



# Development of a hybrid simulation approach for dose rate calculations in nuclear reactor dismantling operations

Ettore Guadagni

## ► To cite this version:

Ettore Guadagni. Development of a hybrid simulation approach for dose rate calculations in nuclear reactor dismantling operations. Modeling and Simulation. Université Paris-Saclay, 2022. English.  
NNT : 2022UPASP003 . tel-03641287

**HAL Id: tel-03641287**

<https://theses.hal.science/tel-03641287>

Submitted on 14 Apr 2022

**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 a hybrid simulation approach for dose rate calculations in nuclear reactor dismantling operations

*Développement d'une approche de simulation hybride pour les calculs de débits de dose dans les opérations de démantèlement de réacteurs nucléaires*

## 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 : Aval du cycle nucléaire, radioprotection et radiochimie

Graduate School : Physique. Référent : Faculté de Sciences d'Orsay

Thèse préparée dans l'unité de recherche **Service de Physique des Réacteurs et du Cycle** (CEA Cadarache),  
sous la direction de **Cheikh M'Backé DIOP**, Directeur de recherche,  
le co-encadrement de **Cindy LE LOIREC**, Docteure,  
et de **Jean-Michel LÉTANG**, Maître de conférences

**Thèse soutenue à Cadarache, le 18 Janvier 2022, par**

**Ettore GUADAGNI**

## Composition du Jury

**Pierre DÉSESQUELLES**

Professeur, Université Paris Saclay

Président

**Abdel-Mjid NOURREDDINE**

Professeur, IPHC Strasbourg

Rapporteur & Examinateur

**Lydia MAIGNE**

Professeure, Université Clermont Auvergne

Rapporteur & Examinatrice

**Sandra DULLA**

Professeure, Politecnico di Torino

Examinatrice

**Abdallah LYOUSSE**

Professeur, CEA Cadarache

Examinateur

**Cheikh M'Backé DIOP**

Directeur de recherche, CEA Saclay

Directeur de thèse

**Titre :** Développement d'une approche de simulation hybride pour les calculs de débits de dose dans les opérations de démantèlement de réacteurs nucléaires

**Mots clés :** Monte Carlo, Démantèlement, Réduction de variance, Transport de gamma, estimateur e-TLE

**Résumé :** Lors du démantèlement d'installations nucléaires, le niveau du débit de dose doit être estimé à l'aide d'outils de calcul vérifiés, validés et qualifiés afin d'être en mesure de planifier le scénario de démantèlement le plus optimisé en terme de radioprotection. Il existe aujourd'hui deux méthodes classiques pour estimer le débit de dose : les méthodes Monte-Carlo (MC), précises mais nécessitant un grand nombre d'heures de calcul, et la technique du noyau ponctuel (ou PKI pour *Point Kernel Integration*), moins précise mais beaucoup plus rapide. Une approche intermédiaire est de développer une méthode dite « hybride » combinant la précision de la simulation MC à la rapidité d'un algorithme déterministe de dépôt de dose. Une telle hybridation du parcours des particules peut fournir une alternative efficace à une simulation MC classique. D'un point de vue algorithmique, cette approche est fondée sur la même physique que celle implémentée dans les simulations MC et prend donc en compte les différentes hétérogénéités rencontrées lors du transport des particules. Dans le cadre de cette thèse, une approche hybride basée sur l'estimateur « *exponential track length* »

(e-TLE) a été implémentée dans le code Monte Carlo du CEA, TRIPOLI-4®, et mise en œuvre pour le transport des rayonnements gamma dans des configurations typiques d'installations nucléaires en phase de démantèlement. Les rayonnements gamma contribuent en effet majoritairement au débit de dose à l'arrêt de l'installation. L'estimateur e-TLE a en premier lieu été implémenté dans sa version de base dans le but de vérifier la cohérence des résultats avec les autres estimateurs déjà disponibles dans TRIPOLI-4®. Dans un second temps, afin d'optimiser les gains en performance observés pour la version de base de l'estimateur, deux versions spécifiques ont été développées : une version locale combinée à un algorithme de détection forcée permettant un gain en performance de plusieurs ordres de grandeur pour des détecteurs dits locaux, et une version globale associée à une technique de « *splitting* » permettant l'obtention d'une carte de dose de façon plus rapide. Dans chaque cas de figure, des configurations-types d'installations en cours de démantèlement ont été étudiées et le gain en performance de l'estimateur estimé.

**Title :** Development of a hybrid simulation approach for dose rate calculations in nuclear reactor dismantling operations

**Keywords :** Monte Carlo, Dismantling, VRT, Gamma transport, e-TLE

**Abstract :** When decommissioning nuclear facilities, the dose rate level should be estimated using calculation tools which are verified, validated and qualified to be able to plan the optimal decommissioning scenario in terms of radioprotection. At the present time, there are two classic methods for estimating dose rate: Monte Carlo methods (MC), accurate but requiring long computation times, and the point-kernel integration (PKI), less accurate but much faster. An intermediate approach is to develop a so-called "hybrid" method combining the precision of the MC simulation with the speed of a deterministic algorithm for dose deposition. Such hybridization of the particle path may provide an efficient alternative to the classic MC simulation. From an algorithmic point of view, this approach is based on the same physics implemented in the MC simulations and therefore takes into account the different heterogeneities encountered during particle transport. As part of this thesis, it is proposed to develop a similar hybrid approach for gamma radiation transport in

configurations that are typically found in nuclear facilities during decommissioning, these radiations contributing for the larger part to dose rate at the shutdown of the facility. The hybridization has been implemented through the "*exponential track length estimator*" (e-TLE), first in its basic version in order to analyse the consistency of the results with the other estimators already available in TRIPOLI-4®. Secondly, in order to optimize the performance gains observed for the basic version of the estimator, two specific versions have been developed: a local version combining the basic e-TLE with a forced detection algorithm and allowing a performance gain of several orders of magnitude for so-called local detectors and a global version combined with a "*splitting*" technique and allowing a faster determination of dose meshes. In each case, typical configurations of installations being dismantled were studied to assess the performance gain of the estimator.



# Acknowledgments

A few words of gratitude towards the people that accompanied me through this three-year-long quest are due.

First of all, of course, I would like to thank my supervisors: Cindy Le Loirec, whose help was simply invaluable, for assisting me from my first days as a confused non-French-speaker student up until the very last moments of this thesis; Jean-Michel Létang, for always being available to help discuss my doubts and providing essential insight; and my thesis director, Cheikh Diop, for being always in touch and equally willing to help on theoretical scientific matters as well as bureaucratic problems (which can take a surprisingly large portion of time during a work of this kind).

Next are all my colleagues at LPN in Cadarache, especially the VIP section who goes by the name of *LPN low-cost*: thank you for making everyday life at work more fun with countless discussions around a dining table, a coffee, or the crucially important pre-Covid tradition of the *gâteau du vendredi*.

Thank you also to my “remote” colleagues at SERMA Saclay, and especially to Davide Mancusi, François Hugot and Odile Petit, for their unending contributions and for being always close to my work even with a considerable geographical distance dividing us.

I am grateful to all the members of the jury who saw me complete the last steps of this journey: thank you to Lydia Maigne and Abdel-Mjid Nourreddine for taking the time to read and review this thesis; and thank you to all the other members, Pierre Désesquelles, Abdallah Lyoussi, Sandra Dulla, Khalil Amgarou and Yannick Pénéliau, for their participation and interesting contributions.

Finally, to all my friends, my family, and my beautiful girlfriend, I say this: a simple thank-you on an insignificant thesis manuscript, no matter how heartfelt, will never be enough to express my appreciation to you. Rather, I hope I have been able to show you my gratitude daily, and I hope I can continue to do so in the future.

# Table of contents

<b>Acknowledgments</b>	<b>IV</b>
<b>Table of contents</b>	<b>V</b>
<b>List of figures</b>	<b>VIII</b>
<b>List of tables</b>	<b>XII</b>
<b>List of acronyms</b>	<b>XIV</b>
<b>Introduction</b>	<b>XVIII</b>
Nuclear energy in France	XIX
Decommissioning and dismantling (D&D)	XX
Role of simulation tools in dismantling applications	XXII
Core calculation (determination of fission sources)	XXIII
Flux calculations	XXIII
Activation calculations	XXIV
Shielding calculations	XXIV
Thesis objectives and summary	XXV
<b>PART I</b>	<b>1</b>
<b>Chapter 1</b>	<b>5</b>
<b>Particle transport</b>	<b>5</b>
1.1. The Boltzmann equation for particle transport	5
1.2. Modelling deterministic approach	7
1.3 Stochastic approach: the Monte Carlo method	12
<b>Chapter 2</b>	<b>21</b>
<b>Heuristic approach</b>	<b>21</b>
2.1. Study configuration	21
2.2. Deterministic solution	22
2.3. Stochastic solution	23
2.4. Practical example: numerical 1D Monte Carlo	26
<b>PART II</b>	<b>29</b>
<b>Chapter 3</b>	<b>33</b>
<b>Limits of modelling deterministic and stochastic methods</b>	<b>33</b>

3.1 Complex configurations encountered in dismantling problems	34
3.2 “Odano” benchmarks	35
3.3 Reactor-specific scenarios	38
3.4 The need for accelerating Monte Carlo calculations	40
<b>Chapter 4</b>	<b>45</b>
<b>Variance reduction techniques for Monte Carlo transport</b>	<b>45</b>
4.1 Efficiency Enhancing Techniques	46
4.2 Variance Reduction Techniques (VRT)	48
4.3 The TRIPOLI-4® Monte Carlo code	50
4.4 Performance study on dismantling applications	53
4.5 Conclusions	59
<b>PART III</b>	<b>61</b>
<b>Chapter 5</b>	<b>65</b>
<b>Theoretical basis of the exponential track length estimator (e-TLE)</b>	<b>65</b>
5.1 Heuristic approach: 1D mono-directional transport using e-TLE	65
5.2 State of the art: current and past applications of the e-TLE	70
5.3 e-TLE implementation in TRIPOLI-4®	72
5.4 Validation and performance analysis	74
5.5 Conclusions	78
<b>Chapter 6</b>	<b>81</b>
<b>The e-TLE as a “local” estimator</b>	<b>81</b>
6.1 e-TLE with forced detection for local reduction of variance	82
6.2 Code implementation	84
6.3 Validation and performance analysis	86
6.4 Conclusions	89
<b>Chapter 7</b>	<b>91</b>
<b>The e-TLE as a “global” estimator</b>	<b>91</b>
7.1 e-TLE for global reduction of variance	91
7.2 Splitting multiplicity study	92
7.3 Validation and performance analysis	93
7.4 Conclusions	98
<b>Conclusions and perspectives</b>	<b>101</b>
<b>References</b>	<b>103</b>

**ANNEX A: KERMA AND DOSE DEFINITIONS** \_\_\_\_\_ A

**ANNEX B: RIGOROUS MATHEMATICAL DERIVATION OF THE EXPONENTIAL TRACK-LENGTH  
ESTIMATOR** \_\_\_\_\_ C

**ANNEX C: RÉSUMÉ SUBSTANTIEL EN FRANÇAIS** \_\_\_\_\_ G

# List of figures

Figure I.1: French nuclear facilities undergoing dismantling in 2019 (source: ASN website [6])	XIX
Figure I.2: Diagram of the most important legislative steps of D&D in France	XXI
Figure 1.3: Comparison of computation results (TRIPOLI4® vs PANTHEREv2) with measurement results for the «slant penetration benchmark»	9
Figure 1.4: Photon reflection in streaming configurations	10
Figure 1.5: Geometric arrangement for the definition of the albedo	10
Figure 1.6: 2D visual representation of a) linear track-length estimator, and b) exponential track-length estimator	15
Figure 2.7: 1D transport problem configuration	22
Figure 2.8: Average total (left) and uncollided (right) particle flux in the detector calculated through Monte Carlo. The red bar shows the value of the analytical solution. Note the difference in the x axis between the pictures, highlighting the much slower convergence of the uncollided simulation.	27
Figure 3.9: Scattering-intensive configuration from [41]	34
Figure 3.10: Source-shield-detector configuration for the study of multiple-scattering with the RANKERN PKI code, from [43]	35
Figure 3.11: Results for the statistical error on a photon transport shielding MC calculation, from [64]	35
Figure 3.12: Geometry of the back-scattering (a) and slant penetration (b) benchmark from [55]. Red points represent the detector locations, red lines represent some of the possible photon paths.	37
Figure 3.13: Geometry of the duct streaming benchmark from [55]. Red points represent the detector locations, red lines show some of the possible photon paths.	37
Figure 3.14: Geometry of the strong attenuation slab (a) and bunker (b) configurations.	39
Figure 3.15: Geometry of the complex bunker configuration, adapted from [68]	40
Figure 4.16: Importance maps obtained for Odano's configurations. Red indicates high importance, whereas blue indicates low importance.	54
Figure 4.17: C/E ratios obtained for Odano's configurations; red lines correspond to the validity domain comprised between $\pm 2\sigma$ .	55
Figure 4.18: TRIPOLI-4® /MCNP dose rate ratio obtained for Odano's configurations; red lines correspond to the validity domain comprised between $\pm 2\sigma$ .	56
Figure 4.19: $C_{VRT}/C_{ANALOG}$ ratio obtained for Odano's configurations; red lines correspond to the validity domain comprised between $\pm 2\sigma$ .	57
Figure 4.20: FOM gain obtained for Odano's configurations.	58
Figure 4.21: FOM as a function of water thickness for the strong attenuation slab configuration	58
Figure 4.22: FOM gain as a function of shield height ( $h$ ) and depth ( $d$ ) for the simple and complex bunker configurations.	59
Figure 5.23: 1D transport problem configuration (from Chapter 2)	66

<i>Figure 5.24: Comparison of collision estimator, track-length estimator, and exponential track-length estimator over the 1D slab. The red bar on the left represents the analytical value of the flux.</i>	70
<i>Figure 5.25: Flowchart of e-TLE implementation for photon transport in TRIPOLI-4®'s transport chain</i>	74
<i>Figure 5.26: KERMA comparison: a) Odano backscattering configuration; b) Odano slant penetration configuration; c) simple bunker configuration, slim wall; d) simple bunker configuration, thick wall; e) complex bunker configuration. Error bars of <math>\pm 2\sigma</math> are shown.</i>	75
<i>Figure 5.27: FOM comparison: a) Odano backscattering configuration; b) Odano slant penetration configuration; c) simple bunker configuration, slim wall; d) simple bunker configuration, thick wall; e) complex bunker configuration. Results above the red line mark an acceleration of the simulation with respect to analog TLE.</i>	76
<i>Figure 5.28: KERMA comparison: a) Odano backscattering configuration; b) Odano slant penetration configuration; c) Simple bunker configuration, slim wall; d) Simple bunker configuration, thick wall; e) Complex bunker configuration. All results are compared to the relative TRIPOLI-4® analog simulation with TLE. Error bars of <math>\pm 2\sigma</math> are shown.</i>	77
<i>Figure 5.29: FOM comparison for INIPOND: a) Odano backscattering configuration; b) Odano slant penetration configuration; c) simple bunker configuration, slim wall; d) simple bunker configuration, thick wall; e) complex bunker configuration. Results above the red line mark an acceleration of the simulation with respect to analog TLE.</i>	77
<i>Figure 5.30: FOM comparison for AMS: a) Odano backscattering configuration; b) Odano slant penetration configuration; c) simple bunker configuration, slim wall; d) simple bunker configuration, thick wall; e) complex bunker configuration. Results above the red line mark an acceleration of the simulation with respect to analog TLE.</i>	78
<i>Figure 6.31: The basic implementation of the e-TLE fails to increase the number of valid tallies when detectors are small and localized (shown on the right)</i>	81
<i>Figure 6.32: e-TLE with FD. The virtual particles p1, p2 and p3 are sent to the bounding spheres and have a much higher probability of crossing the detectors</i>	83
<i>Figure 6.33: Flowchart of the e-TLE algorithm with FD; w is the particle's weight, E is its energy and <math>\Omega</math> is its direction</i>	85
<i>Figure 6.34: Validation of the estimator over all tested configurations (for details on the geometries, see Chapter 3). The y-axis shows the ratio between the equivalent dose rate calculated with the e-TLE estimator (with and without forced detection) and the one calculated with TRIPOLI-4®'s TLE.</i>	86
<i>Figure 6.35: Performance of the estimator over all tested configurations (for details on the geometries, see Chapter 3). The y axis shows the ratio between the figure of merit of the e-TLE estimator (with and without forced detection) and the figure of merit of TRIPOLI-4®'s TLE.</i>	87
<i>Figure 6.36: Validation and performance of the e-TLE-FD in single-detector simulations.</i>	88
<i>Figure 7.37: Relative KERMA error (left) and average standard deviation <math>\sigma</math> (right) dependence on splitting multiplicity</i>	93

<i>Figure 7.38: 2D, xy view of the six bunker configurations studied: R1, top left; R2, top center; R3, top right; R4, bottom left; R5, bottom center; R6, bottom right.</i>	94
<i>Figure 7.39: Comparison of KERMA maps for the R2 bunker configuration: TLE (left), global e-TLE without splitting (center), se-TLE with M=5 (right)</i>	94
<i>Figure 7.40: Normalized frequency of <math>\zeta</math> values for the R2 bunker configuration, fitted with a standard normal distribution: global e-TLE without splitting (shown left, <math>\mu = 0.057, \sigma = 0.888</math>) and se-TLE with M=5 (shown right, <math>\mu = 0.001, \sigma = 0.893</math>)</i>	95
<i>Figure 7.41: FOM gain maps for the R1 bunker configuration, M=5 (left) and M=0 (right)</i>	96
<i>Figure 7.42: FOM gain maps for the R2 bunker configuration, M=5 (left) and M=0 (right)</i>	97
<i>Figure 7.43: FOM gain maps for the R3 bunker configuration, M=5 (left) and M=0 (right)</i>	97
<i>Figure 7.44: FOM gain maps for the R4 bunker configuration, M=5 (left) and M=0 (right)</i>	97
<i>Figure 7.45: FOM gain maps for the R5 bunker configuration, M=5 (left) and M=0 (right)</i>	98
<i>Figure 7.46: FOM gain maps for the R6 bunker configuration, M=5 (left) and M=0 (right)</i>	98
<i>Figure B.47: Geometric configuration for the mathematical derivation of e-TLE</i>	C



# List of tables

<i>Table I.1: Comparison between different dismantling procedures</i>	XX
<i>Table 3.2: Lead shield height (<math>h</math>) and depth (<math>d</math>) variation inside the bunker</i>	39
<i>Table 3.3: Results for the back-scattering configuration</i>	41
<i>Table 3.4: Results for the slant penetration configuration</i>	41
<i>Table 3.5: Results for the bunker configuration</i>	42
<i>Table 5.6: Results for increasingly larger absorption cross-sections. The last two columns show the gain in efficiency of the e-TLE with respect to the other two estimators, calculated as a ratio between their variances. The errors with respect to the analytical solution <math>\Phi_{D,\text{anal}}</math> are also reported.</i>	70
<i>Table 7.7: Average FOM ratio for air cells behind the lead shield</i>	96



# List of acronyms

1D	1-Dimensional
2D	2-Dimensional
3D	3-Dimensional
ACAB	ACtivation ABacus, Inventory Code for Nuclear Applications (computer program developed by the University of Madrid and designed to perform activation and transmutation calculations for nuclear applications)
ALARA	As Low As Reasonably Achievable
AMS	Adaptive Multilevel Splitting
ANS	American Nuclear Society
ANSI	American National Standards Institute
APOLLO2	Software developed at CEA for deterministic neutronic lattice calculations
ASN	Autorité de Sûreté Nucléaire (Nuclear Safety Authority)
BU	Build-up
CEA	Commissariat à l'énergie atomique et aux énergies alternatives (Commission for Atomic Energy and Alternative Energies)
CIDEC	Software for the calculation of dose rates in scenarios with complex geometries and gamma radiation sources.
COLL	COLLision estimator of TRIPOLI-4®
CPU	Central Processing Unit
CRISTAL	Outil de calcul CEA de la criticité
CRONOS2	Software developed at CEA for deterministic neutronic core calculations
D&D	Decommissioning and Dismantling
DARWIN	CEA calculation tool for radioprotection studies

DEMAIN	DEMantlement et Assainissement des Installations Nucléaires (Decommissioning and Dismantling of Nuclear Installations)
DER	Dose Equivalent Rate
DXTRAN	Deterministic eXponentially-attenuated TRANsport
EAF	European Activation File
EDF	Eléctricité De France (France Electricity)
EET	Efficiency Enhancing Techniques
ET	Exponential Transform
e-TLE	Exponential Track-Length Estimator
FISPACT	Inventory code for neutron-induced activation calculations, mostly used in fusion applications
FLUX_PT	FLUX PoinT (next-event estimator of TRIPOLI-4®)
FOM	Figure Of Merit
G33-GP2	Radiation shielding calculation code based on the point kernel integration method with buildup factors
GEANT	GEometry ANd Tracking (Particle transport code)
GPU	Graphical Processing Unit
IAEA	International Atomic Energy Agency
ICRP	International Commission on Radiological Protection
INB	Installation Nucleaire de Base (Basic Nuclear Installation)
INIPOND	TRIPOLI-4® module for the automatic computation of importance maps
IRSN	Institut de Radioprotection et de Sûreté Nucléaire (Institute for Radioprotection and Nuclear Safety)
JEFF	Joint Evaluated Fission and Fusion
KERMA	Kinetic Energy Released per unit MAss

LINAC	LINear ACcelerator
MC	Monte Carlo
MCNP	Monte Carlo N-Particles (Monte Carlo software for particle transport)
MCNPX	Monte Carlo N-Particles eXtended
MCODE	MCNP-ORIGEN DEpletion program
MFP	Mean Free path
MLC	Multi-Leaf Collimator
NARMER	CEA software for shielding calculations, based on PKI approximation with buildup and albedo factors
OPDEM	Opérations de Préparation au Démantèlement (Operations for Dismantling Preparation)
ORIGEN	Nuclear Fuel Inventory Module for Fuel Cycle Assessment (University of Tennessee at Knoxville)
PANTHERE	Predictions ANd THeoretical analyses of Exposure in REactors
PKI	Point-Kernel Integration
PWR	Pressurized Water Reactor
RAM	Random Access Memory
RANKERN	Radioprotection shielding code based on PKI method, with buildup and albedo factors implementation
SAPHYB	Inventory tool for auto-protected cross section data generated with APOLLO2
SHEM	Santamarina-Hfaiedh Energy Mesh (281-group energy mesh recommended for nuclear cycle studies)
SS	Stainless Steel

TLD	Thermo-Luminescent Dosimeter
TLE	Track-Length Estimator
TRACK	TRACK-length estimator of TRIPOLI-4®
TRIPOLI-4®	CEA software for solving Boltzmann's transport equation with Monte Carlo method
V&V	Validation and Verification
VR	Virtual Reality
VRT	Variance Reduction Technique
VVQI	Validation, Vérification et Quantification des Incertitudes (Validation, Verification and Uncertainties Quantification)

# Introduction

In about a year from the writing of this thesis, on the 2 December 2022, the world will celebrate the 80<sup>th</sup> anniversary of the first-ever controlled chain reaction in an artificial fission reactor. That crucial experiment, carried on by Enrico Fermi and his team, marked the beginning of the nuclear era of energy production. Nowadays, nuclear fission still represents a significant part of the global energy mix, accounting for more than 10% of worldwide electricity generation [1] and thus being the second most utilized zero-emission energy source, right after hydroelectric. However, 65 years after the first grid-connected nuclear plant began activity, the nuclear industry needs to put more and more attention towards an important challenge that will become even more prominent in the following years: **decommissioning**.

With almost 70% of the reactors worldwide being over 30 years of age, and more than 20% being over 40 years old [2], lifetime extensions are surely a great answer to the age problem, but surely not a definitive one. In fact, thanks to their solidity and security, most nuclear plants can continue to function well beyond their initial planned life span of 35-40 years, and the operations of lifetime extension are becoming a standard in the industry. However, in their 2020 annual report [3], the IAEA (*International Atomic Energy Agency*) estimates that between 47 and 130 GW of nuclear electrical generating capacity will have to be decommissioned before the year 2030.

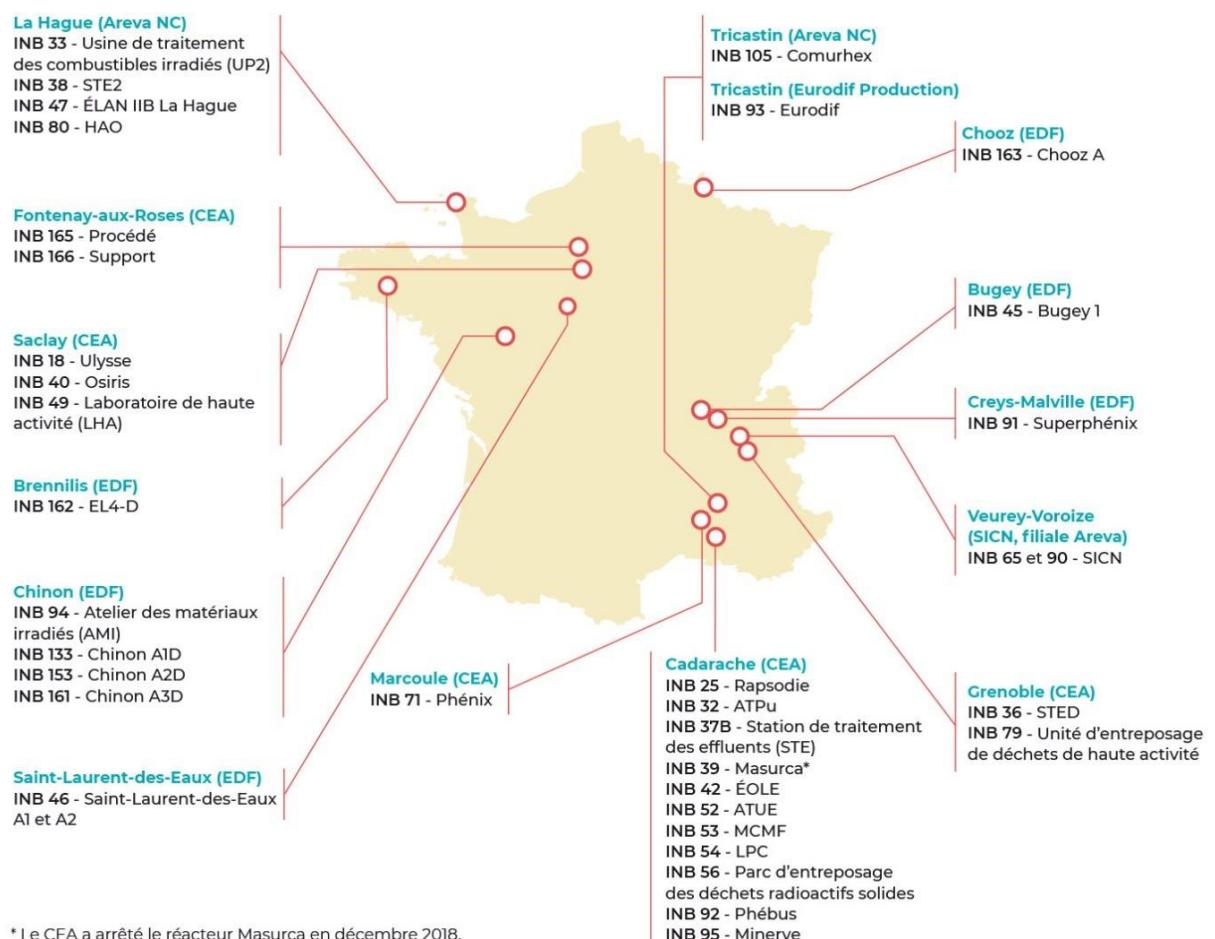
The D&D (*decommissioning and dismantling*) of a nuclear facility [4] begins as soon as the operational phase of the facility ends, and covers all the technical and administrative actions that take place until the facility is either removed or completely requalified for other uses. These actions may involve some or all of the activities associated with dismantling of plant and equipment, decontamination of structures and components, remediation of contaminated ground and disposal of the resulting wastes. It is an extremely long and complex task, involving many different technical aspects (such as economics, safety, neutronics, radiation physics), and it is heavily regulated. That is why researching new methods that would allow simplifying and speeding up the process is a subject of particular interest.

To offer a complete view of the context of this thesis, the first part aims to offer an exhaustive description of the decommissioning framework, with a particular focus on its implementation in France and on the role of simulation tools in D&D activities.

## Nuclear energy in France

With its 56 operating nuclear reactors producing 379 TWh of electricity in 2019 [5], France is the second country in the world for total nuclear energy production. If we consider the nuclear share in the total amount of electricity generated, France is the first country in the word, relying on nuclear for a little more than 70% of its electricity production. This choice of relying heavily on nuclear fission for the country's energy needs dates back to 1974, when a government program laid the basis for the shift from oil to nuclear energy that took place in the following years. Nowadays, France has a cost of electricity well below the European Union's average as well as one of the most decarbonized electricity grids in Europe.

However, with the majority of French nuclear reactors being built in the 80s, the nuclear authorities now have to decide which ones can see their activities prolonged, and which ones are to face decommissioning. In any case, most of the country's nuclear fleet will have to be decommissioned in the next 5 to 10 years. Figure I.1 shows the situation in 2019: 35 nuclear facilities (including reactors, reprocessing facilities and research facilities) were either under dismantling or waiting to be dismantled.



**Figure I.1: French nuclear facilities undergoing dismantling in 2019 (source: ASN website [6])**

Therefore, it is evident that the challenges posed by the decommissioning of nuclear facilities are of particular importance at this moment, in France as in the rest of the world. In the next section, we will offer a more in-depth view of the D&D process: first from an administrative point on view, centered on French legislature, then from a more technical point of view.

## Decommissioning and dismantling (D&D)

As already noted, the D&D of a nuclear installation is an extremely long process, which can take up to 30-40 years for the biggest plants. The process effectively starts when the installation is still functioning (two to three years before the complete shutdown of operations) and it ends when the installation area is either completely free of buildings, or with the remaining buildings fully covered and shielded by radiation, ready to be requalified for other uses. Depending on both the workflow and the final state of the installation, three types of dismantling can be defined, as highlighted in Table I.1:

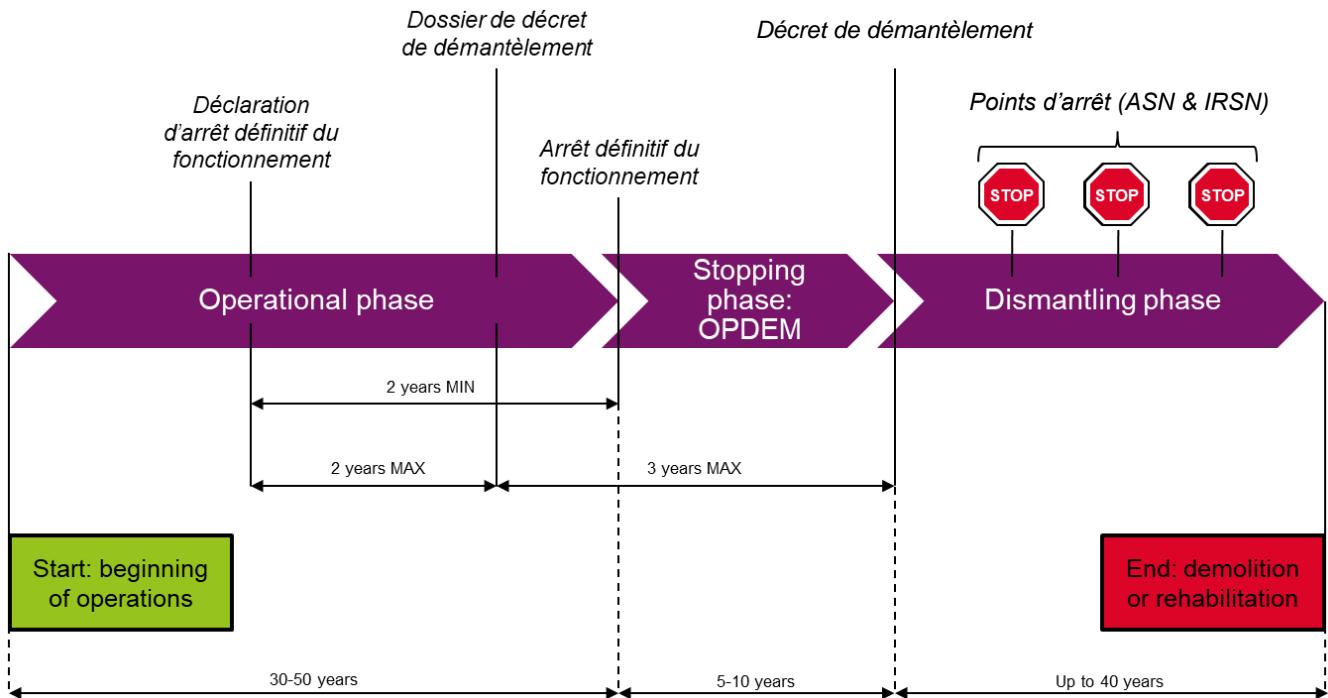
- **Immediate dismantling:** it begins shortly after the shutdown of operations, sometimes after a brief transition period aimed at planning and preparing the dismantling process. This solution is fast and cost-effective, as there are no added cost arising from the maintenance of the facility after the shutdown. Knowledge of the facility is readily available, which simplifies the safety calculations and the planning of operations; the aging of buildings and structures is kept to a minimum; the zone is quickly available to be used again.
- **Deferred dismantling:** between the end of operations and the beginning of dismantling, the installation is monitored and put in a safe state, allowing for a natural decrease of radiation. This procedure is slower and more expensive than the immediate dismantling, due to the added costs of the monitoring period and the slow loss of knowledge of the facility over time. However, it is often safer for the workers, due to the fact that radiation levels are allowed to decrease before the start of operations. Moreover, the birth of new technologies during the monitoring period could lead to better operations.
- **Entombment:** in this limit case, the target installation is just shielded with concrete and possibly buried. A very well-known example is Chernobyl's nuclear plant, which was enclosed in a concrete sarcophagus immediately after the famous incident. Of course, in that case, the extraordinarily dangerous conditions limited the options available for dismantling. Entombment is generally a cost-effective and extremely fast option; however, due to its high unsustainability, it should only be used if absolutely necessary.

**Table I.1: Comparison between different dismantling procedures**

Dismantling procedure	Time	Cost	Safety	Public impact	Sustainability
<b>Immediate dismantling</b>	Fast	Low	High	Good	Very good
<b>Deferred dismantling</b>	Slow	High	Very high	Medium	Good
<b>Entombment</b>	Very fast	Very low	High	Bad	Very bad

Due to its many advantages, the immediate dismantling option is the preferable one and it is usually adopted in France. From the legislative point of view, the procedure is codified in specific laws [7–9] and decrees [10–12]. These documents define in a rigorous way the nuclear installations to which they apply (INB, *Installation Nucléaire de Base*), the various constraints to which the D&D operations are subjected (economic constraints, safety constraints, time constraints), as well as the requirements on the transport, storage and management of radioactive waste. Laws and decrees about nuclear dismantling are enforced by the ASN (*Autorité de sûreté*

*nucléaire*), which emits guidelines for the contractors to follow and operates as a control organ to ensure that all the requirements are met.



**Figure I.2: Diagram of the most important legislative steps of D&D in France**

The timeline for the D&D of an INB in France, as defined by the law and the ASN guidelines, is schematized in Figure I.2, and can be summarized as follows.

1. At least two years before the shutdown of operations, the contractor must send to the ASN a declaration of final shutdown (*déclaration d'arrêt définitif du fonctionnement*). Together with this document, the contractor must include the proposed date of shutdown, a summary of the preliminary operations to be conducted before dismantling, and an updated version of the dismantling plan (*plan de démantèlement*).
2. No more than two years after the *déclaration d'arrêt*, a “dossier” (*dossier de décret de démantèlement*) is examined by the ASN. This document must include an up-to-date version of the dismantling plan, a study of the environmental impact, and a risk analysis.
3. The preliminary operations (OPDEM - *Opérations Préparatoires au DÉMantèlement*) may begin soon after the final shutdown, or even before in some cases. These operations include, for example, the radiological characterisation of the installation, the emptying of the pipe system, the evacuation of the removable dangerous substances in the installation, the removal of most part of the radiological source term (*i.e.* unloading of the core in the case of a nuclear plant).
4. The activity of the installation is declared permanently over (*arrêt définitif du fonctionnement*).
5. If the *dossier de décret de démantèlement* is approved by the ASN, the dismantling operations may begin after the ASN issues a decree (*décret de démantèlement*). The ASN keeps monitoring the operations to ensure that the initial plan is followed and all requirements are met. Several “stopping points” (*points*

*d'arrêt)* may be required both by the ASN and by the IRSN (*Institut de Radioprotection et de Sécurité Nucléaire*) between the different phases of dismantling, to facilitate the control operations.

6. Finally, after the dismantling is completed, if no residual pollution - chemical or radiological - is found on the installation ground, the ASN declares the decommissioning of the zone; this is officialised with the ratification of a *dossier de déclassement* conjointly approved with the French Government.

Besides regulatory constraints, a decommissioning project is also subject to four other curbs: (i) technical, (ii) financial, (iii) organisational and (iv) temporal, and to four main objectives: (i) strategy establishment, (ii) technical choice and feasibility evaluation, (iii) adaptations and modifications of the installations, (iv) definition of a baseline scenario for the security. To answer these different constraints and objectives, a scenario study is applied. The structure of the scenario breaks down in three acts:

- **Knowledge of the initial state** is essential to develop the scenario and manage the radioactive waste stream.
- In a second step, the **elementary operations** and the links between them have to be listed under the form of a flowchart.
- Finally, the **“best” scenario is selected by a multi-criteria analysis**: technical feasibility, waste management method, dosimetric balance sheet, safety and security risks, duration of operations, overall cost of the scenario.

Given the complexity of the whole D&D procedure, it is easy to imagine how important it is for decision-makers to have reliable numerical tools that can help identifying the risks and the opportunities, and ultimately facilitating the choice of the best strategy to pursue.

## Role of simulation tools in dismantling applications

In the dismantling process, the radiological characterization of components is of utmost importance as the isotopic composition will determine the action planning and assign the most appropriate disposal route for nuclear waste. *In-situ* measurement of the radionuclides inventory can be technically difficult because of the high dose rate encountered, access limitations and geometry complexity. In such cases, application of computational approaches based on neutron transport calculations and radionuclide inventory calculation has been shown to be a powerful tool. They allow:

- the control of the evolution of the source term,
- the control and the reduction of the uncertainties through specific measurements,
- the assessment of the classification of waste according to their activities,
- the estimation of the ambient dose rate for decommissioning operations.

From the neutronic and radiological point of view, calculation tools dedicated to D&D have to address two main problems: the evaluation of the so-called **source term**, *i.e.* the spatial and energy distribution of radiation sources in the installation coming from neutronic activation and contamination, and the relative **expected dose rate for operators** inside the domain of interest. The source term evaluation due to activation is usually characterized by

two distinct calculation phases, each one with a dedicated simulation tool: core calculation and flux calculations; results of this step give access to the activation of structures, which gives information about the gamma source distributions; the dose rate evaluation is performed by means of a shielding calculation. Due to the varied nature of this calculation phases, involving transport of different particles (neutrons and photons) and a depletion calculation phase, the complete process usually involves different codes; since the 90s, several calculation systems have been developed by coupling a particle transport code with an isotopic inventory code. Some examples include: MCODE [13], combining MCNP [14] and ORIGEN [15]; a coupling [16] between MCNP and FISPACT [17]; a coupling [18] between MCNP and ACAB [19]. Moreover, a CEA-developed calculation system called DEMAIN (*DEMantèlement et Assainissement des Installations Nucléaires*) [20] was recently established, based on a coupling between TRIPOLI-4® [21] and DARWIN [22] or its successor, MENDEL [23, 24].

We will now look more in detail to the four calculation phases cited above and the main calculation tools associated with each of them in the CEA DEMAIN package.

### **Core calculation (determination of fission sources)**

The first step of a D&D calculation consists in determining the “shape” of the neutron flux inside the core. First of all, a deep knowledge of all relevant data for the core under consideration is needed. This includes: the detailed geometry of the core; the microscopic cross sections of all isotopes inside; the isotopic compositions of fuel and structural elements. Once these factors are known, fission sources can be determined with a full Monte Carlo (MC) core model by using the TRIPOLI-4® [21] CEA code or with a deterministic scheme based on codes such as APOLLO2 [25].

If the deterministic option is chosen, core calculation relies on two steps known respectively as lattice and core calculations [26]. The lattice calculations are performed with the deterministic neutron transport code APOLLO2, 281 SHEM energy groups [27] and CEAV5.1.2 cross section library which comes from JEFF3.1.1 evaluation [28]. The collapsed and homogenized cross sections obtained with APOLLO2, called SAPHYB libraries, are then automatically handled by the CEA 3D deterministic code CRONOS2 [29] for core calculations.

Conversely, if the stochastic option is chosen, a full-core 3D continuous energy MC criticality simulation is performed with TRIPOLI-4® in order to compute the neutron flux distribution.

### **Flux calculations**

Core calculation results are then used in the MC based neutron transport step to describe the reactor core and the associated sources. For this step, additional knowledge of the plant is needed, namely the geometry and isotopic composition of all the structures outside of the core which could be activated. The calculations are performed with TRIPOLI-4® version 10.2 and CEAV5.1.2 data library based on JEFF3.1.1 [28] in order to determine the neutron flux in the structures of interest. The neutron flux is calculated at the nominal power rating conditions and each flux is homogenized in a limited number of energy groups (TRIPOLI-4® 315 groups). The precise knowledge of neutron flux distribution in both energy and space is paramount to activation calculation conducted in the next step.

Once the neutron flux is known with a good level of approximation, this information can be combined with operational data relative to the plant usage and fuel burnup to obtain the **total neutron fluence distribution**, a quantity that is directly proportional to the activation of structural materials.

## Activation calculations

Once the flux is known, the activation of all structural materials must be determined. **Activation** is a process in which a stable material becomes radioactive after exposure to a neutron source. This usually happens after the stable nucleus captures the neutron, therefore entering an excited state. It is quite intuitive to note that, in most cases, activation is directly proportional both to neutron flux and to the total time of exposure; this is why the quantity  $\Phi \cdot t$  (called neutron fluence) is necessary in this phase. In addition to the previously evaluated neutron flux distribution, some more data relative to plant operations are required: notably, a workflow diagram describing the plant workload throughout the years and a detailed description of the isotopic composition of structural materials, including the impurity levels in steel and concrete. This is particularly important because even elements with a very low concentration can greatly contribute to total radioactivity after activation.

In particular, when only the thermal neutron flux is considered for calculations, the most important reaction is  $(n,\gamma)$  and the greatest activities correspond to the following radionuclides:  $^{59}\text{Ni}$ ,  $^{63}\text{Ni}$ ,  $^{55}\text{Fe}$  and  $^{60}\text{Co}$ . If the complete spectrum of neutron flux is considered for calculations, nearly all reactions are considered and the radioisotopes produced are the following:  $^3\text{H}$ ,  $^{14}\text{C}$ ,  $^{28}\text{Al}$ ,  $^{36}\text{Cl}$ ,  $^{41}\text{Ca}$ ,  $^{45}\text{Ca}$ ,  $^{54}\text{Mn}$ ,  $^{46}\text{Sc}$ ,  $^{55}\text{Fe}$ ,  $^{60}\text{Co}$ ,  $^{59}\text{Ni}$ ,  $^{63}\text{Ni}$ ,  $^{65}\text{Zn}$ ,  $^{94}\text{Nb}$ ,  $^{93}\text{Mo}$ ,  $^{108m}\text{Ag}$ ,  $^{110m}\text{Ag}$ ,  $^{152}\text{Eu}$ ,  $^{154}\text{Eu}$  and  $^{178}\text{Hf}$ . As we will see in Chapter 3,  $^{60}\text{Co}$ , in particular, has a major role in the studies presented in this thesis. This isotope is produced by capture reaction on  $^{59}\text{Co}$ . In steels,  $^{59}\text{Co}$  is present from few hundred of part per million (ppm) in carbon steels (reactor pressure level for example) to thousands of ppm in stainless steels.  $^{60}\text{Co}$  is a  $(\beta,\gamma)$  emitter (1173, 1332 keV  $\gamma$  rays) and plays a major role in the external exposition of operators, especially during D&D activities.

If the detailed composition of materials is not available, a maximisation criterion is adopted to over-estimate the concentration of activation-sensible materials due to safety reasons. Depletion codes such as CEA's DARWIN or MENDEL package [22, 23], which solve Bateman's differential equations, are employed at this stage of the calculation. The nuclear data library used for activation calculation is EAF-2001 [30].

After this step, results are compared with measurements data coming from the plant, the assumptions done in the simulation setup (usually the impurity level in structures or the historical data) are re-assessed if necessary, then the calculation is updated iteratively until a reasonable accordance between experimental data and simulation results is achieved.

## Shielding calculations

Once all the radiation sources have been identified, and their spatial and energy distribution is known, the final part of the radiological analysis can be performed. Shielding calculations allow to retrieve the expected dose rate in specific points of the domain considered, knowing the source term distribution and the geometrical and material details (attenuation coefficients, compositions) of the installation. This is a crucial step in the planning phase of the dismantling process, as it allows taking safety-related decisions on the working personnel. For example, if the

dose rate levels in a certain room are too high, remote intervention through teleoperation may be chosen over direct human action.

The simulation methods used to conduct shielding calculations can be divided into two main groups: stochastic methods and deterministic methods. These two approaches to the transport problem will be presented in a rigorous way in the following chapters. As a general introduction, we can describe them like this:

- **Deterministic methods** are based on a set of assumptions and simplifications that greatly reduce the complexity of the transport problem, allowing for a computationally fast analytical solution. The drawback lies in the fact that, together with the assumptions, we introduce a domain of applicability, outside of which the analytical solution becomes wrong.
- **Stochastic/probabilistic methods, or Monte Carlo methods**, are extremely accurate and can give a result within any level of confidence specified by the user, provided they run for a sufficient amount of time. However, particularly in complex geometries, this amount of time can easily become too large to be considered feasible, and some additional techniques have to be employed to reduce it.

The main work of this thesis, which will be introduced in the following chapter, consists precisely in the development of one of these techniques for the acceleration of Monte Carlo shielding calculations.

## Thesis objectives and summary

Having introduced the notions of decommissioning and dismantling, shielding calculations, and MC methods, we can now describe the context and objective of this thesis. Working on CEA's reference Monte Carlo code TRIPOLI-4®, the main objective is to develop a new calculation tool with the purpose of reducing computation times for shielding calculations in complex geometries. This will provide an alternative solution approach for problems that are difficult to treat with deterministic methods, while still retaining an advantage in terms of computation time with respect to classic Monte Carlo calculations.

This new tool, called exponential track-length estimator (*e*-TLE), is inspired by the one developed in the code GEANT4 for medical applications [31].

The thesis is organized as follows.

In Part I, the main theoretical notions required to understand the scope of the thesis are introduced. Chapter 1 is devoted to the problem of particle transport, with an in-depth look to how the problem is usually treated in the nuclear field and, most notably, a presentation of deterministic and stochastic approaches. Chapter 2 demonstrates these approaches by applying them to the solution of a simple mathematical model.

Part  focuses on the limits of traditional deterministic and stochastic approaches when dealing with

particular geometries. In Chapter 3, some of these problematic configurations are presented. Chapter 4 tackles the problem of acceleration techniques for Monte Carlo simulations, by giving a broad presentation of the most important ones and ultimately focusing on the code used in this work, TRIPOLI-4®.

Finally, Part III presents the bulk of the work, the development of the *e*-TLE. In Chapter 5, the estimator is described both from a theoretical point of view and from a practical one, with its implementation in TRIPOLI-4®. Chapter 6 and Chapter 7 introduce two slightly different versions of the algorithm, optimized respectively for local and global dose computations.

Conclusions and perspectives are resumed in the final section of the document.



---

PART I

THEORETICAL CONTEXT

---



The aim of this first part is to present and describe the theoretical foundation upon which this thesis work is built. Of course, it is not meant to be an exhaustive dissertation on particle transport physics, a vast and complex topic about which entire books have been written. Rather, it aims at providing the reader with tools that can help them to better understand the rest of the thesis. The first chapter presents the theory behind particle transport and the two main approaches adopted to study it, the deterministic one and the stochastic one. The second chapter offers a mathematical description of the problem and a first approach to a simplified model solution.



# Chapter 1

## Particle transport

The study of how uncharged particles move through matter, generally called particle transport, is a field of primary importance in nuclear reactor science. Actually, a nuclear reactor can be described in a very elementary way as an assembly of fuel elements heated by a swarm of neutrons rapidly moving inside the core and causing fissions. The knowledge of the physical laws governing these movements is of course paramount to operating the reactor. On the other hand, when dealing not with the active operations, but with the end of life of a nuclear installation, neutrons are no longer the important particles that need to be studied and tracked as they move around the facility. In this case, we are rather interested in photons, as they are the primary particles emitted from activated materials in the buildings (see Introduction).

Photons and neutrons, however, are both uncharged particles and they are therefore unaffected by electromagnetic forces such as Coulomb force. This fundamental common characteristic allows us to study their transport in a similar way, by using a powerful mathematical tool called the **Boltzmann (or transport) equation**. This chapter offers an introduction to this important equation and a closer look at the two main approaches for solving it: deterministic and probabilistic methods.

### 1.1. The Boltzmann equation for particle transport

To study the transport of uncharged particles through matter, a convenient approach could be to look at the microscopic quantities related to each single particle. We start from two main assumptions:

- the problem is linear, meaning that particles do not interact with one another but only with the environment they move in;
- particles have no dimensions, and they coincide with a point in space.

Since we want to write a general equation, we are not interested in the *type* of particle (neutron, photon) but rather in the basic quantities that allow us to univocally identify it and differentiate it from other particles. As such, we will define:

- $\mathbf{r}$  as the particle's position in space,
- $\boldsymbol{\Omega}$  as the particle's direction of movement, represented by a unit vector,
- $E$  as the particle's energy, which is directly related to its velocity  $\mathbf{v}$ :  $E = 1/2 m\mathbf{v}^2$  (non-relativistic hypothesis).

With these three quantities, consisting of seven unknowns, we can univocally identify a particle's position in the phase space, which represents the universe of all possible states a particle can be in. It should be noted that here we also introduce a formalism that will be used throughout the rest of this thesis: all vector quantities are written in boldface. If we introduce the concept of **particle density**  $n(\mathbf{r}, \boldsymbol{\Omega}, E)$  as the number of particles per unit volume in  $\mathbf{r}$ , per unit energy in  $E$ , and per unit solid angle in direction  $\boldsymbol{\Omega}$ , we can introduce one of the most important quantities in particle transport, the **angular flux**  $\Phi_{\boldsymbol{\Omega}}(\mathbf{r}, \boldsymbol{\Omega}, E)$ :

$$\Phi_{\boldsymbol{\Omega}}(\mathbf{r}, \boldsymbol{\Omega}, E) = n(\mathbf{r}, \boldsymbol{\Omega}, E)\mathbf{v} \quad (1.1)$$

where  $\mathbf{v}$  is the particles' velocity. By integrating over all possible directions, we define the **scalar flux**  $\Phi(\mathbf{r}, E)$  as:

$$\Phi(\mathbf{r}, E) = \iint \Phi_{\boldsymbol{\Omega}}(\mathbf{r}, \boldsymbol{\Omega}, E) d^2\Omega \quad (1.2)$$

It is worth mentioning that these quantities would depend, in principle, also on time. However, we are assuming of working with a system that has reached equilibrium (which is usually the case for radiation shielding problems), so time will not appear in our equations.

Now that we have defined the angular and scalar particle flux, we can proceed to write a balance equation by looking at an elementary volume in phase space centred around  $P(\mathbf{r}, \boldsymbol{\Omega}, E)$  and considering all the particles entering and leaving the volume. We will therefore define:

- **a streaming term**, given by the net sum of particles entering and leaving the elementary volume *via* their displacements; it can be written as:

$$-\boldsymbol{\Omega} \cdot \nabla \Phi_{\boldsymbol{\Omega}}(\mathbf{r}, \boldsymbol{\Omega}, E) \quad (1.3)$$

- **a disappearance term**, given by all particles undergoing an interaction inside the elementary volume and therefore changing their energy and/or direction thus leaving the volume: it can be written as :

$$-\Sigma_t(\mathbf{r}, E)\Phi_{\boldsymbol{\Omega}}(\mathbf{r}, \boldsymbol{\Omega}, E) \quad (1.4)$$

where  $\Sigma_t(\mathbf{r}, E)$  is the **macroscopic total cross-section**, representing the probability per unit distance for a particle of energy  $E$  of undergoing an interaction at point  $\mathbf{r}$ ;

- **an interaction term** corresponding, conversely, to a contribution given by particles that had a different energy  $E'$  and/or direction  $\boldsymbol{\Omega}'$ ; by undergoing an interaction inside the elementary volume, they end up having energy  $E$  and direction  $\boldsymbol{\Omega}$ . This term can be written as :

$$\iiint \Sigma_s(\mathbf{r}, E') f_s(\mathbf{r}, \Omega' \rightarrow \Omega, E' \rightarrow E) \Phi_\Omega(\mathbf{r}, \Omega', E') d^2\Omega' dE' \quad (1.5)$$

with  $\Sigma_s(\mathbf{r}, E') f_s(\mathbf{r}, \Omega' \rightarrow \Omega, E' \rightarrow E)$  being the macroscopic differential scattering cross section, defined as the product between the macroscopic cross section  $\Sigma_s(\mathbf{r}, E')$ , representing the probability per unit distance for a particle of energy  $E'$  of undergoing a scattering at point  $\mathbf{r}$  and  $f_s(\mathbf{r}, \Omega' \rightarrow \Omega, E' \rightarrow E)$ , representing the probability density for said particle of changing direction from  $\Omega'$  to  $\Omega$  and changing energy from  $E'$  to  $E$ ;

- finally, a **source term** accounting for any external sources spontaneously generating particles inside the elementary volume: it can be written as  $S(\mathbf{r}, \Omega, E)$ .

By equating arrival and departure terms, we can now write what is called the **integro-differential form of the Boltzmann transport equation**:

$$\Omega \cdot \nabla \Phi_\Omega(\mathbf{r}, \Omega, E) + \Sigma_t(\mathbf{r}, E) \Phi_\Omega(\mathbf{r}, \Omega, E) = \iiint \Sigma_s(\mathbf{r}, E') f_s(\mathbf{r}, \Omega' \rightarrow \Omega, E' \rightarrow E) \Phi_\Omega(\mathbf{r}, \Omega', E') d^2\Omega' dE' + S(\mathbf{r}, \Omega, E) \quad (1.6)$$

Unfortunately, with the mathematical tools that we know today it is not possible to find an exact solution to an equation of this kind, including both a derivative and an integral of the unknown quantity  $\Phi_\Omega(\mathbf{r}, \Omega, E)$ . To search for an analytical solution, one must introduce some assumptions or simplifications: this approach is the one followed by deterministic methods, which will be presented in Section 1.2. On the other hand, stochastic methods, presented in Section 1.3, search for an approximation of the solution by directly replicating the physics of the problem under consideration.

## 1.2. Modelling deterministic approach

As we said, the Boltzmann equation cannot be solved analytically except in simplified physical configurations. Depending on the type of problem studied, one can introduce some assumptions to re-write the equation in a simplified form, which allows to search for an approximation of the solution. Any approach following this method is generally called a **modelling deterministic approach**, because solutions found in this way are purely analytical and do not depend on probability.

Of course, different problems require different sets of assumptions, that lead to different solutions. For example, in the nuclear field it is common to search for the neutron flux distribution in the core of the reactor, as this determines the power distribution and, consequently, the temperature distribution in the reactor. To perform this calculation, one needs to solve the transport equation for neutrons in a multiplying medium. This problem is usually solved through some kind of discretization of the phase space, turning the integral of Equation (1.6) in a simple summation described by a system of linear equations.

For shielding calculations, however, the problem to be solved is different, notably because there are no fission events creating new particles to be transported. This means that interactions can either be absorptions or scatterings and, depending on the specific application, particle transport can be reduced to a straight-line attenuation problem. The usual method to solve this kind of problems is the Point-Kernel Integration (PKI) method, an approach that takes a completely different route to find a solution that, albeit approximated, can be extremely fast to compute.

For the purpose of this thesis, we will focus mainly on photon transport for shielding applications: therefore, a more in-depth view of the PKI method is offered in this section.

### 1.2.1 The Point-Kernel Integration (PKI) method

The most widely used modelling deterministic method in shielding calculations is the so-called Point-Kernel method [32]. It is a macroscopic approach that enhances calculation performance by considering any radiation source as an ensemble of independent beam-like point sources. Interaction between radiation and medium is modelled with the use of macroscopic coefficients, notably the linear attenuation coefficient  $\mu(E, m)$ , which depends on the radiation energy and the medium composition. The direct contribution to the total photon flux at point  $P$  given by a monokinetic point isotropic photon source with energy  $E$  and intensity  $A(E)$  measured in [photons · cm<sup>-3</sup> · s<sup>-1</sup>] is:

$$\Phi_{\gamma}^u(E) = A(E) \frac{e^{-\sum_{m=1}^N \mu(E, m) x_m}}{4\pi r^2} \quad (1.7)$$

where  $r$  is the distance between the source and point  $P$ , and  $x_m$  is the distance travelled in each material. The uncollided flux  $\Phi_{\gamma}^u(E)$ , where the apex  $u$  stands for “uncollided”, corresponds to the photons which arrived at point  $P$  without having suffered any scattering. It can be shown that the  $\Phi_{\gamma}^u(E)$  expression can be rigorously derived from the resolution of the Boltzmann transport equation in a purely absorbing medium [see Chapter 2].

The dose equivalent rate (*DER*) for a mono-energetic source of intensity  $A(E)$  in a volume is thus determined as [33]:

$$DER(E) = k(E) \times \iiint_V BU(E, \mu(E) \times x) \times \Phi_{\gamma}^u(E) \times d^3V \quad (1.8)$$

with  $k(E)$  the conversion factor from gamma flux to dose equivalent rate taken from the ICRP (*International Commission on Radiological Protection*) [34],  $x$  the shield thickness encountered by  $\gamma$  rays along a straight line and *BU* the **build-up factor** which models the scattering of photons in the shield.

### 1.2.2 Build-up factors

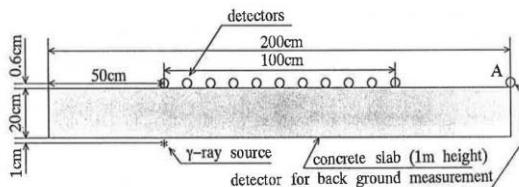
The build-up factors are used to estimate the scattered radiations in matter. They are defined as the ratio between the total flux and the direct (uncollided) flux. In CEA applications [35], they are tabulated for simple elements and for 195 energy groups between 15 keV and 15 MeV and up to 40 mean free paths in the ANSI/ANS-6.4.3 [36]. The knowledge of the build-up factor for a reference geometry and radiation energy allows to retrieve the total dose rate starting from the direct (*i.e.* uncollided) one.

The NARMER point-kernel code [37], developed at CEA as the successor of the MERCURE-6 code, includes a routine for the calculation of build-up factors in the case of multi-layer shields. The algorithm, initially developed by Assad *et al.* [38] and improved by Suteau and Chiron [39], essentially applies a procedure that calculates the equivalent build-up of two consecutive layers as if they were a single, homogeneous layer. Then, the procedure is applied iteratively until all the layers have been homogenized.

The use of build-up factors, shows its limits in the case of high-scattering environments: for instance, when the radiation path has a strong inclination with respect to the shield. For example, the NARMER-based code

PANTHERE becomes extremely unreliable in problems with high inclination angles between radiation and shield [33], especially with slant angles of more than  $73^\circ$  (as shown in Figure 1.3). Indeed, the usage of NARMER and NARMER-based codes is restricted to domains where the assumptions of equations (1.7) and (1.8) are pertinent, thus:

- **a minimum distance** between source and target,
- **a regular geometry** between source and target,
- only **direct contribution** is taken into account (reflection on walls, ceiling and floor can be neglected),
- the domain of validity of BU factor is limited in energy and in penetration lengths up to 50 MFP [40].



Slant angle ( $^\circ$ )	Measure (mR/h)	TRIPOLI4 (mR/h)	FSD (%)	MCNP (mR/h)	FSD (%)	PANTHEREv2 (mR/h)	PV2/measure (%)
0	1,24E+03	1,43E+03	0,23	1,37E+03	3,42	1,35E+03	9%
25	9,38E+02	9,63E+02	0,26	9,78E+02	3,85	9,50E+02	1%
43	3,85E+02	3,77E+02	0,40	3,64E+02	4,09	4,03E+02	5%
54	1,16E+02	1,16E+02	0,66	1,27E+02	2,24	1,38E+02	19%
62	3,55E+01	3,38E+01	1,06	3,93E+01	3,27	4,28E+01	21%
67	1,10E+01	1,02E+01	2,18	1,09E+01	4,54	1,28E+01	16%
70	3,52E+00	2,86E+00	2,77	3,56E+00	4,94	3,83E+00	9%
73	1,23E+00	1,09E+00	5,51	1,27E+00	4,22	1,17E+00	-5%
77	2,58E-01	1,89E-01	7,63	2,46E-01	3,02	1,01E-01	-61%
78	1,20E-01	9,92E-02	9,34	1,54E-01	4,32	3,18E-02	-74%

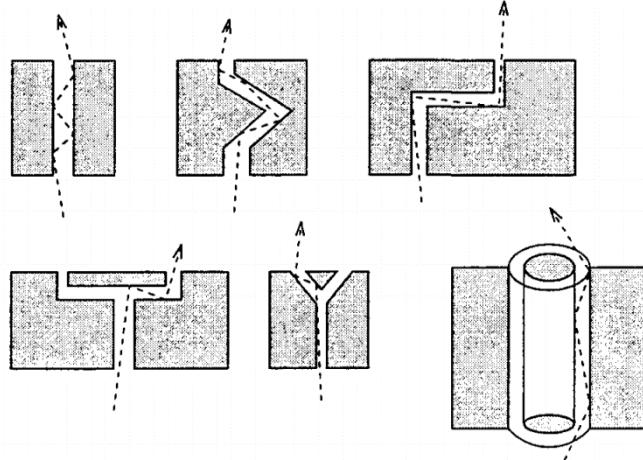
**Figure 1.3: Comparison of computation results (TRIPOLI4® vs PANTHEREv2) with measurement results for the «slant penetration benchmark»**

Vela *et al.* [41] developed a point-kernel code called CIDECA with greatly improved geometric modeling capabilities, able to treat complex geometries. CIDECA allows to evaluate dose rate in realistic reactor environments and with a good approximation, but it retains the drawbacks of point-kernel methodology, resulting unreliable in some cases where scattering or back-scattering is particularly relevant – for example, near the corners of a thick concrete wall.

### 1.2.3 Albedo factors

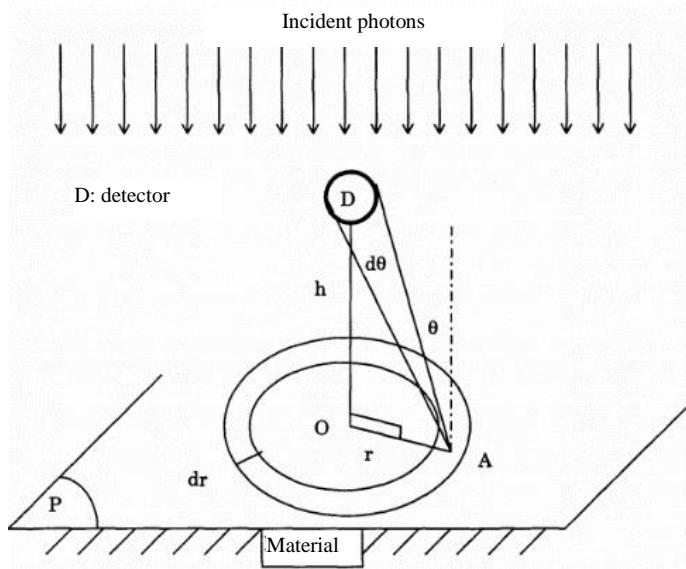
When dealing with nuclear power plants (reactor pit, primary pumps, steam generators) and with other nuclear facilities, such as reprocessing plants (piping systems, gaps between shielding protections, ventilation ducts), we can often find configurations characterized by large empty spaces filled with air. These configurations, usually called **streaming configurations** (see Figure 1.4), often constitute a problem from a radioprotection perspective, as they present a preferred passageway for photons and decrease the efficiency of radiation shields. As we saw in the previous chapter, the build-up approximation is based on the assumption that the main contribution to the dose in the detector arises from photons that travel in a straight line from the source and are attenuated throughout. However, in the case of streaming, this assumption does not hold, as the dominant contribution to detector dose comes instead from photons traveling along weaknesses in the attenuating materials, changing their direction multiple times by bouncing on the walls around them.

In this type of configurations, a different calculation factor is used, called **albedo factor**.



**Figure 1.4: Photon reflection in streaming configurations**

We could therefore say that the straight-line attenuation method with build-up factors is well suited for dense environments, while the albedo calculations are to be preferred in the case of partially-void configurations.



**Figure 1.5: Geometric arrangement for the definition of the albedo**

Similarly to the build-up factor, the albedo is defined as the ratio between the particle flux reflected by a surface and the particle flux impinging on that surface. Initially, the albedo was conceived to study the reflection of light waves on surfaces; this process shares some similarities with the process of gamma photons traveling through matter and being scattered, changing direction and energy.

To define the albedo more rigorously [42] (see Figure 1.5), we start from a monodirectional, monokinetic stream of photons emitted from an infinite plane. We then consider a detector  $D$  placed in front of the source, and another infinite plane parallel to the source. The number of photons arriving to detector  $D$  after being reflected by an annulus of radius  $r$  can be written as:

$$dN_{ref} = n_0 \cdot 2\pi r dr \frac{\Delta\sigma}{r^2 + h^2} \int_0^{E_0} N(\theta, E) dE \quad (1.9)$$

where  $n_0 \cdot 2\pi r dr$  is the average number of photons in the annulus,  $\Delta\sigma/(r^2 + h^2)$  is the solid angle subtended by the detector from point  $A$ , and  $N(\theta, E)$  is the distribution function of the scattered photons, *i.e.* the percentage of photons reflected with angle  $\theta$  and energy  $E$ . Integrating over  $r$  we get:

$$N_{ref} = n_0 \cdot 2\pi \cdot \Delta\sigma \int_0^\infty \int_0^{E_0} N(\theta, E) \frac{r}{r^2 + h^2} dr dE \quad (1.10)$$

In a study by Chucas and Curl [43] it is shown how the domain of application of point-kernel codes (in particular, the RANKERN code) can be enlarged by approximating reflection and scattering events. In this case, reflection is taken into account through an albedo coefficient; scattering events are treated as immediate events which change the energy and direction of the beam, so they divide the beam trajectory in two or more parts which are then treated separately by the code. Results show good approximation in the case of a single scattering event being considered, but the method rapidly becomes unfeasible from a computational point of view in the case of more than two scatterings.

The NARMER code includes an albedo module which can efficiently model a single reflection of gammas on a surface; the albedo coefficient in this case is split into two different contributions, calculated separately, accounting respectively for two processes, Compton scattering and pair production. The total albedo coefficient is the sum of these two contributions. Recently, a double-differential albedo module has been included in the code; it is defined as double-differential as it depends on both reflection angle and energy, as opposed to the simpler albedo coefficient which is only dependent on reflection angle. This more refined albedo calculation allows the code to deal with multiple reflections in sequence, which would not be possible with the single-differential albedo, as the incident energy distribution of gammas at the  $n^{\text{th}}$  reflection is directly related to the reflected energy distribution at the  $n-1^{\text{th}}$  reflection.

NARMER also allows the conjoint use of BU and albedo to evaluate cases where photons are attenuated and reflected, in any order. However, this can have a heavy impact on the computation time, as well the precision of the results.

#### **1.2.4 Decommissioning and dismantling applications**

Due to the very low computational burden of the point-kernel method, allowing for fast calculations, in recent years many researchers have investigated its possible implementation in real time and virtual reality tools. The use of VR (*Virtual Reality*) tools [44, 45] for the training of specialized radioprotection workers, or for the planning of dismantling operations, has proven to be quite effective, as it provides an immersive experience allowing the user to operate in a realistic, yet completely safe environment.

Caracena *et al.* [46] proposed an algorithm for real time VR dose computation, with a mesh utility able to create non-regular, optimized meshes. Szoke *et al.* [47] presented a similar tool, both in VR and as a 3D visualizer. More recently, Chao *et al.* [48, 49] developed a method specialized for decommissioning operations which is able to dynamically change the geometry of the model. This is an important asset in decommissioning problems, where a large number of cutting and demolition operations can frequently change the geometry of the studied domain and produce objects with irregular shapes.

Another interesting research by Chao *et al.* [50] presents an algorithm for the calculation of the less-exposed walking path for workers in radioactive environments. While this does not directly involve dose rate calculation

(in this case, the radiation field is given as an input to the algorithm), it is a good example of how VR technology and real time dose calculation could greatly improve the training experience of specialized workers.

It should be noted that, since the VR implementation is quite recent and still far to be implemented in real D&D applications, these real time dose calculations have yet to be fully validated and verified in realistic dismantling-related work scenarios. Therefore, their limits of applicability still have to be defined in a formal way.

### 1.3 Stochastic approach: the Monte Carlo method

As noted at the beginning of this chapter, to solve the problem of photon transport we need to find a solution to Equation (1.3), which is impossible to do in an analytical way without introducing approximations. However, a reformulation of the Boltzmann equation leads to another type of numerical solution.

In fact, we could also see the angular flux  $\Phi_{\Omega}(\mathbf{r}, \Omega, E)$  as the result of the addition of two terms:

- The first term consists of particles emitted from a generic source term in  $\mathbf{r}' = \mathbf{r} - s\Omega$  with energy  $E$ , arriving in  $\mathbf{r}$  without undergoing any interaction. We will denote them as  $S(\mathbf{r} - s\Omega, \Omega, E)$ . The probability of traveling from  $(\mathbf{r} - s\Omega, \Omega, E)$  to  $(\mathbf{r}, \Omega, E)$  can be written, using macroscopic cross sections, as  $\exp(-\int_0^s \Sigma_t(\mathbf{r} - s'\Omega, E) ds')$ . By integrating over every possible distance  $s$ , we can write the first term as:

$$\int_0^\infty \exp\left(-\int_0^s \Sigma_t(\mathbf{r} - s'\Omega, E) ds'\right) S(\mathbf{r} - s\Omega, \Omega, E) ds \quad (1.11)$$

- The second term consists of particles of energy  $E'$  and direction  $\Omega'$  that underwent an interaction in  $\mathbf{r}' = \mathbf{r} - s\Omega$ , exiting the interaction with energy  $E$  and direction  $\Omega$ . We can write these particles down as  $\iiint \Sigma_s(\mathbf{r} - s\Omega, E') f_s(\mathbf{r} - s\Omega, \Omega' \rightarrow \Omega, E' \rightarrow E) \Phi_{\Omega}(\mathbf{r} - s\Omega, \Omega', E') d^2\Omega' dE'$  (see equation (1.5)). By following the same reasoning as before, we multiply by the probability of arriving in  $(\mathbf{r}, \Omega, E)$  without undergoing other interactions and integrate over every  $s$ . We can therefore write the second term as:

$$\int_0^\infty \exp\left(-\int_0^s \Sigma_t(\mathbf{r} - s'\Omega, E) ds'\right) \iiint \Sigma_s(\mathbf{r} - s\Omega, E') f_s(\mathbf{r} - s\Omega, \Omega' \rightarrow \Omega, E' \rightarrow E) \Phi_{\Omega}(\mathbf{r} - s\Omega, \Omega', E') d^2\Omega' dE' ds \quad (1.12)$$

We can now re-write the equation for the angular flux in what is called the **integral form of the Boltzmann transport equation**:

$$\begin{aligned} \Phi_{\Omega}(\mathbf{r}, \Omega, E) &= \int \exp\left(-\int_0^s \Sigma_t(\mathbf{r} - s'\Omega, E) ds'\right) S(\mathbf{r} - s\Omega, \Omega, E) ds \\ &+ \int \exp\left(-\int_0^s \Sigma_t(\mathbf{r} - s'\Omega, E) ds'\right) \iiint \Sigma_s(\mathbf{r} - s\Omega, E') f_s(\mathbf{r} - s\Omega, \Omega' \rightarrow \Omega, E' \rightarrow E) \Phi_{\Omega}(\mathbf{r} - s\Omega, \Omega', E') d^2\Omega' dE' ds \end{aligned} \quad (1.13)$$

Now, just to improve readability, we make a few notation changes:

- we introduce the **collision density**  $\psi(\mathbf{r}, \Omega, E)$  :

$$\psi(\mathbf{r}, \Omega, E) = \Sigma_t(\mathbf{r}, E) \Phi_{\Omega}(\mathbf{r}, \Omega, E) \quad (1.14)$$

indicating the total number of interactions at point  $(\mathbf{r}, \Omega, E)$  ;

- and the **displacement operator**  $\mathbb{T}(\mathbf{r}' \rightarrow \mathbf{r}, \Omega, E)$  [51]:

$$\mathbb{T}(\mathbf{r}' \rightarrow \mathbf{r}, \Omega, E) d\mathbf{r} = \Sigma_t(\mathbf{r}, E) \exp\left(-\int_0^s \Sigma_t(\mathbf{r} - s' \Omega, E) ds'\right) ds' \quad (1.15)$$

which can be seen as the product of two probabilities: the probability for a particle to fly without interactions from  $\mathbf{r}'$  to  $\mathbf{r}$ , represented by  $\exp(-\int_0^s \Sigma_t(\mathbf{r} - s' \Omega, E) ds')$ , and the probability density of undergoing an interaction in  $\mathbf{r}$ , represented by the macroscopic total cross section  $\Sigma_t(\mathbf{r}, E)$ . In other words, it is the probability for a particle colliding in  $\mathbf{r}'$  to experience its next interaction in  $\mathbf{r}$ .

- Next, we define the **collision operator**  $\mathbb{C}(\mathbf{r}, \Omega' \rightarrow \Omega, E' \rightarrow E)$  as:

$$\mathbb{C}(\mathbf{r}, \Omega' \rightarrow \Omega, E' \rightarrow E) = \frac{\Sigma_s(\mathbf{r}, E') f_s(\mathbf{r}, \Omega' \rightarrow \Omega, E' \rightarrow E)}{\Sigma_t(\mathbf{r}, E')} \quad (1.16)$$

indicating the probability for a particle colliding in  $\mathbf{r}$  of changing its direction from  $\Omega'$  to  $\Omega$  and its energy from  $E'$  to  $E$ .

- Finally, if we define the **transport operator**  $\mathbb{K}(P' \rightarrow P)$  as the product of the displacement and collision operators,

$$\mathbb{K}(P' \rightarrow P) = \mathbb{T}(\mathbf{r}' \rightarrow \mathbf{r}, \Omega, E) \mathbb{C}(\mathbf{r}, \Omega' \rightarrow \Omega, E' \rightarrow E) \quad (1.17)$$

we can finally re-write Equation (1.13) in a more compact form:

$$\psi(P) = \int \mathbb{K}(P' \rightarrow P) \psi(P') dP' + \int \mathbb{T}(P' \rightarrow P) S(P') dP' \quad (1.18)$$

This final formulation underlines very well that the collision density in a given point  $P$  can be deduced by the source particles traveling to  $P$  without interacting, and by all particles which had undergone one or more interactions before arriving in  $P$ . This suggests that by simulating the transport process of large numbers of single particles, interaction after interaction, the solution to this equation can be approximated.

### 1.3.1 Introduction to Monte Carlo methods

Monte Carlo methods are a class of numerical statistical methods originally developed in the frame of the Manhattan Project to treat the problem of neutron transport. In general, they are based on the law of large numbers, which states that the sample average of a certain number  $n$  of trials will converge to the expected value when  $n \rightarrow \infty$ . The essential idea behind Monte Carlo methods is to be able to numerically evaluate arbitrarily complex stochastic processes by starting from basic known probability distributions.

This is also true for Monte Carlo applied to particle transport, more specifically photon transport in our case. As we saw in Equation (1.18), we can decompose the solution to the transport equation by considering the contributions of single particles to the total flux. In the MC method, the full transport mechanism is “disassembled” into its most basic parts – namely a series of straight paths and photon-matter interactions –, which can be efficiently modeled as stochastic events. Below, a brief explanation of the procedure is given.

Photons are simulated one at a time, independently. First, the photon is “created” with some initial properties: a spatial position  $r$ , a direction of flight  $\Omega$ , and an energy  $E$ . These properties all depend on the source specification, which can be a point or a volume source, isotropic or oriented, monokinetic or energy-dependent. Then, the length

of the photon's free path is sampled, determining the distance it will travel before undergoing an interaction of any type. It can be shown that the probability for a photon to travel to point  $x$  with no interaction and then to have an interaction between  $x$  and  $x + dx$  is given by:

$$\mathcal{P}(x) = \Sigma_t e^{-\Sigma_t x} dx \quad (1.18)$$

where  $\Sigma_t$  (expressed in  $\text{cm}^{-1}$ ) is the total macroscopic cross section, or total attenuation coefficient, of the material in which the photon is traveling.

After the free path length has been determined, the interaction undergone by the photon must be sampled. Depending on the photon energy and the material with which it is interacting, one out of all the possible interactions is randomly selected. For the energy range treated in this thesis, ranging from a few MeV to a few keV, the most important interactions are elastic (Rayleigh) scattering, inelastic (Compton) scattering, pair production, and photoelectric effect. Once one of these interactions has been sampled, the interaction "happens" meaning that some properties of the photon are modified, namely its direction and energy, before sampling a new free path and repeating the process. The life of the photon ends when it exits the domain of the simulation, or when it is absorbed. This is, in short, the simplest possible implementation of Monte Carlo simulation for particle transport, and it is usually called **analog Monte Carlo**.

The assumption that a particle's movement through matter can be modelled by a series of free flights and interactions holds particularly well with particles that don't have an electrical charge, like neutrons and photons. However, if charged particles (typically electrons) are to be considered, things can become more problematic. Due to the fact that electrons' interactions with matter are governed by long-range Coulomb forces, these particles tend to have an enormous number of interactions, several orders of magnitudes higher than that a neutral particle would have in the same medium. This means that while 20 to 30 Compton scatterings would be sufficient to reduce a photon energy from several MeV to a few keV, the number of interactions that an electron would need to experience to have the same deceleration is in the order of  $10^4$ - $10^5$ . If we add the consideration that a Monte Carlo simulation usually involves several millions of particles, the problem becomes evident.

In the radioprotection field, electrons resulting from photon-matter interactions are simply discarded. If the quantity to be calculated is the dose, the KERMA approximation is usually applied, which considers all the energy lost from a photon after an interaction to be deposited in the interaction point. This approach neglects both Bremsstrahlung losses and electron transport, but is usually applicable due to the short range of the low-energy electrons produced by photon scattering. The KERMA approximation is also frequently applied in the medical imaging field, provided the voxel size is large compared to the electron range and the condition of electron equilibrium is met (usually true when far from material interfaces).

### **1.3.2 The choice of an estimator**

Let us consider a simple case where we want to estimate some quantity (say, for example, the KERMA) in a point, using Monte Carlo method. The standard way of doing that would be to consider a small volume around the point, to tally all interactions happening inside the volume. Then, by looking at the particle's energy before and after the interaction, it is straightforward to determine the energy deposited in the volume as the difference between the two energies. This corresponds to an energy balance between energy entering the volume and energy exiting it. Finally, by knowing the mass density of the material, we can calculate the point average, retrieving an estimation of the

energy deposited in the point under consideration. Obviously, this can be a very slow-converging method, especially if we are dealing with low-density media (such as air), where the probability for an interaction to happen inside a small volume is very low.

We can then estimate the KERMA at point  $P$  and at interaction  $j$  as:

$$k_j(P) = \begin{cases} \frac{w_{j-1}E_{j-1} - w_jE_j}{\rho V}, & \text{if } P \text{ is inside } V \\ 0, & \text{if not} \end{cases} \quad (1.19)$$

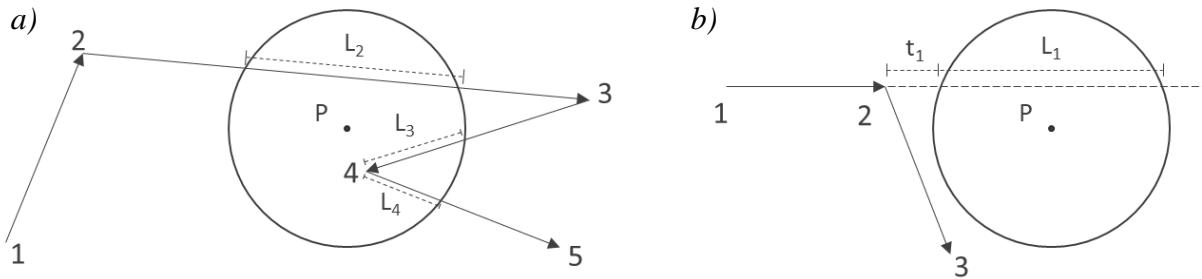
where  $j$  is the interaction number,  $w$  is the particle's weight (see Section 4.2),  $E_{j-1}$  and  $E_j$  are the particle's energies before and after collision,  $\rho$  is the medium density and  $V$  is the scoring volume. This is usually called **collision estimator**, as the estimation of a quantity in a volume is related to the collisions (or, more properly, interactions) taking place inside the volume.

However, this kind of estimation can have a very low efficiency when the number of collisions in the detector is low, for example in the case of low-scattering media or optically thin detection volumes. In these cases, most MC codes use another class of estimators called **track-length estimators (TLE)**. This type of estimator, already described by Gelbard *et al.* [52] in 1966, is based on the equivalence between particle flux and total particle path length per unit volume [51, 53], which can be written as:

$$\Phi = \frac{L}{V} \quad (1.20)$$

where  $L$  is the average total distance travelled by particles inside volume  $V$ .

Equation (1.20) tells us that, for a given volume  $V$ , all particle trajectories that cross the scoring volume contribute to the flux estimate, regardless if they had an interaction inside  $V$  or not. Depending on how this is implemented, and how the distance  $L$  is calculated, several types of estimators exist. In particular, we give here a brief description of the linear and exponential track-length estimators, schematized in Figure 1.6, which constitute the basis of the new method we want to implement.



**Figure 1.6: 2D visual representation of a) linear track-length estimator, and b) exponential track-length estimator**

#### a) Linear track-length estimator

This estimator is the mathematical application of what we described above. Referring to Figure 1.6a, we can estimate the flux at point  $P$  as the average of the path lengths crossing volume  $V$ , divided by  $V$ . The estimation of the KERMA then becomes:

$$k_j(P) = \begin{cases} w_j E_j \frac{\mu_{en}}{\rho} \frac{L_j}{V}, & \text{if } L_j > 0 \\ 0, & \text{if not} \end{cases} \quad (1.21)$$

where  $\mu_{en}/\rho$  is the mass energy absorption coefficient evaluated at energy  $E_j$ ;  $w_j$  is the particle's weight; and  $L_j$  is the distance traveled by the particle inside volume  $V$  between interactions  $j$  and  $j + 1$ . For simplicity, we are assuming the medium to be homogeneous inside  $V$ .  $L_j$  may be defined in a more rigorous way as:

$$L_j = \int_0^{|\mathbf{r}_{j+1} - \mathbf{r}_j|} \Pi_V(\mathbf{r}_j + s \cdot \boldsymbol{\Omega}_j) ds \quad (1.22)$$

In this formulation,  $\Pi_V(\mathbf{r})$  is a characteristic function associated with the scoring volume  $V$ , equal to 1 if the point  $\mathbf{r}$  is inside the volume and equal to 0 if not. This estimator would give a non-zero value for collisions 2, 3 and 4 in Figure 1.6a, whereas the collision estimator of Equation (1.19) would only register collision 4.

### b) Exponential track-length estimator

We can derive the exponential track-length estimator (Figure 1.6b) or  $e$ -TLE from its linear correspondent by extending the definition of quantity  $L_j$ , as explained in the following equations:

$$k_j(P) = \begin{cases} w_j E_j \frac{\mu_{en}}{\rho} e^{-\Sigma_t s_V} \frac{(1 - e^{-\Sigma_t L_j})}{\Sigma_t V}, & \text{if } L_j > 0 \\ 0, & \text{if not} \end{cases} \quad (1.23)$$

$$L_j = \int_0^{\infty} \Pi_V(\mathbf{r}_j + s \cdot \boldsymbol{\Omega}_j) ds \quad (1.24)$$

where  $s_V$  is the distance between the interaction and the volume following the particle's line of flight, and  $\Sigma_t$  is the total macroscopic cross section of the medium. For simplicity, we are assuming the medium to be homogeneous everywhere.

From a mathematical point of view, we are simply integrating over  $|\mathbf{r}_{j+1} - \mathbf{r}_j|$ , thus substituting a random variable ( $L_j$ ) with its expected outcome<sup>1</sup>. Practically, we are virtually extending the particle's path to infinity after every collision, and then considering the particle's linear attenuation to take into account the fact that we are not really simulating the flight. Again, the efficiency of this method can be well observed in Figure 1.6b, where we can see that the particle is not even required to cross the scoring volume to contribute to the score.

### c) Statistical analysis of dose rate estimation

We have seen how the MC estimation of KERMA in a volume is affected by the choice of the estimator. Since we are dealing with a stochastic approach, every calculated quantity will in fact be a random variable; it will therefore have an expected value and a variance associated to it. To continue with the KERMA example, for any given estimator  $\epsilon$  we will have:

<sup>1</sup> For a rigorous mathematical derivation of the  $e$ -TLE, obtained by averaging the collision estimator or the track length estimator over all possible flight lengths, refer to Annex B.

$$\bar{K}_\epsilon = \frac{1}{n} \sum_{i=1}^n k_{i,\epsilon} \quad (1.25)$$

where  $\bar{K}_\epsilon$  is the estimate of KERMA expected value,  $n$  is the number of histories simulated, and  $k_{i,\epsilon}$  is the contribution of the single particle history to the KERMA estimate, as expressed in Equations (1.19), (1.21) and (1.23). We can also write an estimate for the variance  $\sigma^2(\bar{K}_\epsilon)$  associated to  $\bar{K}_\epsilon$  as:

$$\sigma^2(\bar{K}_\epsilon) = \frac{\sigma^2(K)}{n} = \frac{1}{n(n-1)} \sum_{i=1}^n (k_{i,\epsilon} - \bar{K}_\epsilon)^2 = \frac{1}{n-1} \left( \frac{1}{n} \sum_{i=1}^n k_{i,\epsilon}^2 - \left( \frac{1}{n} \sum_{i=1}^n k_{i,\epsilon} \right)^2 \right) \quad (1.26)$$

where we substituted Equation (1.25) to get the final expression [54].

Now, if we look back at the definition of  $k_{i,\epsilon}$  given in Equations (1.19), (1.21) and (1.23), we see that they can either be zero (if the particle does not contribute to the estimate) or greater than zero. Depending on the estimator chosen and on the specifics of the problem, the ratio of zero to non-zero contributions can greatly vary. For example, for estimations in low-density media, where collisions are extremely rare, a collision estimator will mostly have null contributions, with some rare but very high contributions due to photoelectric effect. In the same configuration, a track-length estimator will likely have many more non-zero contributions of much lower value, which tends to lower the variance.

To test this hypothesis we search for an expression for the variance alternative to that of Equation (1.25). If we consider only histories with non-zero contributions to our estimate, we can write the two following relations (the superscript “ $n^0$ ” standing for “non-zero”):

$$\bar{K}_\epsilon = p^{n^0} \overline{K_\epsilon^{n^0}} \quad \text{and} \quad \overline{K_\epsilon^2} = p^{n^0} \overline{K_\epsilon^{n^0}}^2 \quad (1.26)$$

where  $p^{n^0}$  is the probability of having a non-zero contribution and  $\bar{K}_\epsilon$  and  $\overline{K_\epsilon^2}$  are the first and second order moment of the KERMA estimate  $K$ . Using the relations in (1.26) and remembering the definition of variance, we can write:

$$\sigma^2(\bar{K}_\epsilon) = \frac{\sigma^2(K)}{n} = \frac{p^{n^0}}{n} (\sigma^2(K^{n^0}) + (1 - p^{n^0})(\overline{K_\epsilon^{n^0}})^2) = \frac{n^{n^0}}{n^2} \left( \sigma^2(K^{n^0}) + \left(1 - \frac{n^{n^0}}{n}\right) (\overline{K_\epsilon^{n^0}})^2 \right) \quad (1.27)$$

where we estimated the probability  $p^{n^0}$  as the number of non-zero contributions  $n^{n^0}$  divided by the total number of histories  $n$ . We can use this first simple formulation to compare *a priori* the variance of different estimators.

### 1.3.3 Applications in D&D

Due to their ability of giving a precise solution of the transport problem, Monte Carlo calculations are usually adopted when accuracy is the priority. In the framework of decommissioning and dismantling, this is usually the case for criticality and transport studies, which constitute the basis for subsequent activity and shielding calculations. However, these are neutron-based calculations, while for the purpose of this work we are mostly interested in photons. For shielding calculations with photon transport, PKI methods are often preferred, as they provide a much faster answer and the lack of accuracy is usually not a problem, as it allows for a safety margin in the radioprotection calculations. However, there are several instances in which MC codes for photon transport are chosen over deterministic ones.

For example, MC codes are largely used as a numerical validation tool to test the accuracy of PKI codes [33, 37, 55, 56]. A Monte Carlo simulation is faster and cheaper than a real-life experiment, as well as more reproducible. Furthermore, PKI codes are usually designed to give a pessimistic answer [43] when an exact one is not attainable: this is in line with the ALARA (*As Low As Reasonably Achievable*) principle prescribed for radioprotection studies. However, MC codes can be used to perform accurate shielding calculations allowing to evaluate more precise and less conservative safety margins, which can lead to more cost-effective and organisationally efficient solutions from an industrial point of view.

When dealing with shielding calculations applied to smaller geometries, computation times become more manageable and MC codes remain the golden standard for calculating accurate radiation transport. A good example of this can be found in [57]. The objective of this study was to verify the shielding capabilities of a storage container for VLLW (Very Low Level Waste), measuring about one cubic meter. These kind of containers are used to store low-radioactivity nuclear waste and they are subject to regulations fixing the maximum radiation dose that can penetrate the container's walls. In this case, the MC code MCNP was used to evaluate the maximum dose rate at the container outer surface and at a distance of 2m from the container's surface.

Another example of MC transport applied to shielding calculations over smaller geometries is given by [58]. This analysis falls under the modelling efforts surrounding the design and construction of the ITER fusion reactor. In particular, the focus of the analysis is one of the remote-handled units of the reactor, the IVT (In-Vessel Transporter), a remote-controlled machine that will be used to perform operations in radioactive areas of the plant. When the IVT will be subject to maintenance, it will be covered by radioactive dust that could pose a health risk for the operating personnel. MC simulations performed with the code MCNP allowed to evaluate the expected dose rate in a set of points surrounding the IVT, to establish safety margins for the operators.

As a last example of MC codes being used in shielding calculations, in a study by Park *et al.* [59] the MC code MCNP was used to accurately calculate a dose rate map in a dismantling facility; this map was then used as the basis for a 3D visualization simulation, in which a worker moving inside the environment could see in real time the dose coming from the radioactive source. These kind of 3D visualization simulations can be useful for training purposes, allowing workers to accurately plan their interventions in radioactive environments. MC transport codes are also used in shielding calculations for the qualifications of new radiation shielding materials. In a study by Lakshminarayana *et al.* [60], six types of glasses were analysed with MCNP to assess their shielding capabilities for gamma radiation. The attenuation coefficients for the six glasses were tabulated for a range of gamma energies going from  $10^5$  to  $10^{-3}$  MeV. A similar analysis, but on concrete instead of glass, was conducted in [61], where the MCNPX code was used to evaluate the shielding properties of two special concretes enhanced with the insertion of  $\text{WO}_3$  and  $\text{Bi}_2\text{O}_3$  particles. In this case, the concrete attenuation coefficient was evaluated for photon energies ranging from 0.142 to 1.33 MeV. The interest of this kind of analyses is double: it allows to qualify materials for various applications in radiation shielding, and it evaluates coefficients (like the attenuation coefficient, see Section 1.2.1) that can then be used by deterministic radiation transport codes for shielding calculations.

A similar example in the medical field [62] involved a MC simulation to calculate radioprotection quantities for human soft tissue. In this case, the calculated quantity was the build-up factor (see Section 1.2.2), to be used by

PKI codes in specific medical applications. Build-up factors were calculated for incident photon energies of 0.2 MeV, 0.5 MeV, 1 MeV and 2 MeV.

We have now seen some examples of applications of both deterministic and stochastic particle transport. In the next chapter, both methods are applied to a simple, model case, to give the reader a better idea of how they work, albeit in a very elementary way.



# Chapter 2

## Heuristic approach

We have seen how the problem of photon transport is treated mathematically, and the different approaches that exist to solve it in practice. This chapter offers a look at a practical example: a very simple transport problem which is solved first analytically, then through a Monte Carlo approach. We will show that the results converge, demonstrating the validity of the two approaches. The same simple configuration will be used later, in Chapter 5, to show the unbiasedness of the  $e$ -TLE estimator.

### 2.1. Study configuration

The configuration is presented in Figure 2.7. All the simplifications are introduced to allow us to get to an exact analytical solution to the Boltzmann equation, which would not be possible in the general case, as seen in Chapter 1. We consider a monoenergetic, monodirectional, stationary surface source of particles of intensity  $S$  (particles · cm<sup>-2</sup> · s<sup>-1</sup>), penetrating a homogeneous, non-multiplying, one-dimensional material of thickness  $L$ . The source is entirely localized on the left margin of the medium; we can write:

$$S(x = 0) \equiv S\delta(x - 0) = S\delta(x) \quad (2.1)$$

where  $\delta(x)$  is a Dirac's delta function.

For the material we will define a macroscopic scattering cross section  $\Sigma_s$ , a macroscopic absorption cross section  $\Sigma_a$ , and a total macroscopic cross section  $\Sigma_t$ . Therefore:

$$\Sigma_t = \Sigma_s + \Sigma_a \quad (2.2)$$

To further simplify the problem, we suppose that the scatterings never cause any angular displacement. Subsequently, all particles fly in the same direction, which is chosen to be left to right in Figure 2.7, with discrete flights of variable length.

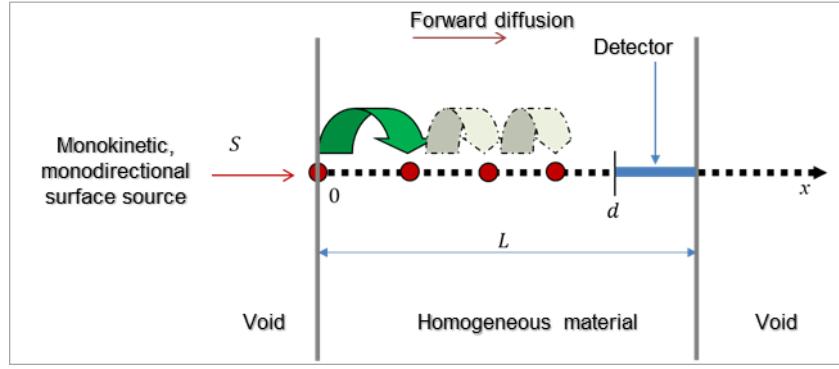


Figure 2.7: 1D transport problem configuration

We want to compute the mean particle flux,  $\phi_D$ , in the interval  $D = [d, L]$ , which constitutes the “detector” in this configuration. In the following, two solutions to this problem are proposed. The first solves the Boltzmann equation deterministically, taking advantage of the geometric simplifications; the second method reproduces a Monte Carlo calculation by subdividing the problem into its elementary components, *i.e.* the single particles and their displacements.

## 2.2. Deterministic solution

For the configuration under consideration, we start from the integro-differential formulation of the stationary Boltzmann equation for the flux (see Equation (1.6) in Chapter 1):

$$\boldsymbol{\Omega} \cdot \nabla \Phi_{\Omega}(\mathbf{r}, \boldsymbol{\Omega}, E) + \Sigma_t(\mathbf{r}, E) \Phi_{\Omega}(\mathbf{r}, \boldsymbol{\Omega}, E) = \iiint \Sigma_s(\mathbf{r}, E') f_s(\mathbf{r}, \boldsymbol{\Omega}' \rightarrow \boldsymbol{\Omega}, E' \rightarrow E) \Phi_{\Omega}(\mathbf{r}, \boldsymbol{\Omega}', E') d^2\Omega' dE' + S(\mathbf{r}, \boldsymbol{\Omega}, E) \quad (2.3)$$

Due to the isotropy and 1D hypothesis, the first term is reduced to a simple derivative over  $x$ , and the other terms can be written in a much more compact form. Equation (2.3) is reduced to a second order differential equation:

$$\frac{d\phi(x)}{dx} + \Sigma_t \phi(x) = \Sigma_s \phi(x) + S \delta(x) \quad (2.4)$$

The solution is given by:

$$\phi(x) = K \times e^{-(\Sigma_t - \Sigma_s)x} = K \times e^{-\Sigma_a x} \quad (2.5)$$

where  $K$  denotes a constant.

Due to the source definition, we can write the boundary condition for  $x = 0$  as:

$$\phi(x = 0) = S \Rightarrow K = S \quad (2.6)$$

So, we can rewrite Equation (2.5) as:

$$\phi(x) = S e^{-\Sigma_a x} \quad (2.7)$$

It is now straightforward to find the average particle flux  $\phi_D$  in the detector  $D = [d, L]$ :

$$\phi_D = \frac{1}{l} \int_d^L S e^{-\Sigma_a x} dx = S \frac{[e^{-\Sigma_a d} - e^{-\Sigma_a L}]}{l \Sigma_a} = S e^{-\Sigma_a d} \frac{[1 - e^{-\Sigma_a (L-d)}]}{l \Sigma_a} = S e^{-\Sigma_a d} \frac{[1 - e^{-\Sigma_a l}]}{l \Sigma_a} \quad (2.8)$$

where we defined  $l = L - d$  as the detector's dimension.

This concludes the analytical solution of the problem. We now turn to the Monte Carlo resolution of the simplified 1D Boltzmann equation to see if we can get to the same result with a different approach.

## 2.3. Stochastic solution

We now want to solve this problem by following a stochastic procedure. To do this, we recall (see Equation (1.13)) that it is convenient to start from the integral formulation of the Boltzmann equation. We will now show that it is possible to derive this formulation directly from the integro-differential one of Equation (2.3).

### 2.3.1 Derivation of the integral form of the Boltzmann equation

We follow, for this step, the procedure outlined in [63], applied to our 1D case. We define:

$$s = x - x' \quad (2.9)$$

By taking advantage of the fact that we know the flux will have an exponential shape, we write:

$$-\frac{d}{ds}[\phi(x') \exp(-\Sigma_t(x - x'))] \equiv -\frac{d}{ds}[\phi(x') \exp(-\Sigma_t s)] \quad (2.10)$$

Developing and using the properties of the derivative we write:

$$-\frac{d}{ds}[\phi(x') \exp(-\Sigma_t s)] = \exp(-\Sigma_t s) \left[ -\frac{d}{ds}\phi(x') + \Sigma_t \phi(x') \right] \quad (2.11)$$

Then, taking advantage of (2.10), we can write:

$$\frac{d}{ds} \equiv -\frac{d}{dx'} \quad (2.12)$$

And re-write (2.12) as:

$$-\frac{d}{ds}[\phi(x') \exp(-\Sigma_t s)] = \exp(-\Sigma_t s) \left[ \frac{d}{dx'}\phi(x') + \Sigma_t \phi(x') \right] \quad (2.13)$$

Now we can substitute (2.3) in the right-hand side and get:

$$-\frac{d}{ds}[\phi(x') \exp(-\Sigma_t s)] = \exp(-\Sigma_t s)[\Sigma_s \phi(x') + S(x')] \quad (2.14)$$

Integrating over  $s$  from 0 to *infinity*:

$$\int_0^\infty -\frac{d}{ds}[\phi(x') \exp(-\Sigma_t s)] ds = \int_0^\infty \Sigma_s \exp(-\Sigma_t s) \phi(x') ds + \int_0^\infty S(x') \exp(-\Sigma_t s) ds \quad (2.15)$$

$$\phi(x) = \int_0^\infty \exp[-\Sigma_t s] \Sigma_s \phi(x') ds + \int_0^\infty S(x') \exp[-\Sigma_t s] ds \quad (2.16)$$

which, by expanding  $x' = x - s$  can be re-written as:

$$\phi(x) = \int_0^\infty \Sigma_t \exp[-\Sigma_t s] \frac{\Sigma_s}{\Sigma_t} \phi(x - s) ds + \int_0^\infty S(x - s) \exp[-\Sigma_t s] ds \quad (2.17)$$

We find in Equation (2.17) the same formulation found in Section 1.3 for the integral form of the Boltzmann equation.

Now, we take advantage of the fact that the source is a Dirac's Delta to rewrite the second integral.

$$\phi(x) = \int_0^\infty \Sigma_t \exp[-\Sigma_t s] \frac{\Sigma_s}{\Sigma_t} \phi(x-s) ds + S \exp[-\Sigma_t x] \quad (2.18)$$

Moreover, to find the exact same equation of Section 1.3, we multiply by  $\Sigma_t$  on both sides of the equation to pass from the flux  $\phi(x)$  to the collision density  $\psi(x)$ :

$$\psi(x) = \int_0^x \Sigma_t e^{-\Sigma_t(x-x')} \frac{\Sigma_s}{\Sigma_t} \psi(x') dx' + S \Sigma_t \exp[-\Sigma_t x] \quad (2.19)$$

where we put:

$$\psi(x) = \Sigma_t \phi(x) \quad (2.20)$$

Finally, if we recall the definitions of **collision operator**  $\mathbb{C}$ , **displacement operator**  $\mathbb{T}(x' \rightarrow x)$  and **transport operator**  $\mathbb{K}(x' \rightarrow x)$  given in Chapter 1, and rewrite them for this simplified case:

$$\mathbb{C} \equiv \frac{\Sigma_s}{\Sigma_t} \quad (2.21)$$

$$\mathbb{T}(x' \rightarrow x) \equiv \int_0^x \Sigma_t e^{-\Sigma_t(x-x')} dx' \quad (2.22)$$

$$\mathbb{K}(x' \rightarrow x) \equiv \mathbb{T}(x' \rightarrow x) \mathbb{C} \quad (2.23)$$

we can then rewrite (2.20) in compact form to find the *ID* formulation of Equation (1.12)

$$\psi(x) = \int_x^\infty \mathbb{K}(x' \rightarrow x) \psi(x') dx' + \mathbb{T}S \quad (2.24)$$

### 2.3.2 Monte Carlo solution

We will now show why this formulation is particularly convenient to introduce Monte Carlo solutions to the equation. Let us first write the unknown we search, the average flux in the detector's interval  $D = [d, L]$ , as:

$$\phi_D = \int_d^L \frac{\psi(x)}{\Sigma_t} dx = \int_d^L \phi(x) dx \quad (2.25)$$

Then, we re-write the collision density  $\psi(x)$  as a Neumann series expansion in the form:

$$\psi(x) = \sum_{n=0}^{\infty} \psi_n(x) \quad (2.26)$$

We now look at the life of a particle as a chain of events, consisting of free flights of variable lengths, divided by collisions. By remembering the definition of cross section, we know that we can write the probability for a particle to travel a **path of length**  $x_n$  and then having a **collision in**  $dx_n$  as:

$$e^{-\Sigma_t x_n} \Sigma_t dx_n$$

Therefore, for a single particle interacting  $n$  times, with  $n > 0$ , we can write the probability of arriving at point  $x$  after having collided exactly in  $x_1, x_2, \dots, x_n$  as:

$$g_n(x_1, x_2, \dots, x_n, x) dx_1 dx_2 \dots dx_n = e^{-\Sigma_t x_1} \Sigma_t dx_1 e^{-\Sigma_t(x_2-x_1)} \Sigma_t dx_2 \dots dx_n e^{-\Sigma_t(x-x_n)} \Sigma_t \quad (2.27)$$

where we implicitly put  $x_0 = 0$  so that  $x - x_0 = x$  and  $x_I - x_0 = x_I$ .

If we introduce implicit capture<sup>2</sup>, then the weight of the particle after  $n$  collisions is:

$$\varpi_n = \left[ \frac{\Sigma_s}{\Sigma_t} \right]^n \quad (2.28)$$

We can now define the **collision estimator for the collision density** as:

$$\eta_0(x) = S e^{-\Sigma_t x} \Sigma_t \equiv \hat{S} e^{-\Sigma_t x} \quad (2.29)$$

for incident source particles that contribute to collision density; and as:

$$\eta_n(x, x_n) = S \left[ \frac{\Sigma_s}{\Sigma_t} \right]^n e^{-\Sigma_t(x-x_n)} \Sigma_t \equiv \hat{S} \varpi_n e^{-\Sigma_t(x-x_n)}, \quad n > 0 \quad (2.30)$$

which represents the contribution given to collision density by a particle that has collided  $n$  times. In Equations (2.29) and (2.30) we introduce  $\hat{S} = S \Sigma_t$ .

Now, by recalling the Neumann series decomposition of (2.25) and proceeding by increasing  $n$ , we have, for a generic point  $x$ :

$$\psi_0(x) = \eta_0(x) = \hat{S} e^{-\Sigma_t x} \quad (2.31)$$

which represents the contribution of uncollided particles; and

$$\psi_1(x) = \int_0^x g_1(x_1) \eta_1(x, x_1) dx_1 = \int_0^x e^{-\Sigma_t x_1} \Sigma_t S \frac{\Sigma_s}{\Sigma_t} e^{-\Sigma_t(x-x_1)} \Sigma_t dx_1 = \hat{S} \Sigma_t x \frac{\Sigma_s}{\Sigma_t} e^{-\Sigma_t x} \quad (2.32)$$

which represents the contribution of particles that have collided exactly one time; and

$$\begin{aligned} \psi_2(x) &= \int_0^x dx_1 \int_{x_1}^x g_2(x_1, x_2) \eta_2(x, x_2) dx_2 = \int_0^x e^{-\Sigma_t x_1} \Sigma_t dx_1 \int_{x_1}^x e^{-\Sigma_t(x_2-x_1)} \Sigma_t S \left[ \frac{\Sigma_s}{\Sigma_t} \right]^2 e^{-\Sigma_t(x-x_2)} \Sigma_t dx_2 \\ \psi_2(x) &= \hat{S} \Sigma_t^2 \left[ \frac{\Sigma_s}{\Sigma_t} \right]^2 e^{-\Sigma_t x} \int_0^x dx_1 \int_{x_1}^x dx_2 = \hat{S} \Sigma_t^2 \left[ \frac{\Sigma_s}{\Sigma_t} \right]^2 e^{-\Sigma_t x} \int_0^x dx_1 [x_2]_{x_1}^x \\ \psi_2(x) &= \hat{S} \Sigma_t^2 \left[ \frac{\Sigma_s}{\Sigma_t} \right]^2 e^{-\Sigma_t x} \int_0^x [x - x_1]_{x_1}^x dx_1 = \hat{S} \Sigma_t^2 \frac{x^2}{2} \left[ \frac{\Sigma_s}{\Sigma_t} \right]^2 e^{-\Sigma_t x} \left[ x \int_0^x dx_1 + \int_0^x x_1 dx_1 \right] \\ \psi_2(x) &= \hat{S} \Sigma_t^2 \frac{x^2}{2} \left[ \frac{\Sigma_s}{\Sigma_t} \right]^2 e^{-\Sigma_t x} \left[ x^2 - \frac{x^2}{2} \right] = \hat{S} \Sigma_t^2 \frac{x^2}{2} \left[ \frac{\Sigma_s}{\Sigma_t} \right]^2 e^{-\Sigma_t x} \end{aligned} \quad (2.33)$$

which represents the contribution of particles that have collided exactly two times, etc.

By induction, for particles having collided  $n$  times, we have:

$$\psi_n(x) = \hat{S} \Sigma_t^n \frac{x^n}{n!} \left[ \frac{\Sigma_s}{\Sigma_t} \right]^n e^{-\Sigma_t x} \quad (2.34)$$

and:

<sup>2</sup> Implicit capture is a technique which allows to implicitly simulate a capture interaction by acting on the weight of the particle. For an in-depth explanation, see Section 4.3.2

$$\psi(x) = \sum_{n=0}^{\infty} \psi_n(x) = \hat{S} \sum_{n=0}^{\infty} \frac{\left(\Sigma_t \frac{\Sigma_s}{\Sigma_t} x\right)^n}{n!} e^{-\Sigma_t x} \quad (2.35)$$

Now, considering that :

$$\sum_{n=0}^{\infty} \frac{\left(\Sigma_t \frac{\Sigma_s}{\Sigma_t} x\right)^n}{n!} = \sum_{n=0}^{\infty} \frac{(\Sigma_s x)^n}{n!} = e^{-\Sigma_s x} \quad (2.36)$$

we can finally write the collision density as :

$$\psi(x) = \hat{S} e^{-\Sigma_s x} e^{-\Sigma_t x} = \hat{S} e^{-(\Sigma_t - \Sigma_s)x} = \hat{S} e^{-\Sigma_a x} = S \Sigma_t e^{-\Sigma_a x} \quad (2.37)$$

and the corresponding particle flux as :

$$\phi(x) = \frac{\psi(x)}{\Sigma_t} = \frac{\hat{S} e^{-\Sigma_a x}}{\Sigma_t} = \frac{\Sigma_t S}{\Sigma_t} e^{-\Sigma_a x} = S e^{-\Sigma_a x} \quad (2.38)$$

The average flux in the detector,  $\phi_D$ , is therefore:

$$\phi_D = \frac{1}{l} \int_d^L S e^{-\Sigma_a x} dx = S e^{-\Sigma_a d} \frac{[1 - e^{-\Sigma_a l}]}{l \Sigma_a} \quad (2.39)$$

which is exactly the same result obtained in Equation (2.8) with the deterministic approach.

As a final remark, it is appropriate to note that the particle flux at point  $x$ ,  $\phi(x) = S e^{-\Sigma_a x}$ , should not be confused with the **uncollided particle flux**  $\phi_0(x)$ , i.e. the flux given only by particles that have not collided before arriving in  $x$ , which can be written as:

$$\phi_0(x) = \hat{S} e^{-\Sigma_t x} \quad (2.40)$$

This quantity is of particular interest in radioprotection, as it is always possible to calculate it analytically by knowing the total macroscopic cross sections of the attenuating material. It is also called **primary flux**, as opposed to the secondary flux, coming from particles that had at least one collision before arriving to the detector. Similarly, we can calculate the **average uncollided flux in the detector**,  $\phi_{D,0}$ , as:

$$\phi_{D,0} = \frac{1}{l} \int_d^L \hat{S} e^{-\Sigma_t x} dx = \hat{S} \frac{[e^{-\Sigma_t d} - e^{-\Sigma_t L}]}{l \Sigma_t} = \hat{S} e^{-\Sigma_t d} \frac{[1 - e^{-\Sigma_t (L-d)}]}{l \Sigma_t} = \hat{S} e^{-\Sigma_t d} \frac{[1 - e^{\Sigma_t l}]}{l \Sigma_t} \quad (2.41)$$

This expression for the uncollided flux will be useful to define a new estimator, which will be shown to converge to the same result as the collision estimator, in Chapter 5.

## 2.4. Practical example: numerical 1D Monte Carlo

It is easy to test the validity of these calculations by developing a simple 1D Monte Carlo script. In this implementation, a particle is generated at  $x = 0$  and displaced by a quantity  $r$  defined as:

$$r = -\frac{\log(\rho)}{\Sigma_t} \quad (2.42)$$

where  $\rho$  is a pseudo-random number uniformly distributed between 0 and 1 and  $\Sigma_t$  is the macroscopic cross section as defined in Equation (2.2). After the displacement, the particle is killed with a probability of  $\Sigma_a/\Sigma_t$ ; if it survives, it is displaced again. For a first implementation, we use the collision estimator (see Section 1.3.2): whenever a collision happens inside the detector, the code scores a value of  $\kappa_{coll}^i = 1/l\Sigma_t$ . Finally, if the particle dies or exits the 1D slab, its history is terminated and the value of the average flux is calculated as the sample mean:

$$\phi_{D,coll} \approx \bar{\kappa}_{coll} = \frac{1}{N} \sum_{i=1}^N \kappa_{coll}^i \quad (2.43)$$

The associated sample variance is calculated as:

$$\sigma_{D,coll}^2 = \frac{1}{N-1} \sum_{i=1}^N (\kappa_{coll}^i - \bar{\kappa}_{coll})^2 \quad (2.44)$$

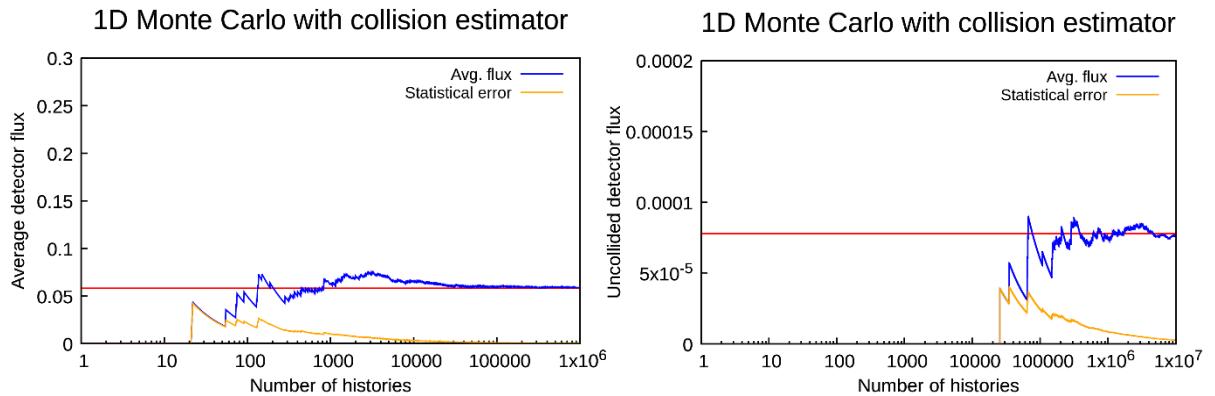
For this test, we chose a scattering cross section  $\Sigma_s$  of  $0.7 \text{ cm}^{-1}$ , an absorption cross section  $\Sigma_a$  of  $0.3 \text{ cm}^{-1}$ , a total length  $L$  of 10 cm, and a detector length  $l$  of 1 cm. With these data, the value for the average particle flux in the detector  $\phi_D$ , calculated with Equation (2.8), is:

$$\phi_D = 5.806148 \cdot 10^{-2} \left[ \frac{\text{particles}}{\text{cm} \cdot \text{s}} \right] \quad (2.45)$$

And the value for the uncollided flux  $\phi_{D,0}$  is:

$$\phi_{D,0} = 7.800974 \cdot 10^{-5} \left[ \frac{\text{particles}}{\text{cm} \cdot \text{s}} \right] \quad (2.46)$$

Figure 2.8 shows that, after a sufficient number of particle histories, the estimation given by the Monte Carlo solution converges to the analytical one, while the associated variance approaches zero. We can also run the simulation searching for the uncollided flux, by only scoring particles that have their first collision inside the detector and not counting the others. Even in this case, the calculation is shown to converge to the analytical value of Equation (2.46); however, the convergence is much slower, due to the fact that the simulated event (a particle not colliding until the detector and then colliding in it) is extremely rare.



**Figure 2.8: Average total (left) and uncollided (right) particle flux in the detector calculated through Monte Carlo. The red bar shows the value of the analytical solution. Note the difference in the x axis between the pictures, highlighting the much slower convergence of the uncollided simulation.**



---

PART II

STOCHASTIC SIMULATIONS IN

DECOMMISSIONING AND

DISMANTLING PROBLEMS

---



In this second part, we will give a closer look at how state-of-the-art deterministic and stochastic codes are used to solve decommissioning and dismantling problems, and explore their current limits. We will then focus on the stochastic Monte Carlo approach and see how its efficiency can be improved to better deal with this kind of problems, ultimately describing the motive behind this thesis work and identifying its framework.



# Chapter 3

## Limits of modelling deterministic and stochastic methods

We have seen that, when dealing with photon transport for shielding calculations, one of two main approaches can be followed to find a solution: PKI method with build-up and/or albedo factors, and MC techniques. We have also hinted at the fact that each of the two methods has its strengths and its weaknesses: now, we will focus on the shortcomings of each method, to explain the need for new, more efficient calculation tools that can help us overcome these shortcomings.

As a general rule, the PKI method has a strict domain of application outside of which it becomes unreliable. In the field of D&D this is usually accepted, as long as the estimations given by the method are conservative: if the radiation dose levels predicted by the method are shown to always be equal to or higher than the experimental values, then the code can be used for radioprotection purposes. However, there are cases in which this does not happen and PKI codes are shown to underestimate dose values.

On the other hand, the limits of MC simulations are more technical in nature. As already pointed out in Chapter 1, the main problem with most MC simulations is the total computation time, which can quickly become prohibitive in the case of large, complex geometries and especially with the estimation of rare events. These are circumstances that can often be encountered in shielding calculations for dismantling, where the geometries can include the whole reactor building and the deep penetration of photons through radiation shields and concrete walls can lead to rare event estimations in the farthest points of the domain.

This chapter presents in a general way some examples of difficult configurations found in D&D, and then focus on some specific test configurations, which have been chosen in the scope of this thesis as a benchmark to assess the limits of analog MC simulation and test the efficiency of our new estimator, the *e*-TLE.

### 3.1 Complex configurations encountered in dismantling problems

As we have seen in Chapter 1, depending on the type of problem to be solved, deterministic codes use some semi-empiric correction factors to determine a solution: the BU coefficient for attenuation-dominated problems, the albedo coefficient for reflection-dominated problems, and sometimes a combination of both of them. Depending on the configuration studied, a number of problems can greatly diminish the accuracy of PKI codes. For example, this happens when the configuration lies outside the limits imposed by the correction factors, thus, when the distance to travel in an attenuating media is too long (BU factors are usually tabulated until a maximum length expressed in mean free path units (MFP) or when there are too many reflections in a streaming configuration between source and detector. Moreover, PKI codes usually struggle to produce a valid estimate when the two approaches, BU and albedo, have to be used together.

We now look at some actual configurations found in the literature, highlighting the limitation of PKI codes. As a first example we consider the geometry of Figure 3.9, first studied in [41]. In this very simple case, a  $^{60}\text{Co}$  source is placed in an empty concrete room, very close to the ceiling. The detector, is placed right next to a corner of the room. This configuration was designed to enhance as much as possible the contribution of scattered radiation from walls and ceiling to the detector. It is however a not-so-uncommon configuration, as it can be encountered for example when searching for dose rate values inside a reactor building close to the floor/ceiling and walls. In this instance, the authors conclude that the errors given by the PKI code CIDEC can be as high as 31%, due to the effect of back-scattering which is neglected by the calculation method, leading to an underestimation in the results.

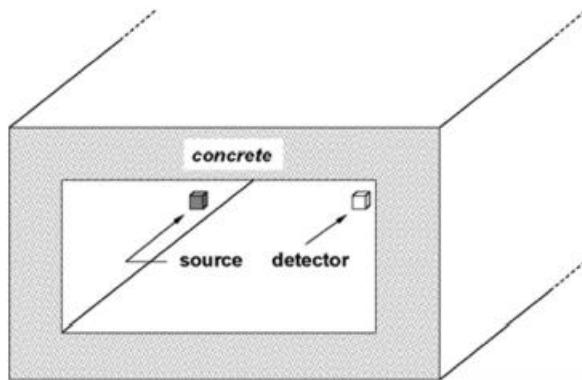
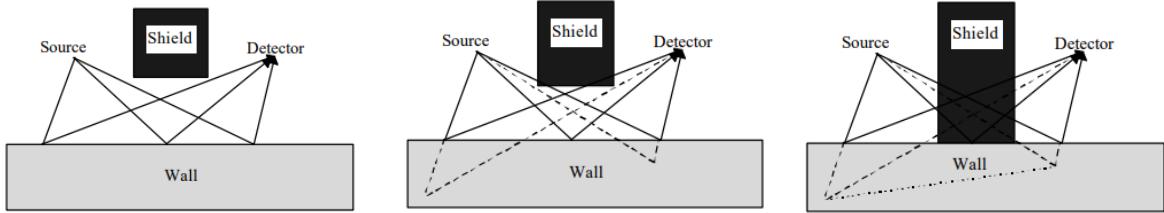


Figure 3.9: Scattering-intensive configuration from [41]

As another example, we look at Figure 3.10, presenting some configurations studied in [43]. As for the first example, we have a somewhat straightforward configuration, in this case composed by a  $^{60}\text{Co}$  source, a radiation shield, a concrete scattering floor, and a detector. The shield is treated as an ideal black body which absorbs all radiation passing through it.

The position of the shield changes identifying three similar configurations that have to be treated differently, due to the different relative contribution of scattered particles in the three cases. In the first case, the use of an albedo coefficient is sufficient to correctly evaluate the solution within a 20% accuracy. However, for the second case, the calculations conducted with the albedo coefficient led to an underestimation of the result, and another calculation algorithm simulating a single scattering point and two ray-tracing computation had to be used. Finally, for the third case, an even more complicated multiple-scattering algorithm was employed, but it still led to extreme overestimation of the results (between 150% and 400% error) and computation times that were comparable to a

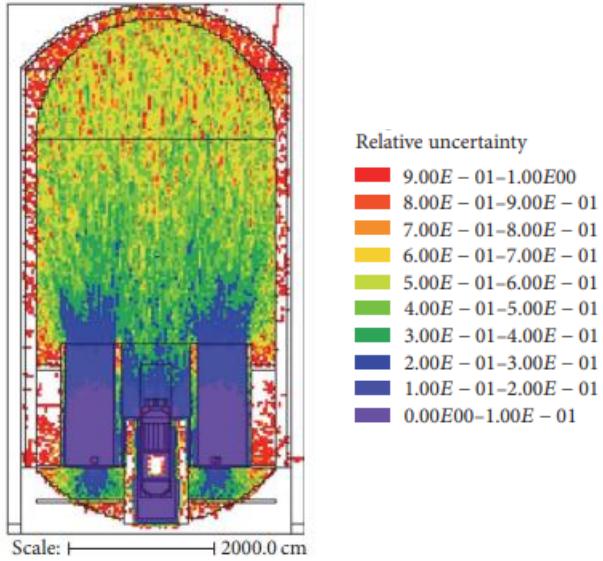
Monte Carlo run on the same geometry, therefore too long for a PKI calculation. This is a good example of how a relatively simple configuration can pose serious problems to PKI-based codes.



**Figure 3.10: Source-shield-detector configuration for the study of multiple-scattering with the RANKERN PKI code, from [43]**

Figure 3.11 gives a very good example of the limitations of an analog MC simulation. In this paper by Matijevic *et al.* [64], the authors perform a shielding calculation over a PWR building. Results of an analog MC calculation for the dose rate coming from photons emitted by the coolant is shown in Figure 3.11. We report only the statistical error on the results, which shows how, when moving far from the source term (identified by the coolant in the steel generators and pumps) the code is completely unable to give meaningful results, with statistical errors ranging from around 30% up to 100%. In the white parts of the picture, identifying regions outside of the radiation shield of the reactor, the code is not able to compute any solution. It is relevant to note that the calculation took 1.64 days with 32 Gigabytes of RAM and a Core i5 CPU.

In the next sections, we present some additional configurations that were chosen for the purpose of this thesis to validate the newly proposed *e*-TLE estimator. All configurations were chosen as they present some specificities that make them difficult to treat with classic PKI or MC codes.



**Figure 3.11: Results for the statistical error on a photon transport shielding MC calculation, from [64]**

### 3.2 “Odano” benchmarks

The first group of configurations is based on the Odano Benchmarks [55], a set of experiment-based shielding benchmarks devised by Odano *et al.* to assess the performance and accuracy of the point-kernel code G33-GP2

[65], and of the MC code MCNP [14], on classical problems that would be problematic to evaluate with the PKI methodology. To this end, the configurations are explicitly designed to highlight the limits of applicability of the kernel method, with each configuration representing a specific problem.

In the original experiment by Odano, these three configurations are reproduced in the lab and the results of the two codes are compared with the experimental data. Experimental dose rates are calculated with thermo-luminescent dosimeters (TLD) and no information on experimental uncertainty is given in the original report. For the purpose of this thesis, the results obtained with TRIPOLI-4® are compared both with the experimental data and with the results of MCNP, which is nowadays considered as a well-established reference Monte Carlo code. Results obtained with the PKI code NARMER in a VVUQ (*Verification, Validation and Uncertainty Quantification*) study performed in 2021 are also included in this chapter for reference.

The Odano benchmarks include a back-scattering setup, a slant penetration setup, and a duct streaming setup.

### **3.2.1 Back-scattering slab**

The back-scattering configuration (Figure 3.12a) assesses the performance of an estimator concerning the phenomenon of photon back-scattering, or reflection, on concrete. As noted before, PKI codes use albedo coefficients to mimic the effect of photon reflection, which could lead to errors if the coefficients are not chosen correctly by the user. In the original experiment by Odano *et al.*, the deterministic code G33-GP2 showed a somewhat important underestimation of the results, between 40% and 50%, therefore failing to evaluate the effect of scattering in the simulation.

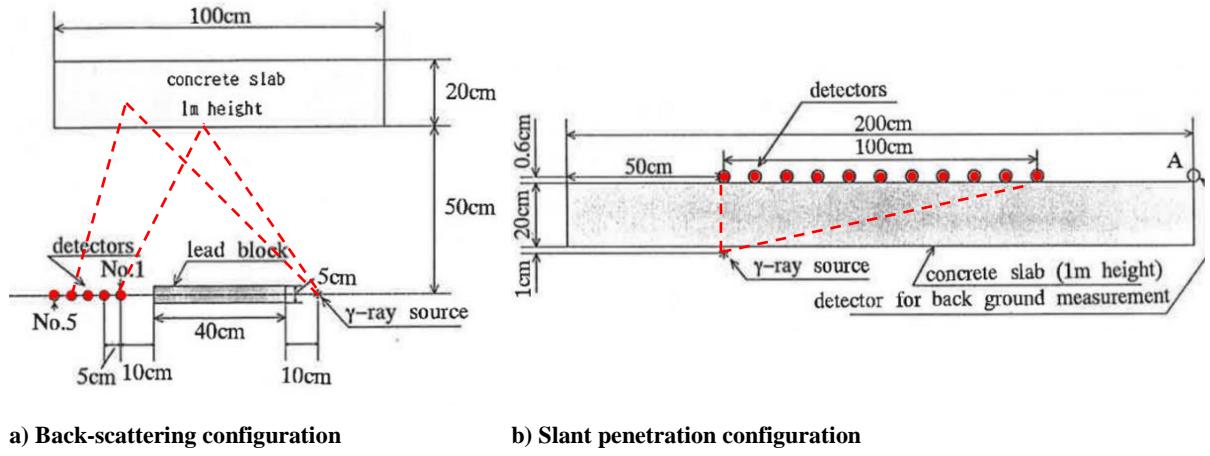
The geometric configuration is quite simple. The direct path between the  $^{60}\text{Co}$  point-wise source and the detector volumes is shielded by a lead slab of  $10 \times 40 \times 5 \text{ cm}^3$ . A concrete slab measuring  $100 \times 100 \times 20 \text{ cm}^3$  is placed at a distance of 50 cm from the photon source, allowing photons to scatter towards the detectors bypassing the shield. The five detectors are modeled as 1 cm radius spheres spaced 5 cm from each other.

### **3.2.2 Slant penetration slab**

The slant penetration configuration (Figure 3.12b) determines the estimator's response to photons traveling inside an attenuating medium with increasing slant angles. With this geometric configuration, the problem for a classical PKI code arises from the treatment of multiple photon scattering in the attenuating. The contribution of scattering becomes more important as the angle of inclination increases.

In the original experiment two different approaches were chosen to deal with the problem: a BU-based calculation and a single-scattering approximation. However, due to the complex nature of the problem, the deterministic code always underestimated the results with errors up to 24% for the detectors further from the source (*i.e.* the ones with the greatest slant angles). This configuration was also used in the validation of the modelling deterministic code PANTHERE [66], as briefly seen in Chapter 1.

As for the back-scattering case, this configuration is also quite simple from a purely geometrical point of view. The attenuating medium is a  $100 \times 200 \times 20 \text{ cm}^3$  concrete slab placed between photon source and detectors at a distance of 1 cm from either side. The first detector is placed vertically above the source; the remaining ten are spaced 10 cm from each other, creating increasing slant angles with respect to the source, up to an angle of  $78^\circ$ . The detectors are modeled as 1 cm radius spheres.



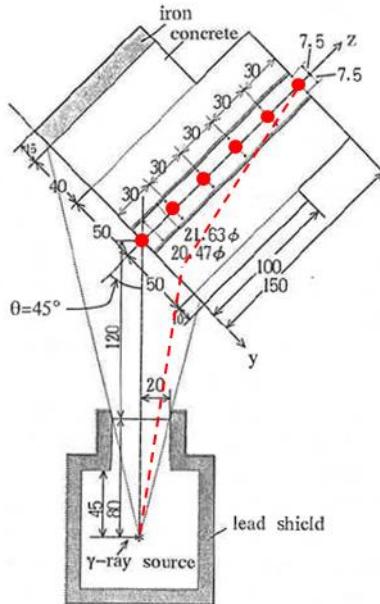
a) Back-scattering configuration

b) Slant penetration configuration

**Figure 3.12:** Geometry of the back-scattering (a) and slant penetration (b) benchmark from [55]. Red points represent the detector locations, red lines represent some of the possible photon paths.

### 3.2.3 Streaming duct

The streaming duct configuration (Figure 3.13) tests the performance of the estimator in the case of a streaming problem, *i.e.* an empty tube traversing a scattering medium thus creating a preferential way for particle transport. This kind of phenomenon is particularly important in shielding calculations, as it can be found quite often in nuclear installations, notably when dealing with air ducts and water pipes [67]. Due to its complex nature, this problem needs the application of both albedo and BU coefficients to be efficiently tackled, which can add a certain level of complexity to the simulation. Again, in this case, the PKI code G33-GP2 showed important underestimations in the results in the original experience, with errors up to almost 50% in the worst case.



**Figure 3.13:** Geometry of the duct streaming benchmark from [55]. Red points represent the detector locations, red lines show some of the possible photon paths.

This configuration is also more complex from the geometric point of view. The pointwise isotropic  $^{60}\text{Co}$  source is placed in a lead collimator which effectively turns it into a conical-emitting source. A concrete structure with an empty steel tube traversing it is placed at a distance of two meters from the source with a  $45^\circ$  angle. Eighteen 1 cm radius spherical detectors are placed along the duct at six different axial positions. Each axial position has three detectors, spaced 7.5 cm from each other.

### 3.3 Reactor-specific scenarios

The next three configurations are designed in order to mimic, albeit in a simplified way, geometries that can be found in an actual dismantling scenario when dealing with a nuclear installation. Unlike the Odano benchmarks, these are not based on experimental data. Moreover, due to their added complexity, these reactor-specific scenarios are designed to highlight the limits of MC codes as well as deterministic ones. Even if they remain quite simple from a purely geometric point of view, some of the configurations are too complex to be treated with a normal, un-accelerated Monte Carlo calculation in a reasonable time frame.

#### 3.3.1 Deep penetration slab

The deep penetration slab configuration (Figure 3.14a) takes inspiration from a hypothetical near-core configuration, with photons attenuated by a large mass of water. As the thickness of the water slab increases in subsequent simulations, it becomes progressively harder for particles to reach the detector placed at the end of the water tank. Therefore, in this situation we deal with the estimation of a rare event, one of the cases in which MC calculations struggle to give a solution in reasonable computation times. Due to the presence of multiple scattering, the problem would be also difficult to treat with a deterministic code.

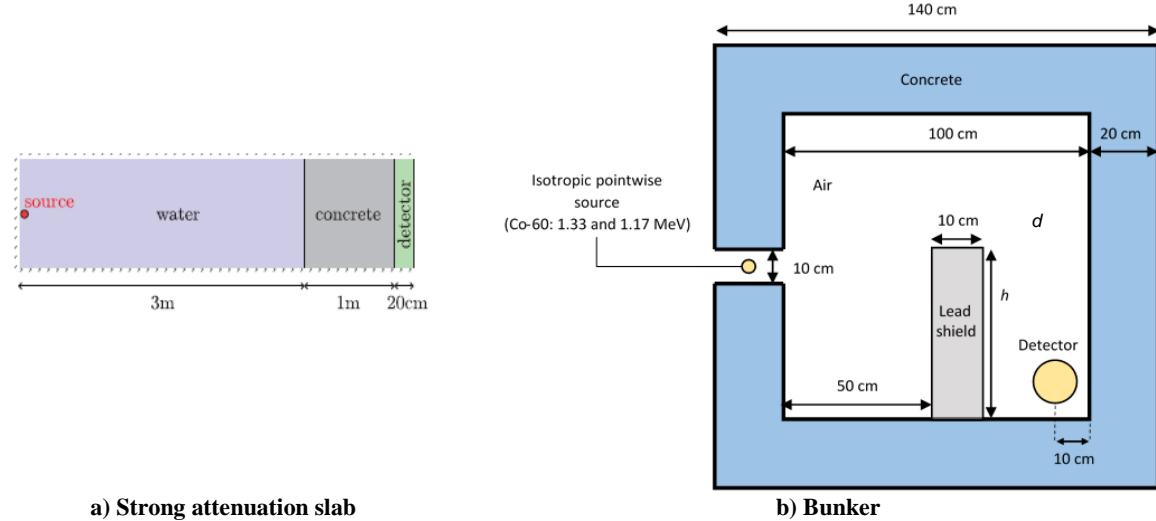
The geometry is fairly simple. It is composed of a water slab with an adaptive thickness varying between 50 cm and 300 cm, a fix-length concrete slab (100 cm) placed at the rightmost edge of the tank, and a second concrete slab on the right (20 cm) which serves as the detector volume. The  $^{60}\text{Co}$  pointwise source is located in the water tank, 1 cm from the left border. Reflection conditions are imposed on all surfaces except for the distal surface of the detector to the plane, to make this a 1D problem.

#### 3.3.2 Simple bunker

The simple bunker configuration (Figure 3.14b) is originally inspired from a paper by Burn [68]. It reproduces a classic shielding problem: an empty room with a photon-emitting source, closed by concrete walls, a radiation shield made of a highly absorbing material, and a detection volume placed behind the shield and close to a corner of the room. In the version proposed here, the dimensions were reduced to simplify the geometry and lower simulation times, and the geometric configuration was slightly simplified with respect to the original version.

Nonetheless, this simple configuration includes all the complexities of photon transport that we have presented until now. The radiation shield of variable thickness causes strong attenuation; the concrete walls provide multiple scattering and reflection at the interface with air; finally, the position of the detector volume (behind the shield and close to a corner) ensures that both these effects have a strong impact on the estimation of the results. Due to these features, this configuration presents challenges both for stochastic and deterministic codes.

The detailed geometry is composed of a cubic concrete box of  $1\text{ m}^3$  filled with air, a  $^{60}\text{Co}$  pointwise source, placed inside a 10 cm hole in the wall, a lead shield of variable height and depth, and a 10 cm radius sphere as the detector volume. Six cases are studied, with different values for the height and depth of the shield, as reported in Table 3.2.



**Figure 3.14: Geometry of the strong attenuation slab (a) and bunker (b) configurations.**

**Table 3.2: Lead shield height ( $h$ ) and depth ( $d$ ) variation inside the bunker**

Name	$h$ (cm)	$d$ (cm)
R1	50	10
R2	80	10
R3	100	10
R4	50	20
R5	80	20
R6	100	20

### 3.3.3 Complex bunker<sup>3</sup>

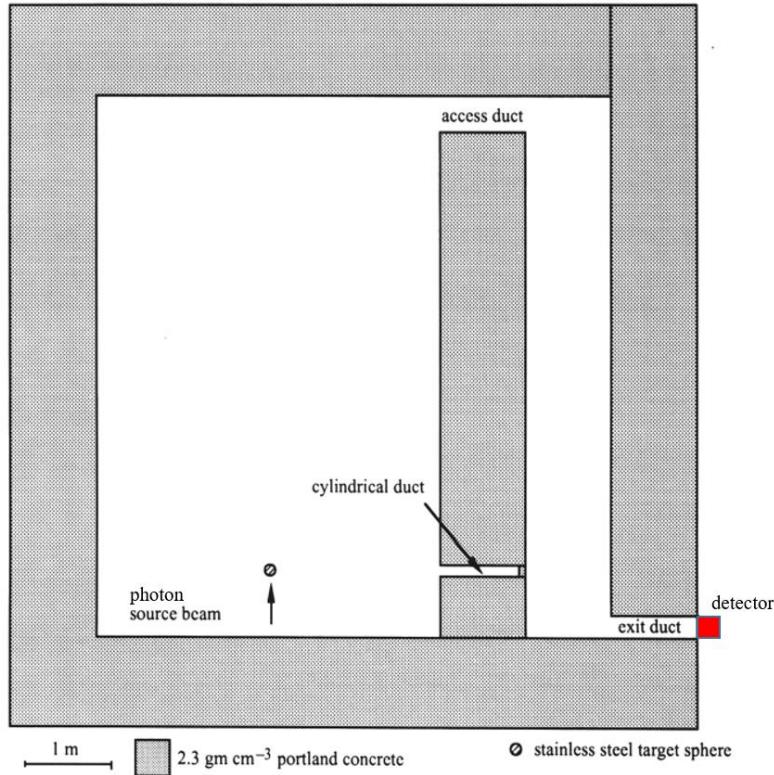
This version of the bunker configurations is the exact reproduction of the one used in [68]. It is similar to the simplified version of Section 3.3.2, but it presents some key differences that make it more complex to treat with an MC simulation. The geometry is larger: the bunker's external dimensions are 8 m x 8 m x 2 m and the concrete walls are 1 m thick. The configuration is studied with photons: the pointwise source of the simplified configuration is replaced by a scattering stainless steel (SS) sphere with a  $^{60}\text{Co}$  photon beam impinging on it. The shield in this case is made of concrete and it presents a horizontal hole right in front of the SS sphere, offering a preferential

---

<sup>3</sup> The analysis on this configuration was performed by Nicolas Guillevic during his Master's thesis (M2 stage)

way for photons. Finally, the detector is placed right outside the bunker, in front of a small opening constituting an exit path for particles.

The configuration is presented in Figure 3.15.



**Figure 3.15: Geometry of the complex bunker configuration, adapted from [68]**

The complex bunker configuration is also used in [69] to test a new acceleration technique applied to MC neutron transport. The solution with an analog MC simulation took, in this case, almost 19 hours to compute, showing the need for some kind of acceleration.

### 3.4 The need for accelerating Monte Carlo calculations

We present in this section the results obtained with the deterministic code NARMER and with the MC code TRIPOLI-4® for some of the configurations previously described. The aim of these simulations is to show the accuracy limits of deterministic codes as previously detailed, but also the excessively large amount of time that are sometimes required by MC simulations to obtain a meaningful result. NARMER's results for the back-scattering and slant configuration are taken from the code's latest VVQI tests<sup>4</sup>.

Table 3.3 shows the results obtained for the back-scattering slab presented in 3.2.1. The quantity calculated is the DER (*Dose Equivalent Rate*, see Annex A), as the albedo module of NARMER does not allow to calculate KERMA. When using the single-differential albedo, NARMER underestimates the result by about 10% with

---

<sup>4</sup> Visonneau T. (2020) NARMER-1 version 5.1 : Vérification et Validation. CEA Cadarache, communication interne

respect to TRIPOLI-4®; when using double-differential albedo, results are overestimated by a factor ranging from 5% to 7%.

**Table 3.3: Results for the back-scattering configuration**

Detector	TRIPOLI-4®			NARMER (simple albedo)			NARMER (double differential albedo)			
	number	DER ( $\mu\text{Sv/h}$ )	$\sigma$ (%)	t (s)	DER ( $\mu\text{Sv/h}$ )	t (s)	Deviation (%)	DER ( $\mu\text{Sv/h}$ )	t (s)	Deviation (%)
1	1.05E+02	0.85			9.430E+01		-10.19	1.151E+02		9.62
2	1.01E+02	0.87			9.009E+01		-10.80	1.090E+02		7.92
3	9.61E+01	0.91	70491		8.565E+01	Few seconds	-10.87	1.027E+02	Few seconds	6.87
4	9.12E+01	0.96			8.104E+01		-11.14	9.624E+01		5.53
5	8.49E+01	0.99			7.633E+01		-10.09	8.980E+01		5.77

Results obtained with double-differential albedo are conservative from a safety point of view, however we can still note a certain error with respect to the MC results. Finally, it is worth mentioning that the TRIPOLI-4® simulation of such a geometrically simple configuration took almost 20 hours to converge to the desired statistical error of 0.1%.

**Table 3.4: Results for the slant penetration configuration**

Detector's horizontal distance from source (cm)	Experiment		TRIPOLI-4®			NARMER		
	KERMA ( $\mu\text{Gy/h}$ )	KERMA ( $\mu\text{Gy/h}$ )	$\sigma$ (%)	C/E	t (s)	KERMA ( $\mu\text{Gy/h}$ )	t (s)	Deviation (%)
0	1.24E+04	1.38E+04	0.06	1.11		1.45E+04		16.94
10	9.38E+03	9.47E+03	0.08	1.01		1.01E+04		7.68
20	3.85E+03	3.81E+03	0.12	0.99		4.08E+03		5.97
30	1.16E+03	1.22E+03	0.21	1.05		1.32E+03		13.79
40	3.55E+02	3.65E+02	0.39	1.03		3.92E+02		10.42
50	1.10E+02	1.10E+02	0.70	1.00	650011	1.13E+02	Few seconds	2.73
60	3.52E+01	3.51E+01	1.22	1.00		3.24E+01		-7.95
70	1.23E+01	1.21E+01	2.02	0.98		9.20E+00		-25.20
80	5.40E+00	4.93E+00	3.00	0.91		2.64E+00		-51.11
90	2.58E+00	2.40E+00	4.18	0.93		7.53E-01		-70.81
100	1.20E+00	1.50E+00	4.99	1.25		2.17E-01		-81.92

Table 3.4 shows the results for the slant penetration slab of Section 3.2.2. In this case, it is possible to calculate the KERMA, so a comparison with the original experimental results of Odano *et al.* is included. Unfortunately, the experimental results do not include any information about uncertainty.

Results obtained with NARMER are conservative with respect to the experimental values on the first six detectors, with errors up to 17%; however, they vastly underestimate the values for the furthest detectors, with negative errors up to 82%. TRIPOLI-4® values are in agreement with experimental results; however, the simulation took 7.52 days to converge to the required target statistical error of 5% for the furthermost detector.

Finally, Table 3.5 lists results obtained for the R1 bunker configuration presented in Section 3.3.2. For this preliminary test, the detector is replaced by nine point detectors to increase the statistics obtained with a single code run. When using NARMER with BU but without albedo, all results are underestimated, with errors ranging from 5% to 15%. We can note that the errors grow for detectors which are closer to the concrete walls: in this case the contribution due to wall reflection is higher, so results without albedo are progressively less accurate, as expected.

**Table 3.5: Results for the bunker configuration**

Detector position relative to center of geometry (cm)			TRIPOLI-4®			NARMER		
x	y	z	KERMA ( $\mu\text{Gy/h}$ )	$\sigma$ (%)	t (s)	KERMA ( $\mu\text{Gy/h}$ )	t (s)	Deviation (%)
35	-35	0	1.06E-09	0.42		1.00E-09		-5.66
35	-40	0	9.49E-10	0.42		8.62E-10		-9.17
35	-45	0	8.63E-10	0.54		7.31E-10		-15.30
40	-35	0	1.04E-09	0.39		9.46E-10		-9.04
40	-40	0	9.27E-10	0.48	26704	8.27E-10	Few seconds	-10.79
40	-45	0	8.36E-10	0.59		7.12E-10		-14.83
45	-35	0	9.83E-10	0.51		8.94E-10		-9.05
45	-40	0	8.99E-10	0.51		7.91E-10		-12.01
45	-45	0	8.07E-10	0.82		6.89E-10		-14.62

The examples proposed in this chapter show that, even for some relatively simple and small geometric configurations, MC calculations can sometimes struggle to give a solution in an acceptable amount of time. As mentioned before, the solution to this problem lies in a class of algorithms and techniques devised to enhance the performance of stochastic calculations, generally called acceleration techniques.

The concept of acceleration techniques is presented and explained in detail in the next chapter. At first, we will give a general overview of the various type of acceleration techniques, by listing the most important and most widely used. Then, we will focus on the techniques implemented in the TRIPOLI-4® code, which was used for the purpose of this work.



# Chapter 4

## Variance reduction techniques for Monte Carlo transport

A “classic” MC simulation, where the physics of the problem under consideration is reproduced in the most exact way possible, is called an **analog simulation**. It is however possible to somewhat “tweak” the laws of physics in a way that favours the simulation, paying attention that this manipulation does not bias the result in any significant way: in this case, we enter the vast domain of **non-analog Monte Carlo techniques**.

Since a MC simulation tries to solve a problem through a probabilistic approach, every result is actually an estimate of an exact value, to which a **statistical variance** is associated. Therefore, to compute a solution by a MC approach actually means to achieve a sufficiently accurate estimate, *i.e.* to reduce the variance to a certain predefined limit. Moreover, it can be demonstrated that in an analog simulation, for a sufficiently large number of histories  $N$ , the variance (see Section 1.3) is inversely proportional to  $N$ . In our specific case, a “history” corresponds to the full life of a particle, with its interactions, the path that it travels, and all the relevant information that it carries.

One could assume that, to improve the efficiency of a simulation, it would be sufficient to reduce the variance associated to a certain number of histories. In fact, it is also necessary to account for the additional time required by the code to implement the variance reduction itself. For example, let us suppose we have devised a method that, for a given number of histories, reduces the variance by a factor 2 with respect to the analog simulation. Another way of seeing this is that the “improved” simulation will need half as much histories as the analog one to attain the same final variance. However, this new method involves some additional calculations to be performed by the code at each step of the particle’s history, which increases the computation time so that now every history takes 4 times longer to be completed. It is easy to see how the new “improved” simulation will actually take double the time of the analog one to reach the same result.

To account for this, we define the **Figure Of Merit (FOM)** of a Monte Carlo simulation, sometimes also called *efficiency* [70], as follows:

$$FOM = \frac{1}{\sigma^2 \cdot T} \quad (4.1)$$

where  $\sigma^2$  is the variance and  $T$  is the time required for the simulation to reach said variance, which is directly proportional to the number of histories. The higher the FOM is, the more efficient the calculation is. Therefore, we can categorize acceleration approaches based on which of these two quantities – the time or the variance – they aim at reducing.

Techniques that focus on reducing the time elapsed for a given number of histories are sometimes called Efficiency Enhancing Techniques (EET). We can further subdivide them into:

- **hardware approaches**, which simply act on the calculator-related part of the code and accelerate it typically through CPU (or, more recently, GPU) parallelization,
- **algorithmic approaches**, which aim to optimize the code, either perfecting the implemented algorithms or introducing new ones that can save calculation time by means of some simplification.

The other approach, aiming to minimize the variance for a given number of histories, is the most adopted and researched, because of the many possibilities it includes. These techniques always follow an algorithmic approach, usually introducing biases in the simulation's physics. They are generally called Variance Reduction Techniques or VRT.

In this Chapter, we briefly detail the most used acceleration techniques in Monte Carlo codes; then, we focus on the TRIPOLI-4® code and the VRT implemented in it. Finally, we present a study investigating the efficiency of TRIPOLI-4®'s VRT when applied to some test cases relevant to the scope of this thesis (taken from the ones presented in Chapter 3).

## 4.1 Efficiency Enhancing Techniques

### 4.1.1. Hardware approaches

Concerning hardware approaches, the best example is the universally recognized technique of code parallelization. Actually, MC simulations are extremely well suited to this type of implementation, because of the independent nature of the simulated histories. For the process to work, each history must be statistically independent from the others, therefore it is a good idea to assign different histories to different processors of a calculator. As a matter of fact, if we don't consider initialization time (the time necessary at the begin of the simulation to initialize the geometry, the materials etc., which often needs to be treated separately), the gain in efficiency resulting from parallelization is directly proportional to the number of processors. Some care should be taken when choosing the random seeds used to initialize the random number sequence for each processor, as the same seed would imply the non-independence of the generated histories.

Another technique that has been gaining more and more attention is the use of Graphical Processing Units (GPU) for the parallelization of the MC simulation. This approach is based on the recent evolution of GPU from specific tools for graphical rendering to powerful computation devices, which increased the interest of researchers in them. GPU parallelization algorithms usually assign to the GPU the bulk of the simulation, *i.e.* the simulation of the particles' random walks, while the CPU usually manages the geometry and other parts of the simulation. In a

specific implementation of GPU parallelization to photon transport [71], researchers found acceleration factors ranging from 15 to 35 with respect to a CPU simulation. In another implementation [72], the GPU was used for deterministic transport in a new estimator for the code MCATK, called VRC (*Volumetric Ray Casting*) estimator, very similar to the *e*-TLE developed in this work. The VRC showed acceleration factors ranging from 2 to about 24, depending highly on the problem geometry.

#### 4.1.2 Algorithmic approaches

On the algorithmic side, EET usually consist in truncating a part of the phase space that does not contribute significantly to the solution. They reduce the computation time for a given number of histories but can introduce a bias, because removing a part of the simulation domain always implies an approximation in the results. In the field of particle tracking, these approaches are mainly devoted to optimizing the simulation by operating some “cuts”, which are numerical values below or above which particles are no longer tracked. Depending on the code and on the problem, we can decide not to follow particles under a certain energy (because they do not contribute significantly to the solution). We could also decide that particles that are in an area far enough from the detection volume will never reach it, and therefore operate a “geometrical” cut of the domain. Another option is to combine energy cuts and geometry cuts, by selecting multiple cut values that depend on the region where the particle is. As an example, the code GEANT4 [73] allows the user to define “region-specific cuts”. With this functionality, a user could deactivate the electromagnetic shower in an uninteresting region of the geometry, while still retaining the maximum accuracy in more important areas, such as areas close to detectors.

An example in which some “physical” cuts are introduced in the simulation is given by [74], somewhat inspired by another technique presented in [75], called **range-rejection**. In this paper, the authors apply the technique to a Monte Carlo simulation of a tungsten Multi-Leaf Collimator (MLC)<sup>5</sup>. The technique presented by Brualla *et al.* [74] consists in defining some regions of the leaves, called *skin regions*, and then applying electron and positron transport only in said regions. In short, while photons are transported everywhere in the MLC, electrons and positrons are only transported in skin regions, and instantly absorbed elsewhere. To do this the material “tungsten” is defined twice in the input file with different cuts for electrons and positrons: the tungsten used to define the skin regions presents small energy cut for electrons and positrons while the tungsten used for the rest of the collimator presents higher energy cuts for electrons and positrons transport. This idea arises due to the reduced range of electrons (and positrons) in tungsten, which makes it difficult for such particles to escape the collimator if they are not very close to the surface. Since only the particles that exit the collimator and reach the target are relevant to the simulation, particles that have a small probability of doing so are simply neglected. This method has to be adapted, case by case, to the simulation: specifically, the thickness of skin regions depends on the geometry under consideration. However, it shows very good results; the ratio between the efficiency of the accelerated simulation and the reference analog simulation ranges from 2.28 to more than 22, in the case of rare events.

As said before, algorithmic EET methods inevitably introduce some bias in the simulation, which have to be carefully quantified and considered when assessing the final results.

---

<sup>5</sup> A collimator is an object used in the medical nuclear field to “collimate”, or “shape”, the photon beam coming (usually) from a LINAC (LINEar ACcelerator). An MLC is a type of collimator that uses several movable pieces (the “leaves”) to accurately shape the beam according to the target.

## 4.2 Variance Reduction Techniques (VRT)

To illustrate the concept of VRT [76, 77], particularly applied to particle transport, in a very broad and intuitive sense, we could say that such techniques attempt to follow **interesting particles** more often than **non-interesting ones**. Interesting particles are defined as those who contribute a large amount to the quantity that needs to be estimated by the simulation. To describe this in a more rigorous way, we need to explain in greater detail the meaning of a quantity related to the MC particle, briefly introduced in Chapter 1: **the weight**.

The **weight** does not have a direct physical meaning. It is a number carried along with each simulation particle representing the particle's relative contribution to the final tallies. Its magnitude is determined to ensure that whenever the Monte Carlo simulation deviates from an exact simulation of the physics, the expected physical result is preserved in the sense of statistical averages. It is initialized at  $1$  and changes during the particle's life. All VRT are based on some kind of algorithm associated to a modification of the weight of the particles.

Depending on how the algorithm impacts the simulation, we can distinguish three broad groups of VRT: population control, modified sampling, and partially deterministic methods.

### 4.2.1 Population Control Methods

VRT of this type aim to control the number of particles based on their position in the phase space. It is preferable to have more particles in the high importance regions of the phase space (*i.e.* particles close to the detector, or with an energy close to some target energy) and less particles in low importance regions. The weight of the particles has to be adjusted accordingly, to preserve statistics.

A well-known example is the **splitting technique**, which can be implemented both in geometry and in energy. Its goal is to favour useful events. Suppose we have a region of particularly high importance situated in the proximity of a detector. If a particle enters that region, it suddenly becomes “more interesting” from the point of view of the simulation, as it has a greater probability to contribute to whichever quantity we are trying to estimate. With splitting, the particle is divided into  $N$  particles, each one of weight  $1/N$ , when it enters the region. Thus, the size of the statistical sample increases, and the variance is reduced. The same thing can be implemented for energy: when a particle approaches a certain energy, it is split.

If splitting aims to increase the number of important particles, a very similar technique called **Russian roulette** does the opposite. It consists in randomly killing, with a given probability, some photons of lesser interest. If the weight of a particle drops under a certain limit value  $w_0$ , the Russian roulette routine is called. Then, the particle can either jump up to a weight  $w_1 > w_0$ , with probability  $w_0 / w_1$ , or be killed, with probability  $1 - w_0 / w_1$ . This prevents the code from having to follow too many particles with too small weights, which could slow down the simulation. Splitting and Russian Roulette are often used together, since they are complementary: particles with a high weight are split, while particles with a low weight undergo Russian Roulette. Sometimes, the splitting/Russian Roulette process also takes under consideration other quantities than weight, like the particles' position or direction of motion, to increase statistics in important areas of the simulation. A well-established example of this process is the treatment of Bremsstrahlung photons emitted at the tungsten target of LINAC or X-ray tube [78]; depending on the problem studied, secondary Bremsstrahlung photons can be split or killed to improve the convergence of the simulation.

### **4.2.2 Modified Sampling Methods**

VRTs of this type act whenever the Monte Carlo code is required to sample a quantity from a certain probability distribution, by replacing the physical distribution with another one. This quantity can be, for example, an interaction type, a mean free path length, a direction of flight. If the weight of the particle is accordingly adjusted, any distribution can be arbitrarily chosen. While population control methods aim to increase the number of particles, these methods, instead, try to increase the number of useful tallies per particle.

**Implicit capture** acts on the capture event, *i.e.* when the simulated particle is absorbed by a target atom. In a purely analog simulation, this event “kills” the particle and terminates its history. It is easy to see how, especially in a highly absorbing medium, this would slow the convergence of the solution. With implicit capture, the absorption is never simulated – in fact, it is removed from the list of possible interactions. Instead, at every interaction the weight of the particle is multiplied by the quantity  $1 - \sigma_a$ ; since  $\sigma_a$  is the absorption cross section of the medium, this quantity represents the probability that the particle is not absorbed in the interaction. In this way, the calculation is accelerated with no bias on the result.

Another well-known VRT belonging to the group of modified sampling is the **exponential transform**, which will be better explained in Section 4.4.1.

### **4.2.3 Partially Deterministic Methods**

The last type of acceleration methods is somewhat linked to the hybrid methods, presented in the next paragraph. In fact, “partially deterministic methods” aim to speed-up the normal random walk process by using deterministic-like techniques. Some examples are the FLUX\_PT option [21] and the DXTRAN method [76] implemented in TRIPOLI-4® and MCNP, respectively.

The **DXTRAN method** in MCNP improves sampling near detectors or other tallies by deterministically transporting particles on collision to some arbitrary, user-defined sphere in the neighbourhood of a tally and then calculating contributions to the tally from these particles. The source, or collision event, is sampled in the usual manner except that the particle is killed if it tries to enter the sphere because all particles entering the sphere have already been accounted for deterministically. As DXTRAN makes it possible to obtain many particles in a small region of interest that would otherwise be difficult to sample, it is widely used in detector design studies [79, 80].

The **FLUX\_PT** in TRIPOLI-4® is a next-event estimator to tally quantities in a point or on a surface. The next-event estimator is included in many MC codes as it is very useful to estimate integral quantities in a point that can be anywhere in the geometry, even very far from the source where it would be difficult for the particles to arrive. It is implemented in TRIPOLI-4® in the following way: whenever a particle undergoes a collision, the code also samples a “fictive” collision with fixed direction towards the point of interest. Then, a fictive particle is deterministically transported to the point, and the code calculates the score that the particle would have given if the fictive collision had really happened. In other words, the score is calculated by instantly moving the fictive particle to the point of interest and changing its weight by considering both material attenuation (related to the total cross sections of materials in the geometry) and geometrical attenuation (related to the distance between collision and point of interest).

## 4.3 The TRIPOLI-4® Monte Carlo code

TRIPOLI-4® [21] is the reference MC and industrial code for CEA (laboratories and reactors), EDF (operating 58 PWRs), ORANO, FRAMATOME and TechnicAtome. It is also the reference code for several packages developed at CEA such as the CRISTAL criticality safety package [81] or the DEMAIN package for dismantling [20]. For historical reasons, the main focus of TRIPOLI-4® is on fission energy, with applications in shielding, reactor physics with and without depletion, criticality, safety, and nuclear instrumentation.

### 4.3.1 Brief overview of TRIPOLI-4®

TRIPOLI-4® is a 3D, continuous-energy MC code for the transport of charged and uncharged particles, developed by the SERMA group at CEA Saclay. It is mainly coded in C++, with a few parts in C and Fortran. Detailed information about the code can be found in [21]. A brief overview is presented here, focused on the features of interest for this thesis.

TRIPOLI-4® is able to track neutrons in the energy range comprised between 20 MeV and  $10^{-5}$ eV, as well as photons, electrons and positrons from 20 MeV to 1 keV. Neutron-photon coupling is handled by default, and photonuclear reactions may be simulated as well, if requested. The continuous-energy data for cross sections can be accessed by TRIPOLI-4® from any file written in ENDF or PENDF format, including JEFF-3.1.1, ENDF/B-VII.1, JENDL-4.0 and FENDL-2.1. The code can also treat multi-group homogenized cross sections in the GENDF format.

TRIPOLI-4® has a built-in geometry module allowing for two kinds of geometric representations: a surface-based one and a combinatorial one, with predefined shapes and logical Boolean operators. TRIPOLI-4® is also fully compatible with ROOT-built geometries [82].

The transport simulation is carried on with batches of particles, the dimensions of which can be specified by the user. Tallies are stored in memory and after every batch the code calculates the mean and the variance for all the quantities requested by the user. TRIPOLI-4® can operate both in criticality mode and in fixed-source mode. Criticality mode is used to solve for the  $k_{\text{eff}}$  eigenvalue equation, which means to find the neutron flux shape distribution corresponding to criticality for a given geometric configuration. This mode of operations requires neutrons as the main particles to transport and it is particularly useful for reactor design studies, for example. Fixed-source mode is used for a number of different studies, among which are the shielding calculations performed in this thesis. For this mode of operation, a particle source (pointwise, surface-based, or volumetric) is imposed by the user, and particles are transported from the source until their disappearance from the spatial or energetic domain. Subsequent batches start with the same user-imposed source, and the simulation continues until either a predefined number of batches have been simulated or when statistical uncertainties fall below a certain threshold. Fixed-source operation can be used to transport any kind of particles: in shielding calculations, it is usually photons.

The code offers two types of estimators for the calculation of quantities over volumes: the collision estimator COLL and the track-length estimator TRACK. The volumes over which the estimation is performed can either be defined in the geometry, or be part of a mesh structure which is super-imposed to the geometry.

As mentioned before, TRIPOLI-4® can simulate photons in the range 20 MeV – 1 keV. Differential cross sections (DCS) and cross sections (CS) are read from ENDF format evaluation files for all relevant photon-matter

interactions, notably [83]: coherent (Rayleigh) scattering, incoherent (Compton) scattering, photoelectric effect with K, LI, LII and LIII shell interaction, and pair effect. A detailed description of photon physics implementation in the code is given in [84].

The electron and photon physics in TRIPOLI-4® has been tested against other established MC codes (MCNP [14], PENELOPE ) and satisfactory accordance was found [84]. For the purpose of this thesis, electron and positron transport is disregarded.

### 4.3.2 VRT in TRIPOLI-4®

TRIPOLI-4® supports a varied choice of VRT, some of which are implicit, which means that they are included in the simulation by default, while some have to be explicitly requested and put in place by the user. In photon calculations, implicit capture (see Section 4.3.2) and implicit pair production are active by default.

User-requested VRT, on the other hand, are all based on the concept of importance and on the creation of importance maps. Hence, it is convenient to first explain how the importance generation is handled by the code, before giving a more detailed explanation of the VRT under consideration.

In TRIPOLI-4®, the importance function is assumed to be factorized in four parts, as detailed in [21, 85]:

$$\mathcal{I}(\mathbf{r}, \Omega, E, t) = \mathcal{I}_1(\mathbf{r}) \cdot \mathcal{I}_2(\Omega) \cdot \mathcal{I}_3(E) \cdot \mathcal{I}_4(t) \quad (4.2)$$

with

- **$\mathcal{I}_1(\mathbf{r})$ , the spatial contribution.** The user has to define the most important sites of the problem either as discrete attractors or as an analytical attractor surface; the spatial importance function is determined by solving for the shortest optical distance from each point to these attractors. To get an efficient variance reduction, the user has to manually adjust the biasing coefficients, which are automatically determined by the code by solving Placzek's equation [86, 87].
- **$\mathcal{I}_2(\Omega)$ , the directional contribution.** It is calculated automatically by the code during the simulation based on particle's direction.
- **$\mathcal{I}_3(E)$ , the energetic contribution.** The energetic importance function is defined over a few groups that can capture most of the energy dependence of the problem; the user can modify the assumed energy dependence within the group by fixing the exponent of a power law of the form  $E^\alpha$ ; this user modification is optional and, by default,  $\alpha$  is set to 0.
- **$\mathcal{I}_4(t)$ , the time contribution.** The dependence of importance on time takes the form of  $1 + \lambda t$ , where  $\lambda$  can be defined by the user; this is also optional, and by default  $\lambda=0$ . It is relevant to note that for the purpose of this thesis, the transport problem was always assumed to be stationary, therefore this contribution is included just for the sake of completeness.

The module in charge of the preliminary computation of the importance map is called INIPOND [85]. It has found large applications in shielding calculations along the years and it is still extensively used for neutron transport in fission [88] and fusion applications [89]. INIPOND is able to produce approximated solutions of the adjoint problem<sup>6</sup> on a domain discretized in space and energy with a negligible computational time compared to the

---

<sup>6</sup> The adjoint solution of the transport equation constitutes a measure of the particles' importance; therefore, it is widely used in VRT. For a more detailed explanation see for example [90] or [91].

simulation time of a MC code. Detectors defined in INIPOND do not represent actual, physical detectors; rather, they act as fictitious particle attractors. They have to be defined in the spatial domain together with a coefficient, quantifying their attraction strength. During the first batches of the simulation, TRIPOLI-4® operates by default an adjustment of the importance map in order to avoid the particle population explosion due to an excessive splitting and an excessive weight discrepancy between the statistical weights of the particles and the reference one.

Once the importance map has been determined, the user can request one of two VRT to improve the simulation's efficiency: the Exponential Transform (ET), or the Adaptive Multilevel Splitting (AMS).

### Exponential transform

The so-called **exponential transform method** [51, 92] is a modified sampling method that modifies the total cross section of the materials in which particles are transported, making it “easier” for the particles to travel towards a certain direction. The modified macroscopic cross-section is defined by:

$$\Sigma^* = \Sigma_t - k\Omega \cdot \Omega^* \quad (4.3)$$

where  $\Sigma_t$  is the true total cross section,  $\Omega$  is the particle’s direction of travel,  $\Omega^*$  is the preferred direction (in our case, towards the right) and  $k$  is a parameter between 0 and 1 which establishes the degree of biasing. Theoretically  $k$  and  $\Omega^*$  depend on the choice of the importance function  $\mathcal{J}$ . The result of this transformation is that, after every collision, any transport step of the particle towards the direction of interest will be stretched, while any step in the opposite direction will be made shorter, effectively making it easier for the particle to travel in the direction of interest. Of course, the particle’s weight has to be adjusted after every collision to preserve the validity of the results. Moreover, since the exponential transform method is known to produce very large fluctuations in the weight distribution, other techniques (splitting, Russian roulette) are usually implemented to keep the particles’ weight sufficiently close to a reference value.

The ET in TRIPOLI-4® is based on the importance map calculated by INIPOND.

### Adaptive multilevel splitting

In the TRIPOLI-4® AMS algorithm applied to shielding study, particle transport between collisions is kept analog and splitting operations are performed in an adaptive and iterative way. At a given iteration, particles are sorted accordingly to their importance, the best particles (according to the importance ranking, set of size  $N-k$ ) are split and replace those that are ranked worst (keeping the same total number of particles, set of size  $k$ ). This ensures that, in the next iteration, all particles will start in a region of higher importance with respect to the beginning, therefore a little bit closer to the target region. Finally, the weight of all the particles is updated by considering the probability  $p$  of surviving the grouping process:

$$p = 1 - \frac{k}{N} \quad (4.4)$$

The splitting level allowing for this operation is adaptively set at each iteration and depends on the percentage of particles to split (1% by default). Consequently, particles approach to the target over the iterations. The iterations stop when a given number of particles (depending on the number of particles to be split) reach the target and a weighting correction is applied accordingly.

Aside from the possibility of using a pre-calculated importance map with INIPOND, the AMS in TRIPOLI-4® can also be used with simpler, geometry-based importance maps, where the importance in every point of the domain is calculated as the inverse of the distance from some user defined point of interest, usually coinciding with the estimation volume.

## 4.4 Performance study on dismantling applications

In this preliminary study, we test TRIPOLI-4®’s VRT on some of the configurations presented in Chapter 3. The aim of this first study is to evaluate TRIPOLI-4® capabilities in dealing with photon shielding problems in geometries representative of dismantling configurations. Results of this study are published in [93].

### 4.4.1 Methodology

The configurations considered in the study are all presented in Chapter 3. Specifically, they are: the three Odano benchmarks (back-scattering slab, slant penetration slab, duct streaming configuration); the deep penetration water slab, with four increasing water thicknesses; the simple bunker configuration, with six different values of shield thickness and height; the complex bunker configuration, with three increasing shield thicknesses. For each of these configurations, four distinct simulations are performed:

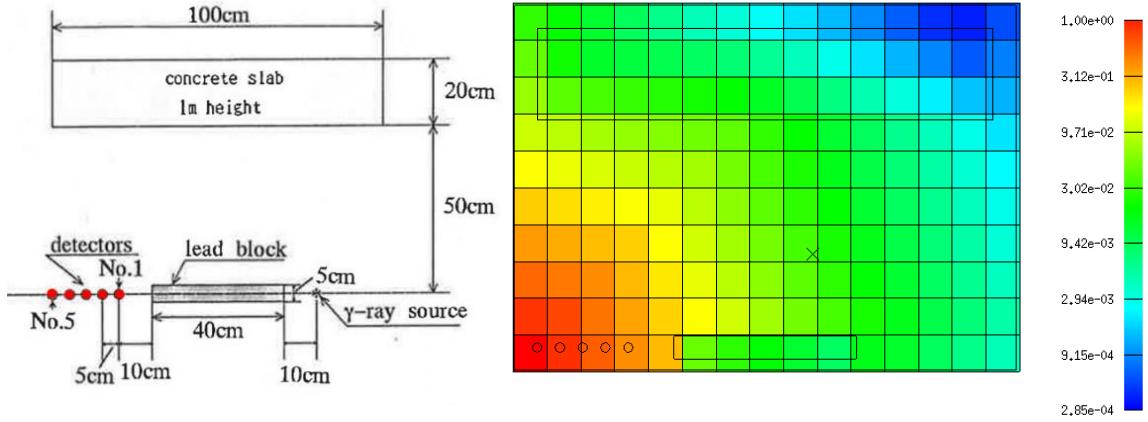
- the **ANALOG simulation** is performed in semi-analog mode, with implicit variance-reduction options only (implicit capture and implicit gamma pair production);
- the **ET-INIPOND simulation** uses the exponential transform, based on an importance map generated by the INIPOND module and manually adjusted by the user, by fine-tuning the importances of different compositions in the simulation;
- the **AMS-GEOMETRIC simulation** is performed with the AMS technique, with an importance map generated by geometric considerations;
- the **AMS-INIPOND simulation** is performed with the AMS technique, based on the same manually-adjusted importance map as the ET-INIPOND simulation.

For each simulation, the track-length estimator is used. The quantity scored is the KERMA for the Odano benchmarks, to allow a comparison with experimental results, and the photon flux for the slab and bunkers configurations. A figure of merit (FOM) is calculated for each detector, to be able to compare the different simulations. The FOM definition is the one given in Equation (4.1). The “gain” of each simulation is calculated as the ratio between its FOM and the FOM of the ANALOG simulation.

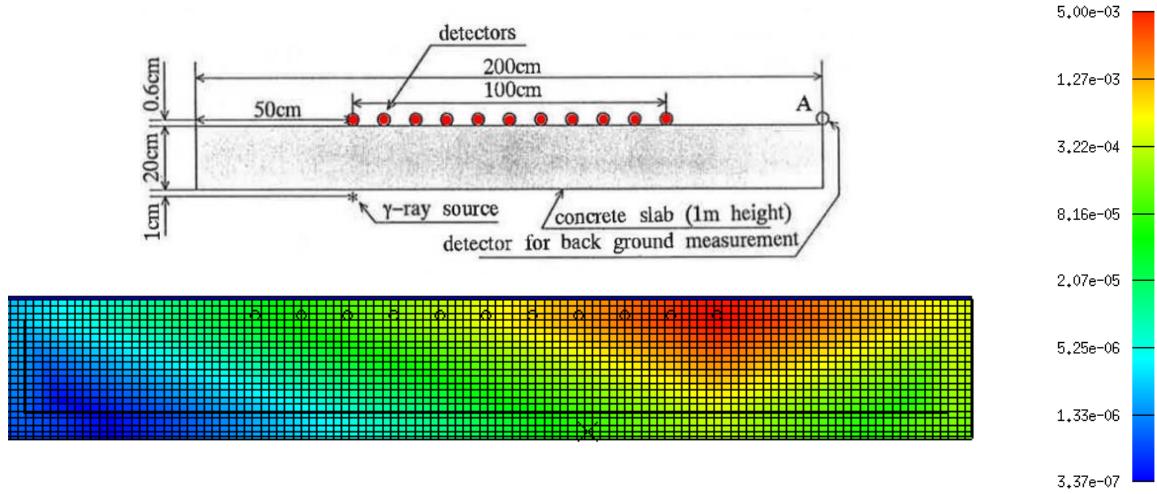
For all the simulations, an average target variance of 5% over the detectors is set as limit. Batches of 5000 particles are used for the ANALOG and ET-INIPOND simulations, while the number of particles per batch is reduced to 500 for the AMS (GEOMETRIC and INIPOND) simulations, as reduced batch sizes have been shown to improve performances [88]. For the Odano benchmarks, the results given by TRIPOLI-4® are also compared both to the experimental measurements (analysis of C/E values, *Calculation/Experiment*) and to the MCNP results provided by [55].

### 4.4.2 Importance maps adjustments

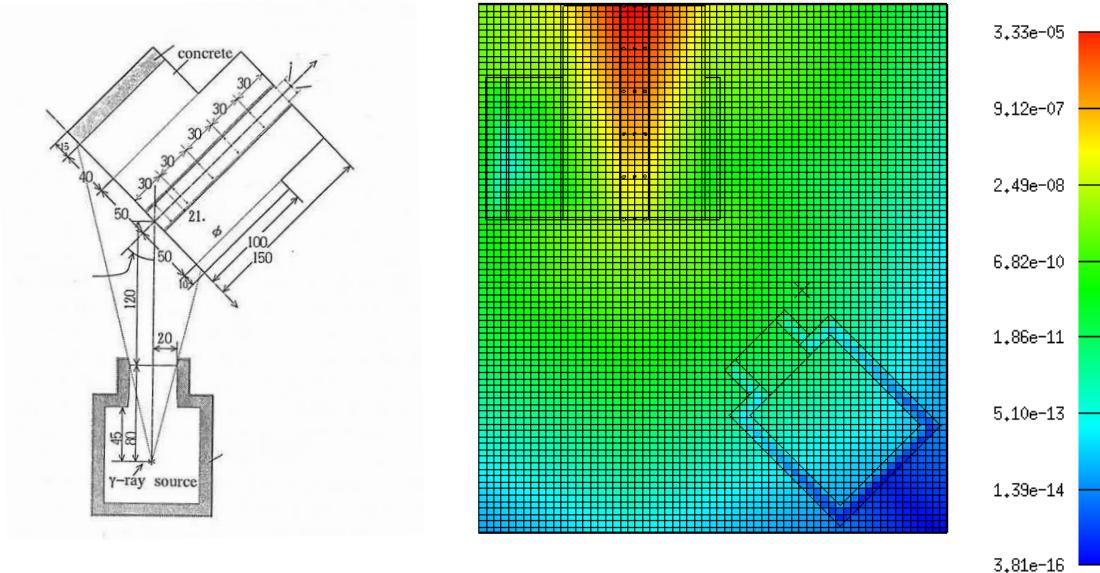
In Figure 4.16, importance maps obtained for Odano’s configurations are reported.



(a) Odano's backscattering configuration: the particle attractor is placed in the leftmost detector.



(b) Odano's slant penetration configuration: the particle attractor is placed in the rightmost detector.



(c) Duct Odano's configuration: the particle attractor is placed in the uppermost detector.

**Figure 4.16:** Importance maps obtained for Odano's configurations. Red indicates high importance, whereas blue indicates low importance.

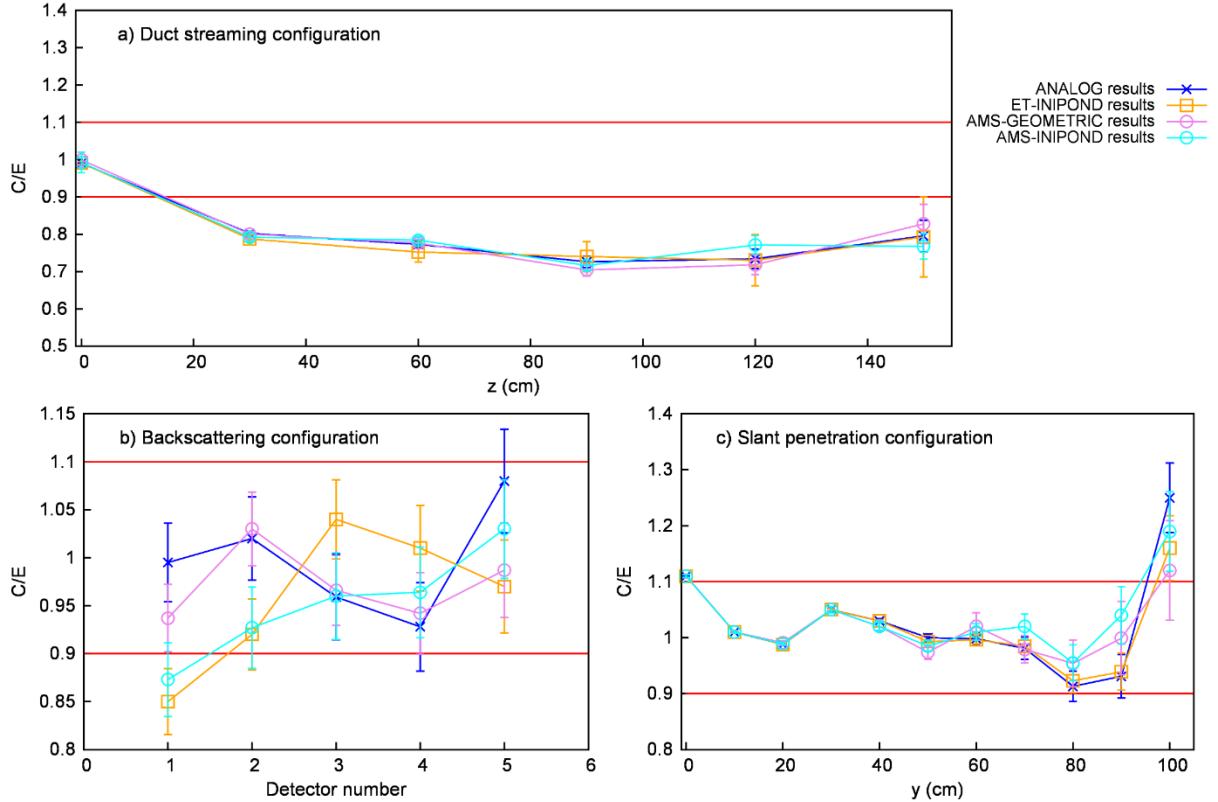
The importance maps are created by the INIPOND module, with manually adjusted biasing parameters for each material. All maps are created by using a single particle attractor, which was placed in the detector farthest from the source.

#### 4.4.3 Comparison with experimental results

C/E values obtained for Odano's configurations are reported in Figure 4.17. Statistical uncertainties are those obtained for MC simulations as no data about experimental uncertainties are given for Odano's configurations. As MC convergence is about 5%, C/E varying between 0.9 and 1.1 ( $\pm 2\sigma$ ) are considered in good agreement with experimental results.

Figure 4.17a shows results for the duct streaming benchmark. The experiment presents 18 detectors, distributed over 6 points along the duct; at each of these 6 points, three detector are placed, with a distance of -7.5, 0 and 7.5 cm perpendicular to the duct axis. In an effort to improve readability, the averaged response of the three detectors at each point is reported in the graph. The C/E plot shows a slight underestimation of the code with respect to experimental data; however, this underestimation is consistent with the one already found by Odano *et al.* with the code MCNP, and attributed to ambiguities in the materials' composition.

Results for the back-scattering configuration are reported in Figure 4.17b. C/E varies between 0.92 and 1.06 for the ANALOG simulations, between 0.85 and 1.04 for ET-INIPOND simulations, between 0.90 and 1.06 for AMS-GEOMETRIC simulations and between 0.87 and 1.03 for AMS-INIPOND simulations. All results, except the ones obtained for detector 1 with ET-INIPOND and AMS-INIPOND simulations, are thus validated.



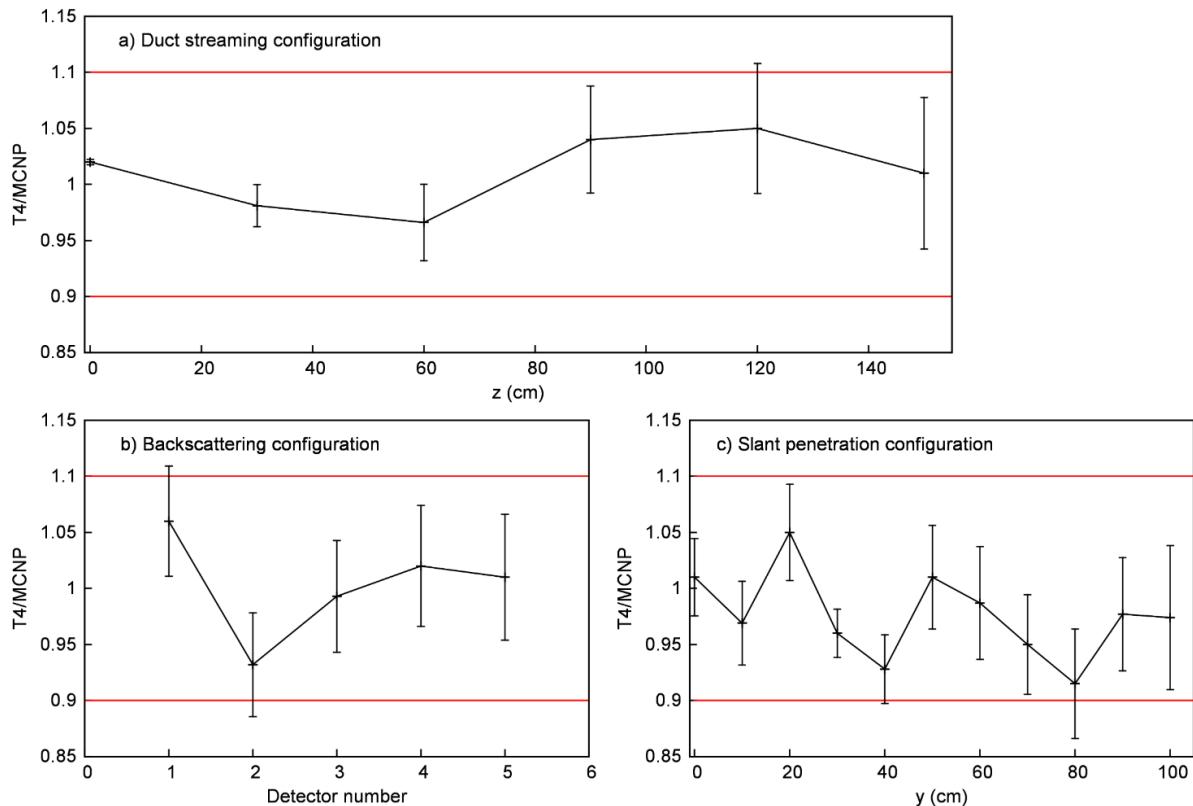
**Figure 4.17:** C/E ratios obtained for Odano's configurations; red lines correspond to the validity domain comprised between  $\pm 2\sigma$ .

Results for the slant penetration benchmark, reported in Figure 4.17c, show a good agreement between simulations and measurements except for the last detector at  $y = 100$  cm.

Deviations observed for detector 1 in Figure 4.17b could be due to the attractor chosen to determine the INIPOND map, as this deviation is observed only for simulations performed with the importance map. Difference between simulation and measurement at  $y = 0$  cm in Figure 4.17c is observed for all simulations.

#### 4.4.4 Comparison with MCNP

Comparisons between TRIPOLI-4® and MCNP results ( $T_4/\text{MCNP}$  ratio) are reported in Figure 4.18 for Odano's configurations. MCNP version 4.2 with photon cross sections data from [94] and [95] is considered here as the reference. Only results obtained for ANALOG simulations are here considered for the comparison. Error bars reported in the plots correspond to the combination of MCNP and TRIPOLI-4® statistical uncertainties. As MC convergence is about 5% for both codes and for all configurations, ratio of TRIPOLI-4® results on MCNP results varying between 0.9 and 1.1 ( $\pm 2\sigma$ ) are considered in good accordance with experimental results. As expected, we can note a good agreement between both MC codes.

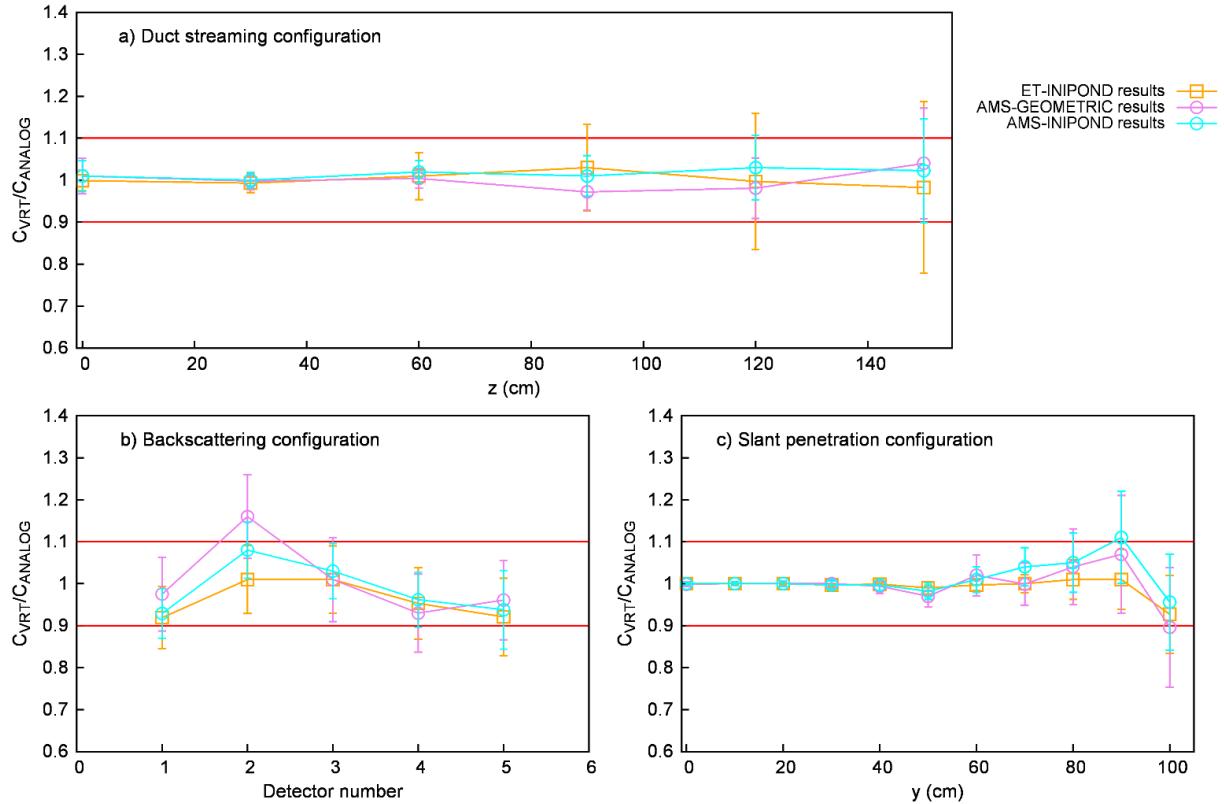


**Figure 4.18:** TRIPOLI-4® /MCNP dose rate ratio obtained for Odano's configurations; red lines correspond to the validity domain comprised between  $\pm 2\sigma$ .

#### 4.4.5 Comparison between VRT-accelerated and analog simulations

Comparisons between analog TRIPOLI-4® and VRT-accelerated TRIPOLI-4® simulations results ( $C_{\text{VRT}}/\text{CANALOG}$  ratios) are reported in Figure 4.19. We report here the results of the comparison obtained over the Odano configurations. We can observe that in all three configurations, all points fall inside the tolerance margin of  $\pm 2\sigma$ , if we consider the uncertainty of the results showed by the error bars in the plots. This is an important result, as it ensures that the VRT-based simulations used in this analysis are not biased and can indeed provide

meaningful results. Therefore, the process of comparing their ability to accelerate the simulations is validated by these results.



**Figure 4.19:  $C_{VRT}/CANALOG$  ratio obtained for Odano's configurations; red lines correspond to the validity domain comprised between  $\pm 2\sigma$ .**

#### 4.4.6 Estimation of TRIPOLI-4®'s VRT performances

In Figure 4.20 the gains of FOM for Odano's configurations are reported. AMS-GEOMETRIC and AMS-INIPOND simulations show good behaviors (FOM gain greater than 1) in the case of the duct streaming configuration in Figure 4.20a as there is a preferred path for the photons to follow. However, AMS fails to accelerate the calculation in all the other cases, as can be observed in Figure 4.20 and Figure 4.20. On the contrary, ET-INIPOND simulations exhibit good behavior in other configurations, with gains of about 1.5 and more than 10 for backscattering - Figure 4.20b - and slant penetration configurations - Figure 4.20c - respectively.

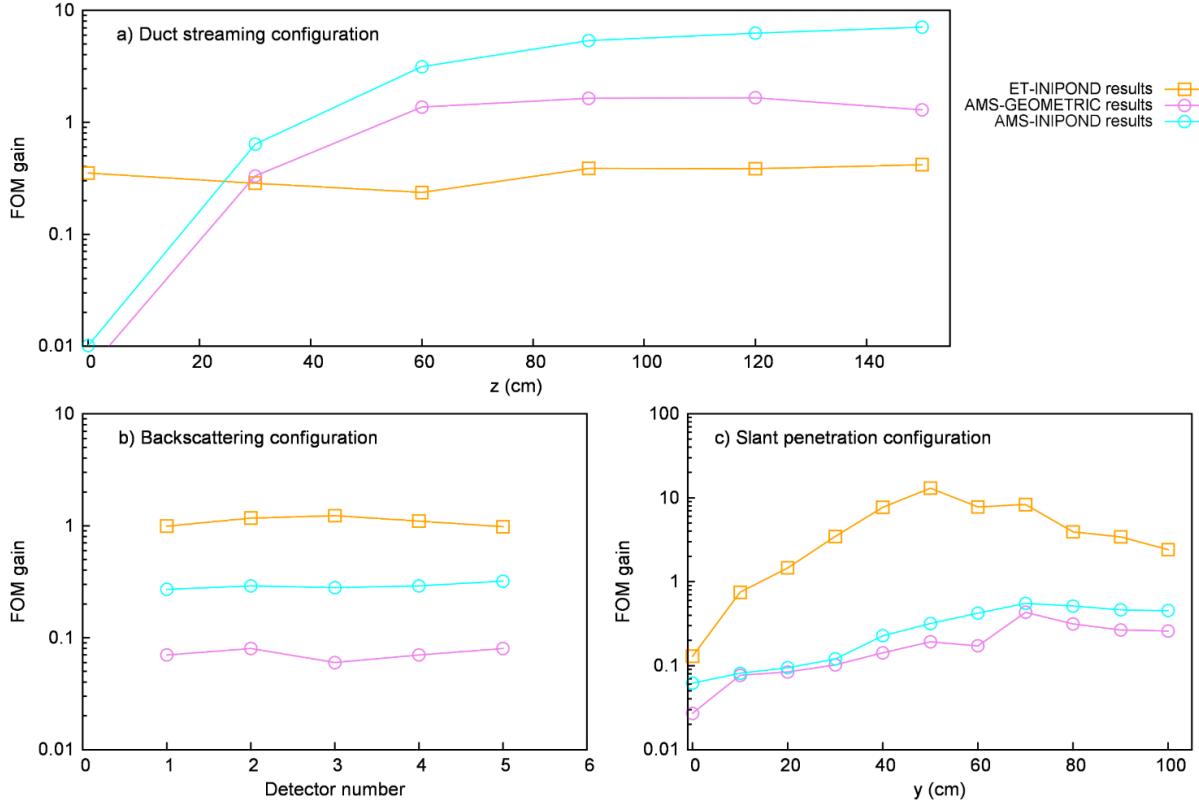


Figure 4.20: FOM gain obtained for Odano's configurations.

For the slab configuration, all VRT simulations lead to a faster convergence of the results with respect to the analog simulation. For a water thickness of 300 cm, ANALOG simulation results are not available because simulation computation time was too long: therefore, since plotting the gain was impossible for this last configuration, Figure 4.21 shows the FOM for the different simulations. In general, the AMS-INIPOND is the most efficient technique as the water thickness increases, with a FOM gain close to 300 for the 200 cm water thickness, as can be seen in Figure 4.21.

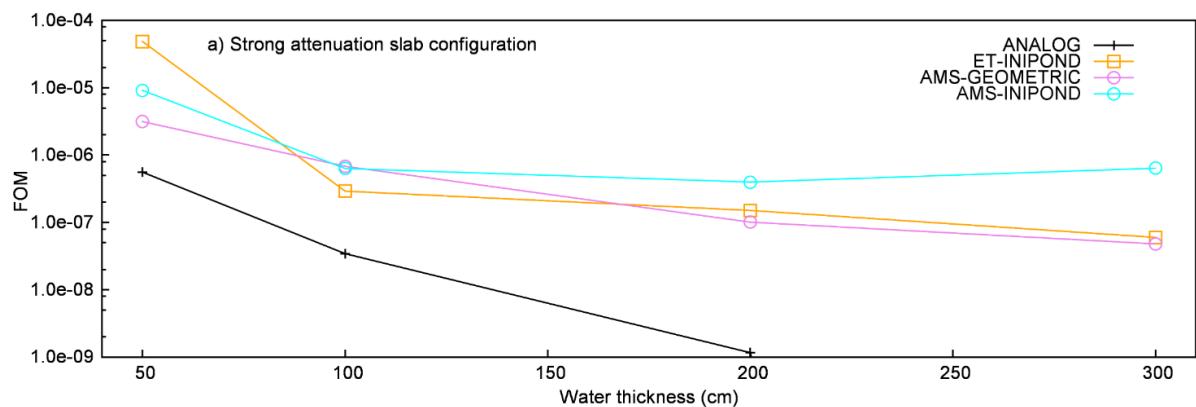
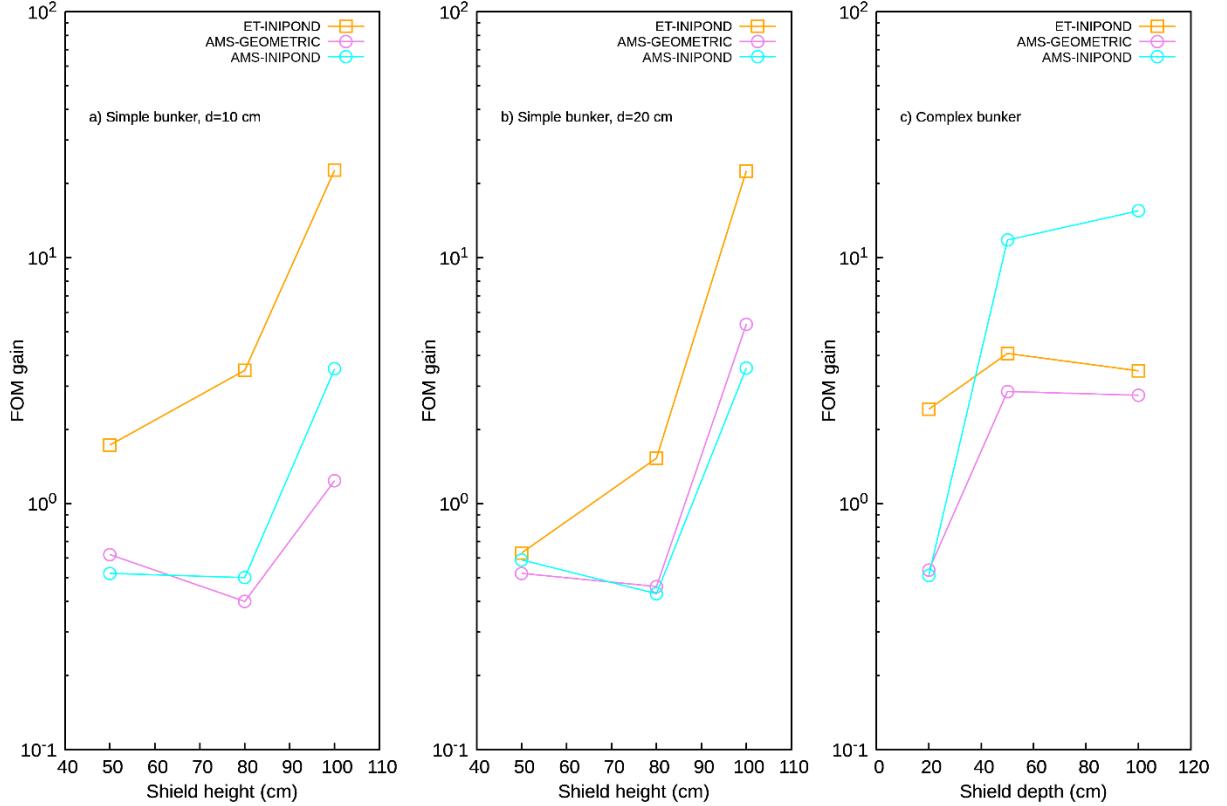


Figure 4.21: FOM as a function of water thickness for the strong attenuation slab configuration

Results for the simple bunker and complex bunker configurations are reported in Figure 4.22.. The acceleration observed strongly differs depending on the VR methodology used and on the geometric configuration, as shown by the gain plot. For the simple bunker case (Figure 4.22a and Figure 4.22b) the ET-INIPOND simulation achieves the best performances for all shield heights and thicknesses, with a maximum acceleration ratio of ~20. On the

other hand, AMS shows good behaviour only for the two cases with no gap between wall and ceiling ( $h=100$  cm), failing to accelerate the simulation in all other cases. In the complex bunker configuration (Figure 4.22c) we see that, for the lowest shield thickness of 20 cm, results are confirmed: AMS fails to accelerate the simulation, while INIPOND achieves a gain of about 2. However, as we explore configurations with very large shield thicknesses of 50 and 100 cm, the results change: as the thickness increases, the AMS-INIPOND becomes the better performing VRT, showing a gain of more than 10. Results of INIPOND and AMS-GEOMETRIC become comparable at large thicknesses, with a gain between 2 and 3.



**Figure 4.22: FOM gain as a function of shield height ( $h$ ) and depth ( $d$ ) for the simple and complex bunker configurations.**

## 4.5 Conclusions

This chapter gives a comprehensive resume of the most important acceleration techniques for MC codes, listing some of the precautions that have to be taken when using such techniques in a MC calculation. Some of them introduce a bias that needs to be carefully assessed, some others can be useless or even counterproductive (*i.e.* slow down the simulation) if not used in the correct way or in the correct framework. The study presented in the last part of the chapter provides an analysis of the performance of the variance-reduction methods implemented in TRIPOLI-4® in dealing with some classical shielding problems for photons in dismantling and the associated dose management. The exponential transform is a reliable choice to deal with various type of problems; however, it requires a good amount of user experience in the creation of the importance map. The implementation of Consistent Adjoint Driver Importance Sampling (CADIS) like methods [96] in TRIPOLI-4® could be a precious help for TRIPOLI-4® users in a near future to automatically determine efficient importance maps, as it has already been

shown for neutron transport in [88]. This study seems to show also that AMS can achieve excellent performances in well adapted problems, but it retains the same drawback of exponential transform, as it requires the same importance map to achieve the best results.

In conclusion, both methods (exponential transform and AMS) behave better in deep penetration problems. This could be due to the higher number of collisions happening, which make it “easier” for the particles to be redirected towards the points of interest. Moreover, the analysis of the complex bunker shows that the AMS can be a very efficient acceleration method in problems presenting a preferential path for particles to follow. Nevertheless, to help for instance the real time dismantling operations in streaming environments, it is also necessary to develop a new type of algorithm capable to tackle configurations characterized simultaneously by both attenuation and reflection of radiation in geometries with a high percentage of low-density material (*i.e.* air) where the lack of collisions can limit the efficiency of the current VRT. In the next and final part of this thesis, we will present such algorithm and its implementation in TRIPOLI-4®.

---

PART III

A NEW TRACK LENGTH  
ESTIMATOR IN TRIPOLI-4®

---



This part presents the core of the thesis work: the development of a new track length estimator and its implementation in the Monte Carlo code TRIPOLI-4®. The estimator is defined mathematically and applied to a simplified problem, then we present its first implementation in the code along with some validation tests. Two improved versions of the estimator are also introduced, implemented and validated in the final chapters.



# Chapter 5

## Theoretical basis of the exponential track length estimator (*e*-TLE)

As discussed until now, the currently adopted calculation standards for shielding and photon transport studies leave plenty of space open to innovation. The gap between deterministic codes, providing quick approximate solutions to the complex problem of radiation transport, and stochastic MC simulations, granting an indefinitely precise solution with calculation times that are often prohibitive, offers an interesting opportunity to develop new methodologies. In the third part of this thesis, we present the development and code implementation of the exponential track-length estimator (*e*-TLE), an advanced MC estimator successfully used in the medical field to improve the efficiency of MC radiation transport simulations. In this chapter, the estimator is defined mathematically and applied to a simplified theoretical configuration already described in Chapter 2. Then, the code implementation of the *e*-TLE in TRIPOLI-4® is introduced. Finally, we present the results of a first validation study, where the convergence and the efficiency of the new estimator are tested and evaluated over the configurations described in Chapter 3.

### 5.1 Heuristic approach: 1D mono-directional transport using *e*-TLE

The idea behind the *e*-TLE is that a particle's contribution to any estimation in a volume is represented by the contribution the particle would have had by traveling without collisions to the volume. It can be derived mathematically from the basic expression of the TLE, as shown in Chapter 1. We now recall the simplified configuration presented in Chapter 2 (Figure 5.23) to show the unbiasedness of such estimator when applied to a simple 1D case. We recall that, for a monokinetic, monodirectional, stationary surface source of intensity  $S$  (particles · cm<sup>-2</sup> · s<sup>-1</sup>) impinging on a 1D homogeneous slab of length  $L$ , we can write the uncollided or primary flux as:

$$\phi_0(x) = S e^{-\Sigma_t x} \quad (5.1)$$

which represents the flux of particles that travelled from the source to point  $x$  without undergoing any interaction.

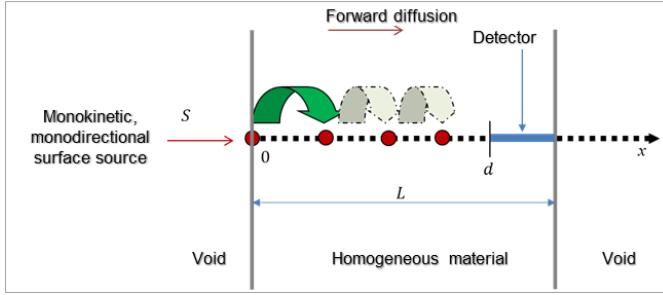


Figure 5.23: 1D transport problem configuration (from Chapter 2)

In the same way, if we define as “detector” as a segment of length  $l$  placed at a distance  $d$  from the source, taking into account Equation (2.41) we can write an expression for the average uncollided flux in the detector  $\phi_{D,0}(x)$ :

$$\phi_{D,0} = S e^{-\Sigma_t d} \frac{[1 - e^{-\Sigma_t l}]}{l \Sigma_t} \quad (5.2)$$

Starting from here, we can write a new estimator  $\kappa_n(x)$  for the average detector flux  $\phi_D$  defined by the following equations:

$$\kappa_n(x) = S \varpi_n e^{-\Sigma_t(d-x)} \Sigma_t \frac{\Sigma_s [1 - e^{-\Sigma_t l}]}{l \Sigma_t} \quad , \quad x \in [0, d] \quad (5.3)$$

$$\kappa_n(x) = S \varpi_n(x) \Sigma_t \frac{\Sigma_s [1 - e^{-\Sigma_t(L-x)}]}{l \Sigma_t} \quad , \quad x \in [d, L] \quad (5.4)$$

with  $\varpi_n = (\Sigma_s / \Sigma_t)^n$  being the weight of the particle at collision  $n$ . We recall that the cross sections are supposed to be constant and independent from energy. We now divide the total contributions to the estimation in three terms: one for particles having a collision before the detector, called  $\vartheta_1(x)$ ; the other for particles colliding inside the detector, called  $\vartheta_2(x)$ ; finally, the contribution given by particles coming directly from the source with no collisions (primary particles), called  $\vartheta_s$ .

The expression of the latter is straightforward, as it corresponds to the average uncollided flux of Equation (5.2):

$$\vartheta_s = \phi_{D,0} = S e^{-\Sigma_t d} \frac{[1 - e^{-\Sigma_t(L-d)}]}{l \Sigma_t} \quad (5.5)$$

For the other two contributions,  $\vartheta_1(x)$  and  $\vartheta_2(x)$ , we will start from a Neumann series decomposition (see Chapter 2):

$$\vartheta_1(x) = \sum_{n=0}^{\infty} \vartheta_{1,n}(x) \quad (5.6)$$

$$\vartheta_2(x) = \sum_{n=0}^{\infty} \vartheta_{2,n}(x) \quad (5.7)$$

The quantity searched,  $\phi_D$ , will then be given by:

$$\phi_D = \int_0^d \vartheta_1(x) dx + \int_d^L \vartheta_2(x) dx + \vartheta_s \quad (5.8)$$

### 5.1.1 Evaluation of the first term, $\vartheta_1(x)$

Remembering Equation (5.6) we develop the series, writing each term  $\vartheta_{1,n}(x)$  starting from  $n = 0$  and increasing the collision order:

$$\vartheta_{1,0}(x) = S e^{-\Sigma_t x} e^{-\Sigma_t(d-x)} \sum_t \frac{\Sigma_s}{\Sigma_t} \frac{[1 - e^{-\Sigma_t l}]}{l \Sigma_t} = S e^{-\Sigma_t d} \sum_s \frac{[1 - e^{-\Sigma_t l}]}{l \Sigma_t} \quad (5.9)$$

$$\vartheta_{1,1}(x) = S \int_0^x e^{-\Sigma_t x_1} \sum_t e^{-\Sigma_t(x-x_1)} \frac{\Sigma_s}{\Sigma_t} \sum_t \frac{\Sigma_s}{\Sigma_t} e^{-\Sigma_t(d-x)} \frac{[1 - e^{-\Sigma_t l}]}{l \Sigma_t} dx_1 = S x \sum_s \frac{\Sigma_s}{\Sigma_t} e^{-\Sigma_t d} S \sum_s \frac{[1 - e^{-\Sigma_t l}]}{l \Sigma_t} \quad (5.10)$$

$$\begin{aligned} \vartheta_{1,2}(x) &= S \int_0^x dx_1 \int_{x_1}^x e^{-\Sigma_t x_1} \sum_t e^{-\Sigma_t(x_2-x_1)} \sum_t e^{-\Sigma_t(x-x_2)} \sum_t \left[ \frac{\Sigma_s}{\Sigma_t} \right]^2 \frac{\Sigma_s}{\Sigma_t} e^{-\Sigma_t(d-x)} \frac{\Sigma_s}{\Sigma_t} \frac{[1 - e^{-\Sigma_t l}]}{l \Sigma_t} dx_2 \\ \vartheta_{1,2}(x) &= S \Sigma_t^2 \frac{x^2}{2} \left[ \frac{\Sigma_s}{\Sigma_t} \right]^2 e^{-\Sigma_t d} \sum_s \frac{[1 - e^{-\Sigma_t l}]}{l \Sigma_t} \end{aligned} \quad (5.11)$$

By induction, we can write :

$$\vartheta_{1,n}(x) = S \Sigma_t^n \frac{x^n}{n!} \left[ \frac{\Sigma_s}{\Sigma_t} \right]^n e^{-\Sigma_t d} \sum_s \frac{[1 - e^{-\Sigma_t l}]}{l \Sigma_t} \quad (5.12)$$

So we can rewrite the Neumann series and solve:

$$\vartheta_1(x) = \sum_{n=0}^{\infty} \vartheta_{1,n}(x) = S \sum_{n=0}^{\infty} \frac{\left( \Sigma_t \frac{\Sigma_s}{\Sigma_t} x \right)^n}{n!} e^{-\Sigma_t d} \frac{[1 - e^{-\Sigma_t l}]}{l \Sigma_t} = S e^{\Sigma_s x} e^{-\Sigma_t d} \sum_s \frac{[1 - e^{-\Sigma_t l}]}{l \Sigma_t} \quad (5.13)$$

### 5.1.2 Evaluation of the second term, $\vartheta_2(x)$

Similarly to what we did for  $\vartheta_1(x)$ , we start from the Neumann series of Equation (5.7) and solve for each term:

$$z \vartheta_{2,0}(x) = S e^{-\Sigma_t x} \sum_t \frac{\Sigma_s}{\Sigma_t} \frac{[1 - e^{-\Sigma_t(L-x)}]}{l \Sigma_t} = S e^{-\Sigma_t x} \sum_s \frac{[1 - e^{-\Sigma_t(L-x)}]}{l \Sigma_t} \quad (5.14)$$

$$\vartheta_{2,1}(x) = S \int_0^x e^{-\Sigma_t x_1} \sum_t e^{-\Sigma_t(x-x_1)} \frac{\Sigma_s}{\Sigma_t} \sum_t \frac{\Sigma_s}{\Sigma_t} \frac{[1 - e^{-\Sigma_t x}]}{l \Sigma_t} dx_1 = S_d x \sum_s \frac{\Sigma_s}{\Sigma_t} e^{-\Sigma_t x} \sum_s \frac{[1 - e^{-\Sigma_t(L-x)}]}{l \Sigma_t} \quad (5.15)$$

$$\begin{aligned} \vartheta_{2,2}(x) &= S \int_0^x dx_1 \int_{x_1}^x e^{-\Sigma_t x_1} \sum_t e^{-\Sigma_t(x_2-x_1)} \sum_t e^{-\Sigma_t(x-x_2)} \left[ \frac{\Sigma_s}{\Sigma_t} \right]^2 \sum_t \frac{\Sigma_s}{\Sigma_t} \frac{[1 - e^{-\Sigma_t(L-x)}]}{l \Sigma_t} dx_2 \\ \vartheta_{2,2}(x) &= S \Sigma_t^2 \frac{x^2}{2} \left[ \frac{\Sigma_s}{\Sigma_t} \right]^2 e^{-\Sigma_t x} \sum_s \frac{[1 - e^{-\Sigma_t(L-x)}]}{l \Sigma_t} \end{aligned} \quad (5.16)$$

By induction:

$$\vartheta_{2,n}(x) = S \Sigma_t^n \frac{x^n}{n!} \left[ \frac{\Sigma_s}{\Sigma_t} \right]^n e^{-\Sigma_t x} \sum_s \frac{[1 - e^{-\Sigma_t(L-x)}]}{l \Sigma_t} \quad (5.17)$$

And finally, solving for the Neumann series:

$$\vartheta_2(x) = \sum_{n=0}^{\infty} \vartheta_{2,n}(x) = S \sum_{n=0}^{\infty} \frac{\left(\Sigma_t \frac{\Sigma_s}{\Sigma_t} x\right)^n}{n!} e^{-\Sigma_t x} \Sigma_s \frac{[1 - e^{-\Sigma_t(L-x)}]}{l\Sigma_t} = S e^{\Sigma_s x} e^{-\Sigma_t x} \Sigma_s \frac{[1 - e^{-\Sigma_t(L-x)}]}{l\Sigma_t} \quad (5.18)$$

$$\vartheta_2(x) = S e^{-\Sigma_a x} \Sigma_s \frac{[1 - e^{-\Sigma_t(L-x)}]}{l\Sigma_t} \quad (5.19)$$

### 5.1.3 Evaluation of the average detector flux $\phi_D$

We now have an expression for each of the terms of Equations (5.5), (5.6) and (5.7). Referring to Equation (5.8), we see that it is possible to find  $\phi_D$  by performing two integrations on  $\vartheta_1(x)$  and  $\vartheta_2(x)$ , and adding the term  $\vartheta_s(x)$ .

We can define the two integrals as:

$$I_1 = \int_0^d \vartheta_1(x) dx, \quad I_2 = \int_d^L \vartheta_2(x) dx \quad (5.20)$$

and finally:

$$\phi_D = I_1 + I_2 + \vartheta_s \quad (5.21)$$

#### Evaluation of the first integral $I_1$

Substituting the expression for  $\vartheta_1(x)$  from equation (5.9) we have:

$$I_1 = \int_0^d \vartheta_1(x) dx = \int_0^d S e^{\Sigma_s x} e^{-\Sigma_t d} \Sigma_s \frac{[1 - e^{-\Sigma_t l}]}{l\Sigma_t} dx = S e^{-\Sigma_t d} \Sigma_s \frac{[1 - e^{-\Sigma_t l}]}{l\Sigma_t} \int_0^d e^{\Sigma_s x} dx \quad (5.22)$$

Now, remembering that the detection length is  $l = L - d$  we can write:

$$\begin{aligned} I_1 &= S e^{-\Sigma_t d} \Sigma_s \frac{[1 - e^{-\Sigma_t l}]}{l\Sigma_t} \frac{1}{\Sigma_s} [e^{\Sigma_s d} - 1] = S e^{-\Sigma_t d} \frac{[1 - e^{-\Sigma_t l}]}{l\Sigma_t} [e^{\Sigma_s d} - 1] \equiv S_d \frac{[1 - e^{-\Sigma_t l}]}{l\Sigma_t} [e^{\Sigma_s d} - 1] \\ I_1 &= S \frac{[e^{-\Sigma_t d} - e^{-\Sigma_t L}]}{l\Sigma_t} [e^{\Sigma_s d} - 1] \\ I_1 &= \frac{S}{l\Sigma_t} [e^{-\Sigma_a d} - e^{-\Sigma_t L} e^{\Sigma_s d} - e^{-\Sigma_t d} + e^{-\Sigma_t L}] \\ I_1 &= \frac{S}{l\Sigma_t} [e^{-\Sigma_a d} - e^{-\Sigma_t L} e^{\Sigma_s d} - e^{-\Sigma_t d} (1 - e^{-\Sigma_t(L-d)})] \end{aligned} \quad (5.23)$$

#### Evaluation of the second integral $I_2$

Similarly to what we did above, we substitute Equation (5.17) in the formula for  $I_2$ :

$$I_2 = \int_d^L \vartheta_2(x) dx = \int_d^L S_d e^{-\Sigma_a x} \Sigma_s \frac{[1 - e^{-\Sigma_t(L-x)}]}{l\Sigma_t} dx = \frac{S\Sigma_s}{l\Sigma_t} \int_d^L e^{-\Sigma_a x} dx - \frac{S\Sigma_s}{l\Sigma_t} e^{-\Sigma_t L} \int_d^L e^{\Sigma_s x} dx \quad (5.24)$$

and:

$$\begin{aligned} I_2 &= \frac{S_d}{l\Sigma_t} \frac{\Sigma_s}{\Sigma_a} [e^{-\Sigma_a d} - e^{-\Sigma_a L}] - \frac{S_d}{l\Sigma_t} \frac{\Sigma_s}{\Sigma_a} e^{-\Sigma_t L} \frac{1}{\Sigma_s} [e^{\Sigma_s L} - e^{\Sigma_s d}] \\ I_2 &= \frac{S_d}{l\Sigma_t} \left\{ \frac{\Sigma_s}{\Sigma_a} [e^{-\Sigma_a d} - e^{-\Sigma_a L}] - e^{-\Sigma_a L} + e^{-\Sigma_t L} e^{\Sigma_s d} \right\} \end{aligned} \quad (5.25)$$

Finally, we can rewrite Equation (5.21) and solve for the flux  $\phi_D$ :

$$\begin{aligned}
\phi_D &= \frac{S}{l\Sigma_t} \left[ e^{-\Sigma_a d} - e^{-\Sigma_t L} e^{\Sigma_s d} - e^{-\Sigma_t d} (1 - e^{-\Sigma_t(L-d)}) + \frac{\Sigma_s}{\Sigma_a} [e^{-\Sigma_a d} - e^{-\Sigma_a L}] - e^{-\Sigma_a L} + e^{-\Sigma_t L} e^{\Sigma_s d} \right. \\
&\quad \left. + e^{-\Sigma_t d} (1 - e^{-\Sigma_t(L-d)}) \right] \\
\phi_D &= \frac{S}{l\Sigma_t} \left[ e^{-\Sigma_a d} + \frac{\Sigma_s}{\Sigma_a} [e^{-\Sigma_a d} - e^{-\Sigma_a L}] - e^{-\Sigma_a L} \right] \\
\phi_D &= \frac{S}{l\Sigma_t} \left[ \frac{\Sigma_s}{\Sigma_a} [e^{-\Sigma_a d} - e^{-\Sigma_a L}] + e^{-\Sigma_a d} - e^{-\Sigma_a L} \right] \\
\phi_D &= \frac{S}{l\Sigma_t} \left[ \frac{\Sigma_s [e^{-\Sigma_a d} - e^{-\Sigma_a L}] + \Sigma_a [e^{-\Sigma_a d} - e^{-\Sigma_a L}]}{\Sigma_a} \right] \\
\phi_D &= \frac{S}{l\Sigma_a} [e^{-\Sigma_a d} - e^{-\Sigma_a L}] \\
\phi_D &= S e^{-\Sigma_a d} \frac{[1 - e^{-\Sigma_a L}]}{l\Sigma_a}
\end{aligned} \tag{5.26}$$

which is the same equation as (2.8), obtained by the analytic solution, and (2.39), obtained with the collision estimator. This proves that the  $e$ -TLE is, in fact, an unbiased estimator of  $\phi_D$ .

#### 5.1.4 Practical example: numerical 1D Monte Carlo

As we did in Chapter 2, we can test the good functioning and the efficiency of this new estimator by numerically reproducing the 1D slab case of Figure 5.23. We use the same code described in Section 2.4, but to the collision estimator already described, we add a TLE and an  $e$ -TLE. Using the same quantities defined in Chapter 2, the contribution scored by the TLE at the end of the  $i$ -th particle history is described by the following algorithm:

$$\kappa_{TLE}^i = \begin{cases} 1 & \text{if the particle travels past } L, \\ \frac{x-d}{L-d} & \text{if the particle is absorbed at } x, \text{ with } d \leq x \leq L, \\ 0 & \text{otherwise} \end{cases} \tag{5.27}$$

Similarly, the contribution scored by the  $e$ -TLE after each collision  $j$  at point  $x$  is defined as:

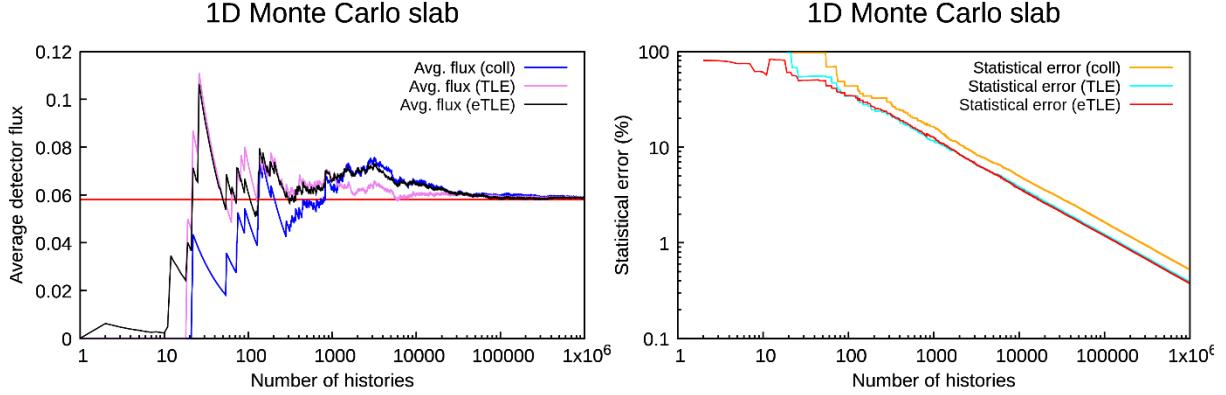
$$\kappa_{e-TLE}^j = \begin{cases} \frac{e^{-\Sigma_t(d-x)}(1 - e^{-\Sigma_t(L-d)})}{\Sigma_t(L-d)} & \text{if } 0 \leq x < d, \\ \frac{(1 - e^{-\Sigma_t(L-x)})}{\Sigma_t(L-d)} & \text{if } d \leq x \leq L, \end{cases} \tag{5.28}$$

And the contribution at the end of the  $i$ -th particle history, after  $J$  interactions, is simply:

$$\kappa_{e-TLE}^i = \sum_{j=1}^J \kappa_{e-TLE}^j \tag{5.29}$$

The sample mean and variance are then calculated as detailed in Chapter 2.

With these new estimators in place, we can now see how they compare with each other in terms of accuracy and performance. Figure 5.24 shows results after 1 million particle histories. We can see that the three estimators all converge to the analytical value; on the right, we can see that the TLE and  $e$ -TLE both perform better than the collision estimator, in that their variance is consistently lower after a certain number of histories. The  $e$ -TLE has a performance comparable to that of the TLE, but it is slightly better.



**Figure 5.24:** Comparison of collision estimator, track-length estimator, and exponential track-length estimator over the 1D slab. The red bar on the left represents the analytical value of the flux.

We can further investigate the behavior of the three estimators by changing the value of the absorption cross section  $\Sigma_a$ . The higher it is, the harder it is for particles to reach the detector: in this case, we expect the performance of the *e*-TLE to get progressively better with respect to the other two estimators. Table 5.6 shows that, for  $\Sigma_a \geq 0.3\Sigma_t$ , the *e*-TLE provides the most accurate results, as expected.

**Table 5.6:** Results for increasingly larger absorption cross-sections. The last two columns show the gain in efficiency of the *e*-TLE with respect to the other two estimators, calculated as a ratio between their variances. The errors with respect to the analytical solution  $\Phi_{D,anal}$  are also reported.

$\Sigma_a$	Collision estimator			Track-length estimator			Exponential track-length estimator			Gain/ coll ( $\sigma_{coll}^2 / \sigma_{e-TLE}^2$ )	Gain/ TLE ( $\sigma_{TLE}^2 / \sigma_{e-TLE}^2$ )
	$\Phi_{D,coll}$	$\sigma_{coll}$ (%)	Error (%)	$\Phi_{D,TLE}$	$\sigma_{coll}$ (%)	Error (%)	$\Phi_{D,e-TLE}$	$\sigma_{coll}$ (%)	Error (%)		
<b>0.1</b>	3.87E-01	0.20	0.00	3.87E-01	0.12	0.03	3.87E-01	0.15	0.03	1.78	0.64
<b>0.2</b>	1.50E-01	0.33	0.07	1.50E-01	0.23	0.07	1.50E-01	0.24	0.00	1.89	0.92
<b>0.3</b>	5.79E-02	0.53	0.21	5.79E-02	0.39	0.36	5.80E-02	0.38	0.05	1.95	1.05
<b>0.4</b>	2.24E-02	0.83	0.40	2.24E-02	0.64	0.36	2.26E-02	0.57	0.27	2.12	1.26
<b>0.5</b>	8.77E-03	1.28	0.32	8.80E-03	1.02	0.70	8.71E-03	0.81	0.40	2.50	1.59
<b>0.6</b>	3.33E-03	1.99	1.94	3.41E-03	1.62	0.44	3.38E-03	1.14	0.38	3.05	2.02
<b>0.7</b>	1.32E-03	3.11	0.08	1.33E-03	2.57	0.98	1.32E-03	1.54	0.38	4.08	2.78
<b>0.8</b>	5.55E-04	4.62	8.00	5.34E-04	4.03	3.93	5.19E-04	1.98	0.97	5.44	4.14
<b>0.9</b>	2.04E-04	7.24	1.95	1.87E-04	6.72	6.40	1.97E-04	2.21	1.45	10.73	9.25

## 5.2 State of the art: current and past applications of the *e*-TLE

As previously noted in Chapter 1, the *e*-TLE was first proposed by Gelbard *et al.* [52] in 1966, and the first mention of the name “exponential track-length estimator” was by Williamson [97] in 1987. Already in Williamson’s publication, the main calculation bottleneck of this technique was identified by the author to be in the straight-line transport of the particle. In complex, heterogeneous geometries, the straight-line attenuation of the particle can require lots of computation time, decreasing the estimator’s efficiency. That is why the *e*-TLE was never implemented in any MC code until recent years, whereas its linear version (the TLE) is nowadays included in virtually every MC code.

In the field of medical imaging, geometries are very often voxelized, which can constitute a partial solution to the tracking problem, especially if an adapted ray-casting algorithm is used. In this framework, Freud *et al.* [98, 99] developed a hybrid methodology where a combination of deterministic and MC transport were joined to create something similar to the *e*-TLE. In this approach, the dose calculation is divided in two stages:

- The first one, purely deterministic, aims at calculating the dose due to direct (*i.e.* uncollided) radiation, by means of a purely deterministic computation based on a ray-tracing algorithm.
- In the second stage, a hybrid combination of Monte Carlo simulation and deterministic ray tracing gives an estimate of the dose due to scattered radiation. First, the Monte Carlo module tracks a group of photons and determines a set of interaction points, interaction types, and (in the case of scattering) directions of propagation. Then, each of these points is used as a secondary source for the ray-casting algorithm. Specifically, in the Freud *et al.* approach, this last deterministic part of the simulation can be viewed as a sort of “forced detection” approach, as the rays are automatically cast towards the voxel of the detector.

The proposed method, even if just in a very preliminary way, showed good results when compared to the classical MC approach with TLE. Starting from these preliminary developments, Smekens *et al.* [31, 100] slightly modified and improved the initial approach chosen by Freud *et al.*, notably in the last ray-casting part of the algorithm. In fact, several disadvantages arose from the forced detection scheme, such as aliasing artifacts or efficiency losses due to the direction of the rays being uncorrelated with the physics of the problem. The new method proposed assigns the generation of secondary photon to the MC module. After each interaction, a splitting routine is called, which generates  $N$  different particles with a weight of  $1/N$  each to avoid the introduction of a statistical bias. The particles directions are chosen based on the differential cross sections (DCS) of the respective interactions. Thus, the physics of the problem is preserved. Finally, these secondary particles are transported in a fully deterministic way, with continuous attenuation along their path. This technique, called **split-exponential track length estimator** (*se*-TLE), showed extremely good results both in its first test applications and in more realistic dosimetric frameworks, where it achieved an improvement in efficiency of roughly one order of magnitude with respect to MC calculations with classic TLE.

Moreover, it is interesting to note that by slightly changing the “ray-casting” part of the algorithm that manages the sampling of split particles’ directions, we can switch from a “global” estimator to a “local” one. In other words, the *e*-TLE (especially its version with splitting, the *se*-TLE), can be equally used to estimate quantities distributed on the whole simulation domain, as well as to conduct highly localized tallies on detectors.

A remarkably similar approach to the *se*-TLE was developed by Sweezy [72] for neutron transport with MonteRay, a library to be used with the Monte Carlo code MCATK [101]. In his work, he develops a next-event estimator called Volumetric Ray Casting (VRC), which works essentially in the same way as the one implemented by Smekens *et al.*. The main innovation in the approach by Sweezy, besides the application of the estimator to neutron transport, was the introduction of GPU calculations. In the VRC implementation, the random walk of the particle as well as the creation of deterministic rays is managed by the CPU. However, the rays are then completely processed by the GPU, which grants an additional acceleration to the process. This implementation showed to achieve good acceleration with respect to a classic MC + TLE simulation, with gains of around one order of magnitude.

This use of a *se*-TLE-like estimator to neutron flux calculations also opens the door to a possible interesting application in the dismantling field. As seen in the Introduction, a complete computational approach to a dismantling study involves several simulation steps including in-core neutron transport, out-of-the-core neutron transport, and photon transport. In a long-term perspective, an estimator like the *e*-TLE or *se*-TLE could be used in all of these instances to improve calculation performances.

These encouraging results ranging from photons to neutrons and from the medical field to the reactor field show the interest of developing a TRIPOLI-4® version of the *e*-TLE, as well as opening the way to future improvements and further developments.

### **5.3 *e*-TLE implementation in TRIPOLI-4®**

To implement any type of track-length estimator in a MC code, it is necessary to act on the transport part of the random walk. This is because, for any given TLE, the contribution to the score is not given by the particle interacting with the medium (through a scattering or an absorption), but rather by the particle traveling inside the detector volume. The *e*-TLE adds another specific challenge, as it operates not on the actual particle's path, but on a fictive straight-line path that extends, in principle, up to the end of the domain. The way this is implemented in our algorithm is through the creation of a fictive particle, which retains the original particle's direction of motion, energy, and simulation weight, and keeps traveling in the particle's direction until it reaches a boundary.

#### **5.3.1 Position in the transport chain**

The main steps of TRIPOLI-4®'s algorithm for the random walk of a photon are described below:

1. The photon is created, by sampling its initial properties (position, direction, and energy) from a given source distribution.
2. The length of the photon's free travel  $\ell$  is sampled, starting from initial data about the photon itself (position, direction, energy) and the medium (material composition and cross section).
3. The position of the photon is changed by adding  $\ell$  to its original position.
4. At this stage, any track-length estimator is updated with the information of the photon's free flight.
5. Based on the new position of the photon and on the isotopic composition of the material, a nucleus is sampled to undergo the interaction.
6. The interaction is sampled, based on material's cross section and the energies of both photon and interacting nucleus.
  - 6.a If the interaction is a photoelectric absorption, the photon is killed and its random walk is terminated.
  - 6.b If the interaction is a scattering, photon quantities are re-sampled (direction in the case of a Rayleigh scattering, direction and energy in the case of a Compton scattering).
7. We go back to point 2 and start again, repeating until the photon is killed or leaves the simulation domain.

The algorithm for the new estimator takes place immediately before the calculation of the photon's free flight, *i.e.* just before point 2 in the list above. In fact, at this stage the code has already sampled all the quantities needed for the *e*-TLE to take place: the photon's direction, energy, position, and statistical weight. The reader should keep in mind that in this first, basic implementation of the *e*-TLE, there is no splitting neither direction sampling to implement after the interaction.

### 5.3.2 Detailed implementation of the e-TLE

The implementation of the basic version of *e*-TLE is described below and summarized in Figure 5.25:

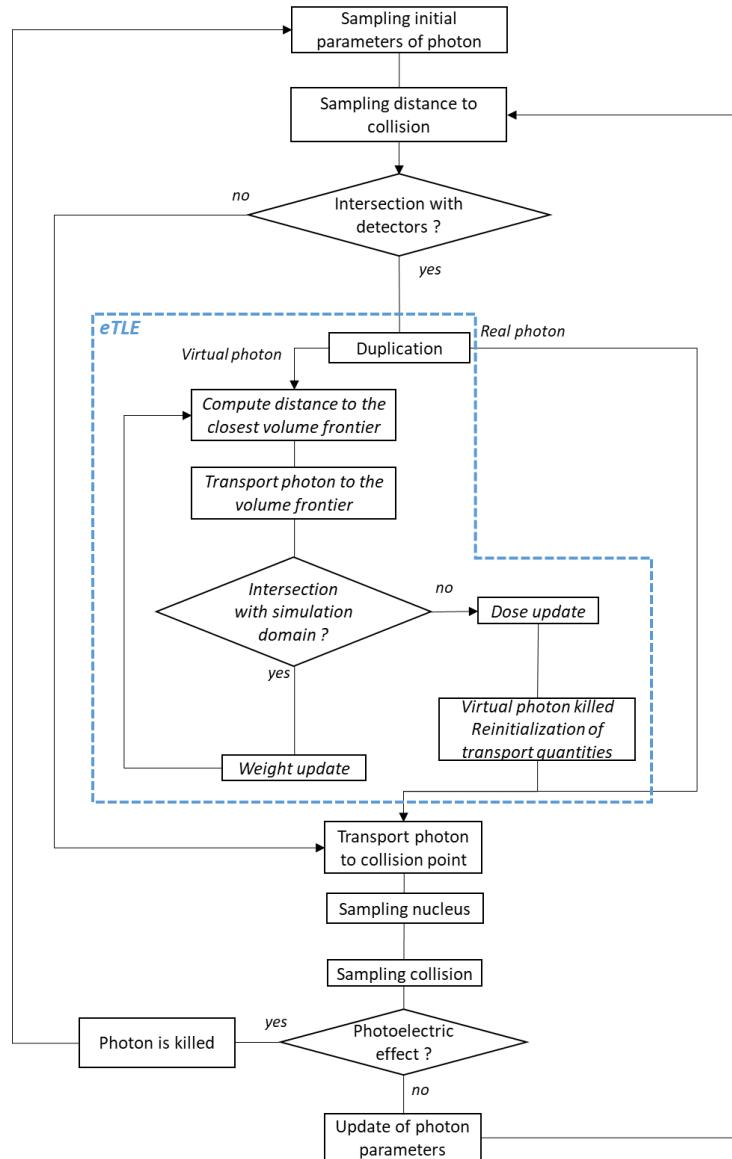
1. First, a check in the geometry takes place: the code checks if the straight-line path of the photon will intersect any of the volumes defined as detectors. If it does not, there is no point in transporting the photon, so the whole process is skipped.
2. Once we know that the particle is going to cross at least one detector, we start by duplicating it into a “virtual” photon, created as an exact double of the original, with the same properties.
3. We calculate the straight-line distance  $d_i$  between the virtual photon and the closest volume frontier, *i.e.* the distance travelled by the particle inside the  $i$ -th volume.
4. The virtual photon is displaced to its new position on the volume frontier; a value  $\kappa_i = \frac{w_i * (1 - e^{-\Sigma_t d_i})}{\Sigma_t}$  is stored in memory for the volume just crossed. Here,  $w_i$  is the weight of the virtual particle at the entrance of the  $i$ -th volume and  $\Sigma_t$  is the total macroscopic cross section.
5. The weight of the particle  $w_i$  is updated to take into account material attenuation:  $w_{i+1} = w_i \cdot e^{-\Sigma_t d_i}$ .
6. We go back to point 3 and repeat the process until the virtual photon reaches a domain boundary.

Once the virtual photon leaves the domain, the volumes marked as detectors contribute to the tally of the estimator. The tally is then weighted with the appropriate response function to produce the estimate for the quantity that is being computed. For example, if the required quantity is the KERMA  $K_i$ , the result is (see also Equation (1.23)):

$$K_i = \frac{\kappa_i}{V} E_i \frac{\mu_{en}}{\rho} = w_i E_i \frac{\mu_{en}}{\rho} \frac{(1 - e^{-\Sigma_t d_i})}{V \Sigma_t} = w_0 E_i \frac{\mu_{en}}{\rho} e^{-\Sigma_t s_V} \frac{(1 - e^{-\Sigma_t d_i})}{V \Sigma_t} \quad (5.30)$$

where  $E_i$  is the photon’s energy,  $\mu_{en}/\rho$  is the mass energy absorption coefficient of the material,  $V$  is the detector’s volume,  $w_0$  is the particle’s initial weight and  $s_V = d_0 + d_1 + \dots + d_{i-1}$  is the total distance the particle has to travel to get to the  $i$ -th volume frontier.

Finally, the virtual particle is deleted and all transport quantities are reinitialized, before going back to the transport of the “real” photon.



**Figure 5.25: Flowchart of *e*-TLE implementation for photon transport in TRIPOLI-4®'s transport chain**

## 5.4 Validation and performance analysis

A first set of tests aimed at assessing the unbiasedness and the efficiency of the new *e*-TLE is conducted in this section, starting from the results of the VRT study presented in Chapter 4. The configurations considered for the study are the Odano backscattering and slant penetration, as well as the simple and complex bunker configurations (see Chapter 3).

For each of the four configurations, several simulations are performed:

- in semi-analog mode, with implicit variance reduction options only;
- with the exponential transform method used with a user-generated importance map;
- with the AMS method with the same manually-adjusted importance map.

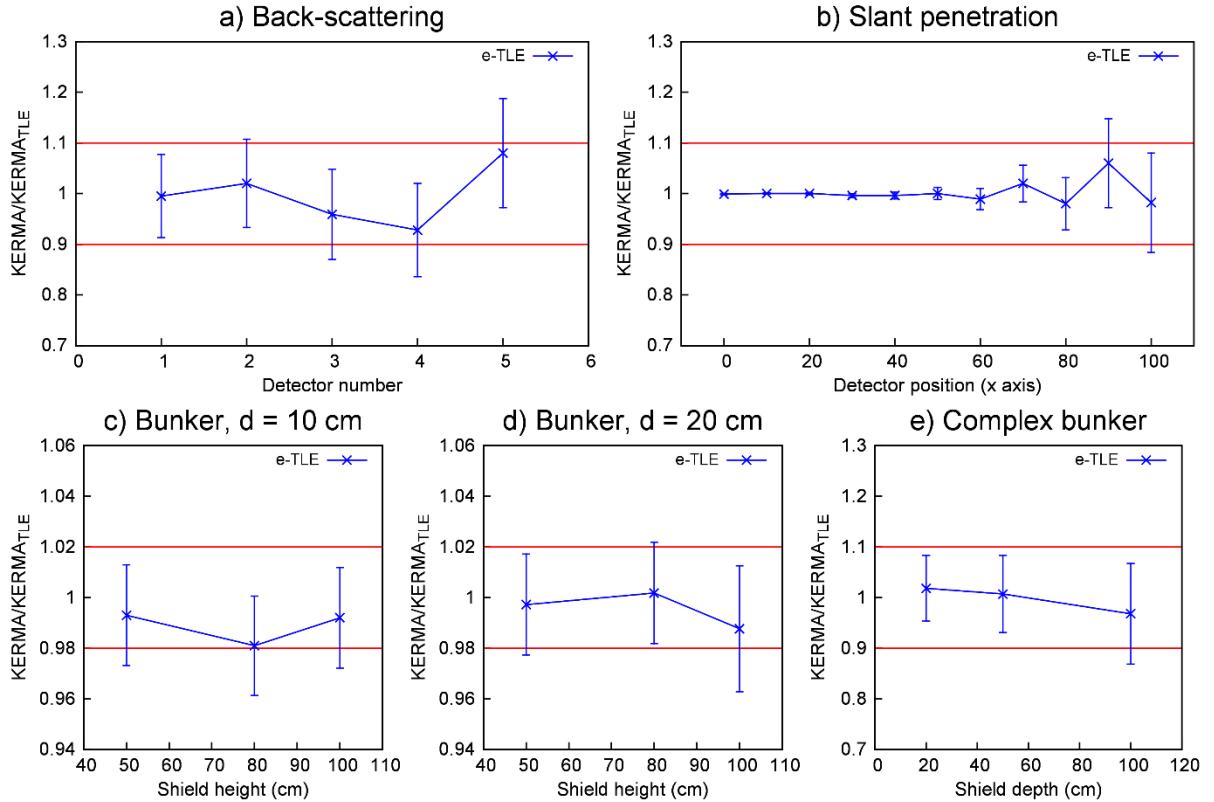
In each case, two estimators are tested and compared: the classical TRIPOLI-4® TLE and the newly implemented *e*-TLE. In all simulations,  $^{60}\text{Co}$  is used as a photon source, and the calculated quantity is the KERMA. A figure of merit (FOM) defined as:

$$FOM_{det} = \frac{1}{T \cdot \sigma_{det}^2} \quad (5.31)$$

where  $T$  is the total simulation time (including any importance generation time, which was always negligible in the cases under consideration) and  $\sigma_{det}^2$  is the estimate of the variance of the detector response (in %), is calculated for each detector, to be able to compare the different simulation performances. For the back-scattering and slant penetration simulation, an average target variance of 5% over all the detectors is set as limit; for the simple bunker, the average target variance is set at 1%. Batches of 10000 particles are used for all simulations except for the AMS ones, where the batches size is 1000 particles.

#### 5.4.1 Analog simulation results

Results for the validation of *e*-TLE in the analog simulations are presented in Figure 5.26. The plots show, for every detector, the ratio between the KERMA calculated with *e*-TLE and the one obtained with TRIPOLI-4®'s TLE. The red lines mark the error margin of  $\pm 2\sigma$ . We note a perfect accordance between results, thus validating results obtained with the *e*-TLE.

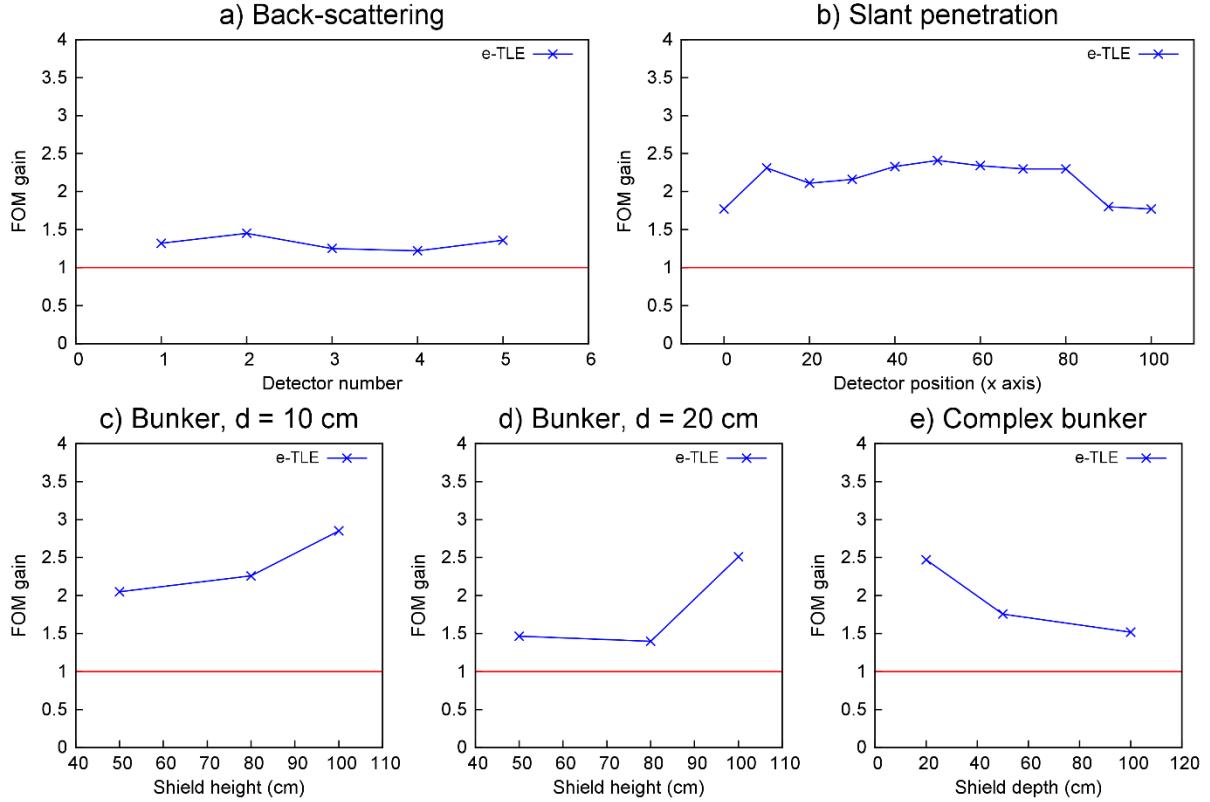


**Figure 5.26:** KERMA comparison: a) Odano backscattering configuration; b) Odano slant penetration configuration; c) simple bunker configuration, slim wall; d) simple bunker configuration, thick wall; e) complex bunker configuration. Error bars of  $\pm 2\sigma$  are shown.

To assess the performance of the new estimator, we compare the FOM of the simulations with the *e*-TLE to that obtained with the TLE. Therefore, a ratio higher than one means that the new estimator performs better than the

TLE, by reducing total computation time – since the standard deviation is fixed in all these cases. Results of this comparison are shown in Figure 5.27.

As we can see by the plots, the ratio is always greater than 1, meaning the new estimator is able to outperform the normal track-length estimator in all simulations and for all detectors. The gain in performance ranges from around 1.2 in the back-scattering problem to over 2.8 in the R3 bunker configuration ( $h = 100 \text{ cm}$ ,  $d = 10\text{cm}$ ).

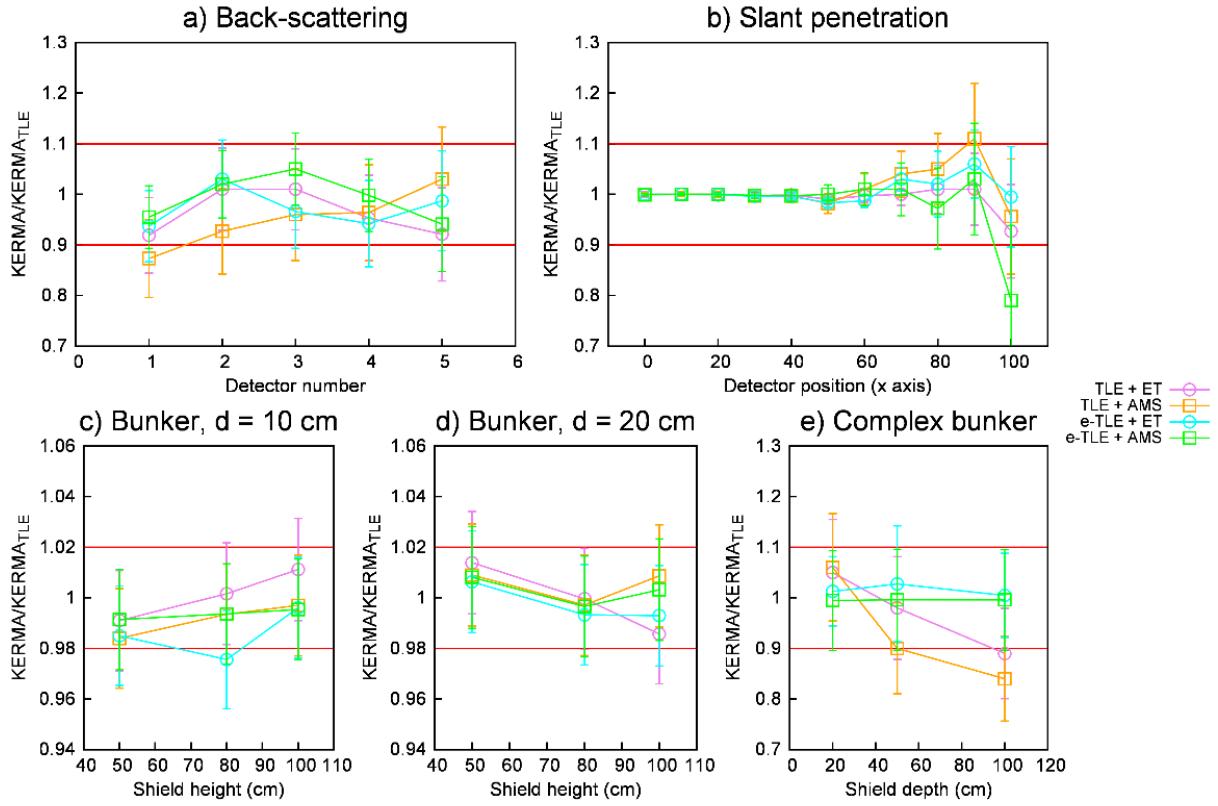


**Figure 5.27: FOM comparison:** a) Odano backscattering configuration; b) Odano slant penetration configuration; c) simple bunker configuration, slim wall; d) simple bunker configuration, thick wall; e) complex bunker configuration. Results above the red line mark an acceleration of the simulation with respect to analog TLE.

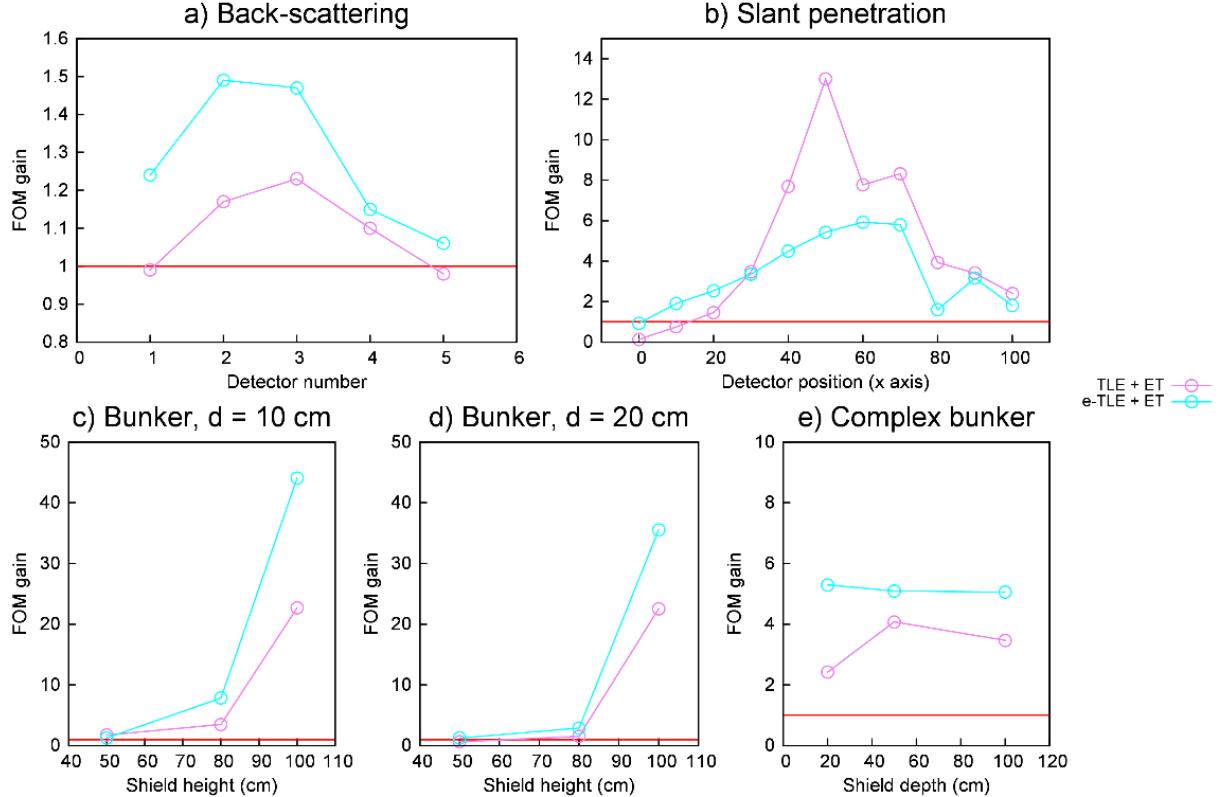
#### 5.4.2 VRT-accelerated simulation results

Figure 5.28 shows the validation of the VRT-accelerated simulations, with both TLE and *e*-TLE. The plots show, for every detector, the ratio between the computed KERMA and the one obtained with TRIPOLI-4®’s TLE in analog mode. The red lines mark the error margin of  $\pm 2\sigma$ . We note a perfect accordance between results, thus validating all results obtained.

Since the *e*-TLE showed good performances in analog simulations (see Figure 5.27), we now want to test its capabilities when coupled with variance reduction techniques. Again, we compared the FOM of all simulations to the FOM of the analog simulation with classic track-length estimator. Figure 5.29 shows results obtained with INIPOND, with and without the new estimator. For the simple bunker, complex bunker and backscattering configurations, we can see that the new estimator is consistently able to improve the gain in performance obtained by the exponential transform. Simulations performed with *e*-TLE show a performance gain ranging from a little over 1 to more than 2 for the best case (R3 bunker configuration). This leads to critical improvements with respect to the analog simulation, with a maximum performance gain of more than 40.

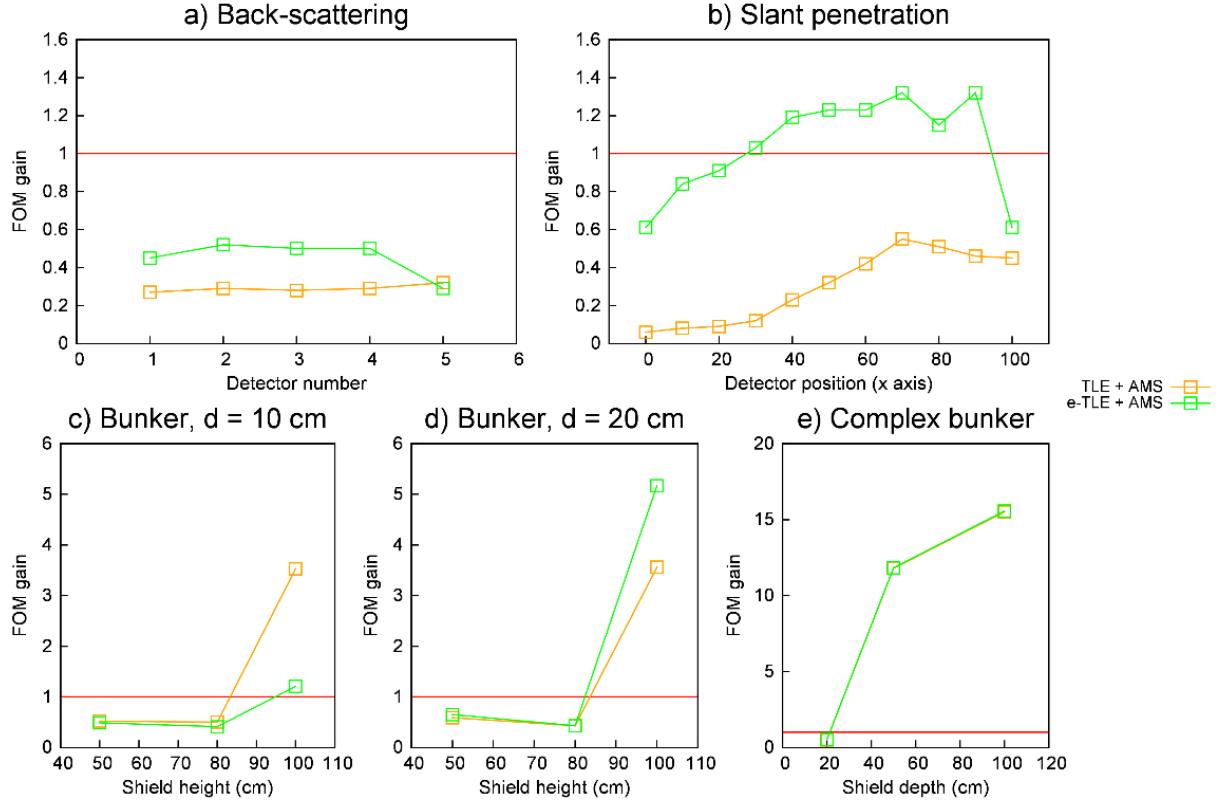


**Figure 5.28: KERMA comparison:** a) Odano backscattering configuration; b) Odano slant penetration configuration; c) Simple bunker configuration, slim wall; d) Simple bunker configuration, thick wall; e) Complex bunker configuration. All results are compared to the relative TRIPOLI-4® analog simulation with TLE. Error bars of  $\pm 2\sigma$  are shown.



**Figure 5.29: FOM comparison for INIPOND:** a) Odano backscattering configuration; b) Odano slant penetration configuration; c) simple bunker configuration, slim wall; d) simple bunker configuration, thick wall; e) complex bunker configuration. Results above the red line mark an acceleration of the simulation with respect to analog TLE.

For the Odano slant penetration benchmark, we see a different behavior. Except for the first three detectors, the simple track-length estimator outperforms the new estimator by a factor of more than 2 in the best case. Figure 5.30 shows results obtained with AMS, with and without the new estimator.



**Figure 5.30: FOM comparison for AMS:** a) Odano backscattering configuration; b) Odano slant penetration configuration; c) simple bunker configuration, slim wall; d) simple bunker configuration, thick wall; e) complex bunker configuration. Results above the red line mark an acceleration of the simulation with respect to analog TLE.

Notably, for both the Odano benchmarks and the simple bunker configuration, many points fall below the red line marking a gain of 1, the value that denotes a performance equal to that of the analog simulation. This means that the AMS, in these cases, is not able to accelerate the simulations. This remains mostly true if we apply the new estimator, although the performances are often improved, as shown by the two Odano cases and by the thick-wall simple bunker configurations. This behaviour could be related to the fact that, for very simple geometries like the Odano cases, the acceleration introduced by the AMS algorithm is not enough to outweigh the additional computational burden of the algorithm itself. Moreover, the good results (FOM gain up to more than 15) obtained in the complex bunker configuration seem to corroborate this hypothesis. The complex bunker simulations also seem to suggest that, when the AMS is well-adapted to the problem under consideration, the *e*-TLE fails to introduce any relevant acceleration with respect to the TLE.

## 5.5 Conclusions

We can conclude that the newly implemented *e*-TLE does not introduce any systemic bias in the simulations, and it is therefore validated. Moreover, the estimator shows an improvement in code performance when used in situations where the classic TLE would be chosen, notably to compute a tally in a low-density medium such as air. In the best cases, the new estimator has been shown to reduce computation time by more than a half. Finally, these

first simulations suggest that the exponential track-length estimator can be coupled with importance-based variance reduction techniques to furtherly improve code performance.

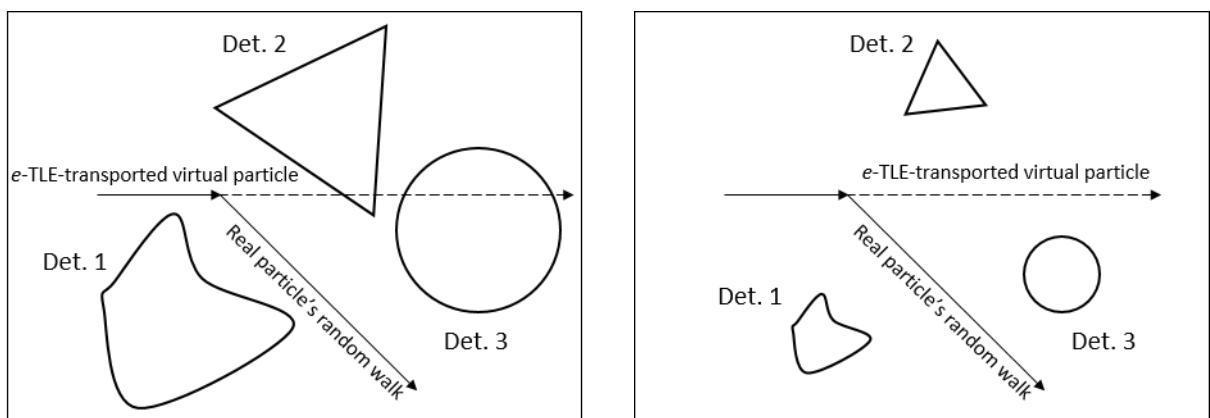
These first results show that the *e*-TLE, even in its most basic implementation, is already able to consistently improve the simulation efficiency in a non-voxelized geometry; this proves the interest of developing more refined versions of the estimator. The next chapters are devoted to present the additional improvements included in the *e*-TLE in the framework of this thesis. Taking inspiration from [99], the capabilities of the exponential track-length estimator as a local estimator are enhanced by introducing a forced detection algorithm, to ensure that each particle transported by the method contributes to the detector tally. Furthermore, the use of *e*-TLE as a global estimator, to provide dose rate maps in the whole problem geometry instead that only in specific detector areas, is also investigated. To this aim, a splitting algorithm is introduced along with the possibility of using the *e*-TLE for estimation on meshes, in an implementation similar to [100] and [72].



# Chapter 6

## The *e*-TLE as a “local” estimator

We have seen how the implementation of the basic *e*-TLE in TRIPOLI-4® aimed to extend the free flight of each particle, to increase the quantity of scores related to the particle history. However, this method is intuitively much less efficient as the number of detectors, or volumes where the tallies are requested, decreases. More generally, the efficiency decreases as the ratio between total volume of the detectors and total volume of the whole geometry decreases. In other words, if the detectors are small and/or localized in a particular zone, the probability of crossing them with a given straight line decreases, which in turn means that the *e*-TLE will lose more time transporting virtual particles that do not cross any detector, as illustrated in Figure 6.31.



**Figure 6.31: The basic implementation of the *e*-TLE fails to increase the number of valid tallies when detectors are small and localized (shown on the right)**

Since the problem of searching for tallies over small, localized volumes is sufficiently common in the nuclear field (in-core detectors, experimental configurations with TLDs), the interest of improving this aspect of the estimator is straightforward. To this aim, a “local” version of the *e*-TLE has been developed, through a forced detection (FD) algorithm able to send the virtual particle towards the detector after every interaction, in a similar fashion to that

described in [98] for a voxelized geometry. This chapter presents the implementation of this local version of the *e*-TLE, called *e*-TLE-FD, and the validation and performance studies realized to test it.

## 6.1 *e*-TLE with forced detection for local reduction of variance

We recall (see Equation (1.23)) that, in its basic implementation, the score  $\kappa_j^{eTLE}$  registered by the *e*-TLE on a volume  $V$  that is crossed by a virtual particle after interaction  $j$  is:

$$\kappa_j^{eTLE} = w_j e^{-\Sigma_t s_V} \frac{(1 - e^{-\Sigma_t L_j})}{\Sigma_t V} \quad (6.1)$$

where  $w_j$  is the weight of the real particle at the moment of the interaction,  $\Sigma_t$  is the total macroscopic cross section of the material (supposed to be homogeneous for simplicity),  $s_V$  is the distance to the detector and  $L_j$  is the length of the intersection between the virtual particle's straight-line path and  $V$ . In this version, we can say that the *e*-TLE is a flight-based estimator, *i.e.* it contributes to the tally when the random particle performs a flight.

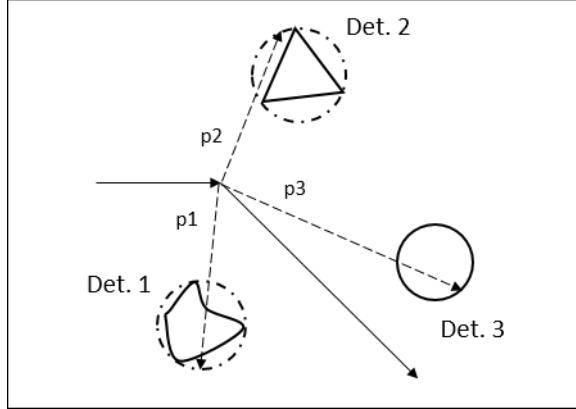
We can develop on this formulation by turning the *e*-TLE into a collision-based estimator: in this way, after each free flight we can sample virtual collisions directed towards the detection volume and weight them by their probability of occurrence. This ensures that every particle will always contribute to the score at every collision. In practice, if we assume that the detection volume  $V$  is a sphere, we sample a uniformly random direction  $\Omega$  under the constraint that the half line from the collision point  $r_j$  directed along  $\Omega$  intersects the sphere. We can then write the score for the *e*-TLE-FD as:

$$\kappa_j^{eTLE,FD} = \kappa_j^{eTLE} p(r_j, \Omega) \Delta\Omega = w_j e^{-\Sigma_t s_V} \frac{(1 - e^{-\Sigma_t L_j})}{\Sigma_t V} p(r_j, \Omega) \Delta\Omega \quad (6.2)$$

where  $p(r_j, \Omega)$  represents the probability density, per unit solid angle, of scattering towards direction  $\Omega$  after an interaction in  $r_j$  and  $\Delta\Omega$  is the solid angle subtended by the scoring volume  $V$  seen from the interaction point. If the detection volume is not spherical, the solid angle can be hard to calculate; however, the problem is easily solved by defining a bounding sphere surrounding the volume and then scoring only the particles that effectively cross the volume (Figure 6.32). In this case, the solid angle of the bounding sphere can be easily calculated by knowing the sphere radius  $R$  and the distance  $D$  between the interaction point and the center of the sphere:

$$\Delta\Omega = 2\pi \left(1 - \frac{\sqrt{D^2 - R^2}}{D}\right) \quad (6.3)$$

Moreover, we can now choose to re-sample the interaction multiple times and, in the case of more than one detection volumes, send a different virtual particle to each one of them. A schematic depiction of the procedure can be found in Figure 6.32. The result is a powerful estimator that, in principle, ensures a score contribution for every particle, at every interaction, in every detection volume. Depending on the shape of the volume it can happen, as shown in Figure 6.32 for the triangular detector, that a virtual particle does not produce a contribution. Therefore, the efficiency of the estimator is maximum when detectors coincide with the bounding spheres, as for detector 3 in Figure 6.32.



**Figure 6.32: *e*-TLE with FD.** The virtual particles  $p_1$ ,  $p_2$  and  $p_3$  are sent to the bounding spheres and have a much higher probability of crossing the detectors

This process of sending a particle towards a detector to improve the simulation’s efficiency shares some important similarities with some other algorithms employed in particle transport; notably, the next-event point estimator [102] on one side, and algorithms like DXTRAN [76] or the one developed by Tickner [103] on the other. However, the *e*-TLE-FD is fundamentally different from these approaches.

The next-event point estimator, also present in TRIPOLI-4® with the name FLUX\_PT, is a well known estimator in the field of MC simulations. As the name implies, it allows to estimate quantities in a point; it also involves deterministic transport and exponential attenuation of the particle’s weight. In practice, it can be viewed as a limit version of the *e*-TLE-FD, where the target detector becomes infinitely small, converging to a single point.

As explained by Cramer in [102], this change from volume to point leads to the appearance, in the estimator, of a  $1/R^2$  term, where  $R$  is the distance between the point of collision and the point of detection. This term causes the point estimator’s variance to drastically diverge whenever a collision takes place very close to the point of detection, and it is the reason why every MC implementation of next-event point estimation usually includes an “exclusion sphere” inside of which any contribution to the estimator is not taken into account. Therefore, the main difference between the *e*-TLE-FD and a point estimator is that the former does not require any exclusion sphere.

As for DXTRAN [76] and DXTRAN-like transport algorithms [103], the difference is more related to the mechanics of the particle. In these algorithms, the particle is split at every interaction, and deterministically transported to the surface of some element: a sphere in the case of DXTRAN, any convex volume in Tickner’s implementation. Then, the transported particle is left free to continue its simulation, until it is absorbed or leaves the simulation domain. Therefore, the transported particle can, in principle, interact many times inside the volume of interest before ending its simulation history. This in turn means that the “mother” particle has to be killed if it tries to enter the volume of interest, as its interactions have already been simulated by the split particle.

The fundamental difference between this approach and the *e*-TLE-FD is that, in the latter, there is no actual particle being transported, but rather a “virtual” particle representing a single contribution of the mother particle to the score. This “virtual” particle gives one and only one contribution, and then it is killed, as will be better detailed in Section 6.2. Moreover, the “real” particle only serves as a mean to simulate collision points, but it never registers any score: all the contributions to the *e*-TLE-FD come from the “virtual” particles. This is also why, in our method, there is no need to kill the “real” particle if it tries to enter the detection volume.

Keeping in mind these important clarifications, we can now give a more in-depth look at the code implementation of the *e*-TLE-FD.

## 6.2 Code implementation

As shown in Figure 6.33, the new part of the implementation adds some calculation steps taking part immediately after the creation of the virtual photon. The difference with respect to the flowchart described in Chapter 5 is that the direction followed by the virtual particle is now sampled from the scattering probability distribution of the interaction. This also implies an additional weight and energy adjustment to preserve the unbiasedness of the simulation. The resampling of the direction is also what allows to repeat the process when multiple detectors are present.

We can schematize the algorithm of the new *e*-TLE-FD as follows:

1. First, before launching the simulation, the user associates a bounding sphere to each detection volume. The sphere must fully enclose the detection volume while still being as small as possible for maximum efficiency.
2. If the simulation has more than one detector, we start a loop over all of them.
3. We create a “virtual” particle as an exact copy of the original particle.
4. Based on the bounding radius and the distance between the interaction point and the sphere center, we calculate the solid angle subtended by the sphere,  $\Delta\Omega$ . Then, we sample a random direction uniformly distributed within  $\Delta\Omega$ .
5. The code samples a “virtual” collision for the fictive particle, adjusting its weight (for implicit capture and for the scattering probability distribution) and assigning the direction determined at the previous step, effectively directing the particle towards the bounding sphere. The energy of the fictive particle is sampled by taking into account the differential cross-section of the collision, according to the direction randomly selected at point 4. This ensures that the physics of the simulation remains unbiased.
6. At this point, the direction of flight is determined and the process of displacing the particle and scoring the tallies is equal to the one described in Section 5.3.2, specifically points 3 to 6.
7. If the particle has crossed the detector volume, we register the score calculated by *e*-TLE during the virtual particle’s displacement. Then, we kill the particle and go back to point 2, selecting the next detector in the list.
8. Finally, the virtual particle is deleted and all transport quantities are reinitialized, before going back to the transport of the “real” particle.

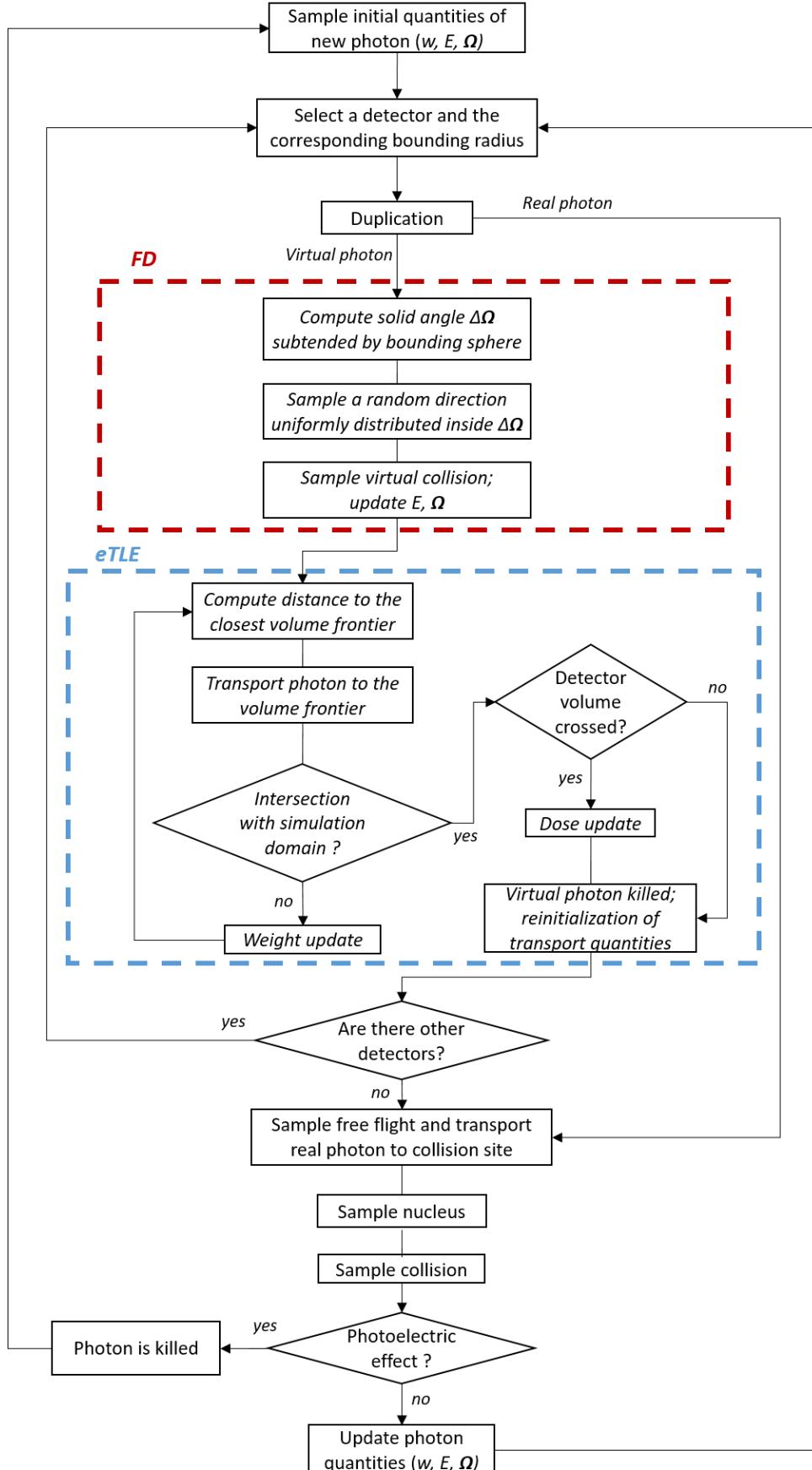


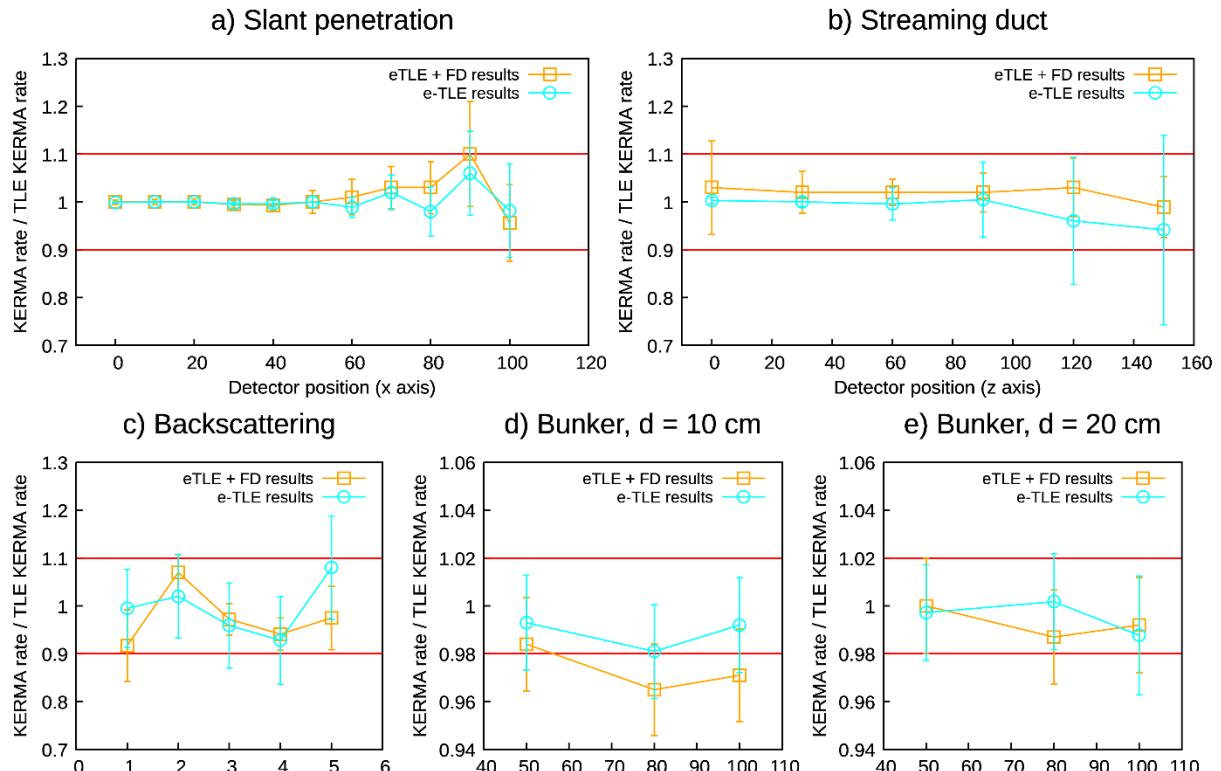
Figure 6.33: Flowchart of the *e*-TLE algorithm with FD;  $w$  is the particle's weight,  $E$  is its energy and  $\Omega$  is its direction

### 6.3 Validation and performance analysis

Once again, the precision and performance of the new estimator are tested on the configurations presented in Chapter 3. The configurations chosen are the three Odano benchmarks and the simple bunker configurations. These configurations, especially the Odano benchmarks, are particularly well suited for testing the *e*-TLE-FD algorithm, as they present a set of very small, spherical detectors in a low-density medium. Results of this study are also published in [104].

For each configuration, three simulations are performed: one with TRIPOLI-4®'s TLE, one with the *e*-TLE without FD (cfr. Chapter 5), and one with the *e*-TLE-FD. This allows us to not only assess the FOM gain of the *e*-TLE-FD with respect to the TLE, but also quantify the acceleration introduced by the FD algorithm, by comparing the *e*-TLE-FD with the *e*-TLE. Similarly to the study presented in Chapter 5, a FOM is associated to each detector (see Equation (5.28)) and then a FOM gain is defined as the ratio between the FOM obtained with the exponential estimator and the one obtained with the TLE. Batches of 10000 particles are used with a target variance of 5% for the Odano benchmarks and 1% for the bunker.

Validation results are presented in Figure 6.34. This step serves to ensure that the new estimator converges to the same results as the classic TLE already implemented in TRIPOLI-4®. The KERMA rate computed by *e*-TLE with and without the FD algorithm is compared with the KERMA rate obtained with the TLE. Figure 6.34 shows the ratio between the two estimates, with tolerance margins of  $\pm 2\sigma$  highlighted on the plots. Each point on the plots shows the dose rate of a single detector, with the exception of the streaming duct case, where results for each triplet of detectors occupying the same axial position were averaged to improve legibility. In general, we note a very good accordance with the TLE results: all results fall inside the error margin, if we consider their relative uncertainty.

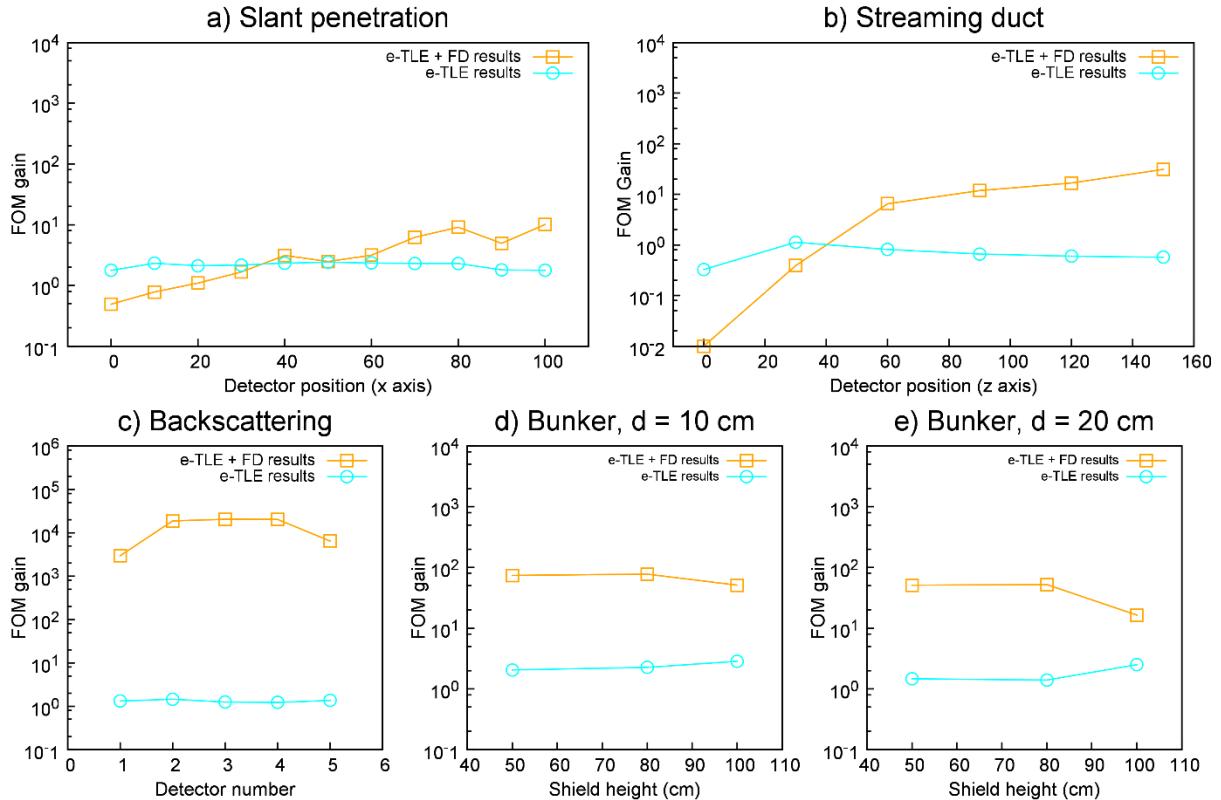


**Figure 6.34: Validation of the estimator over all tested configurations (for details on the geometries, see Chapter 3). The y-axis shows the ratio between the equivalent dose rate calculated with the *e*-TLE estimator (with and without forced detection) and the one calculated with TRIPOLI-4®'s TLE.**

The new *e*-TLE-FD is therefore validated and it shows no systematic bias with respect to either the classic TLE or the *e*-TLE.

To assess the performance of the new estimator, we compare in Figure 6.35 the FOM gain of the simulations with the *e*-TLE-FD to the FOM gain of the simulations performed with the basic *e*-TLE. As before, the FOM gain is defined as the ratio between the tested estimators and the classic TLE without any VRT.

In the back-scattering configuration (Figure 6.35c) the *e*-TLE-FD attains outstanding performances, with acceleration factors up to  $2 \times 10^4$ . This is an enormous improvement with respect to the basic *e*-TLE, which has acceleration ratios around 1.2 for this configuration. In the bunker configuration (Figure 6.35d and Figure 6.35e), all simulations are accelerated by the *e*-TLE with and without FD. Specifically, with the FD algorithm we reach factors of almost 100, while the basic *e*-TLE leads to a gain of 2 in the best case. Results are more varied in the slant penetration (Figure 6.35a) and streaming configurations (Figure 6.35b). In the slant penetration setup, we see that the *e*-TLE without FD is able to uniformly accelerate the results over all the detectors, with acceleration ratios of about 2. With the FD algorithm, the estimator shows increasingly good performances as we move further from the source, with acceleration ratios ranging from lower than 1 (which indicates a drop in performance with respect to the TLE) to around 10.

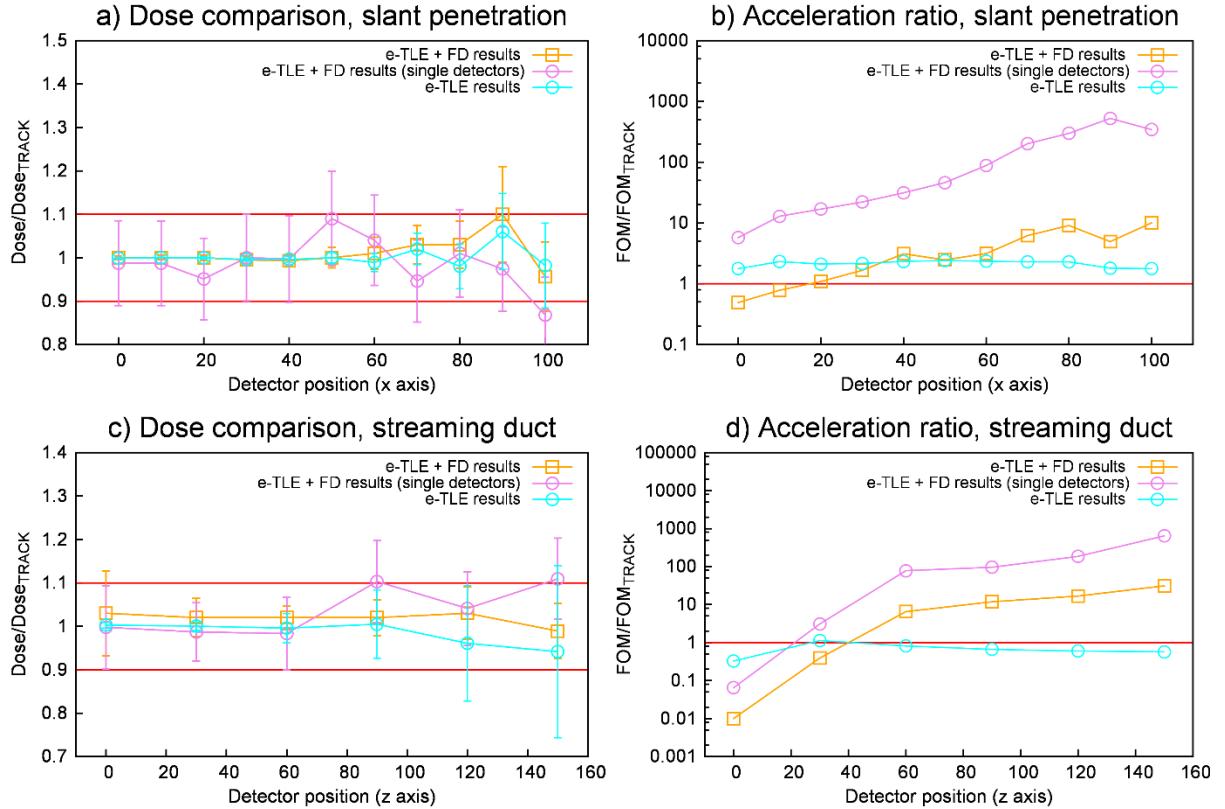


**Figure 6.35: Performance of the estimator over all tested configurations (for details on the geometries, see Chapter 3). The y axis shows the ratio between the figure of merit of the *e*-TLE estimator (with and without forced detection) and the figure of merit of TRIPOLI-4®'s TLE.**

We can find the same behavior in the streaming configuration, with the estimator failing to accelerate the convergence over the first two detectors and progressively increasing its performances as we move to the farthest detectors, with ratios close to 30. This is also the only configuration where the *e*-TLE without FD fails to accelerate the convergence. An explanation for this behavior might be found in the fact that having to simulate and transport virtual particles adds a certain amount of overhead to the transport algorithm. To generate a single particle history,

the *e*-TLE-FD requires the largest amount of time (which increases with the total number of detectors), followed by the *e*-TLE and then by the TLE . This of course means that, given a certain amount of computation time, the simulation with the TLE will generate a larger amount of particle histories than the one using the *e*-TLE. Therefore, if a detector is close to the source and easily reached by the particles, convergence with the TLE will be quicker. As we mentioned before, in the case of multiple detectors, the current implementation of the FD algorithm sends one virtual particle to each of the detectors before restarting the original particle transport. Intuitively, this implies that the algorithm's efficiency is inversely proportional to the number of detectors in the simulation.

Therefore, to try and improve the estimator's performance, we run an additional set of simulations on both the slant penetration and the streaming duct configurations. In this case, only one detector is active, which means that a total of 22 simulations are run in the slant penetration case (11 with the TLE , 11 with the *e*-TLE-FD) and 12 simulations were run in the streaming case (6 with the TLE , 6 with the *e*-TLE-FD). Results of this “single detector” simulations are reported in Figure 6.36. As we can see from the plots, the estimator performance is drastically improved when used with only one detector, with the efficiency consistently gaining one order of magnitude with respect to the multiple-detector case. We can still see the same behavior related to the detector's distance from the source: for the farthest detectors, we register acceleration ratios of around 500 for the slant penetration case and more than 600 for the streaming duct.



**Figure 6.36: Validation and performance of the *e*-TLE-FD in single-detector simulations.**

## 6.4 Conclusions

The results of these tests are very encouraging. First of all, we can say that the FD algorithm is well implemented in that it does not introduce any systemic bias in the results obtained with the *e*-TLE-FD.

Secondly, the *e*-TLE-FD shows excellent accelerating capabilities when dealing with photon transport both in void and scattering-intensive environments, with acceleration factors reaching up to  $10^4$  in the simplest cases. The *e*-TLE-FD proves to be a valid alternative to the TLE, even when dealing with geometries of medium complexity as the streaming one. It can efficiently tackle both reflecting-dominated problems and attenuation-dominated ones, suggesting possible applications as an alternative to deterministic codes for geometries that are difficult to treat with the PKI method.

In conclusion, the “local” implementation of the *e*-TLE is effectively able to optimize performances of MC simulations with small detectors and localized tallies. These results are encouraging also for the development of a “global” option, which could be complementary to the *e*-TLE-FD in situations where a score distributed over the whole domain is required.



# Chapter 7

## The *e*-TLE as a “global” estimator

The problem of calculating global quantities distributions throughout a whole simulation domain is often found in nuclear particle transport problems. For neutrons, some examples include core flux calculations, for reactor design studies, and neutron fluence distribution in structures of interest, for activation calculations. For photons, dose rate maps are required for D&D studies, and other applications can be found in the medical and imaging field. For these kind of problems, the TLE is usually the estimator of choice. In this chapter, we consider the option of using the *e*-TLE for such calculations; more specifically, by implementing a new splitting algorithm and introducing the possibility of using the estimator on meshes rather than volumes, we present a new version of the estimator optimized for global fluence calculations. This “global *e*-TLE” or *se*-TLE is inspired by the *split-exponential track length estimator* by Smekens *et al.* [100] as well as by the *volumetric ray casting estimator* (VRC) developed by Sweezy [72].

### 7.1 *e*-TLE for global reduction of variance

It is easy to see how the *e*-TLE would be well suited for whole-domain estimations, even by looking at its most basic implementation. By extending every particle’s trajectory over the whole geometry, the estimator provides a score over an area much more extended than the collision estimator or the normal TLE would provide. Of course, the increase in computation time relative to the virtual particle’s displacement is always present, and lowers the efficiency of the method.

In an effort to improve both efficiency and usability of the estimator in dealing with global estimations, two modifications were added to the *e*-TLE algorithm: a splitting routine, to increase the statistics recovered from a single particle collision, and the possibility of using the estimator on meshes superimposed on the simulation’s geometry.

The concept of the splitting algorithm is fairly straightforward in principle: after a collision of the “real” particle,  $M$  virtual particles are created instead of only one, each having a weight equal to  $w_0/M$ , where  $w_0$  is the weight of

the real particle. Their energies and directions are sampled according to the scattering cross section of the collision. These particles are then transported in a straight line according to the *e*-TLE routine.

This splitting process may seem similar to the one presented in Chapter 6 with forced detection; however, a big difference lies in how the virtual particles' directions and energies are sampled. In fact, for the global *e*-TLE, the direction of every virtual particle has to be sampled from the same probability distribution, related to the DCS of the collision undergone by the real particle. On the other hand, for the FD splitting the virtual particles' directions are pre-determined, then the weight is adjusted by taking into account the probability of undergoing a certain collision and exiting with that pre-determined direction. In practice, this means that, when sampling virtual particles for the global *e*-TLE, information related to the type of collision undergone by the real particle (Compton or Rayleigh scattering in our case) is needed. This may seem like a small difference, but it leads to a very different implementation, due also to how collisions and transport are coded in TRIPOLI-4®.

The other modification introduced for the global *e*-TLE was the possibility of using the estimator on meshes. In TRIPOLI-4®, a volumetric estimator can be defined either on volumes or on meshes. The estimation on volumes was chosen at the beginning of the thesis, as it is easier to implement: each volume, by definition, contains only one material, therefore cross sections are uniform inside volumes. Moreover, volumes are embedded in the definition of TRIPOLI-4®'s geometry, meaning that some functions that would become useful for the *e*-TLE already existed in the code.

Meshes, on the other hand, are superimposed on the geometry and are only used for calculating scores. They are particularly useful to calculate distributions of quantities over large portions of the simulation domain. Therefore, a mesh-oriented implementation is paramount to really exploit the capabilities of the global *se*-TLE.

## 7.2 Splitting multiplicity study

Any VRT involving some sort of splitting introduces the parameter of splitting multiplicity, which can have a strong impact on the global efficiency of the simulation. Multiplicity is the amount of particles created at each splitting instance, in this case at each collision. In the TRIPOLI-4® implementation of *se*-TLE, the choice of the splitting multiplicity is left to the user. Intuitively, one can say that the more split particles are generated, the more statistics are recovered from each simulated “mother” photon; on the other hand, more time is consumed to track and transport the split particles. Therefore, we can expect the splitting multiplicity  $M$  to have an optimum value, after which the simulation efficiency decreases.

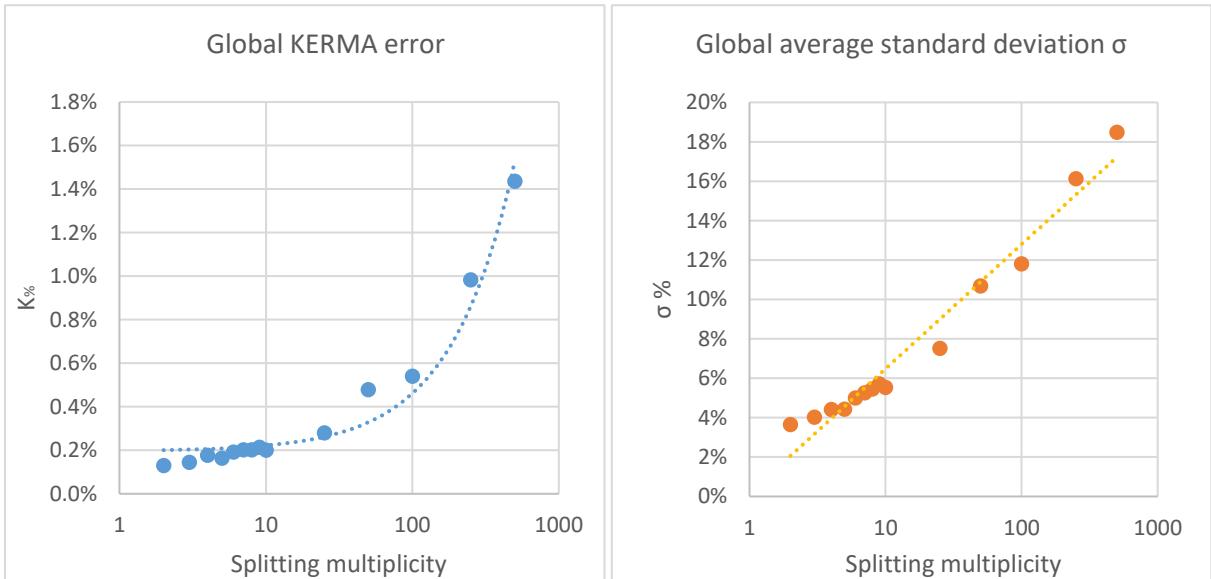
To verify this and to find the optimal value for  $M$ , a set of *se*-TLE simulations are performed by gradually varying the multiplicity. The procedure chosen is the same used by Smekens *et al.* in [31]. The configuration chosen for the study is the Bunker configuration R3, with a 10 cm lead wall extending from floor to ceiling. A cubic mesh of 14x14x14 cells is superimposed to the geometry for scoring.

A high-precision reference simulation is performed with TRIPOLI-4®'s TLE with an extremely high number of histories (ten million batches of 1000 particles, totalling  $10^{10}$  particle histories). Then, a total of 14 *se*-TLE simulations are performed, with values of  $M$  ranging from 2 to 500. Simulations are stopped at time  $T=13900\text{ s}$  to conduct a constant time analysis. Then, to assess the global efficiency of each simulation, we define a *global KERMA error*  $K\%$  as:

$$K\% = \frac{\sum_{i=1}^{N_{mesh}} |K^i - K_{ref}^i|}{\sum_{i=1}^{N_{mesh}} K_{ref}^i} \quad (7.1)$$

where  $N_{mesh}$  is the total number of mesh cells, and  $K^i$  and  $K_{ref}^i$  are the KERMA in the  $i$ -th mesh cell, calculated respectively in the *se*-TLE simulation and in the reference simulation.

Results are shown in Figure 7.37. In this analysis, the correlation between the global KERMA error  $K\%$  and the splitting multiplicity  $M$  seems to follow a linear trend: therefore, we cannot identify an optimal value for  $M$  which would correspond to a minimum error. The fact that  $K\%$  increases does not reflect a bias in the estimation, but rather is related to the fact that for high splitting multiplicity the convergence is slower. Therefore, since the analysis is conducted at constant time, for high splitting multiplicities the final average standard deviation  $\sigma$  increases, as shown by the plot in Figure 7.37 (right). These results, showing the lack of an optimum for the multiplicity, differ from what was observed in [31], where an optimal  $M$  value was indeed found and shown to be independent from the simulation time chosen for the analysis.



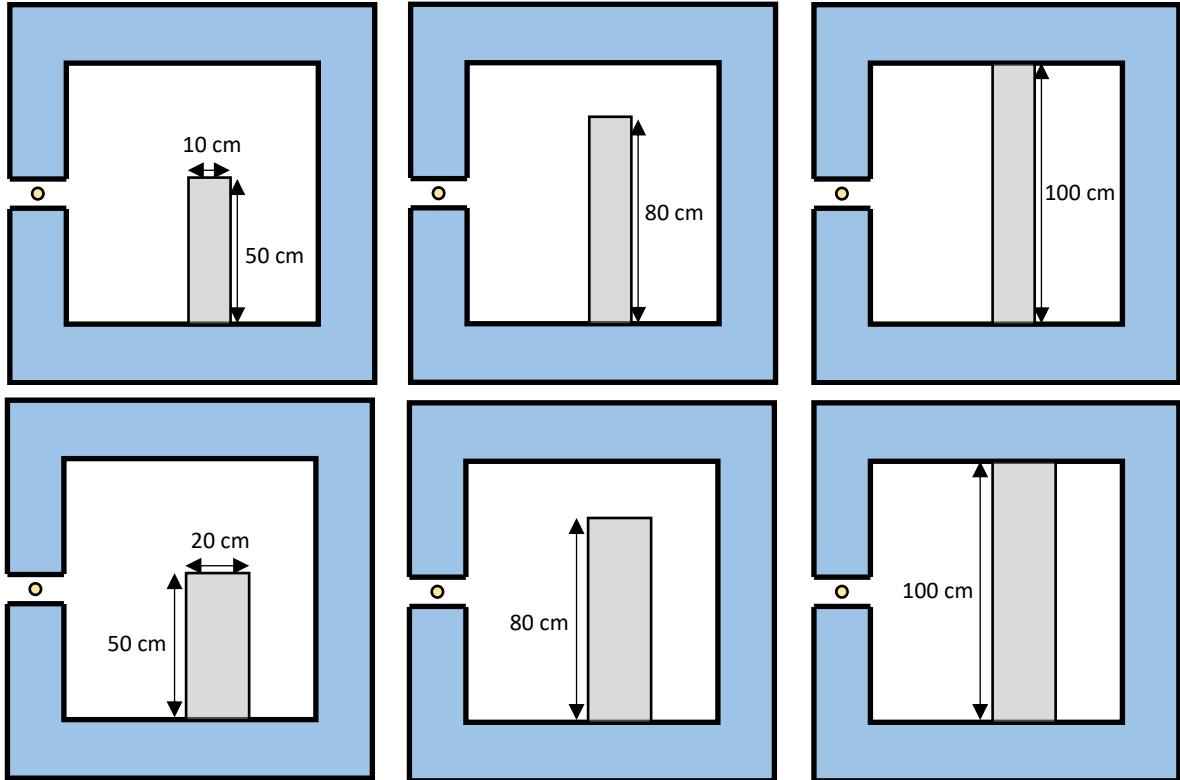
**Figure 7.37: Relative KERMA error (left) and average standard deviation  $\sigma$  (right) dependence on splitting multiplicity**

From the multiplicity analysis, it would seem that the introduction of splitting actually decreases the global efficiency of the *e*-TLE. However, the global KERMA error and the global average  $\sigma$  do not give us insight about the local efficiency of the estimator. To better assess this, we perform a more in-depth analysis on a refined mesh, presented in the next section. Based on the curves of Figure 7.37, we choose to investigate results obtained with a low multiplicity ( $M=5$ ) to avoid losing too much efficiency due to a high number of split particles.

### 7.3 Validation and performance analysis

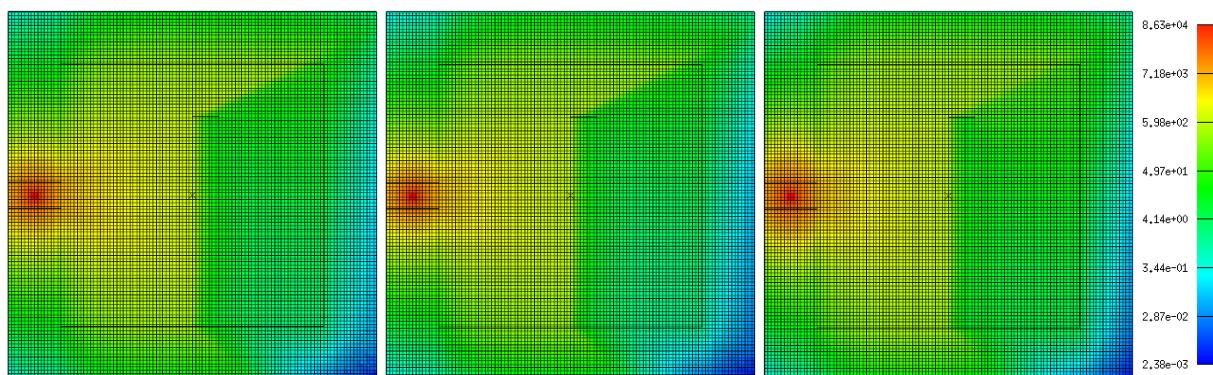
To validate the results obtained with the *se*-TLE, and evaluate its performance, we perform another set of simulations on the six bunker configurations with a highly refined mesh. In Figure 7.38, we recall the bunker configurations with different values for the shield's depth and height. The superimposed mesh is a 112x112x1 grid: the refinement in the  $z$  direction is dropped to have faster simulations. Therefore, this analysis is effectively

a 2D study, even if performed on a 3D geometry. As seen in the previous section, the multiplicity analysis failed to find an optimal value for the splitting multiplicity, suggesting that the actual optimal configuration would be without any splitting. However, we decided to run two *e*-TLE simulations for each bunker configuration, one with no splitting and one with a value of  $M$  set to 5, in the hope of better assessing the differences between the two approaches.



**Figure 7.38:** 2D,  $xy$  view of the six bunker configurations studied: R1, top left; R2, top center; R3, top right; R4, bottom left; R5, bottom center; R6, bottom right.

As usual, a reference simulation is performed with TRIPOLI-4®'s TLE, bringing the total number of simulations to 18, 3 for each of the six configurations. First of all, to test the estimator's validity, we compare a KERMA map obtained with the *se*-TLE and one obtained with the global *e*-TLE without splitting to the map obtained with TRIPOLI-4®'s TLE. For brevity, we report here only the results for the R2 configuration.



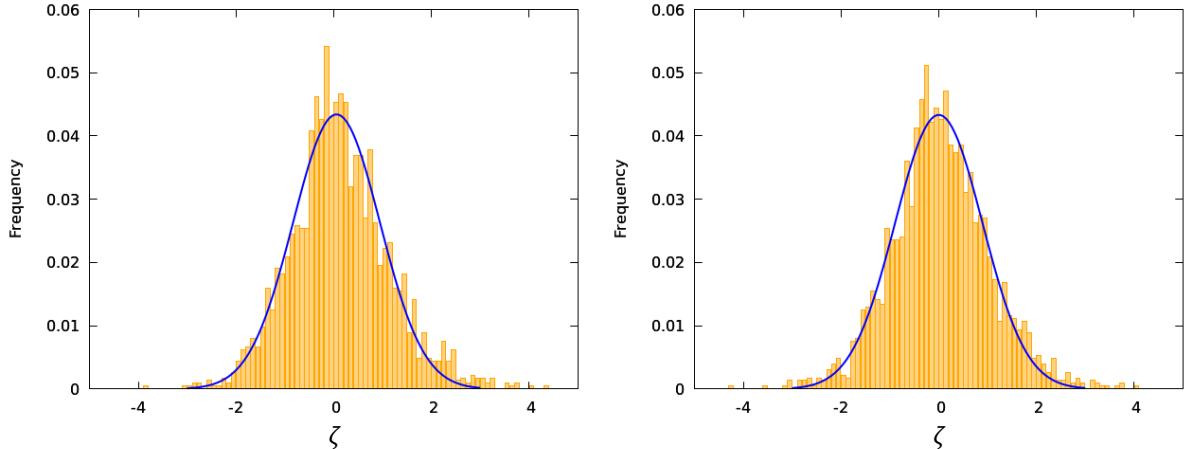
**Figure 7.39:** Comparison of KERMA maps for the R2 bunker configuration: TLE (left), global *e*-TLE without splitting (center), *se*-TLE with  $M=5$  (right)

Figure 7.39 shows a side-by-side comparison of the three maps. The results of the three simulations appear to be consistent. To validate the estimator more rigorously, we follow the procedure proposed in [105]. For every mesh cell, we define a  $\zeta$  variable as:

$$\zeta_i = \frac{K_i^{e-TLE} - K_i^{TLE}}{K_i^{TLE} \sigma^{TOT}} \quad (7.2)$$

where  $K_i^{e-TLE}$  and  $K_i^{TLE}$  are the KERMA values in the  $i$ -th cell calculated respectively by the global  $e$ -TLE and by the normal TLE, and  $\sigma^{TOT}$  is the standard deviation of the distribution  $K_i^{e-TLE} - K_i^{TLE}/K_i^{TLE}$ . We then compare the distribution of  $\zeta$  over the whole mesh to a standard normal distribution. Moreover, to avoid cells where the value of KERMA is very low and its statistical uncertainty very high, for this analysis we only consider cells where the KERMA is at least 0.1% of the maximum value registered.

Figure 7.40 shows the results of the analysis. Both with and without splitting, the statistical distribution of the  $\zeta$  values is shown to follow a standard normal distribution with mean  $\mu$  close to zero and standard deviation  $\sigma$  close to 1. We can conclude that the global  $e$ -TLE does not present any systemic bias.



**Figure 7.40: Normalized frequency of  $\zeta$  values for the R2 bunker configuration, fitted with a standard normal distribution: global  $e$ -TLE without splitting (shown left,  $\mu = 0.057, \sigma = 0.888$ ) and se-TLE with  $M=5$  (shown right,  $\mu = 0.001, \sigma = 0.893$ )**

To assess the efficiency of the  $e$ -TLE, a FOM is defined for every mesh cell and compared to the FOM of the corresponding TLE simulation. In this way, by looking at the cells where the FOM ratio is greater, one can have an idea of the spatial efficiency of the  $e$ -TLE as well as noting in which areas of the geometry the estimator performs better.

Results for the six configurations are shown in Figure 7.41 to Figure 7.46. A constant trend can be seen: the FOM ratio is greater in those areas of the geometry that are more difficult for particles to reach. For all configurations, both with and without splitting, the estimator performs well in the two corners of the room close to the source (which are shielded from direct radiation by the concrete).

When used with splitting, the estimator is also able to consistently improve results for the air behind the shield, while the version without splitting shows good results for configuration R6 only. Table 7.7 shows, for all configurations, the average FOM ratio value for air cells behind the shield.

**Table 7.7: Average FOM ratio for air cells behind the lead shield**

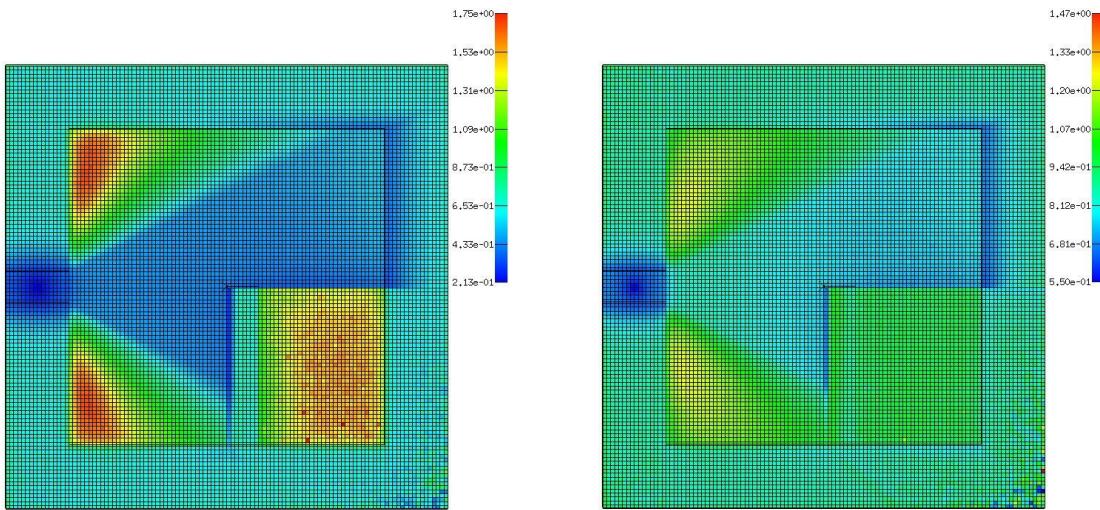
	R1	R2	R3	R4	R5	R6
<b>Global <i>e</i>-TLE</b>	0.94	0.92	0.95	0.91	0.84	1.02
<b>Global <i>se</i>-TLE</b>	1.35	1.22	1.16	1.27	1.24	1.06

We can also see a clear trend in the concrete walls, where the FOM ratio is almost always lower than 1 but increases closer to the edges of the geometry; this becomes more evident for the portions of wall which are also shielded by the lead slab, like the bottom right corner.

We can conclude that the estimator shows its best performance in deep penetration problems, and that the effect would be even clearer with thicker walls. In general, we can say that the estimator manages to improve simulation efficiency in “low-statistics areas”, if we design with this term all those areas that are less likely to be traversed by photons in a normal MC simulation.

Concerning the differences in performance between the version with splitting and the version without splitting, we can see that in general the version with splitting is both more efficient in low-statistics areas and less efficient in the rest of the geometry.

In low-statistics areas, the additional number of photons generated by the splitting algorithm provides additional statistics and lowers the final variance of the result; however, in areas that already have sufficiently high statistics, the additional time required by the algorithm greatly lowers the efficiency. This is why, in general, the FOM map has more contrast in the simulations with splitting, and is more “smoothed out” in no-split simulations. This is also evident by looking at the maximum and minimum values of the FOM in each simulation: if we exclude rare outliers, the simulation with splitting always has a higher maximum FOM and a lower minimum FOM than its no-split counterpart.



**Figure 7.41: FOM gain maps for the R1 bunker configuration, M=5 (left) and M=0 (right)**

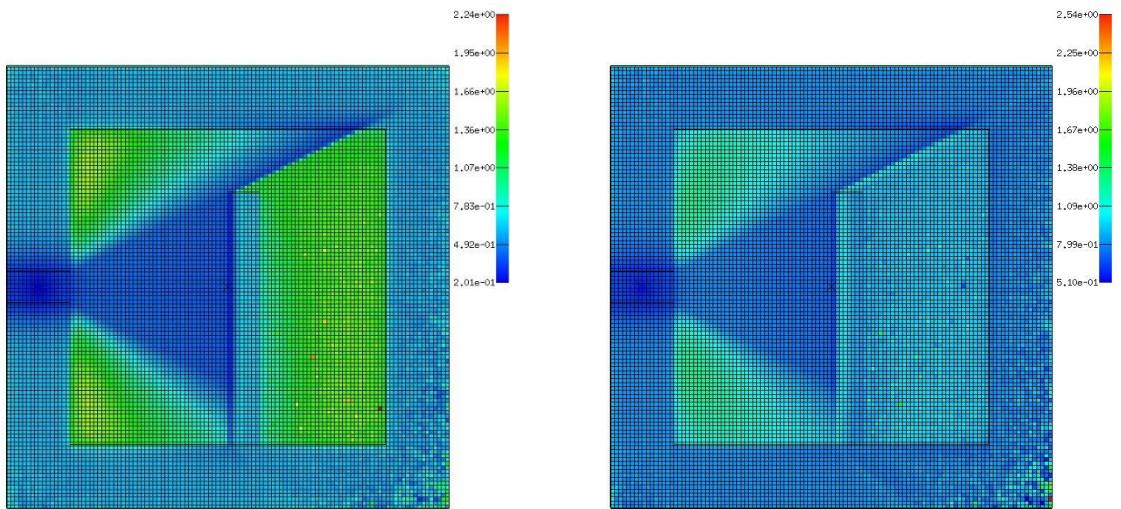


Figure 7.42: FOM gain maps for the R2 bunker configuration,  $M=5$  (left) and  $M=0$  (right)

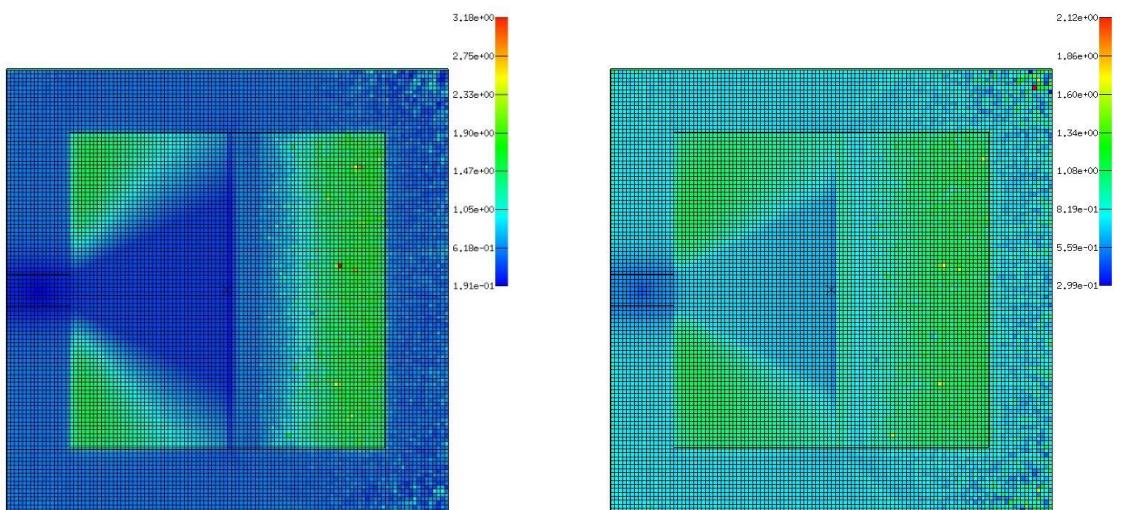


Figure 7.43: FOM gain maps for the R3 bunker configuration,  $M=5$  (left) and  $M=0$  (right)

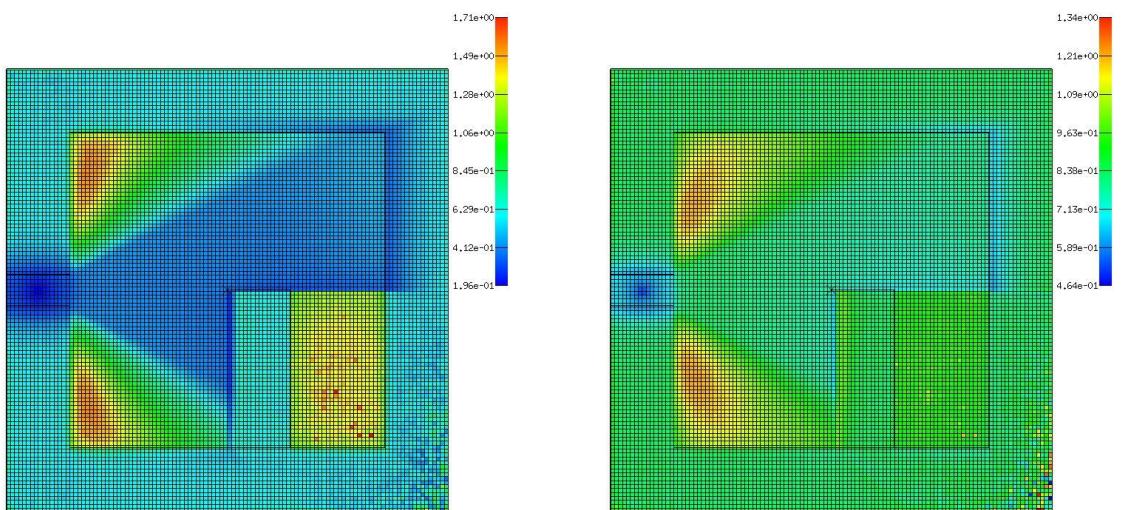
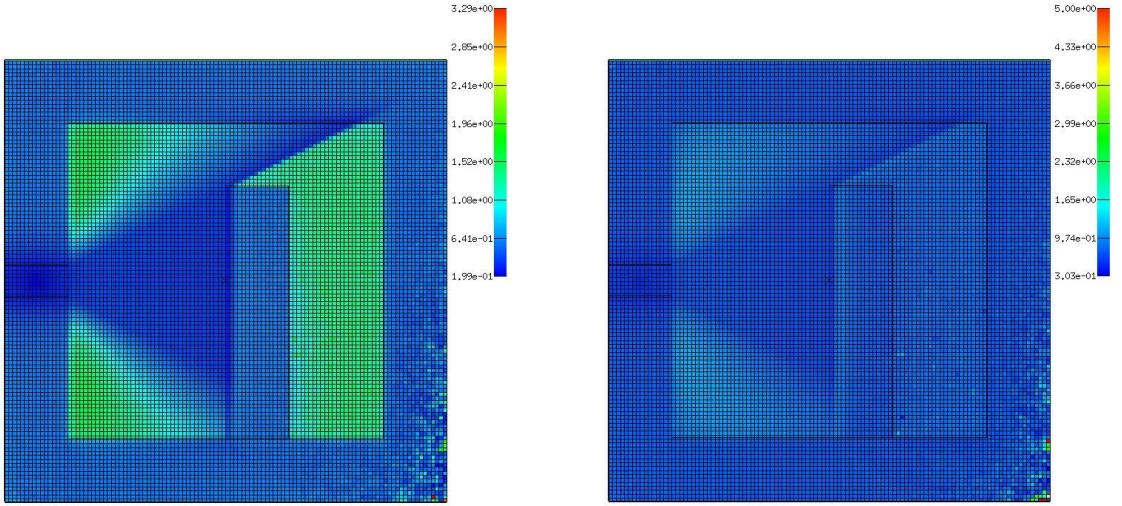
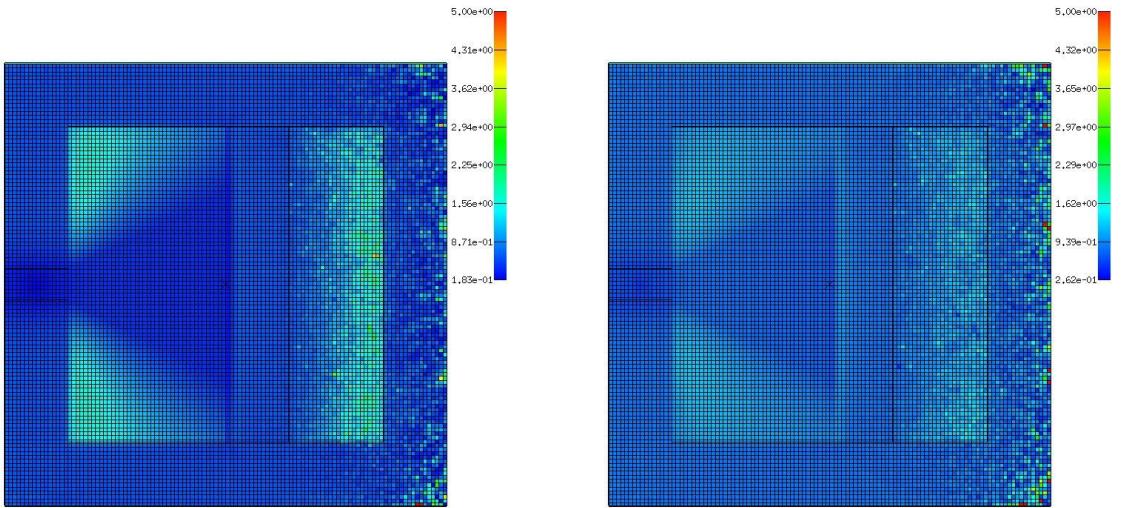


Figure 7.44: FOM gain maps for the R4 bunker configuration,  $M=5$  (left) and  $M=0$  (right)



**Figure 7.45:** FOM gain maps for the R5 bunker configuration,  $M=5$  (left) and  $M=0$  (right)



**Figure 7.46:** FOM gain maps for the R6 bunker configuration,  $M=5$  (left) and  $M=0$  (right)

## 7.4 Conclusions

The global version of the  $e$ -TLE is validated, in that it does not introduce any systemic bias in the simulation results, both with and without splitting.

The performance of this global version shows somewhat mixed results. Due to the added complexity of deterministic transport, the estimator's performance is inferior to that of the normal TLE in areas which already have a sufficient amount of photon statistics. Conversely, the global  $e$ -TLE shows good results in all those areas that are harder for the particles to reach, such as shielded areas.

The addition of a splitting algorithm is shown to produce an increase in both these effects. In high-statistics areas, split particles only add computation time without any significant improvement in the results' precision, *i.e.* with no observable variance-lowering effect. In low-statistics areas, this added computation time is balanced by the increase in statistics, which effectively improves the estimation efficiency.

These results are encouraging and suggest that the global version of the *e*-TLE could find its application in cases where it is important to improve the statistics in large areas of the simulation domain. It could be interesting to assess how the estimator compares to VRT techniques like the exponential transform in reducing the variance over low-statistics portions of the domain. Moreover, the capability of the estimator to push virtual particles in the farthest zones of the domain could be an argument for using it in applications different from dose rate evaluation, like the generation of importance maps through the calculation of adjoint fluxes in neutron calculations.



# Conclusions and perspectives

The objective of this thesis was to implement a new exponential track-length estimator in the code TRIPOLI-4®, taking inspiration from the one developed by Smekens *et al.* [100] for medical imaging applications and extending its domain of use to radioprotection studies.

Since the new estimator was expected to reduce the variance of the simulation, the first study that we performed was an analysis of the other Variance Reduction Techniques already implemented in TRIPOLI-4®, in the domain of gamma shielding calculations. This first study showed how these already existing VRT often failed to accelerate the simulations consistently, or in some cases (*i.e.* the adjustment of importance maps with INIPOND) required a certain amount of user experience to work correctly. This proved the interest of developing a new, fully automatized methodology able to consistently accelerate shielding simulations.

The first approach we followed was to develop a simple exponential track-length estimator, similar to the ones described in [52] and [97]. This first version of the *e*-TLE was defined as a volume-based estimator, due to the simpler implementation of this kind of estimators in TRIPOLI-4®. This first implementation was successful, as we showed that the new estimator was unbiased, required little to no user experience to be used in a simulation, and was consistently able to accelerate computations by a factor ranging from a little above 1 to 2 ~ 3 in the best cases. The estimator also showed mostly good performance when coupled with TRIPOLI-4®’s VRT.

From this first approach, we went on to perfect and refine the estimator by effectively creating two “versions” of it. The first one, called *e*-TLE-FD, was more suited to local variance reduction applications, *i.e.* the estimation of some quantity localized in a specific point of the domain. The second one was a refined version of the *e*-TLE, able to compute scores on meshes as well as volumes and to include a splitting algorithm to improve performances; this version was more suited to global variance reduction applications, *i.e.* the estimation of a quantity in the whole simulation domain.

The *e*-TLE-FD is a volume-based estimator that introduces a Forced Detection (FD) algorithm to the estimator’s routine. At each collision, a virtual particle is created in the collision point and deterministically transported towards the detectors, then it is killed after the score. In practice, the estimator computes the score that any particle *would have given* if it collided towards the detector, travelled up to it, and then registered a score. The *e*-TLE-FD showed excellent performance in all the test cases. In its final implementation, this estimator is able to deal with

multiple detectors (*i.e.* estimation volumes), although it achieves its best performance when used in simulations with only one detector. The choice of the number of detectors is left to the user as a free parameter.

The second version of the estimator, the global *e*-TLE, can be used both for volume tallies and for mesh tallies. It is conceived to accelerate simulations whose objective is to calculate a quantity defined over the whole geometric domain, or over a large part of it. The user can choose to use the estimator without particle splitting (*e*-TLE) or introduce any number of split virtual particles after every collision (*se*-TLE) to try and improve the estimator's performance. However, the user should keep in mind that the increased accuracy in low-statistics areas comes at the expense of a globally slower simulation. Therefore, an optimum must carefully be found to determine the best number of split particles, which changes depending on the simulation on a case-by-case basis.

A number of future implementations can be envisioned to improve and develop the results obtained with this thesis work.

First of all, the use of *e*-TLE and *e*-TLE-FD in conjunction with TRIPOLI-4®'s AMS and INIPOND can be further explored. Extensive tests should be done to see in which cases the use of the estimator can improve an already-accelerated simulation, and in which cases the algorithm complexity does not justify its use.

In the short/medium-term, the effects related to the activation of electro-magnetic shower could be evaluated. Since the estimator can be used with high-energy gammas, these effects should be taken into account. More generally, the use of *e*-TLE with particles other than photons could be explored. Preliminary tests showed that the estimator can be used with neutrons, suggesting that applications such as neutron flux estimations in a reactor core could be viable, similar to what was proposed by Sweezy *et al.* in [72]. This also paves the way for a longer-term implementation of *e*-TLE as an adjoint-flux estimator on a mesh, to be used for the generation of importance maps for superior variance reduction. With the application of *e*-TLE to neutron transport, another long-term perspective could be identified in coupled neutron-photon calculations for dismantling studies, such as those performed with the DEMAIN system. In this case, the neutron version of the *e*-TLE would be employed in the initial phases of the simulation, to calculate the source term for the activation code; then, the photon version of the estimator would be used in the final shielding calculation. The porting of the estimator to neutron calculations is currently the object of another thesis, conducted at CEA Cadarache by Henri Hutinet.

Finally, taking inspiration from Sweezy's VRC, the *e*-TLE algorithm could also be modified to include GPU-based acceleration. By porting some parts of the algorithm, like the deterministic transport, to the GPU, better acceleration performances could be achieved.

# References

1. International Energy Agency (IEA), <https://www.iea.org/statistics/electricity/>
2. International Atomic Energy Agency (IAEA),  
<https://pris.iaea.org/PRIS/WorldStatistics/OperationalByAge.aspx>
3. (2020) Energy, Electricity and Nuclear Power Estimates for the Period up to 2050. International Atomic Energy Agency, Vienna
4. Gauché F, Parisot J-F, Éditions le Moniteur (2018) Clean-up & dismantling of nuclear facilities. CEA Nuclear Energy Division, Paris
5. IAEA Country Nuclear Power Profiles. <https://cnpp.iaea.org/countryprofiles/France/France.htm>.
6. ASN, Le démantèlement des installations nucléaires. <https://www.asn.fr/Informer/Dossiers-pedagogiques/Le-demantement-des-installations-nucleaires>
7. (2006) Loi n° 2006-739 du 28 juin 2006 de programme relative à la gestion durable des matières et déchets radioactifs, <https://www.legifrance.gouv.fr/loda/id/JORFTEXT000000819043/>
8. (2006) Loi n° 2006-686 du 13 juin 2006 relative à la transparence et à la sécurité en matière nucléaire, <https://www.legifrance.gouv.fr/jorf/id/JORFTEXT000000240700/>
9. (2015) Loi n° 2015-992 du 17 août 2015 relative à la transition énergétique pour la croissance verte, <https://www.legifrance.gouv.fr/loda/id/JORFTEXT000031044385/>
10. (2007) Décret n° 2007-830 du 11 mai 2007 relatif à la nomenclature des installations nucléaires de base, <https://www.legifrance.gouv.fr/loda/id/JORFTEXT000000428384/>
11. (2007) Décret n° 2007-1557 du 2 novembre 2007 relatif aux installations nucléaires de base et au contrôle, en matière de sûreté nucléaire, du transport de substances radioactives, <https://www.legifrance.gouv.fr/loda/id/JORFTEXT000000469544/>
12. (2008) Décret n° 2008-251 du 12 mars 2008 relatif aux commissions locales d'information auprès des installations nucléaires de base, <https://www.legifrance.gouv.fr/loda/id/JORFTEXT000018315123/>
13. Xu Z, Hejzlar P (2008) MCODE, Version 2.2: An MCNP-ORIGEN DEpletion Program. Massachusetts Institute of Technology. Center for Advanced Nuclear Energy Systems. Nuclear Fuel Cycle Program
14. Werner CJ, Bull JS, Solomon CJ, *et al.* (2018) MCNP Version 6.2 Release Notes
15. Bell MJ (1973) ORIGEN: the ORNL isotope generation and depletion code. Oak Ridge National Lab.

16. Petrovich C (2001) SP-FISPACT2001. A computer code for activation and decay calculations for intermediate energies. A connection of FISPACT with MCNPX. Ente per le Nuove Tecnologie
17. Forrest R, Gilbert M (2005) FISPACT-2005: User manual
18. García-Herranz N, Cabellos Ó, Sanz J (2010) Assessment of the MCNP-ACAB code system for burnup credit analyses
19. Sanz J, Latkowski JF, Tobin MT, *et al.* (2000) ACAB Activation Code for Fusion Applications: User's Manual V5. 0. Lawrence Livermore Natl Lab UCRL-MA-143238
20. Le Loirec C, Soulard M, Ritter G, Pénéliau Y (2020) Benchmark of DEMAIN, a new calculation package for decommissioning purposes. *Prog Nucl Energy* 118:103076. <https://doi.org/10.1016/j.pnucene.2019.103076>
21. Brun E, Damian F, Diop CM, *et al.* (2015) TRIPOLI-4®, CEA, EDF and AREVA reference Monte Carlo code. *Ann Nucl Energy* 82:151–160. <https://doi.org/10.1016/j.anucene.2014.07.053>
22. Tsilanizara A, Diop CM, Nimal B, *et al.* (2000) DARWIN: An Evolution Code System for a Large Range of Applications. *J Nucl Sci Technol* 37:845–849. <https://doi.org/10.1080/00223131.2000.10875009>
23. Lahaye S, Bellier P, Mao H, *et al.* (2014) First verification and validation steps of mendel release 10 cycle code system. Proceedings of the international conference on physics of reactors (PHYSOR2014), Japan, 2014
24. Jaboulay J-C, Brun E, Hugot F-X, *et al.* (2016) Rigorous-two-Steps scheme of TRIPOLI-4® Monte Carlo code validation for shutdown dose rate calculation. In: 13th International Conference on Radiation Shielding (ICRS-13) & 19th Topical Meeting of the Radiation Protection & Shielding Division of the American Nuclear Society -2016 (RPSD-2016). Paris, France, p 02008
25. Sanchez R, Zmijarevic IM, Delclaux C, *et al.* (2010) APOLLO2 YEAR 2010. *Nucl Eng Technol* 42:474–499
26. Faure B, Archier P, Vidal J-F, *et al.* (2018) Neutronic calculation of an axially heterogeneous ASTRID fuel assembly with APOLLO3®: Analysis of biases and foreseen improvements. *Ann Nucl Energy* 115:88–104. <https://doi.org/10.1016/j.anucene.2017.12.035>
27. Hfaiedh N, Santamarina A (2005) Determination of the optimized SHEM mesh for neutron transport calculations. In: Conference: M&C 2005: international topical meeting on mathematics and computation, supercomputing, reactor physics and nuclear and biological applications, Avignon (France), 12-15 Sep 2005
28. Koning A, Forrest R, Kellett M, *et al.* (2006) The JEFF-3.1 Nuclear Data Library - JEFF Report 21. Organisation for Economic Co-Operation and Development
29. Lautard JJ, Loubiere S, Fedon-Magnaud C (1992) CRONOS: A modular computational system for neutronic core calculations. International Atomic Energy Agency (IAEA)
30. Forrest RA (2001) The European Activation File. EAF-2001 decay data library. EURATOM/UKAEA Fusion Association
31. Smekens F, Freud N, Létang JM, *et al.* (2009) Simulation of dose deposition in stereotactic synchrotron radiation therapy: a fast approach combining Monte Carlo and deterministic algorithms. *Phys Med Biol* 54:4671–4685. <https://doi.org/10.1088/0031-9155/54/15/003>
32. Prokopets IM, Prokopets SI, Khazhmuradov MA, *et al.* (2007) Point-kernel method for radiation fields simulation. *Vopr At Nauki Tekhniki* 106–109
33. Longeot M, Dupont B, Zweers M, *et al.* (2014) PANTHERE: simulation software for 3D dose rate calculation in complex nuclear facilities. *Prog Nucl Sci Technol* 4:557–560. <https://doi.org/10.15669/pnst.4.557>
34. ICRP. <http://www.icrp.org/publication.asp?id=ICRP%20Publication%2074>. Accessed 27 Jan 2020

35. Kitsos S, Diop CM, Assad A, *et al.* (1996) Improvement of Gamma-Ray Sn Transport Calculations Including Coherent and Incoherent Scatterings and Secondary Sources of Bremsstrahlung and Fluorescence: Determination of Gamma-Ray Buildup Factors. *Nucl Sci Eng* 123:215–227. <https://doi.org/10.13182/NSE96-A24184>
36. Trubey DK, Eisenhauer CM, Foderaro A, *et al.* (1991) Gamma-Ray Attenuation Coefficients and Buildup Factors for Engineering Materials. 643–1991
37. Visonneau T, Pangault L, Malouch F, *et al.* (2017) NARMER-1: a photon point-kernel code with build-up factors. *EPJ Web Conf* 153:06028. <https://doi.org/10.1051/epjconf/201715306028>
38. Assad A, Chiron M, Claude Nimal J, *et al.* (2000) General Formalism for Calculating Gamma-Ray Buildup Factors in Multilayer Shields into MERCURE-6 Code. *J Nucl Sci Technol* 37:493–497. <https://doi.org/10.1080/00223131.2000.10874935>
39. Suteau C, Chiron M (2005) An iterative method for calculating gamma-ray build-up factors in multi-layer shields. *Radiat Prot Dosimetry* 116:489–492. <https://doi.org/10.1093/rpd/nci192>
40. Suteau C, Chiron M, Arnaud G (2004) Improvement of MERCURE-6's General Formalism for Calculating Gamma-Ray Buildup Factors in Multilayer Shields. *Nucl Sci Eng* 147:43–55. <https://doi.org/10.13182/NSE04-A2417>
41. Vela O, de Burgos E, Perez JM (2006) Dose rate assessment in complex geometries. *IEEE Trans Nucl Sci* 53:304–311. <https://doi.org/10.1109/TNS.2006.869831>
42. Litaize O (2000) Contribution to the determination of the double angular and energy differential neutron albedo. Application to the propagation in lacunar medium. France
43. Chucas S, Curl I (2000) Streaming Calculations using the Point-Kernel Code RANKERN. *J Nucl Sci Technol* 37:515–519. <https://doi.org/10.1080/00223131.2000.10874940>
44. Canberra Industries (2003) MERCURAD - Dose Rate Calculation Software, User's Manual
45. Thevenon JB, Ferey T, Lopez L, *et al.* (2008) Narveos: a virtual reality simulation environment for nuclear radiating working sites. In: Decommissioning challenges: an industrial reality?, Avignon (France); 28 Sep - 2 Oct 2008
46. Caracena TM, Gonçalves JGM, Peerani P, Vidal EV (2013) A Variable Point Kernel Dosimetry Method for Virtual Reality Simulation Applications in Nuclear Safeguards and Security. *IEEE Trans Nucl Sci* 60:3862–3871. <https://doi.org/10.1109/TNS.2013.2279411>
47. Szőke I, Louka MN, Brynteson TR, *et al.* (2014) Real-time 3D radiation risk assessment supporting simulation of work in nuclear environments. *J Radiol Prot Off J Soc Radiol Prot* 34:389–416. <https://doi.org/10.1088/0952-4746/34/2/389>
48. Chao N, Liu Y, Xia H, *et al.* (2018) A dose assessment method for arbitrary geometries with virtual reality in the nuclear facilities decommissioning. *Radiat Phys Chem* 144:238–247. <https://doi.org/10.1016/j.radphyschem.2017.08.020>
49. Chao N, Liu Y, Xia H, *et al.* (2018) Adaptive point kernel dose assessment method for cutting simulation on irregular geometries in nuclear facility decommissioning. *Radiat Phys Chem* 150:125–136. <https://doi.org/10.1016/j.radphyschem.2018.04.035>
50. Chao N, Liu Y, Xia H, *et al.* (2017) A sampling-based method with virtual reality technology to provide minimum dose path navigation for occupational workers in nuclear facilities. *Prog Nucl Energy* 100:22–32. <https://doi.org/10.1016/j.pnucene.2017.05.024>
51. Lux I, Koblinger L, (1991) Monte Carlo Particle Transport Methods: Neutron and Photon Calculations. CRC Press, Boca Raton

52. Gelbard EM, Ondis II LA, Spanier J (1966) A New Class of Monte Carlo Estimators. *SIAM J Appl Math* 14:697–701. <https://doi.org/10.1137/0114058>
53. Barish RJ (1991) The Dosimetry of Ionizing Radiation. Vol 3. *Radiology* 180:120–120. <https://doi.org/10.1148/radiology.180.1.120>
54. Baldacci F, Mittone A, Bravin A, *et al.* (2015) A track length estimator method for dose calculations in low-energy X-ray irradiations: implementation, properties and performance. *Z Für Med Phys* 25:36–47. <https://doi.org/10.1016/j.zemedi.2014.04.001>
55. Odano N, Miura T, Shindo Y (1994) Experimental assessment of gamma-ray shielding transport calculations by Monte Carlo calculation code and Point-Kernel code. In: Proceedings of ICRS-8. Arlington, USA, pp 1301–1308
56. Kodeli IA, Sartori E (2021) SINBAD – Radiation shielding benchmark experiments. *Ann Nucl Energy* 159:108254. <https://doi.org/10.1016/j.anucene.2021.108254>
57. Shin SH, Choi WN, Yoon S, *et al.* (2021) Radiological analysis of transport and storage container for very low-level liquid radioactive waste. *Nucl Eng Technol*. <https://doi.org/10.1016/j.net.2021.06.024>
58. Saito M, Ueno K, Maruyama T, *et al.* (2014) Preliminary assessment for dust contamination of ITER In-Vessel Transporter. *Fusion Eng Des* 89:2352–2356. <https://doi.org/10.1016/j.fusengdes.2014.02.023>
59. Park HS, Kim SK, Lee KW, *et al.* (2008) Visualization of a dismantling environment for an evaluation of a worker's dose during the decommissioning of KRR-1&2. *Ann Nucl Energy* 35:1117–1124. <https://doi.org/10.1016/j.anucene.2007.10.004>
60. Lakshminarayana G, Kumar A, Dong MG, *et al.* (2018) Exploration of gamma radiation shielding features for titanate bismuth borotellurite glasses using relevant software program and Monte Carlo simulation code. *J Non-Cryst Solids* 481:65–73. <https://doi.org/10.1016/j.jnoncrysol.2017.10.027>
61. Tekin HO, Sayyed MI, Issa SAM (2018) Gamma radiation shielding properties of the hematite-serpentine concrete blended with WO<sub>3</sub> and Bi<sub>2</sub>O<sub>3</sub> micro and nano particles using MCNPX code. *Radiat Phys Chem* 150:95–100. <https://doi.org/10.1016/j.radphyschem.2018.05.002>
62. Sardari D, Abbaspour A, Baradaran S, Babapour F (2009) Estimation of gamma- and X-ray photons buildup factor in soft tissue with Monte Carlo method. *Appl Radiat Isot* 67:1438–1440. <https://doi.org/10.1016/j.apradiso.2009.02.033>
63. Irving DC (1971) The adjoint Boltzmann equation and its simulation by Monte Carlo. *Nucl Eng Des* 15:273–293. [https://doi.org/10.1016/0029-5493\(71\)90069-0](https://doi.org/10.1016/0029-5493(71)90069-0)
64. Matijević M, Pevec D, Trontl K (2016) PWR Containment Shielding Calculations with SCALE6.1 Using Hybrid Deterministic-Stochastic Methodology. *Sci Technol Nucl Install* 2016:1–30. <https://doi.org/10.1155/2016/7328131>
65. Sakamoto Y, Tanaka S (1990) QAD-CGGP2 and G33-GP2: revised versions of QAD-CGGP and G33-GP. Japan Atomic Energy Research Inst.
66. Longeot M, Dupont B, Zweers M, *et al.* (2014) PANTHERE: simulation software for 3D dose rate calculation in complex nuclear facilities. *Prog Nucl Sci Technol* 4:557–560. <https://doi.org/10.15669/pnst.4.557>
67. Žohar A, Lengar I, Snoj L (2020) Analysis of water activation in fusion and fission nuclear facilities. *Fusion Eng Des* 160:111828. <https://doi.org/10.1016/j.fusengdes.2020.111828>
68. Burn KW (1997) A New Weight-Dependent Direct Statistical Approach Model. *Nucl Sci Eng* 125:128–170. <https://doi.org/10.13182/NSE97-A24262>

69. Nowak M, Mancusi D, Louvin H, *et al.* (2018) Accelerating Variance Reduction techniques by scoring the importance map in direct Monte Carlo Shielding calculations. In: PHYSOR 2018: Reactor Physics paving the way towards more efficient systems. Cancun, Mexico, April 22-26, 2018
70. Verhaegen F, Seco J (2021) Monte Carlo Techniques in Radiation Therapy: Introduction, Source Modelling and Patient Dose Calculations. CRC Press
71. Tickner J (2010) Monte Carlo simulation of X-ray and gamma-ray photon transport on a graphics-processing unit. *Comput Phys Commun* 181:1821–1832. <https://doi.org/10.1016/j.cpc.2010.07.001>
72. Sweezy JE (2018) A Monte Carlo volumetric-ray-casting estimator for global fluence tallies on GPUs. *J Comput Phys* 372:426–445. <https://doi.org/10.1016/j.jcp.2018.06.032>
73. Agostinelli S, Allison J, Amako K, *et al.* (2003) Geant4—a simulation toolkit. *Nucl Instrum Methods Phys Res Sect Accel Spectrometers Detect Assoc Equip* 506:250–303. [https://doi.org/10.1016/S0168-9002\(03\)01368-8](https://doi.org/10.1016/S0168-9002(03)01368-8)
74. Brualla L, Salvat F, Palanco-Zamora R (2009) Efficient Monte Carlo simulation of multileaf collimators using geometry-related variance-reduction techniques. *Phys Med Biol* 54:4131–4149. <https://doi.org/10.1088/0031-9155/54/13/011>
75. Jenkins TM, Nelson WR, Rindi A (1988) Monte Carlo Transport of Electrons and Photons. Springer US
76. Hendricks JS, Booth TE (1985) MCNP variance reduction overview. In: Alcouffe R, Dautray R, Forster A, *et al.* (eds) Monte-Carlo Methods and Applications in Neutronics, Photonics and Statistical Physics. Springer-Verlag, Berlin/Heidelberg, pp 83–92
77. Vassiliev ON (2017) Monte Carlo methods for radiation transport. *Fundam Adv Top*
78. Kawrakow I, Rogers DWO, Walters BRB (2004) Large efficiency improvements in BEAMnrc using directional bremsstrahlung splitting. *Med Phys* 31:2883–2898. <https://doi.org/10.1118/1.1788912>
79. Guerra BT, Soares AL, Pereira C Analysis of neutron flux distribution using the Monte Carlo method for the feasibility study of the Prompt Gamma Activation Analysis technique at the IPR-R1 TRIGA reactor. In: INAC 2015: international nuclear atlantic conference. Brazilian nuclear program. State policy for a sustainable world; Sao Paulo, SP (Brazil); 4-9 Oct 2015
80. Rodionov R, Nemtcev G, Barnsley R, *et al.* (2021) Optimization of the ITER vertical neutron camera detectors and collimators. *Fusion Eng Des* 166:112341. <https://doi.org/10.1016/j.fusengdes.2021.112341>
81. Gomit Jm, Duhamel I, Richet Y, *et al.* (2017) CRISTAL v2: new package for criticality calculations. In: Nuclear Criticality Safety Division Topical Meeting (NCSD 2017). Nuclear Criticality Safety Division Topical Meeting (NCSD 2017), Carlsbad, United States
82. Brun R, Gheata A, Gheata M (2003) The ROOT geometry package. *Nucl Instrum Methods Phys Res Sect Accel Spectrometers Detect Assoc Equip* 502:676–680. [https://doi.org/10.1016/S0168-9002\(03\)00541-2](https://doi.org/10.1016/S0168-9002(03)00541-2)
83. Joly Y (2009) Interaction Matter-Polarized Light. In: Beaurepaire E, Bulou H, Scheurer F, Kappler J-P (eds) Magnetism and Synchrotron Radiation. Springer, p 77
84. Pénéliau Y (2001) Electron photon shower simulation in TRIPOLI-4 Monte Carlo code. In: Advanced Monte Carlo for Radiation Physics, Particle Transport Simulation and Applications. Springer, pp 129–134
85. Both JP, Nimal JC, Vergnaud T (1990) Automated importance generation and biasing techniques for Monte Carlo shielding techniques by the TRIPOLI-3 code. *Prog Nucl Energy* 24:273–281. [https://doi.org/10.1016/0149-1970\(90\)90046-8](https://doi.org/10.1016/0149-1970(90)90046-8)
86. Weinberg AM, Wigner EP, Cohen ER (1959) The physical theory of neutron chain reactors. *Phys Today* 12:34

87. Dehaye B (2014) Accélération de la convergence dans le code de transport de particules Monte-Carlo TRIPOLI-4® en criticité. Phdthesis, Université Paris Sud - Paris XI
88. Nowak M, Mancusi D, Sciannadrone D, *et al.* (2019) Accelerating Monte Carlo Shielding Calculations in TRIPOLI-4 with a Deterministic Adjoint Flux. *Nucl Sci Eng* 193:966–981. <https://doi.org/10.1080/00295639.2019.1578568>
89. Lei M, Penelaiu Y, Lee Y-K, Song Y (2018) Analysis of Dogleg Duct Experiments With 14-MeV Neutron Source Using TRIPOLI-4 Monte Carlo Transport Code. *IEEE Trans Plasma Sci* 46:1180–1185. <https://doi.org/10.1109/TPS.2017.2786465>
90. Coveyou RR, Cain VR, Yost KJ (1967) Adjoint and Importance in Monte Carlo Application. *Nucl Sci Eng* 27:219–234. <https://doi.org/10.13182/NSE67-A18262>
91. Hoogenboom JE (1979) Optimum Biasing of Integral Equations in Monte Carlo Calculations. *Nucl Sci Eng* 70:210–212. <https://doi.org/10.13182/NSE79-A19656>
92. Gualdrini G, Ferrari P (2011) Monte Carlo variance reduction techniques: an overview with some practical examples. *Radiat Prot Dosimetry* 146:425–433. <https://doi.org/10.1093/rpd/ncr240>
93. Guadagni E, Le Loirec C, Mancusi D (2021) Comparison of variance-reduction techniques for gamma dose rate determination. *Eur Phys J Plus* 136:1–20
94. Hubbell JH, Veigle WmJ, Briggs EA, *et al.* (1975) Atomic form factors, incoherent scattering functions, and photon scattering cross sections. *J Phys Chem Ref Data* 4:471–538. <https://doi.org/10.1063/1.555523>
95. Storm L, Israel HI (1970) Photon cross sections from 1 keV to 100 MeV for elements Z=1 to Z=100. At Data Nucl Data Tables 7:565–681. [https://doi.org/10.1016/S0092-640X\(70\)80017-1](https://doi.org/10.1016/S0092-640X(70)80017-1)
96. Haghigat A, Wagner JC (2003) Monte Carlo variance reduction with deterministic importance functions. *Prog Nucl Energy* 42:25–53. [https://doi.org/10.1016/S0149-1970\(02\)00002-1](https://doi.org/10.1016/S0149-1970(02)00002-1)
97. Williamson JF (1987) Monte Carlo evaluation of kerma at a point for photon transport problems: Monte Carlo calculation of kerma at a point. *Med Phys* 14:567–576. <https://doi.org/10.1118/1.596069>
98. Freud N, Létang J-M, Babot D (2005) A hybrid approach to simulate multiple photon scattering in X-ray imaging. *Nucl Instrum Methods Phys Res Sect B Beam Interact Mater At* 227:551–558. <https://doi.org/10.1016/j.nimb.2004.10.001>
99. Freud N, Letang JM, Mary C, *et al.* (2008) A Hybrid Approach for Fast Simulation of Dose Deposition in Stereotactic Synchrotron Radiotherapy. *IEEE Trans Nucl Sci* 55:1008–1017. <https://doi.org/10.1109/TNS.2008.922831>
100. Smekens F, Létang JM, Noblet C, *et al.* (2014) Split exponential track length estimator for Monte-Carlo simulations of small-animal radiation therapy. *Phys Med Biol* 59:7703–7715. <https://doi.org/10.1088/0031-9155/59/24/7703>
101. Adams T, Nolen S, Sweezy J, *et al.* (2015) Monte Carlo Application ToolKit (MCATK). *Ann Nucl Energy* 82:41–47. <https://doi.org/10.1016/j.anucene.2014.08.047>
102. Cramer SN (1984) Variance reduction methods applied to deep-penetration problems. Oak Ridge National Lab., TN (USA)
103. Tickner JR (2009) Algorithm for forcing scattered radiation to arbitrary convex regions in neutral particle Monte Carlo simulation. *Nucl Instrum Methods Phys Res Sect B Beam Interact Mater At* 267:2361–2364. <https://doi.org/10.1016/j.nimb.2009.05.008>
104. Guadagni E, Le Loirec C, Pénéliau Y, *et al.* (2021) A new hybrid next-event estimator for photon-based Monte Carlo dose rate calculations. *Eur Phys J Plus* 136:1135. <https://doi.org/10.1140/epjp/s13360-021-02120-5>

105. Giménez-Alventosa V, Giménez Gómez V, Oliver S (2021) PenRed: An extensible and parallel Monte-Carlo framework for radiation transport based on PENELOPE. *Comput Phys Commun* 267:108065. <https://doi.org/10.1016/j.cpc.2021.108065>
106. Podgorsak EB (2005) Radiation oncology physics. International Atomic Energy Agency, Vienna.



# ANNEX A: KERMA AND DOSE DEFINITIONS

**KERMA** [106] is an acronym for Kinetic Energy Released in Matter. It is a quantity applicable to indirectly ionizing radiations such as photons and neutrons. It quantifies the average amount of energy transferred from indirectly ionizing radiation to directly ionizing radiation, without concern as to what happens after this transfer. In the discussion that follows, we will limit ourselves to photons.

The energy of photons is imparted to matter in a two-stage process. In the first stage, the photon radiation transfers energy to the secondary charged particles (electrons) through various photon interactions (the photoelectric effect, the Compton effect, pair production, etc.). In the second stage, the charged particle transfers energy to the medium through atomic excitations and ionizations. KERMA is related to the first stage, and it is defined as the mean energy  $d\bar{E}_{tr}$  transferred from the indirectly ionizing radiation (photons) to charged particles (electrons) in the medium per unit mass  $dm$ :

$$KERMA = \frac{d\bar{E}_{tr}}{dm} \quad A.1$$

KERMA is measured in Gray (Gy), where  $1\text{Gy}=1\text{J/kg}$ .

The **absorbed dose**, or simply **dose**, is another quantity related to the energy transfer of ionizing radiation. As said before, when a photon beam interacts with matter, the resulting process of energy transfer can be divided in two phases. The dose is related to the second one, where the charged particles (electrons) excited by the photons transfer some of their kinetic energy to the medium (resulting in absorbed dose) and lose some of their energy in the form of radiative losses (bremsstrahlung, annihilation in flight). The absorbed dose  $D$  is defined as the mean energy  $d\bar{E}_{imp}$  imparted by ionizing radiation to the unit mass  $dm$ :

$$D = \frac{d\bar{E}_{imp}}{dm} \quad A.2$$

Having the same quantities of KERMA, dose is also measured in Gray.

The first important thing to note is that, because electrons travel in the medium and deposit energy along their tracks, this absorption of energy does not take place at the same location as the transfer of energy described by KERMA. Another important distinction between the two quantities is that the dose does not take into account radiative losses undergone by the charged particle during its travel in the medium, while KERMA does.

In fact, KERMA can be divided in two parts:

- The **collision KERMA**  $K_{col}$  is that part of KERMA that leads to the production of electrons that dissipate their energy as ionization in or near the electron tracks in the medium, and is the result of Coulomb force interactions with atomic electrons. Thus, the collision KERMA is the expectation value of the net energy

transferred to charged particles per unit mass at the point of interest, excluding both the radiative energy loss and energy passed from one charged particle to another.

- The **radiative KERMA**  $K_{rad}$  is that part of KERMA that leads to the production of radiative photons as the secondary charged particles slow down and interact in the medium. These interactions most prominently are bremsstrahlung as a result of Coulomb field interactions between the charged particle and the atomic nuclei, but can also result from annihilation in flight.

The total KERMA  $K$  is thus given by the following:

$$K = K_{col} + K_{rad} \quad \text{A.3}$$

Therefore, the dose is directly related only to the collisional part of the KERMA. Specifically, in a condition of ideal charged particle equilibrium (attained when the energy emitted per unit mass of material is the same as the energy absorbed per unit volume), then the dose is exactly equal to the collision KERMA.

The **DER** is an acronym for Dose Equivalent Rate. The dose equivalent is usually measured in Sv (Sievert) and it represents a measure of the biological damage to living tissue as a result of radiation exposure. Since it gives an estimate of biological damage, it is the quantity usually adopted in radioprotection analysis to evaluate safety limits. Specifically, since the total dose is proportional to the time of exposure, the dose equivalent rate (Sv/s or Sv/h) is often used. The dose equivalent is related to the dose by the following relation:

$$D_{eq} = \sum_R \sum_T w_{R,T} D_{R,T} \quad \text{A.4}$$

where  $D_{R,T}$  is the dose of radiation R absorbed by tissue T, and  $w_{R,T}$  is a weighting factor measuring the vulnerability of tissue T to radiation R.

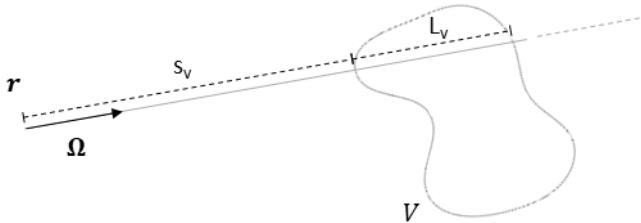
# ANNEX B: RIGOROUS MATHEMATICAL DERIVATION OF THE EXPONENTIAL TRACK-LENGTH ESTIMATOR

We show here that the exponential track-length estimator can be rigorously derived from both the collision estimator and the classic track-length estimator, by averaging their values over all possible particle flight lengths. We take the simple problem described in Figure B.47. A particle is flying from point  $\mathbf{r}$ , with direction  $\Omega$ , towards the estimation volume  $V$ . We want to calculate the average contribution given by the particle to the flux in volume  $V$  by using a collision estimator. We recall that the collision estimator gives a contribution equal to:

$$\kappa_{COLL} = \frac{\Pi_V(\mathbf{r})}{\Sigma_t V} \quad (\text{B.1})$$

where the characteristic function  $\Pi_V(\mathbf{r})$  is defined as:

$$\Pi_V(\mathbf{r}) = \begin{cases} 1 & \text{if } \mathbf{r} \text{ is inside } V, \\ 0 & \text{if not.} \end{cases} \quad (\text{B.2})$$



**Figure B.47: Geometric configuration for the mathematical derivation of e-TLE**

To calculate the average contribution given by the particle in point  $\mathbf{r}$ , we integrate  $\kappa_{COLL}$  over all possible flight lengths  $s$  starting from  $\mathbf{r}$ , and multiply it by the displacement operator  $\Sigma_t e^{\int_0^s -\Sigma_t dt}$  (see also Equation (1.15)), which can be seen as the probability of traveling for a length  $s$  and then having a collision. For simplicity, we choose a homogeneous isotropic material where the total macroscopic cross section is equal to  $\Sigma_t$  everywhere, so the displacement operator simplifies to  $\Sigma_t e^{-s\Sigma_t}$ . We can write:

$$\kappa_{COLL,avg} = \int_0^\infty \frac{\Pi_V(\mathbf{r} + s\Omega)}{\Sigma_t V} \Sigma_t e^{-s\Sigma_t} ds \quad (\text{B.3})$$

By referring to Figure B.47, and remembering the definition of  $\Pi_V(\mathbf{r})$  given in Equation (B.2), we can see that all trajectories for which  $s < s_V$  or  $s > s_V + L_V$  will give a null contribution. We can therefore rewrite the integral as:

$$\kappa_{COLL,avg} = \int_{s_V}^{s_V+L_V} \frac{1}{V} e^{-s\Sigma_t} ds \quad (\text{B.4})$$

To simplify the notation we operate a variable change by considering  $s' = s - s_V$ :

$$\kappa_{COLL,avg} = \int_0^{L_V} \frac{1}{V} e^{-(s'+s_V)\Sigma_t} ds' = \frac{1}{V} e^{-s_V\Sigma_t} \int_0^{L_V} e^{-s'\Sigma_t} ds' \quad (\text{B.5})$$

and finally we solve the integral to obtain:

$$\kappa_{COLL,avg} = e^{-s_V\Sigma_t} \frac{1 - e^{-\Sigma_t L_V}}{V\Sigma_t} \quad (\text{B.6})$$

which is equal to the formulation given for the  $e$ -TLE in Section 1.3.2 and in Chapter 5.

We will now show that, if we start from the track-length estimator instead of the collision estimator, we get to the same final result. We recall that, in its basic formulation, the track-length estimator gives a contribution equal to:

$$\kappa_{TRACK} = \frac{L_j}{V} \quad (\text{B.7})$$

where the characteristic length  $L_j$  (see also Equation (1.22) in Section 1.3.2) is equal to the length travelled by the particle inside volume  $V$  between two successive interactions:

$$L_j = \int_0^s \Pi_V(\mathbf{r} + s'\boldsymbol{\Omega}) ds' \quad (\text{B.8})$$

Therefore, as before, we can write:

$$\kappa_{TRACK,avg} = \int_0^\infty \frac{\Pi_V(\mathbf{r} + s\boldsymbol{\Omega})}{V} \Sigma_t e^{-s\Sigma_t} ds \quad (\text{B.9})$$

In this case, we have three distinct situations. First of all, like for the collision estimator, for all  $s < s_V$  the estimator gives a null contribution. Then, for all  $s_V < s < s_V + L_V$ , the integral of equation (B.9) becomes equal to:

$$\int_{s_V}^{s_V+L_V} \frac{s - s_V}{V} \Sigma_t e^{-s\Sigma_t} ds \quad (\text{B.10})$$

Finally, for all flight lengths  $s > s_V + L_V$ , the particle flies past the volume and  $L_j = L_V$ . The integral now becomes:

$$\int_{s_V+L_V}^\infty \frac{L_V}{V} \Sigma_t e^{-s\Sigma_t} ds \quad (\text{B.11})$$

Therefore, to evaluate the integral of Equation (B.9) we have to sum the two contributions of Equations (B.10) and (B.11). We now operate the same change of variable as before with  $s' = s - s_V$ , and rewrite the integral as the sum of the two contributions:

$$\kappa_{TRACK,avg} = \int_0^{L_V} \frac{s'}{V} \Sigma_t e^{-(s'+s_V)\Sigma_t} ds' + \int_{L_V}^{\infty} \frac{L_V}{V} \Sigma_t e^{-(s'+s_V)\Sigma_t} ds' \quad (\text{B.12})$$

Taking the constants out of the integrals, we have:

$$\kappa_{TRACK,avg} = \frac{\Sigma_t}{V} e^{-s_V\Sigma_t} \int_0^{L_V} s' e^{-s'\Sigma_t} ds' + \frac{\Sigma_t L_V}{V} e^{-s_V\Sigma_t} \int_{L_V}^{\infty} e^{-s'\Sigma_t} ds' \quad (\text{B.13})$$

And finally, solving the integrals:

$$\kappa_{TRACK,avg} = \frac{\Sigma_t}{V} e^{-s_V\Sigma_t} \frac{1 - e^{-L_V\Sigma_t}(L_V\Sigma_t + 1)}{\Sigma_t^2} + \frac{\Sigma_t L_V}{V} e^{-s_V\Sigma_t} \frac{e^{-L_V\Sigma_t}}{\Sigma_t} = e^{-s_V\Sigma_t} \frac{1 - e^{-L_V\Sigma_t}}{V\Sigma_t} \quad (\text{B.14})$$

which is again the same result as Equation (B.6).

F

# ANNEX C: RÉSUMÉ SUBSTANTIEL EN FRANÇAIS

Avec ses 56 réacteurs nucléaires en exploitation, produisant 379 TWh d'électricité en 2019, la France est le deuxième producteur mondial d'énergie nucléaire. Si l'on regarde le pourcentage d'électricité provenant du nucléaire, la France est le premier pays au monde, avec plus de 70%. Le choix de s'appuyer massivement sur l'énergie d'origine nucléaire, et ce dès la transition commencée en 1974, a amené la France à avoir, aujourd'hui, un coût de l'électricité bien au-dessous de la moyenne européenne et surtout un réseau de production parmi les plus décarbonés au monde. Néanmoins, la majorité des réacteurs du parc français ayant été construite dans les années '80, l'Autorité de sûreté nucléaire se trouve confronter aujourd'hui à une situation nouvelle : quels réacteurs maintenir en fonctionnement et lesquels démanteler dans les années à venir. Il est probable qu'une partie significative des réacteurs du parc sera engagée dans la phase de démantèlement dans le courant des 5 ou 10 prochaines années. En 2019, 36 installations nucléaires de base (INB) étaient en cours de démantèlement en France. De manière générale, un nombre croissant de réacteurs voient leur fin de vie se rapprocher. Il est donc évident que les enjeux posés par le démantèlement des installations nucléaires sont de plus en plus d'actualité, tant en France qu'ailleurs dans le monde.

Dans une procédure de démantèlement, la caractérisation radiologique des composants et des lieux joue un rôle très important, tant dans une logique de sûreté des opérations à mettre en œuvre pour démanteler et nettoyer le site, que dans une problématique de gestion des déchets. Les mesures *in-situ* étant souvent compliquées à mener, en raison du risque radiologique associé et de la difficulté d'accès de certaines zones, l'emploi de méthodes numériques, s'appuyant sur des calculs de transport de particules et d'inventaire radiologique, s'avère un outil très efficace. Ces calculs sont traditionnellement menés en quatre étapes :

- Un **calcul de cœur** ayant pour objectif la détermination des sources de fission et du flux neutronique au sein du cœur du réacteur : il consiste à résoudre l'équation de Boltzmann encore dénommée ici équation du transport, à laquelle obéit le flux des neutrons qui se propagent dans le cœur ;
- Un **calcul de flux neutronique** au niveau de toutes les structures internes de la cuve du réacteur (voire de la cuve elle-même) radiologiquement activées pendant l'exploitation du réacteur : il consiste à résoudre l'équation du transport satisfait par le flux des neutrons et les gamma qui se propagent dans les structures du réacteur ;
- Un **calcul d'activation** visant à calculer l'activité des structures irradiées et les sources de rayonnements qui en découlent ; il consiste à résoudre les équations de Bateman généralisées, encore appelées équations d'évolution auxquelles satisfont les concentrations des radionucléides formés dans les structures.

- Un « **calcul de protection** » visant l'estimation des débits de dose induits par les rayonnements émis par les radionucléides formés dans les structures irradiées et le dimensionnement éventuel d'un blindage pour s'en protéger.

Cette thèse se focalise sur la dernière étape de ce schéma, c'est-à-dire le calcul de transport des **photons** pour estimer le débit de dose qu'ils induisent. Notamment, en partant des deux approches utilisées aujourd'hui pour mener ce type de calculs, à savoir l'**approche déterministe de traitement de l'équation du transport**, rapide mais approximative et donc peu précise dans certaines configurations, et l'**approche stochastique**, précise mais pouvant entraîner des temps de calculs très longs, on se propose de développer une **approche hybride**, combinant les points forts de ces deux approches classiques de manière à pallier leurs faiblesses respectives.

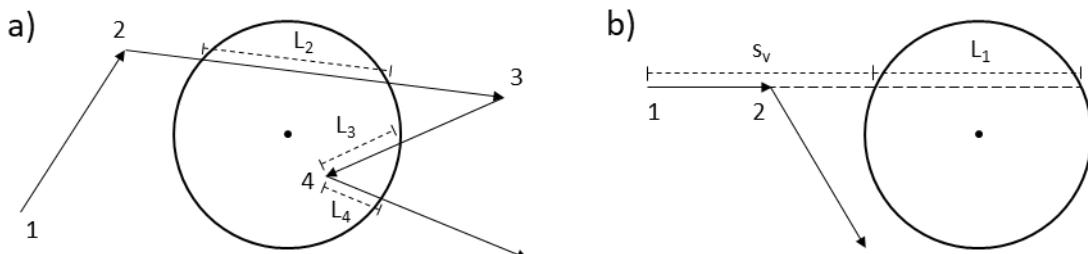
La voie choisie pour le développement de cette approche hybride est basée sur TRIPOLI-4®, un code stochastique Monte-Carlo de transport de particules, développé au CEA. En s'inspirant d'une étude menée dans le domaine de l'imagerie médicale, on a introduit dans TRIPOLI-4® des éléments du transport déterministe pour améliorer l'efficacité de calcul dans des configurations d'intérêt. On a donc développé un nouvel outil, appelé *exponential track-length estimator (e-TLE)*, « estimateur de longueur de trace (ou de corde) exponentiel » en français.

Cet estimateur est fondé sur le concept du *track-length estimator TLE*, « estimateur de longueur de trace (ou de corde) », déjà présent dans TRIPOLI-4® et dans la plupart des codes de transport Monte-Carlo. Cet estimateur est défini à partir de l'équivalence (C.1) qui existe entre le flux de particules et le parcours total des particules par unité de volume :

$$\Phi = \frac{L}{V} \quad (\text{C.1})$$

où  $L$  est le parcours total moyen traversé par les particules dans le volume  $V$ .

Suivant l'équation (C.1), toutes les particules qui croisent le volume  $V$  dans leurs trajets contribuent à l'estimation du flux, même celles qui n'ont pas d'interaction dans  $V$ . En revanche, un estimateur de type « collision » est défini à partir des interactions subies par une particule pour estimer des grandeurs physiques afférentes comme le flux. On peut ainsi décrire plusieurs estimateurs de longueur de trace, en changeant la façon dont la longueur  $L$  de l'équation (C.1) est calculée. La Figure C.48 présente une visualisation de l'estimateur TLE classique et de l'estimateur e-TLE développé dans cette thèse.



**Figure C.48 : visualisation 2D de l'estimateur de longueur de trace linéaire (a) et exponentiel (b)**

#### a) Estimateur de longueur de trace linéaire (TLE)

Cet estimateur est l'application en logique Monte-Carlo de l'équation (C.1). Par rapport à la Figure C.48a, on peut estimer le flux au point  $P$  comme une moyenne des parcours des particules traversant  $V$ , divisée par  $V$ . L'estimation du flux est donc :

$$\phi_j(P) = \begin{cases} \frac{L_j}{V}, & \text{si } L_j > 0 \\ 0, & \text{sinon} \end{cases} \quad (\text{C.2})$$

où  $L_j$  est la distance parcourue par la particule dans le volume  $V$  entre les interactions  $j$  et  $j + 1$ . Pour simplifier, on suppose ici que le volume  $V$  est constitué d'un matériau homogène. On peut définir  $L_j$  de façon plus rigoureuse comme :

$$L_j = \int_0^{|r_{j+1} - r_j|} \Pi_V(r_j + s \cdot \Omega_j) ds \quad (\text{C.3})$$

Dans cette formulation,  $\Pi_V(\mathbf{r})$  est une fonction caractéristique du volume  $V$ , égale à 1 si le point  $\mathbf{r}$  se trouve dans  $V$  et égale à 0 dans le cas contraire. En se reportant à la Figure C.48a, cet estimateur donne une contribution ou « score » de zéro pour les interactions 2, 3 et 4 ; en revanche, un estimateur de type « collision » aurait enregistré seulement l'interaction 4, qui est située dans le volume  $V$ .

#### b) Estimateur de longueur de trace exponentiel (e-TLE)

L'estimateur de longueur de trace exponentiel (Figure C.48b) ou *e*-TLE peut être dérivé directement de son correspondant linéaire en étendant la définition de la quantité  $L_j$ , comme le montrent les équations suivantes :

$$k_j(P) = \begin{cases} e^{-\Sigma_t s_V} \frac{(1 - e^{-\Sigma_t L_j})}{\Sigma_t V}, & \text{if } L_j > 0 \\ 0, & \text{if not} \end{cases} \quad (\text{C.4})$$

$$L_j = \int_0^\infty \Pi_V(r_j + s \cdot \Omega_j) ds \quad (\text{C.5})$$

où  $s_V$  est la distance entre l'interaction et le volume  $V$  dans la direction du vol de la particule, et  $\Sigma_t$  est la section efficace totale macroscopique du matériau traversé. Pour simplicité, le matériau est supposé homogène.

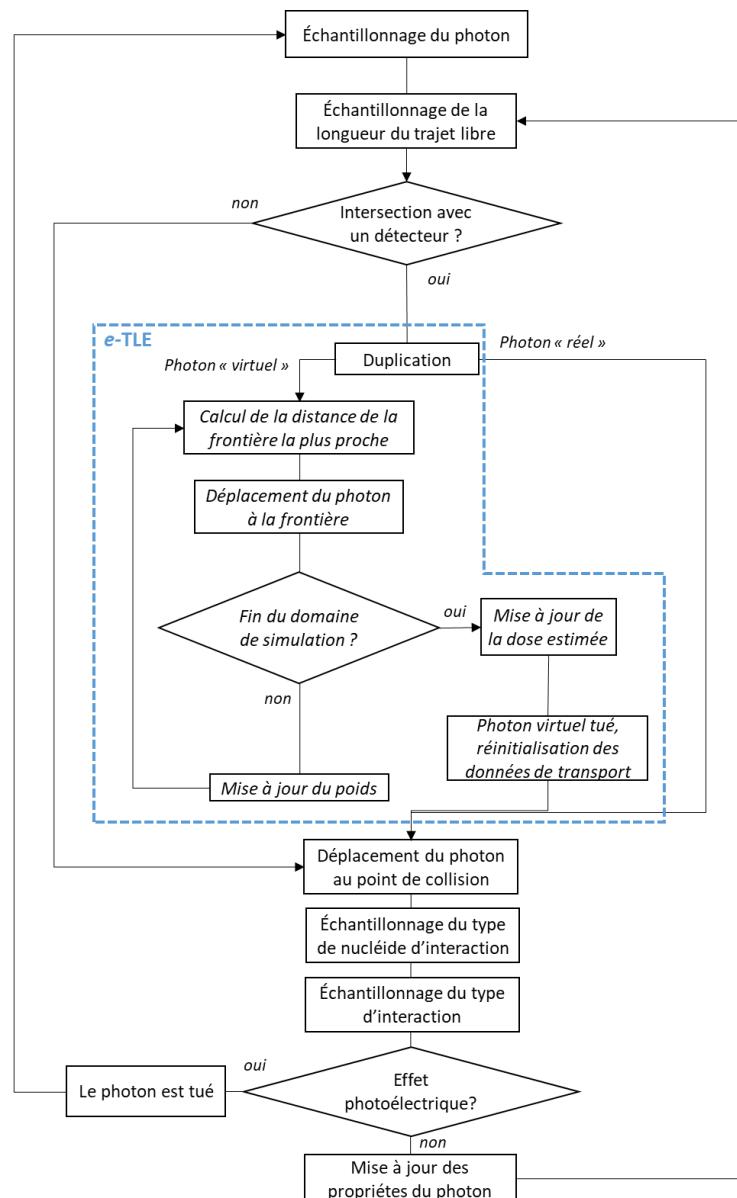
La dérivation mathématique de l'équation C.4 est donné dans l'annexe B. Ici, on se limitera à noter que la méthode consiste en une prolongation à l'infini du trajet de la particule après chaque interaction (d'où l'intégration de 0 à l'infini dans l'équation C.5) et d'une correction apportée du terme  $e^{-\Sigma_t s_V}$  qui prend en compte l'atténuation linéaire de la particule, due au fait que le transport en ligne droite n'est pas simulé en Monte Carlo. L'efficacité du processus est bien montré dans la Figure C.48b, qui montre que l'estimateur *e*-TLE apporte une contribution même si la particule ne traverse pas le volume d'intérêt.

On va maintenant expliquer brièvement comment ce processus a été traduit sous une forme algorithmique et implémenté dans le code TRIPOLI-4®. Comme l'estimation donnée par le *e*-TLE a lieu non pas sur le trajet effectif de la particule, mais bien sur son prolongement virtuel en ligne droite, un procédé doit être trouvé pour implémenter ce type d'estimateur. Pour résoudre ce problème, on crée une particule dite « virtuelle » ou « fictive » à laquelle on attribue les caractéristiques de la particule « réelle » (poids, énergie, direction de mouvement) et qui est transportée en ligne droite jusqu'à la fin du domaine de simulation. On résume ci-après les étapes principales de l'algorithme du *e*-TLE dans son implémentation au sein du code de transport Monte-Carlo TRIPOLI-4® :

1. Tout d'abord, un contrôle géométrique vérifie que la prolongation en ligne droite du trajet de la particule croise effectivement un des volumes de détection. Si ce n'est pas le cas, le transport déterministe de la particule ne donnerait aucune contribution, et de ce fait le processus n'est pas enclenché.

2. Si l'on sait que la particule va croiser au moins un détecteur dans la prolongation de son chemin, on commence par la création d'une particule « virtuelle » créée comme une copie exacte de la particule originale en toutes ses propriétés.
3. On calcule la distance  $d_i$  entre la particule virtuelle et la frontière du volume plus proche, *i.e.* la distance parcourue par la particule dans le  $i$ -ème volume.
4. La particule virtuelle est déplacée dans sa nouvelle position sur la frontière du volume ; une valeur de l'estimateur (ou « score »)  $\kappa_i = \frac{w_i * (1 - e^{-\Sigma_t d_i})}{\Sigma_t}$  est sauvegardée en mémoire pour le volume qui vient d'être parcouru. Ici,  $w_i$  est le poids de la particule virtuelle à l'entrée du  $i$ -ème volume et  $\Sigma_t$  est la section efficace macroscopique totale du milieu traversé.
5. Le poids de la particule virtuelle  $w_i$  est mis à jour pour tenir compte de l'atténuation :  $w_{i+1} = w_i \cdot e^{-\Sigma_t d_i}$ .
6. On retour au point 3 et on répète le processus jusqu'à quand la particule virtuelle n'arrive à la fin du domaine simulé.

Cet algorithme est schématisé sur la Figure C.49.



**Figure C.49 : Implémentation de l'algorithme relatif à l’“exponential track-length estimator”, e-TLE, pour le transport des photons dans le code de transport Monte-Carlo TRIPOLI-4®**

Après vérification de cet algorithme, une première version de l'estimateur a été testée et validée sur un ensemble de configurations d'étude reflétant des cas d'intérêt dans le contexte du démantèlement. Certaines de ces configurations représentent de vraies expériences de laboratoire, d'autres plus complexes sont purement théoriques et ont été conçues à la fois pour être représentatives de situations réelles, et pour être particulièrement difficiles à traiter par une approche de transport déterministe classique.

Sur l'ensemble de ces configurations, le *e-TLE* offre des résultats très satisfaisants. L'écart par rapport à l'estimateur TLE est inférieur à l'écart type de la simulation, ce qui montre la validité de l'estimateur. En terme de performances, les calculs réalisés avec l'*e-TLE* montrent une efficacité de 1.5 à 3 fois supérieure à celle des calculs usuels. L'efficacité est mesurée par un indicateur dénommé « figure de mérite » (FOM pour “*figure of merit*”) qui prend en compte conjointement le temps de calcul et la variance statistique associés aux résultats de la simulation.

Ayant donc vérifié la fiabilité et les bonnes performances du nouvel estimateur, la partie restante de ce travail de thèse a été dédiée à l'optimisation des performances de l'estimateur, en suivant deux voies principales :

- d'un côté, en se proposant d'optimiser l'*e-TLE* pour des évaluations de débits de dose bien localisées dans le domaine de simulation, telles qu'une mesure de dose effectuée à l'aide d'un dosimètre thermo-luminescent ;
- de l'autre, en optimisant l'estimateur pour des mesures de débits de dose distribuées sur tout le domaine de simulation, par exemple pour établir une cartographie de valeurs de débit de dose en soutien aux opérations de démantèlement.

Pour répondre aux problématiques de la première voie, on a implémenté un algorithme de « détection forcée » qui dirige automatiquement la trajectoire « fictive » de chaque particule vers un, ou plusieurs, volumes définis comme détecteurs. Cette modification permet d'augmenter significativement l'efficacité d'une simulation, en assurant que chaque particule fournit une contribution à chaque détecteur cible après chacune de ses interactions dans la matière. Le nouvel estimateur, appelé *e-TLE-FD* (FD pour “*forced detection*”, « détection forcée ») a été validé sur les mêmes configurations utilisées précédemment pour le *e-TLE*, en conduisant toujours un à très bon accord avec le TLE classique. L'estimateur montre ainsi des performances qui varient beaucoup en fonction de la configuration étudiée. Dans les meilleurs cas, l'*e-TLE-FD* atteint des performances 10 000 fois supérieures au TLE ; dans les autres configurations le gain en efficacité est compris entre 10 et 100. L'estimateur se montre en revanche moins adapté pour traiter des détecteurs qui sont très proches de la source de particules. On en conclut que ce type d'estimateur est bien adapté à la détection d'événements rares, avec un nombre de détecteurs pas trop élevé.

La deuxième voie d'implémentation vise à permettre l'estimation sur un large volume maillé, à l'aide d'un algorithme de « fractionnement » ou “*splitting*” après chaque interaction des particules. L'estimateur prend ici le nom de *se-TLE* (*s* pour “*split*”). Avec cet algorithme, lors d'une interaction, la particule réelle, d'un poids  $p_0$  donné est remplacée par un nombre  $M$  de particules fictives, de poids respectif  $p_0/M$ , ayant chacune une direction et une énergie différente, échantillonnée sur la base de la section efficace différentielle de l'interaction. Les particules fictives sont transportées et atténuées en suivant le même algorithme de transport déterministe que les autres versions du *e-TLE*. À la fin de la boucle de transport des particules fictives, le code reprend la simulation de la particule réelle.

Une première étude sur la multiplicité  $M$  a pu montrer qu'il n'est pas aisé d'évaluer l'efficacité générale de la méthode pour une multiplicité  $M$  quelconque. Aussi, une étude de validation et performance avec une multiplicité fixée à  $M = 5$  particules fictives montre que le *se-TLE* améliore les performances de calcul dans certaines zones du domaine de simulation, en la réduisant dans d'autres zones. On en conclut que,似ilairement au cas du *e-TLE-FD*, la méthode est

bien adaptée pour réduire la variance statistique dans les zones qui sont plus difficiles à atteindre par les particules (zones derrière une protection radiologique, zones loin de la source, événements rares) mais la simulation d'un nombre de particules additionnelles entraîne un ralentissement de la simulation qui diminue l'efficacité dans des zones proches de la source à plus haute statistique.

Les développements réalisés lors de ce doctorat ont permis de compléter des outils de calcul du code TRIPOLI-4®, et de montrer tout leur intérêt dans des configurations complexes d'assainissement-démantèlement. Des pistes de développement sont suggérées pour améliorer davantage les performances des nouveaux estimateurs ainsi que d'étendre leur domaine d'application, par exemple en traitant d'autres particules que des photons. Plusieurs publications ont résulté des travaux de recherche présentés dans le cadre de cette thèse :

- E. Guadagni, C. Le Loirec, D. Mancusi, “Comparison of variance-reduction techniques for gamma dose rate determination”, *The European Physical Journal Plus*, 136, 19, February 2021, 232.
- E. Guadagni, C. Le Loirec, Y. Pénéliau, J. M. Létang, D. Mancusi, C. Diop, “A new hybrid next-event estimator for photon-based Monte Carlo dose rate calculations”, *The European Physical Journal Plus*, <https://doi.org/10.1140/epjp/s13360-021-02120-5>, 136, 13 November 2021, 1135.
- E. Guadagni, Y. Pénéliau, J. M. Létang, F. X. Hugot, C. Le Loirec, “Split exponential track length estimator for Monte-Carlo simulations in reactor dismantling operations”, DEM 2021, Avignon, September 13 to 15, 2021.

