of the SPERT transients using Serpent 2 with time-dependent capabilities,” *Annals of Nuclear Energy*, vol. 125, p. 80–98, 2019.
- [8] D. Ferraro, M. Garcia, U. Imke, V. Valtavirta, J. Leppänen, and V. Sanchez-Espinoza, “Serpent/SCF pin-level multiphysics solutions for the VERA Fuel Assembly benchmark,” *Annals of Nuclear Energy*, vol. 128, p. 102–114, 2019.
- [9] D. Legrady, G. Tolnai, T. Hajas, E. Pazman, T. Parko, and I. Pos, “Full Core Pin-Level VVER-440 Simulation of a Rod Drop Experiment with the GPU-Based Monte Carlo Code GUARDYAN,” *Energies*, vol. 15, no. 8, p. 2712, 2022.
- [10] D. Mancusi, M. Faucher, and A. Zoia, “Monte Carlo simulations of the SPERT III E-core transient experiments,” *The European Physical Journal Plus*, vol. 137, no. 1, p. 127, 2022.
- [11] T. E. Booth, “Computing the Higher k-Eigenfunctions by Monte Carlo Power Iteration: A Conjecture,” *Nuclear Science and Engineering*, vol. 143, no. 3, p. 291–300, 2003.
- [12] P. Zhang, H. Lee, M. Lemaire, C. Kong, J. Choe, J. Yu, F. Khoshahval, and D. Lee, “Practical Monte Carlo simulation using modified power method with preconditioning,” *Annals of Nuclear Energy*, vol. 127, p. 372–384, 2019.
- [13] T. Yamamoto, “Implementation of a frequency-domain neutron noise analysis method in a production-level continuous energy Monte Carlo code: Verification and application in a BWR,” *Annals of Nuclear Energy*, vol. 115, p. 494–501, 2018.
- [14] A. Rouchon, A. Zoia, and R. Sanchez, “A new Monte Carlo method for neutron noise calculations in the frequency domain,” *Annals of Nuclear Energy*, vol. 102, p. 465–475, 2017.
- [15] T. Viitanen and J. Leppänen, “Explicit Treatment of Thermal Motion in Continuous-Energy Monte Carlo Tracking Routines,” *Nuclear Science and Engineering*, vol. 171, no. 2, p. 165–173, Oct 2012.

- [16] G. Yesilyurt, W. R. Martin, and F. B. Brown, "On-the-Fly Doppler Broadening for Monte Carlo Codes," *Nuclear Science and Engineering*, vol. 171, no. 3, p. 239–257, 2012.
- [17] S. P. Hamilton and T. M. Evans, "Continuous-energy Monte Carlo neutron transport on GPUs in the Shift code," *Annals of Nuclear Energy*, vol. 128, p. 236–247, 2019.
- [18] P. K. Romano, S. P. Hamilton, R. O. Rahaman, A. Novak, E. Merzari, S. M. Harper, P. C. Shriwise, and T. M. Evans, "A Code-Agnostic Driver Application for Coupled Neutronics and Thermal-Hydraulic Simulations," *Nuclear Science and Engineering*, vol. 195, no. 4, p. 1–21, 2020.
- [19] L. L. Carter, E. D. Cashwell, and W. M. Taylor, "Monte Carlo Sampling with Continuously Varying Cross Sections Along Flight Paths," *Nuclear Science and Engineering*, vol. 48, no. 4, p. 403–411, 1972.
- [20] D. Legrady, B. Molnar, M. Klausz, and T. Major, "Woodcock tracking with arbitrary sampling cross section using negative weights," *Annals of Nuclear Energy*, vol. 102, p. 116–123, 2017.
- [21] A. Rouchon, "Analyse et développement d'outils numériques déterministes et stochastiques résolvant les équations du bruit neutronique et applications aux réacteurs thermiques et rapides," Ph.D. dissertation, Université Paris-Saclay, 2016.
- [22] E. Brun, F. Damian, C. Diop, E. Dumonteil, F. Hugot, C. Jouanne, Y. Lee, F. Malvagi, A. Mazzolo, O. Petit, J. Trama, T. Visonneau, and A. Zoia, "TRIPOLI-4®), CEA, EDF and AREVA reference Monte Carlo code," *Annals of Nuclear Energy*, vol. 82, p. 151–160, 2015.
- [23] J. Leppänen, M. Pusa, T. Viitanen, V. Valtavirta, and T. Kaltiaisenaho, "The Serpent Monte Carlo code: Status, development and applications in 2013," *Annals of Nuclear Energy*, vol. 82, p. 142–150, 2015.
- [24] H. Belanger, "Papillon Nuclear Data Library," 2021. [Online]. Available: <https://github.com/HunterBelanger/papillon-ndl>
- [25] A. Sood, R. A. Forster, and D. K. Parsons, "Analytical Benchmark Test Set For Criticality Code Verification," *Progress in Nuclear Energy*, vol. 42, no. 1, p. 55–106, 2003.
- [26] H. Belanger, "MGMC," 2022, v0.2.0. [Online]. Available: <https://github.com/HunterBelanger/mgmc>
- [27] E. R. Woodcock, T. Murphy, P. J. Hemmings, and T. C. Longworth, "Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry," Argonne National Laboratory, Tech. Rep., 1965, ANL-7050.
- [28] J. Leppänen, "Performance of Woodcock delta-tracking in lattice physics applications using the Serpent Monte Carlo reactor physics burnup calculation code," *Annals of Nuclear Energy*, vol. 37, no. 5, p. 715–722, 2010.
- [29] P. Zhang, H. Lee, and D. Lee, "A general solution strategy of modified power method for higher mode solutions," *Journal of Computational Physics*, vol. 305, p. 387–402, 2016.
- [30] T. E. Booth and J. E. Gubernatis, "Exact Regional Monte Carlo Weight Cancellation for Second Eigenfunction Calculations," *Nuclear Science and Engineering*, vol. 165, no. 3, p. 283–291, 2010.
- [31] T. Yamamoto, "Monte Carlo algorithm for buckling search and neutron leakage-corrected calculations," *Annals of Nuclear Energy*, vol. 47, p. 14–20, 2012.
- [32] —, "Monte Carlo method with complex weights for neutron leakage-corrected calculations and anisotropic diffusion coefficient generations," *Annals of Nuclear Energy*, vol. 50, p. 141–149, 2012.

- [33] —, “Monte Carlo method with complex-valued weights for frequency domain analyses of neutron noise,” *Annals of Nuclear Energy*, vol. 58, p. 72–79, 2013.
- [34] I. Lux and L. Koblinger, *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations*. CRC Press, 1991.
- [35] J. E. Hoogenboom, “Zero-Variance Monte Carlo Schemes Revisited,” *Nuclear Science and Engineering*, vol. 160, no. 1, p. 1–22, 2008.
- [36] D. Mancusi and A. Zoia, “Zero-variance schemes for kinetic Monte Carlo simulations,” *The European Physical Journal Plus*, vol. 135, no. 6, p. 401, 2020.