

Master's Thesis

**Simulating Long Neutron Beamlines Using Duct Source
Variance Reduction in GEANT4**

Isak SVENSSON
June 18, 2018

Supervisors
Douglas DI JULIO
Joakim CEDERKÄLL

Examiner
Bengt MARTINSSON



LUND UNIVERSITY

Faculty of Engineering
Department of Physics
Division of Nuclear Physics

Abstract

The European Spallation Source, ESS, is set to become the world's brightest spallation neutron source. Spallation neutron sources present unique challenges with regards to radiation shielding due to the creation of a significant number of high energy neutrons during nuclear reactions in the spallation target. These can reach up to the energy of the driving proton beam. Performing Monte Carlo simulations of deep-penetration shielding problems is highly CPU intensive, and with beamlines at ESS reaching up to 150 m in length, it is a necessity to develop methods to utilize the available computing power as efficiently as possible.

In this thesis a source biasing technique called a duct source is implemented in GEANT4. The technique reduces variance by uniformly generating particles along the length of the beamline and altering the particle weights to keep the simulation physically valid. In addition to angular biasing for anisotropic neutron sources, energy biasing is introduced in order to better study the high energy component of the spectrum.

Figures of neutron currents in beamline sections show that the duct source effortlessly transports neutrons arbitrary distances down a guide. A model problem consisting of a 77 m long beamline with multiple sections of shielding show that while analog simulations are effectively unable to populate areas outside the shielding, the duct source—particularly with energy biasing enabled—can provide neutron population in virtually the entire geometry within an acceptable time frame.

Additionally, the duct source is coupled with an algorithm for automatic weight window generation, and simulations show the techniques to be complementary, with the duct source pushing particles down the beamline and the weight windows pushing them laterally toward the edges of the geometry.

Preface and Acknowledgements

This project was carried out at the European Spallation Source in Lund, Sweden, between November, 2017 and June, 2018, with the intent of providing computationally efficient tools for accurate Monte Carlo simulation of long neutron beamlines. The author would like to express his gratitude to the on-site supervisor at ESS, Douglas Di Julio, who provided exemplary guidance and support throughout the project. Further thanks are extended to the university supervisor Joakim Cederkäll and to examiner Bengt Martinsson. Finally the author would like to acknowledge the generous funding received from the European Spallation Source for the completion of this project.

Contents

Preface and Acknowledgements	ii
1 Introduction	1
1.1 The Need for Neutrons	1
1.2 The European Spallation Source	2
1.3 Applications	3
1.4 Radiation Background Issues	3
1.5 Project and Goals	4
2 Physics Overview	5
2.1 Properties of the Neutron	5
2.2 The Spallation Process	6
2.2.1 The Case for Protons	8
2.2.2 Notes on the Accelerator	9
2.3 Moderation	11
2.4 Neutron Optics	12
2.5 Shielding	14
3 Monte Carlo Simulation	16
3.1 Overview	16
3.2 Relative Error	17
3.3 Variance Reduction	19
3.4 Source Biasing	20
3.5 Implicit Capture	20
3.6 Russian Roulette	20
3.7 Geometry Splitting	21
3.8 Weight Windows	22
3.9 . . . and Weight Window Generators	23
4 Implementation in GEANT4	25
4.1 About GEANT4	25
4.2 Structure of a GEANT4 Program	25
4.3 Setting Up the Simulation	26
4.4 Physics List	27
4.5 Neutron Generation	27
4.6 The Duct Source	27
4.7 Passing Through the Duct	29
4.8 Anisotropic Sources	30

4.9	Energy Selection	31
4.10	Problem Geometry	33
4.11	Weight Window Generator	36
5	Results and Discussion	37
5.1	Proof of Concept	37
5.2	Energy Biasing	41
5.3	Beamline Results	43
5.4	Figures-of-Merit and Empty Cells	49
6	Conclusions and Outlook	52

List of Abbreviations

ADS	Accelerator-Driven System
AR	Accumulator (or Compressor) Ring
CC-BY-3.0	Creative Commons Attribution 3.0 Unported
CPU	Central Processing Unit
CSNS	Chinese Spallation Neutron Source
ESS	European Spallation Source
FOM	Figure of Merit
Geant4	GEometry ANd Tracking
GVR	Global Variance Reduction
LH	Long History
LPSS	Long Pulse Spallation Source
LVR	Local Variance Reduction
MC	Monte Carlo
MCNP	Monte Carlo N-Particle Transport Code
MT	Multithreading / Multithreaded
PHITS	Particle and Heavy Ion Transport code System
PRNG	Pseudo-Random Number Generator
SINQ	Swiss Spallation Neutron Source
SNS	Spallation Neutron Source
SPSS	Short Pulse Spallation Source
ST	Singlethreading / Singlethreaded
ToF	Time of Flight
VR	Variance Reduction
WW	Weight Window
WWG	Weight Window Generator

List of Figures

2.1	The neutron and its constituent quarks.	5
2.2	Spallation neutron yield as a function of energy for a lead target, given per charge of the projectile. Figure reproduced from Noboru Watanabe [24]. © IOP Publishing. Reproduced with permission. All rights reserved.	7
2.3	The nuclear spallation process with possible interactions in the intra-nuclear cascade model. Fission only occurs in heavier target such as uranium. Figure reproduced from Noboru Watanabe [24]. © IOP Publishing. Reproduced with permission. All rights reserved.	9
2.4	Comparison between the proton range and the nuclear collision probability. The target material is tungsten with mass number $A = 183.84$. Range data acquired from Berger et al. [25]	10
2.5	Comparison of unmoderated spallation and fission spectra. Credit: N. V. Shetty, 2013 [30]	11
2.6	Effect of moderator temperature on the resulting neutron spectrum. Note that higher energies are shown to the left and lower energies to the right. Figure reproduced from Noboru Watanabe [24]. © IOP Publishing. Reproduced with permission. All rights reserved.	13
2.7	Moderated spectrum from the ESS target station at the entrance of the NMX beamline, normalized to unity.	14
3.1	Relative error vs number of trials with $\sigma = \bar{x} = 1$	18
3.2	A weight window cell.	23
4.1	Surfaces to be sampled, with $A = A_1 + A_2$	29
4.2	The DREAM beamline at ESS as pictured by Schweika et al. in the instrument's proposal [46]. Reprinted under the terms of the CC-BY-3.0 [55].	33
4.3	The simplified model of the ESS DREAM beamline used as a test case for the duct source generator. The coordinate axes are pictured as $x, y, z =$ red, green, blue. Each color in the figure represents a material. The iron floor and monolith is gray, the copper collimators (only one is visible from this angle) are brown, the heavy concrete bunker wall is blue, and the concrete beamline shielding and instrument cave is green. The bunker roof alternates between polyethylene (pink) and steel (black).	34

4.4	The model pictured from the left side. Both collimator blocks in the bunker are visible, as is the layered roof structure above them.	35
4.5	Same view as Figure 4.3 but with wireframes to reveal the inner structures. The instrument cave is visible on the righthand side.	35
4.6	Top wireframe view.	35
4.7	Right side wireframe view.	36
5.1	Graphical representation of the short beamline section with detector tiles in the beamline.	37
5.2	Histogram (normalized) of the wall current with the analog isotropic source. The function shown in red is proportional to $1/r^3$.	38
5.3	The beamline current (normalized) for the analog isotropic source. The function shown in green is proportional to $1/r^2$.	38
5.4	Duct source, isotropic case. Measured wall and beamline currents with functions proportional to $1/r^3$ (red) and $1/r^2$ (green) superimposed.	39
5.5	Wall current for the isotropic analog source (blue) and the isotropic duct source (yellow), unnormalized.	40
5.6	Wall current with the duct source extending from -5 to 0 m and 25 % of the particles passing through the cross section of the guide at $z = 0$ m.	40
5.7	Wall current for the anisotropic analog source (blue), the anisotropic duct source (yellow), and the isotropic duct source (red). Unnormalized.	41
5.8	Unweighted energy spectrum as used in the energy biasing technique.	42
5.9	The energy spectrum from Figure 5.8 with weights applied.	42
5.10	The spectrum from Figure 5.9 after filtering out particles that cross the duct wall within 10 cm of the source.	43
5.11	Flux map in the horizontal plane, analog simulation, anisotropic source, ~ 16 CPU hours.	44
5.12	Vertical plane flux map from same data as Figure 5.11.	44
5.13	Horizontal plane flux map produced using the anisotropic duct source without energy biasing, ~ 16 CPU hours.	45
5.14	Vertical plane flux map from the same data as Figure 5.13.	45
5.15	Horizontal plane flux map produced using the anisotropic duct source with energy biasing, ~ 16 CPU hours.	46
5.16	Vertical plane flux map from the same data as Figure 5.15.	46
5.17	Horizontal plane flux map produced using the anisotropic duct source with energy biasing and weight windows, ~ 16 CPU hours.	47
5.18	Vertical plane flux map from the same data as Figure 5.17.	47
5.19	The combined fluxes of Figures 5.15 and 5.17, ~ 32 CPU hours total.	48
5.20	The combined fluxes of Figures 5.16 and 5.18, ~ 32 CPU hours total.	48

List of Tables

2.1	Neutron classification by energy [18].	6
3.1	Guidelines for judging the precision of Monte Carlo simulations. From the MCNP manual [37].	19
5.1	Figures-of-Merit and empty cells for different runtimes.	50

Chapter 1

Introduction

1.1 The Need for Neutrons

If the neutron had not been discovered by Chadwick in 1932, it would have been invented.

Bertram Brockhouse, 1994 Nobel Prize laureate in physics [1]

There is no one-stop solution for probing the innards of objects. To the general public, looking inside of opaque objects is virtually synonymous with X-rays thanks to their long history of ubiquitous use in medicine. Far less well known is that neutrons are at least as useful as material probes. The main reason for this discrepancy in the public mind is that X-rays are much easier to produce and work with. Neutrons, being electrically neutral and weakly interacting, are difficult to control and manipulate. Even so, the first neutron radiograph was successfully produced as early as 1935 [2], just three years after the discovery of the neutron by James Chadwick [3]. A radiograph is a two-dimensional attenuation map of an object; an X-ray of a broken bone is an example of a radiograph.

Dimness is the Achilles' heel of neutron sources in general. Even the most powerful and sophisticated neutron sources are no brighter than a run-of-the-mill X-ray tube, and advanced facilities such as synchrotrons are many orders of magnitude brighter than their neutronic counterparts [2]. But despite the challenges, the inherent properties of neutrons as material probes justify the development of high brightness neutron sources.

For imaging purposes, X-rays and neutrons are largely complementary [4]. X-rays primarily interact with the electron clouds in materials and the primary factor that determines penetration depth is electron density. Thus X-rays experience more and more attenuation as the atomic number of the probed material goes up. Neutrons by contrast interact with the nuclei of the material, by a variety of mechanisms such as elastic scattering, inelastic scattering, and absorption. Overall, the highest scattering cross sections are found among the low- Z materials of the periodic table, with hydrogen as the most prominent example, leading to strong attenuation for hydrogenous materials. Scattering

cross sections vary not only with element but also with isotope, providing a mechanism by which isotopic composition can be studied or contrast can be enhanced. The neutron can also experience magnetic scattering because of its non-zero magnetic moment [2].

Thermal neutrons have energies of around 25 meV and wavelengths that are on the same order as the spacing of atoms in many common materials. This makes them attractive for use in diffraction studies of materials and objects [2].

High brightness neutron sources broadly come in two varieties, reactors and spallation sources. In a nuclear reactor neutrons are continuously released as a result of a nuclear chain reaction, sustained by nuclear fission. In a spallation source, a target, typically composed of some heavy metal, is bombarded by charged particles such as protons or deuterons, releasing a large number of neutrons in every impact. A “spall” is a fragment of material, hence the term “spallation” for the process.

1.2 The European Spallation Source

The European Spallation Source, ESS, is currently under construction in the fields on the outskirts of Lund, Sweden, where it will become the next door neighbour of the MAX IV synchrotron facility. When it becomes operational it will be the brightest spallation neutron source in the world, delivering higher peak and average fluxes than other neutron sources [5]. The first proton beam-on is set to take place in the early 2020’s, with a suite of fifteen neutron scattering instruments and one test beamline under construction. The user program is scheduled to begin in 2023. The facility will host 22 neutron scattering instruments in the future with the proton beam operating at a power of 5 MW average and 125 MW peak. The operational phase is planned to last until 2065, at which point the facility will be promptly decommissioned and dismantled [5].

ESS is a joint effort by 15 countries, around 40 in-kind partners, and over 130 world-wide institutions [6]. On May 28, 2009, the decision to place the facility in Lund was taken in Brussels, although the final decision to actually go ahead and build it was made at a later date [7]. Several European locations had previously been contenders for the honor of being chosen as the ESS site, including Debrecen in Hungary and Bilbao in Spain [5], which were the last remaining contestants. Lund finally secured overwhelming support, in part due to the area’s strong existing involvement in science and technology.

The original plans for a large European spallation facility, drawn up in 1995, specified the proton beam power as 10 MW [8], a figure that was eventually halved for technical and financial reasons. As of March, 2018, the budget of the project is €1843 million according to the ESS website [6].

As is befitting of a modern large-scale, multi-country collaborative science project, ESS is careful about its environmental impact. The project aims to be carbon neutral over all and has established a collaboration with an energy company to be wind-powered [8]. The area in which it is being built, formerly tranquil Brunnskög in north-eastern Lund, is currently in an era of intense expansion and activity as it becomes home to two major research facilities. The area will become an entirely new part of Lund in its own right, and may eventually become home to as many as 50,000 people [9]. To facilitate efficient and environmentally friendly transportation of people to and from Brunnskög,

the city of Lund is building a 5.5 km long tramway from Lund Central Station to ESS, the ambition being that at least two thirds of the visitors and inhabitants of Brunshög shall arrive by tramway, bike, or on foot [10].

When ESS reaches its end-of-life, 47 years from the time of this writing, the whole facility will be dismantled and the grounds on which it stands will be returned to the state they were in before construction began. This process is planned to take just five years, so that the experience of the employees who has worked there can be utilized. There will be no usage restrictions on the land afterwards [5].

1.3 Applications

ESS can be thought of as a microscope on steroids, revealing the structure and interaction of matter at a wide range of scales in both space and time. Its far-reaching capabilities will be utilized in many fields of research, including material sciences, non-destructive studies of historical artifacts, medicine, food science, life science, imaging, environmental studies, and fundamental physics [5]. Flexibility is a key feature and the instruments have been chosen following input from the extended neutron community to be eminently general-purpose. The initial fifteen instruments are divided into four main categories: diffraction, spectroscopy, large-scale structures, and engineering & imaging [11].

It is possible that ESS could answer a question that has confounded cosmologists for decades: why is there more matter than antimatter in the universe? It is not currently known whether or not the neutron has an electric dipole moment. If it does, it could provide a mechanism by which the charge asymmetry of the early universe arose, resulting in more matter than antimatter [5]. ESS, with its world-leading flux, is a prime candidate to settle this question. Should it turn out that the neutron does not have an electric dipole moment we at least know to look elsewhere.

1.4 Radiation Background Issues

Unique to spallation sources is the presence of very high energy neutrons in the source spectrum [12]. The highest-energy neutrons emanating from nuclear reactors typically show kinetic energies of less than 20 MeV and how to shield such neutrons is well understood [13]. At spallation sources, the neutron energies can run much higher, up to the energy of the proton beam driving the spallation process, which in ESS's case is 2.0 GeV. This radiation is highly penetrating and substantial shielding must be designed and employed for the safe and efficient operation of spallation sources. How to optimize the design of the shielding is a non-trivial task and the present work aims to provide one tool that can be employed in the solving of this puzzle. Shielding typically represents a significant fraction of the budget of a spallation source so figuring out how to get as much value as possible from the material is important [14].

The high energy neutrons can make their way to the instruments, either through the beamline, by scattering in the shielding, or scattering against the sky or ground. This introduces noise, reducing the usability of the instruments (see later chapters for a more detailed discussion about this problem.) They

also pose a biological risk to personnel at and the areas surrounding a spallation source if insufficiently shielded.

1.5 Project and Goals

Computer simulation is crucial tool in the development of efficient shielding, but simulating particle transport through dense material is a highly CPU-intensive task. Full simulation of a process (known as analog simulation) may therefore, depending on the nature of the problem, lead to execution times that are unacceptably long. This is definitely the case when simulating shielding of long neutron beamlines and as such it is of vital importance to employ one or several techniques for intelligently spending the CPU time in a manner that maximizes the value of the resulting data.

To illustrate the severity of the problem, consider a crude and simplistic model of ESS. In this model the target emit spallation neutrons isotropically in 4π . The flux of this source is proportional to $1/r^2$ where r is the distance from the surface of the source, and if we consider a straight beamline the neutron current crossing the beamline wall falls off proportional to $1/r^3$ [15]. Consequently this current, known as the wall current, is six orders of magnitude smaller at $r = 100$ m than at $r = 1$ m. Analog simulation of this situation would produce a huge number of particles crossing the beamline wall very close to the source, and almost no particles making it all the way down the beamline. Beamlines at ESS run up to 150 m in length, further aggravating the problem.

To alleviate this problem Niita et al. introduced a technique they dubbed the duct source [15]. The idea is to generate particles uniformly along the beamline walls and compensate the artificially high statistics further down the guide by proportionally lowering the particle weight. Their implementation was realized in a software package known as PHITS, a general-purpose Particle and Heavy Ion Transport code System [16]. The present work aims to extend and investigate this idea and to implement it in the free software C++-based particle simulation package GEANT4.

Chapter 2

Physics Overview

This chapter will give an overview of the neutrons' journey from source to detector, including moderators, shielding, and neutron optics. The problem of the high energy background at pulsed spallation sources is discussed.

2.1 Properties of the Neutron

Neutrons are not fundamental particles. Like protons, neutrons are hadrons, i.e. particles that are made up of quarks, see Figure 2.1. Both protons and neutrons are composed of up quarks (with electric charge $+2/3 e$) and down quarks (charge $-1/3 e$), with the proton consisting of two up quarks and one down quark and the neutron consisting of one up quark and two down quarks, resulting in electrical charges of $+1 e$ and 0 , respectively. Free neutrons undergo β -decay with a half-life of about 10 minutes [17]. This is in contrast with protons, which are extremely stable.

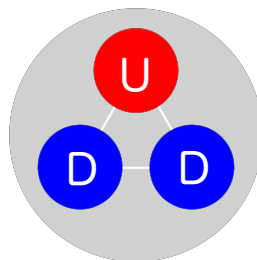


Figure 2.1: The neutron and its constituent quarks.

The primary role in nature of the neutron is to act as a kind of nuclear glue. Protons, being positively charged particles, repel each other via the electromagnetic force. This repulsion would lead to the disintegration of any multi-proton nucleus were it not for the stabilizing presence of neutrons, achieved through the strong nuclear force which acts attractively between nucleons. The strong force is repulsive at very short distances, keeping the nuclear density roughly constant for all nuclei [17].

The Coloumb repulsion will tend to overpower the nucleon-nucleon attraction as Z grows large. This is essentially because a nucleon acts only on its immediate neighbours via the strong force, whereas the Coloumb force between protons acts simultaneously on every proton in the nucleus [17]. The outcome of this is that heavier nuclei are rich in neutrons for the sake of stability.

Neutrons are divided into groups according to their kinetic energy. There is no one definitive way to apply this classification; Table 2.1 details one suggested classification, by L'Annunziata [18].

Energy	Designation
< 3 meV	Cold neutrons
3 - 400 meV	Thermal neutrons
0.4 - 100 eV	Epithermal neutrons
0.1 - 200 keV	Intermediate neutrons
0.2 - 10 MeV	Fast neutrons
> 10 MeV	High energy (relativistic) neutrons

Table 2.1: Neutron classification by energy [18].

Room-temperature neutrons have an energy of 25 meV, and these are frequently labeled thermal neutrons. A particle's de Broglie wavelength is defined as [17]

$$\lambda = \frac{h}{p} = \frac{h}{mv} \quad (2.1)$$

where h is Planck's constant, p is the momentum, m is the mass, and v is the velocity (for high energy neutrons it is necessary to use the relativistic mass.) The velocity v of a 25 meV neutron is 2200 m/s. Using this and the neutron rest mass of $1.675 \cdot 10^{-27}$ kg [17], Equation (2.1) yields the de Broglie wavelength for a 25 meV neutron as 0.18 nm (1.8 Å). This value is close to the interatomic distances in many materials; see for instance Linus Pauling's detailed 1947 article on interatomic distances in metals [19] and Allen et al. [20] for organic compounds. Neutrons at about thermal energies therefore undergo diffraction when passing through materials, and constructive interference results for neutrons that satisfy Bragg's law [21]

$$n\lambda = 2d \sin(\theta), \quad (2.2)$$

where n is a positive integer, λ is the wavelength, d is the interatomic spacing, and θ is the angle between the incoming neutron and the material lattice. ESS makes prominent use of this diffraction to probe material structures, with several beamlines dedicated to diffractometry.

Another attractive feature of thermal neutrons is that their kinetic energy is of the same order as the excitation and deexcitation energies in condensed matter systems [21].

2.2 The Spallation Process

The spallation process is driven by an energetic pulsed beam of protons impinging on a target of high- Z material, such as tungsten or mercury [21]. This process is endothermic [22] and cannot therefore be used as an energy source. However, it can be used in conjunction with a subcritical reactor to drive a fission process in what is known as an accelerator-driven system (ADS). This fission process can in turn generate net energy. ADSs can further be used to transmutate nuclear waste, rendering it inert or into a form that is easier to store safely.

Fission is intrinsically poor in terms of neutron yield, yielding only about 2.5 neutrons per fission event on average [23]. The yield per particle for spallation

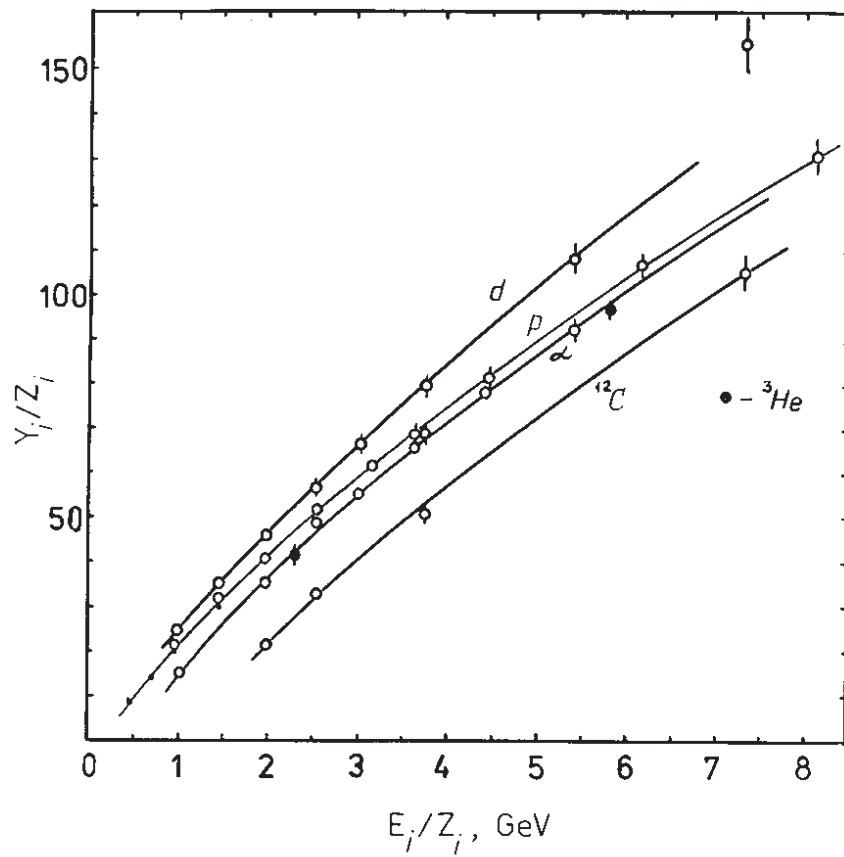


Figure 2.2: Spallation neutron yield as a function of energy for a lead target, given per charge of the projectile. Figure reproduced from Noboru Watanabe [24]. © IOP Publishing. Reproduced with permission. All rights reserved.

is much higher, as can be seen in Figure 2.2; about an order of magnitude for a 2 GeV proton projectile.

The neutron yield increases with the mass of the target nucleus [22], warranting high- Z materials. Uranium targets, with $Z = 92$, yield about twice as many neutrons as the lead target ($Z = 82$) shown in Figure 2.2, but is not used because of complications arising from fission and metallurgical considerations [22].

The target chosen by ESS is a rotating tungsten ($Z = 74$) wheel, 2.5 m in diameter, and helium-gas cooled; using helium as coolant is advantageous because it is inert [5]. The rotation helps distribute both the heat and the mechanical stress incurred by the proton beam. The target wheel's expected lifetime is five years, after which it will have to be replaced.

The proton's de Broglie wavelength is on the order of 10^{-16} m at \sim GeV energies, which is significantly smaller than the diameter of a heavy nucleus. For this reason the interaction between the projectile and the nucleus is treated as a series of collisions between the incident particle and the individual nucleons in the target nuclei [22]. This is called a cascade model. Each collision transfers kinetic energy to the nucleons, heating the nucleus. The struck nucleons may subsequently collide with other nucleons, causing an intra-nuclear cascade, wherein pion production and emission of high energy secondary hadrons occur [24]. These particles can, if sufficiently energetic, cause further intra-nuclear cascades in other nuclei; this is called an inter-nuclear cascade [22].

The intra-nuclear cascade leaves the nucleus in a highly excited state and it deexcites by isotropically evaporating lower-energy (less than 10 MeV) particles, most commonly neutrons [24]. The nuclear spallation process is depicted in Figure 2.3.

2.2.1 The Case for Protons

As Figure 2.2 shows, deuterons—the nuclei of deuterium (^2H)—are ideal for triggering spallation, cost notwithstanding. Protons emerge as the best overall option after also taking financial aspects into account [24].

A positively charged particle entering the target is far more likely to interact with an electron than a nucleus since the nuclei of the target material only account for a tiny fraction (about 10^{-15}) of the volume. Coulomb scattering against electrons is for this reason the dominant mechanism by which the particle loses energy in the material at lower energies [17]. The particle interacts with multiple electrons simultaneously, causing it to lose energy continuously until it—if the slab of material is thick enough—has no kinetic energy at all left. The distance travelled at this point is called the range of the particle and is a well-defined quantity that fluctuates marginally around a mean value [17]. The range is density-dependent and therefore frequently multiplied by the material density, yielding a quantity that is confusingly also known as the “range”.

It is important to note that the concept of range assumes that no hadronic nuclear interactions take place. This is a reasonable assumption at energies below about 100 MeV for a typical spallation target material such as lead [26]. At higher energies, however, the probability for nuclear interactions becomes significant, and consequently the range and the actual penetration depth start to diverge. For a given energy, the average distance travelled between nuclear collisions is called the mean free path and denoted λ . It is typically given

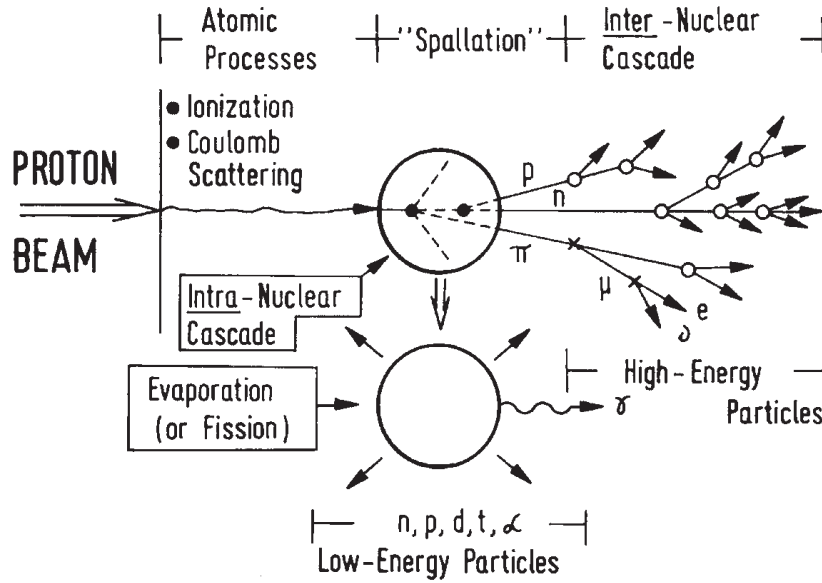


Figure 2.3: The nuclear spallation process with possible interactions in the intra-nuclear cascade model. Fission only occurs in heavier target such as uranium. Figure reproduced from Noboru Watanabe [24]. © IOP Publishing. Reproduced with permission. All rights reserved.

in units of $\text{g}\cdot\text{cm}^{-2}$, in a similar fashion as the range. The mean free path is approximately constant at proton energies above 100 MeV and is given by [26]

$$\lambda = 33 \cdot A^{1/3}, \quad A > 1 \quad (2.3)$$

where A is the mass number of the target material.

It follows from this discussion that the range is required to be greater than the mean free path for spallation to occur reliably. In mathematical terms, the probability of a nuclear collision is given by [24]

$$P_n = 1 - \exp(-R/\lambda), \quad (2.4)$$

so an interaction probability in excess of $> 95\%$ requires $R > 3\lambda$. A comparison between the range and the nuclear collision probability is given in Figure 2.4.

R is proportional to A/Z^2 [24], where A is the mass number, so it is disadvantageous to use particles with $Z > 1$. The deuteron's advantage over the proton stems from its higher mass number, but the difference is not large enough to warrant the increased cost.

2.2.2 Notes on the Accelerator

The job of accelerating the protons falls on the ESS Linac, which will be the world's most powerful linear accelerator. The distance from ion source to the center of the target is just over 600 m, with the actual accelerator accounting for 443 m [5]. It accelerates protons in pulses, with each pulse lasting 2.86

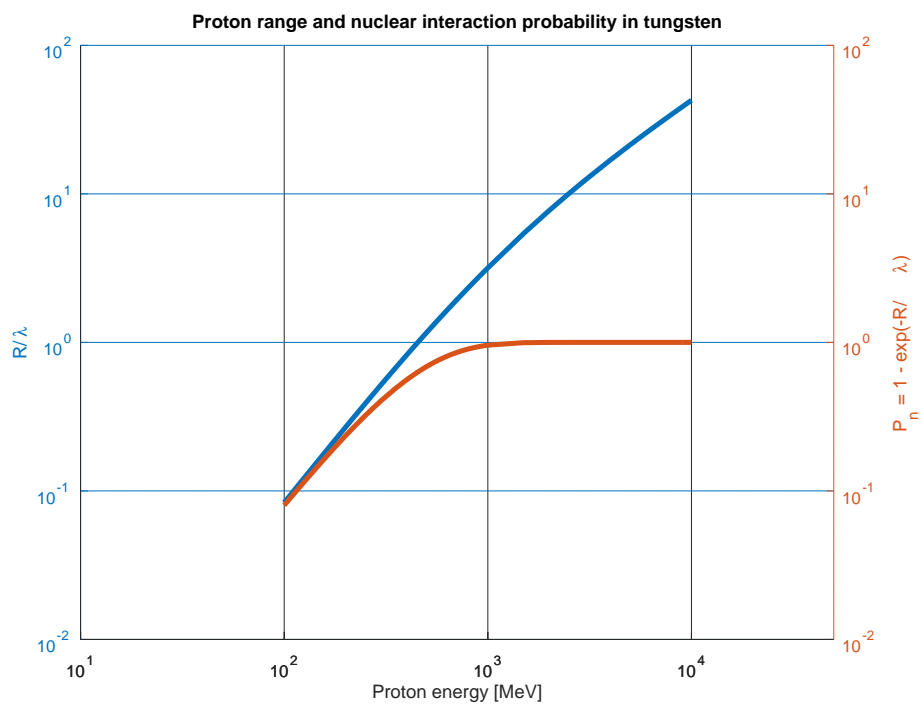


Figure 2.4: Comparison between the proton range and the nuclear collision probability. The target material is tungsten with mass number $A = 183.84$. Range data acquired from Berger et al. [25]

ms and a repetition rate of 14 Hz. The pulse duration is a standout feature of the European Spallation Source, which is the first ever so-called long pulse spallation source (LPSS) [27]. Previous spallation sources are of a type called short pulse spallation sources (SPSS), featuring pulse lengths on the order of μs . In an LPSS the proton pulse lasts considerably longer than the time it takes for the neutrons to thermalize in the moderator [22].

The LPSS concept simplifies things on the accelerator side, as it is easier to achieve high beam power with longer pulses. Thus a high-powered SPSS would require complex beam compression mechanisms in order to achieve μs pulses, a problem that is bypassed by directly utilizing the longer pulses [28]. In particular, unlike SPSSs, an LPSS does not require a so called compressor ring [24]. A compressor ring, also known as an accumulator ring, is a very substantial piece of equipment; for example, the AR at the Spallation Neutron Source (SNS) in Oak Ridge, Tennessee, has a circumference of 221 m [29]. It takes ms-long pulses of H^- ions as input, strips the ions of their electrons, and outputs 0.5 μs pulses of protons.

2.3 Moderation

Figure 2.5 shows a comparison between unmoderated spallation and fission spectra. The spallation spectrum is for a 600 MeV proton impinging on a tungsten target. The ESS unmoderated source spectrum is even harder since the proton energy is 2.0 GeV in that case. The neutrons that are useful for scattering ex-

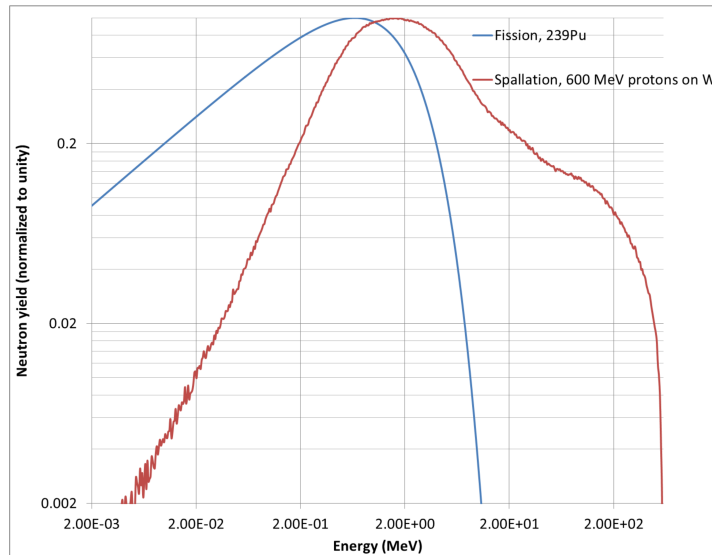


Figure 2.5: Comparison of unmoderated spallation and fission spectra. Credit: N. V. Shetty, 2013 [30]

periments are in the cold and thermal energy ranges, far lower than the “naked” spectrum. The neutrons coming out of the target therefore have to be moderated, i.e. slowed down.

A good moderator decreases the neutron energy without affecting the source brightness. Thus, roughly speaking a good moderator needs to have a high scattering cross-section and low absorption cross-section. Hydrogen, in pure form or in a mixture with high hydrogen content, is an obvious option since it has the highest elastic scattering cross-section for neutrons. It takes, on average, just 14.5 neutron-hydrogen collisions to slow a 2 MeV neutron down to 1 eV [22]. Deuterium is also an option with the advantage that the absorption cross-section for thermal neutrons is lower than for hydrogen, since it already has a neutron.

The best moderator option for an LPSS is para-H₂ [27], one of the two possible so called spin isomers of molecular hydrogen. In para-H₂, one proton is aligned spin-up and the other spin-down. The other spin isomer, where the spins are parallel, is called ortho-H₂. The low energy scattering cross-section differs between the two isomers, with increased mean free path for thermal and, in particular, cold neutrons in para-H₂; this affects directionality of the emerging neutrons [27]. It is essential that the moderator is surrounded by a reflector in order to maximize brightness [27], typically composed of beryllium [21]. Openings in the reflector allow for beam extraction.

Of crucial importance to the resulting spectrum is the temperature of the moderator. The emerging neutron energy spectrum roughly follows a Maxwellian distribution, centered around the temperature of the moderator. This is demonstrated in Figure 2.6 [24] for a methane moderator. To enhance the flux of cold neutrons it is therefore necessary to keep the moderators at low temperatures.

Figure 2.7 shows the neutron spectrum emerging from the moderator assembly at ESS. This figure was produced in GEANT4 using data adapted from an MCNP (Monte Carlo N-Particle Transport Code) input file, which serves as part of the input data for the simulations performed for this thesis¹. MCNP is a widely used proprietary software package for Monte Carlo simulation of neutron, photon, and electron transport problems [31], developed by the Los Alamos National Laboratory.

The source data was calculated for the NMX beamline, a long beamline that is dedicated to crystallography. The data was calculated at the entrance of the beamline, 2 m from the moderator surface. Note that a significant high energy component remains as part of the spectrum.

2.4 Neutron Optics

As with rays of light, neutron beams can be bent and aimed at a certain spot in order to maximize the beam brilliance at the instrument station [14], although the efficiency of neutron optics is lower than their photonic counterpart.

For light, the refractive index n for a medium is defined as

$$n = \frac{c}{v} \quad (2.5)$$

where c is the speed of light in vacuum and v is the speed of light in the medium. As $c \geq v$, it holds that $n \geq 1$ under normal circumstances (although there exist exotic possibilities for $n < 1$ [32].) As per Snell's law, this can lead to the phenomenon of total internal reflection [33]. This means that, when the angle

¹I would like to thank Valentina Santoro of the Neutron Shielding and Optics Group at ESS for supplying the MCNP source data.

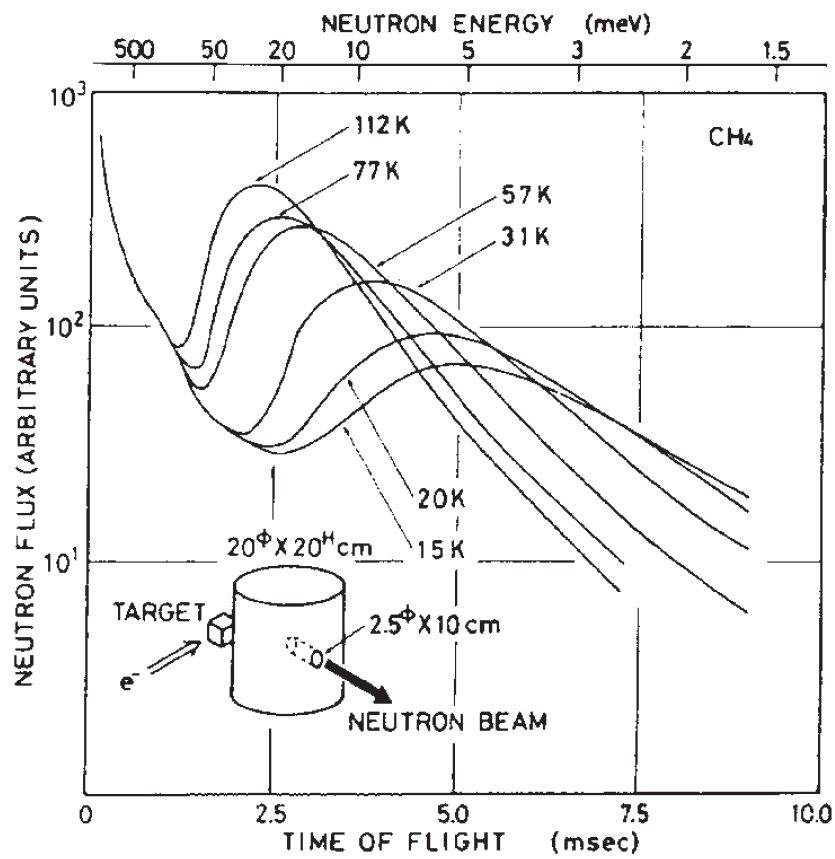


Figure 2.6: Effect of moderator temperature on the resulting neutron spectrum. Note that higher energies are shown to the left and lower energies to the right. Figure reproduced from Noboru Watanabe [24]. © IOP Publishing. Reproduced with permission. All rights reserved.

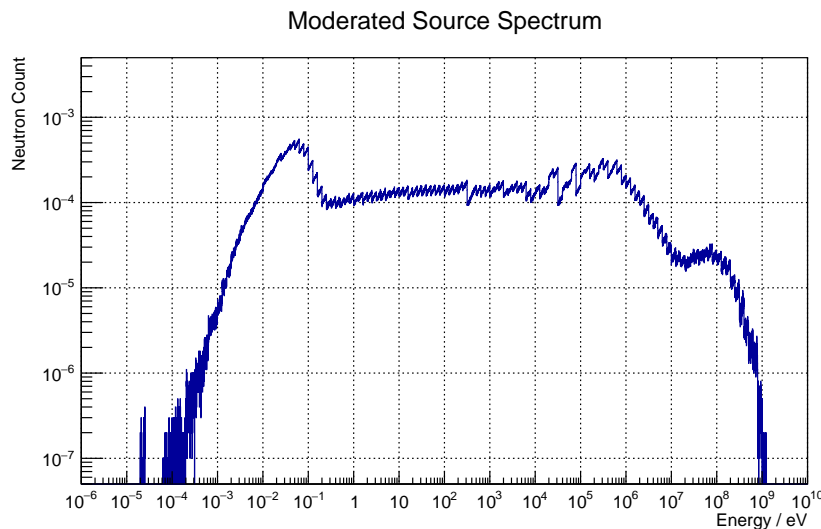


Figure 2.7: Moderated spectrum from the ESS target station at the entrance of the NMX beamline, normalized to unity.

of incidence is small enough, a ray of light coming from the denser medium is completely reflected against the interface between regions of higher and lower refractive indexes. This is the basis of fiber optic cables, which can transmit optical signals over vast distances [33].

Neutrons similarly have a refractive index given by

$$n = 1 - \lambda^2 \frac{Nb}{2\pi} + i\lambda \frac{N\sigma_a}{4\pi}, \quad (2.6)$$

where λ is the neutron wavelength, σ_a is the absorption cross-section, N is the material's atomic number density, and Nb is called the neutron scattering length density [21]. For cold neutrons n is typically smaller than unity in most materials, leading to total reflection below a critical angle when the neutrons coming from air or vacuum encounter a dense material. Using this phenomenon it is possible to construct neutron guides, a kind of neutronic counterpart to the fiber optic cable. Furthermore, using Equation (2.2) (Bragg's law), it is possible to construct layered materials that increase the critical angle for cold and thermal neutrons [21]. This is called a neutron supermirror.

2.5 Shielding

The high energy hadrons causing the inter-nuclear cascade discussed in Section 2.2 have the unfortunate side-effect of forming an intense pulse of high energy neutrons that is very difficult to eliminate. At a pulsed spallation source it is called a prompt pulse because the extremely high velocity of the particles means that the time-of-flight (ToF), as measured by the detector setup, is practically zero [13]. The velocity of a 2.0 GeV neutron is $0.95c$, five orders of magnitude higher than a thermal neutron. Eliminating the high energy

background is important not only for safety but also because it decreases the signal-to-noise ratio, S/N, in the scientific instruments, degrading the results.

The prompt neutrons have the ability to make their way to the detectors by a variety of paths. One is to penetrate the shielding, possibly passing through without interaction, or scattering against the nuclei of the shielding. The bulk shielding around the target must be sufficient to reduce this radiation to an acceptable level. The amount of shielding necessary is substantial; typically, about 10 % of \sim GeV neutrons make it through 1 m of shielding [14]. Common shielding materials include steel/iron and concrete.

The emission of high energy radiation in the spallation process is also strongly angle dependent, peaking around the direction of the proton beam [22]. Thus no instruments at ESS are placed in the area directly behind the target, as seen from the accelerator. Shorter beamlines are also not placed in the vicinity of the beam direction for this reason.

A second possibility is for the high energy neutrons to travel down the beamline. This is a tricky problem to solve, as the beamline obviously must let neutrons pass. Two strategies for dealing with this are to build curved beamlines and to place choppers in the beamline path. The idea behind the curved beamline is to use neutron optics (see Section 2.4) to bend the beam of useful neutrons toward the detector area whilst the high energy neutrons continue along a straight path, into the shielding. This process causes degradation of the neutron signal with curvature, and its use is a matter of optimization—it is not appropriate in every situation.

A chopper is a kind of rotating shutter that is synchronized with the proton beam. A significant reduction of prompt neutrons can be achieved by keeping the chopper closed at the time of the pulse and subsequently opening it when the cold and thermal neutrons arrive.

A third way the prompt neutrons can reach the detector area is a phenomenon known as skyshine. This occurs when an energetic neutron passes through the shielding, completely escaping the facility, and subsequently scatters against the outside air and makes its way back into the detector area. Particles that scatter against the ground are analogously called groundshine.

High energy particles escaping the primary shielding around the target have the ability to create secondary particle cascades anywhere they go, such as in the shielding near the detector area, and have a tendency to travel down gaps and through shielding weak points [14]. Such weaknesses may be difficult to identify and as such it is of great interest to perform accurate and efficient shielding simulations before the fact.

Chapter 3

Monte Carlo Simulation

This chapter will give an overview of the Monte Carlo method and describe how to quantify the errors inherent in the method. The concept of variance reduction (and why it is necessary) is introduced, and several techniques for reducing the variance are presented.

3.1 Overview

Monte Carlo simulation is the process of sampling a large number of stochastic events in order to obtain an average behaviour [34]. For example, determining beforehand where a particle of a certain energy and direction of travel will end up after having scattered through dense material is pretty much impossible as there are simply too many variables involved. In principle, the Monte Carlo method solves this problem rather elegantly by simulating the event several times. The resulting distribution of particles is, if the simulation is done in a physically valid way, what can be expected to happen should the experiment be conducted in real life.

The history of the Monte Carlo method stretches deep into the past, and well pre-dates the computer. But it was the development of the modern computer in the 1940's that sparked the ripening of the method into a tool with extensive applications in particle transport problems and other areas. It's most prominent pioneers were Stanislaw Ulam, Enrico Fermi, and John von Neumann [35].

Direct simulation of the full physical situation is called *analog* simulation. Implementation of an analog simulation is straightforward and dependable, but for many problems the CPU-time required to achieve accurate results with analog simulation is prohibitively long [35]. As an example of this problem, consider the Swiss Spallation Neutron Source, SINQ. The target station is surrounded by a shielding "monolith" consisting of 4.5 m of steel followed by 30 cm of boron-infused concrete [14]. Here the neutron intensity outside the shielding can be as much as 15 orders of magnitude lower than the beam intensity [14], rendering analog simulation entirely unfeasible. Similar situations occur for many geometries. The problem is almost intrinsic to the nature of shielding simulations, as containing radiation is the actual purpose of shielding in the first place. Thus evening out the particle fluxes across the geometry, or parts of it, is central to the success of MC simulations. Variance reduction (see Section 3.3) works to

make these kinds of problems tractable.

A natural consequence of the basic premise of the Monte Carlo method is that a large number of samples, or trials, need to be made in order to achieve a meaningful result. Therefore MC simulation is necessarily CPU-intensive.

3.2 Relative Error

The relative error is used as a measure of the precision of the simulation, and by extension how useful the results are. It is defined in this context as [36]

$$Re = \frac{\sigma_{\bar{x}}}{|\bar{x}|} \quad (3.1)$$

where $\sigma_{\bar{x}}$ is the standard deviation of the sample mean (i.e. a measure of the variation of the mean) and \bar{x} is the average score. If the number of trials is N then \bar{x} is obtained as

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i. \quad (3.2)$$

where x_i is the i th score. The x_i 's are independent and identically distributed.

The real expectation value $\langle x \rangle$ of a tally corresponding to some physical quantity can't be calculated directly and is approximated by the mean value \bar{x} . According to the strong law of large numbers it holds that $\bar{x} \rightarrow \langle x \rangle$ as the number of trials $N \rightarrow \infty$ [36].

The standard deviation of the sample mean $\sigma_{\bar{x}}$ can be calculated from the standard deviation σ of the samples themselves using the properties of variance. The sample variance is given by

$$\sigma^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2 \quad (3.3)$$

and can be calculated directly. Using Equations (3.2) and (3.3) and basic properties of variance it is possible to calculate $\sigma_{\bar{x}}^2$ as

$$\begin{aligned} \sigma_{\bar{x}}^2 &= \text{Var}(\bar{x}) = \text{Var}\left(\frac{x_1 + x_2 + \dots + x_N}{N}\right) \\ &= \text{Var}\left(\frac{x_1}{N} + \frac{x_2}{N} + \dots + \frac{x_N}{N}\right) \\ &= \frac{1}{N^2} \text{Var}(x_1) + \frac{1}{N^2} \text{Var}(x_2) + \dots + \frac{1}{N^2} \text{Var}(x_N) \\ &= \frac{1}{N^2} \left(\text{Var}(x_1) + \text{Var}(x_2) + \dots + \text{Var}(x_N) \right), \end{aligned}$$

and since every x_i has the same variance $\text{Var}(x_i) = \sigma^2$

$$\sigma_{\bar{x}}^2 = \frac{1}{N^2} (N \cdot \sigma^2) = \frac{1}{N} \sigma^2. \quad (3.4)$$

Thus

$$\sigma_{\bar{x}} = \frac{\sigma}{\sqrt{N}} \quad (3.5)$$

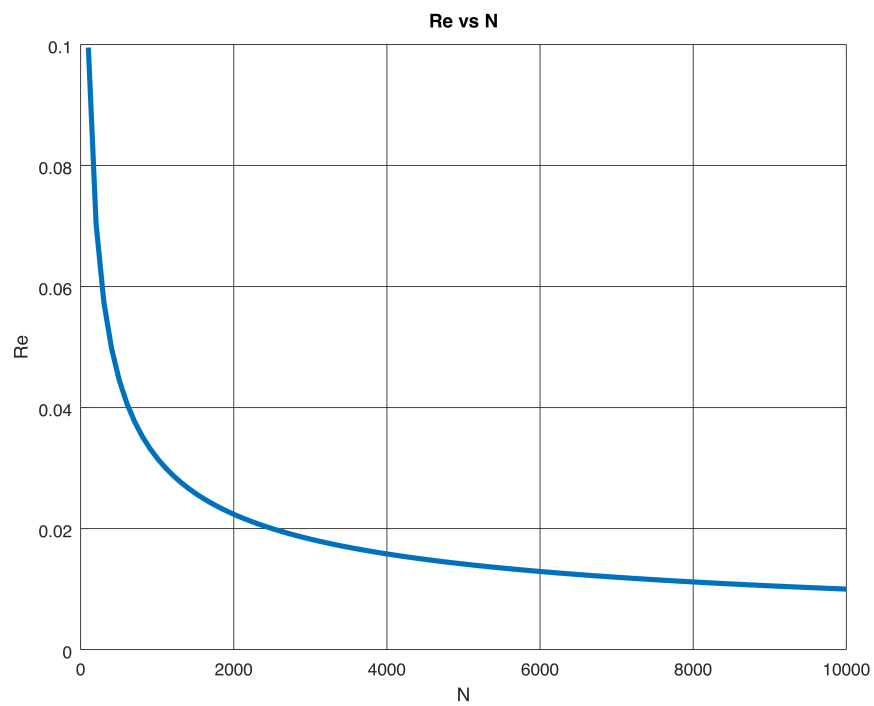


Figure 3.1: Relative error vs number of trials with $\sigma = |\bar{x}| = 1$.

and consequently

$$Re \propto \frac{1}{\sqrt{N}}. \quad (3.6)$$

Figure 3.1 shows a plot of Equation (3.1) for the case $\sigma = |\bar{x}| = 1$. The conclusion is that the relative error decreases slowly with increasing N . Achieving a ten-fold improvement in the relative error requires a hundred-fold increase in N . This is the root cause of why Monte Carlo simulation is CPU-intensive [36]. Table 3.1 presents guidelines for assessing simulation precision.

It is important to note that a high precision doesn't necessarily mean that the simulation is physically valid or accurate. The precision merely measures the uncertainty of \bar{x} . But systematic errors can cause \bar{x} to deviate from the true physical quantity that is sought, due to an incorrect model, flaws in the simulation software, or user error [37]. Lack of convergence in simulations may also cause loss of accuracy, e.g. a rare event carrying very high statistical weight could "dislodge" \bar{x} .

Relative error	Assessment
0.5 - 1	Garbage
0.2 - 0.5	Factor of a few
0.1 - 0.2	Questionable
0.05 - 0.1	Generally reliable except for point detectors
< 0.05	Generally reliable for point detectors

Table 3.1: Guidelines for judging the precision of Monte Carlo simulations. From the MCNP manual [37].

3.3 Variance Reduction

As Figure 3.1 shows, efforts to reduce Re by increasing the number of trials quickly suffer diminishing returns. Solving computationally difficult problems therefore require a more sophisticated approach to variance reduction. This essentially entails "smearing" the relative error over the region of interest, improving the relative error in areas of poor statistics at the expense of some loss of precision in areas of high statistics. Of course, the variance reduction must not lead to changes in the expected result of the simulation—it must remain physically valid.

Variance reduction can be performed locally, i.e. in a delimited section of the geometry, or globally across the entire geometry. The two types are abbreviated as LVR and GVR, respectively. LVR problems also are frequently called source-detector problems. The best choice depends on the type of problem and the sought quantity. A GVR approach is often preferred for deep-penetration shielding problems at spallation sources as it is necessary to know the flux levels throughout the model [38].

Variance reduction has been in use throughout the history of Monte Carlo simulation itself, and over the years a multitude of techniques have been devised. Which technique, or combination of techniques, to choose—if any—for a given problem is not always a trivial decision as it depends on the type of problem [34].

3.4 Source Biasing

An intuitive way to reduce variance is to increase sampling of events that are relatively improbable but important in the context of the problem. This can be done with regard to position, direction of travel, and energy. In the spatial dimension, this can be thought of as artificially “aiming” the source at the areas where increased statistics are desired. The biased particle weights are adjusted to avoid skewing the result, with the governing formula given by [34]

$$w_b \cdot p_b = w_0 \cdot p_0 \quad (3.7)$$

where w_0 and p_0 refer to the unbiased weight and sampling probability, and w_b and p_b refer to the biased dittos. Source biasing is strongly problem-dependent and using it successfully requires a good understanding of the problem at hand. A detailed account of a source biasing implementation for long neutron beam-lines is given in Chapter 4.

3.5 Implicit Capture

Possibly the simplest VR technique is implicit capture. In each collision between the particle and the material there exist a possibility, P_a , of the particle being absorbed. In the analog case it is simply absorbed and the simulation continues with a new particle. But when implicit capture is employed, the particle is instead allowed to continue with a reduced statistical weight. The new weight is calculated as [35, 39]

$$w_{new} = (1 - P_a) \cdot w_{old} \quad (3.8)$$

where w_{old} and w_{new} are the weights before and after the collision, respectively. This allows the particle to contribute to tallies in regions well beyond the area where it would be absorbed in reality. The technique has proven simple, dependable, and successful enough to be enabled by default in the MCNP family of codes [39]. This is not the case in GEANT4, however [38].

3.6 Russian Roulette

Over time, repeated application of implicit capture (or other VR techniques) may lead to very low statistical weight for the Monte Carlo particle. At some point this leads to wasteful use of CPU resources as the particle that is being followed contributes very little useful information to the final tally. A game of russian roulette was devised to deal with this problem [39].

Two options for action are available when the weight of an MC particle falls below a threshold value, w_{th} , for which it is decided that it is no longer meaningful to continue tracking the particle. They are

- (a) kill the particle with probability P_{kill} , and
- (b) continue tracking the particle with increased weight.

The increased weight in option (b) compensates for the weight of the particles that are killed when option (a) is chosen. This is necessary in order to keep the final tally unbiased.

There are two alternative implementations as to how the game is played. One is to fix the probability of survival

$$P_{survive} = 1 - P_{kill} \quad (3.9)$$

to a specified value and adjust the weight of survivors according to [35]

$$w_{new} = \frac{w}{P_{survive}} \quad (3.10)$$

where $w < w_{th}$ is the weight of the particle before the game. Alternatively, $w_{new} > w_{th}$ can be fixed and the survival probability calculated as

$$P_{survive} = \frac{w}{w_{new}}. \quad (3.11)$$

In both cases the fixed variable can be set to different values depending on the importance of the region in which the game is played [35]. It is worth noting that the russian roulette technique decreases CPU utilization at the expense of increased variance.

3.7 Geometry Splitting

As opposed to russian roulette, which deals with low-weight particles, geometry splitting takes higher-weight particles and splits them into several lower-weight particles. The process takes place on the surfaces between regions of disparate assigned importances and allows particles that happen to enter regions of high importance to contribute more statistics than they would in the analog case. If the weight of the original particle is w , and the number of fragments is n , then the weight of each fragment after the splitting is w/n . The direction of travel is decisive: travellers from regions of high importance crossing into regions of low importance play a game of russian roulette rather than splitting [35].

In practice the geometry is divided into cells with each cell having its own importance I (as opposed to defining unique importance values for each and every surface.) As described by Lux and Koblinger [35], the splitting can be performed as follows. Let the weight of each fragment be

$$w' = \frac{w}{n} = \frac{1}{I} \quad (3.12)$$

and a the fraction between the weight of the original particle and the weight of the fragments, i.e.

$$a = \frac{w}{w'}. \quad (3.13)$$

Three possibilities emerge:

1. a is an integer. The original particle is split into $n = a$ fragments (if $a = 1$ nothing happens.)
2. a is a real number greater than 1, i.e. $a = n + v$ with $n \geq 1$ and $0 < v < 1$. The particle is split into either $n + 1$ fragments with probability v , or n fragments with probability $1 - v$.

3. a is a real number smaller than 1, i.e. $0 < a < 1$. Rather than splitting, the incoming particle plays russian roulette with the survival probability a .

Geometry splitting can be an efficient method for performing variance reduction. The main hurdle for its use is that setting up the importance values requires significant skill and effort by the user, and optimization of them proves elusive in many realistic geometries [35].

3.8 Weight Windows ...

Variance reduction by weight windows is conceptually similar to the geometry splitting described in Section 3.7, but allows a range of weights for the fragments in contrast to the geometry splitting requirement that every particle crossing the cell surface must split or rouletted into fragments of a specified weight. The idea of a weight window cell is to make sure that the weight of every particle that dwells in the cell must be between an upper and a lower bound, i.e. $w_L \leq w \leq w_U$ where w is the weight of the particle.

If the initial particle weight is above w_U then it is split until the fragments' weights are inside the window. Conversely, if $w < w_L$ then a game of roulette is played (see Section 3.6). In the case of roulette the survival probability is calculated using a fixed survival weight w_S , as described by Equation (3.11) (with $w_{new} = w_S$). Obviously it must hold that $w_L < w_S < w_U$. Nothing happens in case the weight of the incoming particle is already within the weight window. This is analogous to the case where $a = 1$ in geometry splitting. A conceptual weight window cell with its parameters is depicted in Figure 3.2.

Some general recommendations regarding how to set the window bounds and the survival weight have been in use ever since the weight window method was devised for MCNP by Thomas E. Booth in the late 1970's [40]. Let

$$C_U = \frac{w_U}{w_L} \tag{3.14}$$

$$C_S = \frac{w_S}{w_L}, \tag{3.15}$$

i.e. w_U is C_U times greater than w_L etc. Then it is generally recommended that $C_U = 5$ and $C_S = 3$, although the success of the simulation is relatively insensitive to the exact values of these parameters [40]. C_U should never be set below 2 however, as this would result in particles with initial weights above w_U being split into fragments of weights below w_L , immediately forcing them to play roulette.

Although the weight window method has proven very successful in the many years since its conception, it does have a rather well-documented problem with excessive execution times caused by "oversplitting" in some circumstances [40, 41]. A high-weight particle encountering a cell with a much lower w_U will have to split a large number of times before it fits within the weight window, causing heavy CPU utilization as the program has to track each fragment. The number of MC particles can increase by several orders of magnitude in extreme cases [40]. This leads to so called long histories (LH), where the computer spends hours, days, or even weeks tracking fragments that all originate with a single initial particle, with little value added to the overall result.

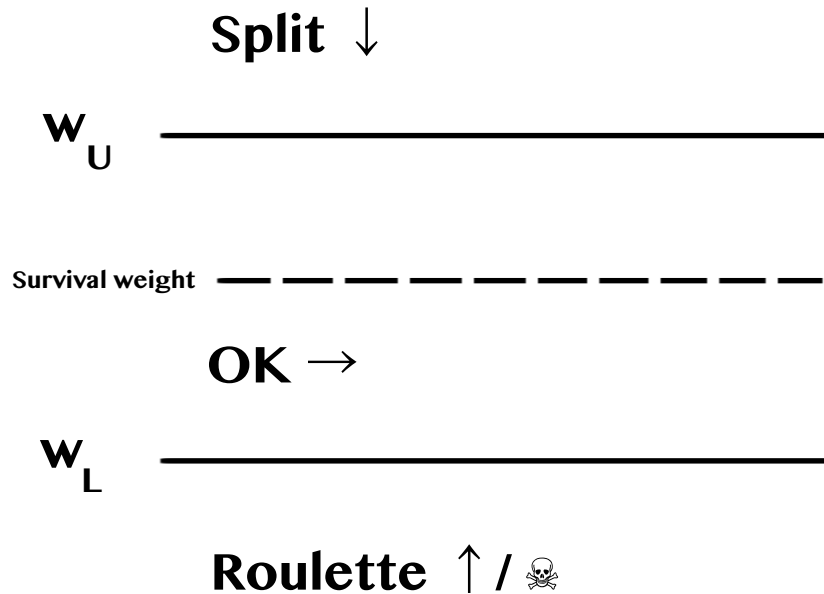


Figure 3.2: A weight window cell.

As part of the general recommendations by Booth, the maximum number of fragments a particle is allowed to split into is 5. This helps mitigate the oversplitting problem in some cases, but hardly helps at all in many other circumstances [40]. Each fragment is itself allowed to split 5 times, and so on; the splitting can be set to take place on the cell boundary or on collision with particles in the material, or both. The phenomenon does not seem to be caused by the weight window method per se, but by particle splitting [41, 42].

This can be a serious problem in MCNP as the simulation cannot be terminated prematurely [41]; the LH must be allowed to terminate naturally. No tally results are saved if the simulation is forcefully aborted. Some methods to combat this problem have been developed, such as dynamically adjusting the weight windows [41]. GEANT4 allows the user to enforce user limits, such as limiting the time an event is allowed to take.

3.9 ... and Weight Window Generators

The previous section made no mention of the precise value of the weight window bounds, notably how to set w_L , which the others are set in relation to. It is possible—if laborious—to set the weights by hand [38, 40]. A much more sophisticated and user-friendly method to automatically set the WW cells for GVR problems was devised by Cooper and Larsen [43], developed for MCNP by van Wijk et al. [39], and implemented in GEANT4 by Stenander and DiJulio [38]. Methods to set the WW cells automatically are called weight window generators (WWG's).

The method is based on running a pre-simulation to obtain flux and relative error data for the cells. Two alternative approaches, termed the flux-based and *Re*-based approaches, for setting the cell lower weight bounds w_L are detailed in References [38, 39]. For the flux-based approach, w_L in cell i is set according to

$$w_L = \left(\frac{C_U + 1}{2} \right)^{-1} \frac{\phi_i}{\text{Max}(\vec{\phi})} \quad (3.16)$$

where ϕ_i is the scalar flux in cell i , $\vec{\phi}$ is a vector containing the fluxes across the geometry, and C_U is the ratio between w_U and w_L as seen in Equation (3.14). For the *Re*-based approach w_L is set as

$$w_L = \left(\frac{C_U + 1}{2} \right)^{-1} \frac{\text{Min}(\vec{Re})}{Re_i} \quad (3.17)$$

where Re_i is the relative error in cell i and \vec{Re} is a vector containing the relative errors across the geometry. An advantage of the *Re*-based approach is that it leads to smaller differences between adjacent cells, since $Re_i \propto 1/\sqrt{\phi_i}$. This helps combat the oversplitting/long history problem described in Section 3.8 [39].

It might turn out that some particularly hard-to-reach cells receive zero flux during the presimulation. van Wijk et al. [39] deals with this situation by setting the relative error to 100 % (for the *Re*-based approach) or letting the flux be zero (for the flux-based approach). In the latter case the simulation is simply analog locally in the cell. Unfortunately, weight windows cells in GEANT4 cannot be turned off; Stenander deals with this problem by setting the lower weight bound according to the the lowest non-zero flux value encountered globally in the cell mesh [44].

Chapter 4

Implementation in GEANT4

This chapter will give an account of the implementation of the duct source generator in GEANT4 and an overview of the geometry used as a test case for the generator. The geometry is a simplified model based on the DREAM beamline at ESS [45, 46].

4.1 About GEANT4

Users of GEANT4 are expected to develop their own, free-standing applications, compiled and executed like any other computer program. This is in contrast to other particle transport codes and provides extensive freedom, flexibility, and responsibility. GEANT4 itself is also free and open source software, allowing competent users to modify it according to their requirements and to distribute their modifications freely under the conditions stipulated by the GEANT4 Software License (see Reference [47]).

Included in the GEANT4 source is a large selection of example programs, showcasing and demonstrating features of the toolkit. The examples range from basic to advanced, and many highlight specialized features. Users are encouraged to use a suitable example as a starting point for their own application instead of starting from scratch. The application presented in this chapter originally evolved from basic example B1, although very little of that code remains.

For more information about GEANT4, or to acquire a copy of the latest source code, see the GEANT4 website (Reference [48]).

4.2 Structure of a GEANT4 Program

The structure of a GEANT4 program is fundamentally based on the concepts of sessions, runs, and events. A session is composed of one or more runs, and a run is composed of many sequential events. A session is started by running the program and concludes when the program is closed. Sessions can be either interactive through a graphical user interface or be conducted in "batch mode" in a terminal without a GUI. The former is suitable for testing/debugging and visual inspection of the geometry while the latter is best suited for long runs as GUIs use up valuable computer resources. Which mode to use is up to the programmer to decide, one common way is to run the program in interactive

mode if it is run without command line arguments and in batch mode if an argument (e.g. a macro file) is specified by the user.

A run is started by issuing a `/run/beamOn` command, in the spirit of real-life experiments. This triggers the generation of a series of events. Events are in turn subdivided into smaller units of simulations such as tracks and steps (steps being the smallest unit of simulation.) By default, GEANT4 performs the simulation silently and it is up to the user to add scorers to the simulation in order to register hits.

GEANT4 relies heavily on object-oriented principles, allowing users to extend base classes as necessary, without interfering with the core functionality. Two important abstract classes are `G4VUserDetectorConstruction` and `G4VUserPrimaryGeneratorAction`, both of which contain virtual functions the user needs to supply implementations of. These classes concern the problem geometry and generation of primary particles, respectively.

4.3 Setting Up the Simulation

The `main` function prepares the simulation. It allows the user to choose what type of source to use (duct source or analog), optionally set a time limit for the simulation, and to choose whether or not to use weight windows.

Random number generation is an important part of Monte Carlo simulations, and the `main` function is responsible for picking a random engine and setting the seed. Random number generation requires some care with regards to thread safety, and instantiating the random engine in `main` and using the `G4Random` library to shoot random numbers guarantees thread safety. As is common practice, the random seed is set using the current Unix time. The random seed is printed in the program output so that a run can be repeated if needed.

Whether GEANT4 runs in multi- or singlethreaded mode is decided when it is compiled. It is useful to acknowledge both possibilities in the user code since some statements, such as the number of worker threads, only make sense in MT mode. This is included in `main` like so:

```
#ifdef G4MULTITHREADED
    auto run_manager = new G4MTRunManager;
    G4int n_cores = G4Threading::G4GetNumberOfCores();
    run_manager->SetNumberOfThreads(n_cores);
#else
    auto run_manager = new G4RunManager;
#endif
```

Here a multithreaded run manager is created when multithreading is enabled, and the number of worker threads is set to the maximum available. Otherwise a singlethreaded run manager is created.

The program utilizes the GEANT4 concept of parallel worlds. Three worlds exist in parallel. The first is the world containing the mass geometry, including shielding. The class containing the mass geometry extends `G4VUserDetectorConstruction` and is dubbed `BeamlineConstruction`. The second world, described in `wwgImportanceDetectorConstruction`, contains the weight window cells, which are filled with vacuum. The third world, `ParallelWorld`, contains a mesh of flux detectors, also vacuum-filled, and roughly sized $25 \times 25 \times 25$ cm to correspond with real dosimeters used in practice at neutron facilities. The parallel worlds are created in `main`.

4.4 Physics List

It is mandatory for the user to tell GEANT4 what physics processes to simulate. This is because GEANT4 cannot possibly simultaneously simulate every type of interaction in every energy range accurately [49]. Also, certain types of interactions may be irrelevant for the problem at hand and valuable CPU time should not be spent simulating such physics.

For these reasons the user must choose which types of particles and energies they deem to be important for the problem at hand. This is done by supplying a physics list in the `main` function. The user may write their own physics list or pick a standard list included in GEANT4. The standard physics list `QGSP_INCLXX_HP` is used in this thesis as it has given reliable results previously [50].

The name stems from the models used in the physics list. `QGSP` means that it uses the Quark-Gluon String model at high energies. The Liège Intranuclear Cascade model for C++ [51] (`INCLXX`) is used at intermediate energies. The `HP` (High Precision) part indicates the use of the NeutronHP model at low energies, for accurate simulation of neutron scattering at energies below 20 MeV [52].

4.5 Neutron Generation

Generation of primary particles (neutrons) takes place in the `DuctSourceGenerator` class, which extends `G4VUserPrimaryGeneratorAction`. The problem geometry is aligned in such a way that the beamline stretches out in the positive z direction while the x axis is parallel to the ground and the y axis is perpendicular to the ground. Or, to put it more simply, y points to the sky.

The neutron source is assumed to be rectangular and located in the xy plane. Neutrons are generated uniformly over the source area using the GEANT4's static `G4RandFlat::shoot` function, which produces random `G4double`'s between given limits. A `G4ThreeVector` is then created containing the starting coordinates:

```
G4double x0 { G4RandFlat::shoot(xmin, xmax) };
G4double y0 { G4RandFlat::shoot(ymin, ymax) };
G4ThreeVector start_pos { x0, y0, z0 };
```

Here `xmin` refers to the leftmost edge of the source, etc, and (x_0, y_0, z_0) are the starting coordinates.

4.6 The Duct Source

The duct is an imagined rectangular cuboid that coincides with the neutron beamline. The source is in the current implementation located at the very beginning of the duct, and is of the same dimensions as the duct's cross section. This isn't required; the source and the duct can be placed independently of each other and do not have to be the same size.

A hit position on the duct wall is generated in order to determine the neutron's momentum direction. This is in contrast with the analog case, where the momentum direction is directly randomized. The hit position is generated by the `DuctSourceGenerator::RandomizeHitPosition` function, which generates

a hit position and a normal vector to the hit surface. The duct has five walls; the cross-sectional surface facing the source is open. Keep in mind that the duct walls are imagined—they do not physically interact with particles.

The generation of a hit position is done in two steps. The algorithm first picks a random wall to hit, setting the normal vector accordingly. This fixes either the x, y, or z coordinate, depending on which wall is chosen. Then a location on the chosen wall is randomized, fixing the remaining two coordinates to give a fully defined hit position. The momentum direction is then determined as the difference of the start and hit position vectors as in the following code snippet (where `start_pos` is as generated in Section 4.5.)

```
G4ThreeVector momentum { hit_pos - start_pos };
G4ThreeVector momentum_unit { momentum/momentum.mag() };
```

Note that `momentum.mag()` represents the distance from the start position to the hit position.

The particle weight must now be altered in order to satisfy Equation (3.7). The duct can be viewed as a large detector. (This was in fact done in GEANT4 for the purpose of proving that the method is sound, see Chapter 5.) Using this fact, and for the moment assuming that the source emits neutrons isotropically, the number of neutrons, S , emitted by the source can be back-calculated as

$$S = N \frac{4\pi}{\epsilon_{ip}\Omega} \quad (4.1)$$

where N is the number of detected neutrons, ϵ_{ip} is the intrinsic peak efficiency of the detector, and Ω is the solid angle subtended by the detector at the source location [53]. ϵ_{ip} can be set to 1 because every particle is counted.

The volume of the source is negligible—in fact it has no volume, it is a rectangle—and thus the solid angle Ω can be calculated as [53]

$$\Omega = \int_A \frac{\cos \alpha}{r^2} dA \quad (4.2)$$

where r is the distance from the source to the detector surface element dA (represented in the code as `momentum.mag()` as noted above), α is the angle between the source direction and the detector surface normal, and A is the surface area of the detector. Using the geometric definition of the dot product

$$\mathbf{a} \cdot \mathbf{b} = \|\mathbf{a}\| \|\mathbf{b}\| \cos \theta \quad (4.3)$$

where \mathbf{a} and \mathbf{b} are Euclidean vectors and θ is the angle between them, Equation (4.2) can be rewritten as

$$\Omega = \int_A \frac{1}{r^2} \frac{\mathbf{r} \cdot \hat{\mathbf{n}}}{\|\mathbf{r}\| \|\hat{\mathbf{n}}\|} dA \quad (4.4)$$

where \mathbf{r} is the vector from the source to the surface element dA and $\hat{\mathbf{n}}$ is the normal to dA . $\hat{\mathbf{n}}$ is a unit vector so $\|\hat{\mathbf{n}}\| \equiv 1$, and $\|\mathbf{r}\| = r$. Thus

$$\Omega = \int_A \frac{\mathbf{r} \cdot \hat{\mathbf{n}}}{r^3} dA = \int_A \frac{1}{r^2} \frac{\mathbf{r}}{\|\mathbf{r}\|} \cdot \hat{\mathbf{n}} dA \quad (4.5)$$

Performing the integration over the detector area A yields

$$\Omega = \frac{A}{r^2} \frac{\mathbf{r}}{\|\mathbf{r}\|} \cdot \hat{\mathbf{n}}. \quad (4.6)$$

Insertion of this expression for Ω into Equation (4.1) gives

$$S = N \frac{4\pi r^2}{A} \frac{\|\mathbf{r}\|}{\hat{\mathbf{n}} \cdot \mathbf{r}} \quad (4.7)$$

Equation (4.7) describes the number of particles S that the source has to emit for the detector to register N hits in the direction \mathbf{r} . The fraction $p_0 = N/S$ thus describes the unbiased sampling probability. Utilizing Equation (3.7) with the unbiased weight $w_0 = 1$ and the biased sampling probability $p_b = 1$ yields the biased weight

$$w_b = \frac{A}{4\pi r^2} \cdot \frac{\mathbf{r}}{\|\mathbf{r}\|} \cdot \hat{\mathbf{n}}. \quad (4.8)$$

This is the weight factor that compensates for the geometrical flux attenuation.

4.7 Passing Through the Duct

It is important that the hit positions are generated uniformly across the duct walls. If this is not the case, the particle weights must be adjusted to reflect the changed sampling.

It may be desired to intentionally oversample the ulterior duct wall, i.e. the cross-sectional area at the end of the duct, in order to increase statistics beyond the end of the duct. In order to achieve this, the weights of all particles must be altered in accordance with Equation (3.7) to reflect the altered sampling probabilities p' .

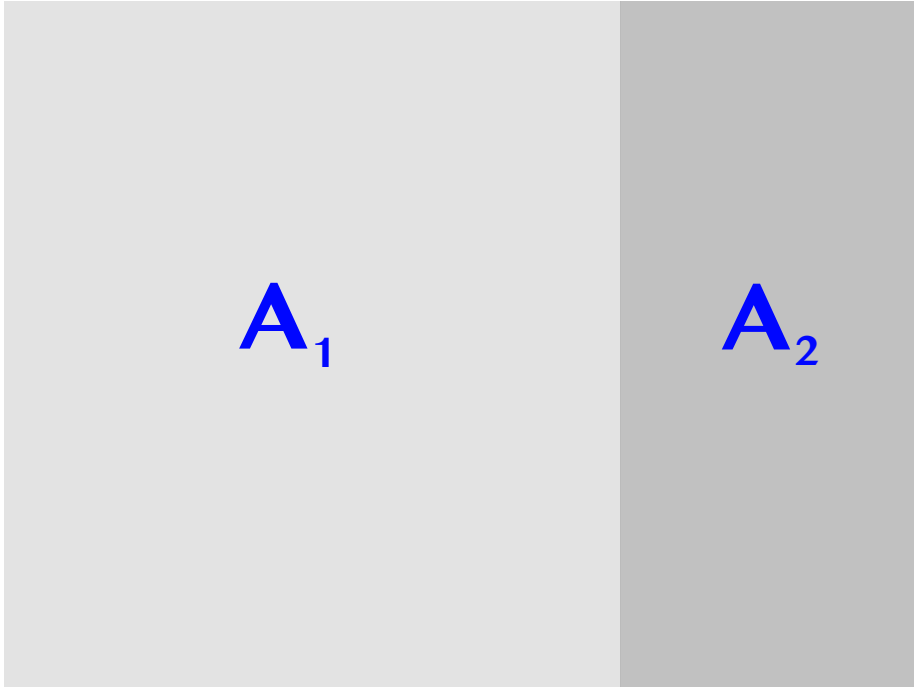


Figure 4.1: Surfaces to be sampled, with $A = A_1 + A_2$.

Consider two surfaces of areas A_1 and A_2 , respectively, and total area $A = A_1 + A_2$, see Figure 4.1. To achieve uniform sampling across A , the sampling probability for each area should be

$$p_1 = \frac{A_1}{A} \quad (4.9)$$

$$p_2 = \frac{A_2}{A}. \quad (4.10)$$

Trivially it holds that $p_1 + p_2 = 1$. Now, suppose that the sampling probability for A_2 is set to a new sampling probability $p'_2 \neq p_2$, in defiance of Equation (4.10). The altered sampling must be compensated for by changing the particle weight using Equation (3.7) so that

$$w'_2 = \frac{p_2}{p'_2} w_2 \quad (4.11)$$

where w_2 and w'_2 are the old and new particle weights, respectively.

The total sampling probability must always be unity, so the new sampling probability p'_1 of A_1 must be

$$p'_1 = 1 - p'_2. \quad (4.12)$$

The new particle weight w'_1 is then found by using Equations (3.7) and (4.12):

$$p'_1 w'_1 = p_1 w_1 \Leftrightarrow \quad (4.13)$$

$$w'_1 = \frac{p_1}{p'_1} w_1 \Leftrightarrow \quad (4.14)$$

$$w'_1 = \frac{p_1}{1 - p'_2} w_1. \quad (4.15)$$

The sampling probability p'_2 can be set arbitrarily on the interval $0 \leq p'_2 < 1$, but deviating too far from uniform sampling could lead to adverse effects such as convergence problems or poor statistics in parts of the geometry.

While the above discussion holds for any general surface, the area A_2 would in the specific case of the duct source correspond to the cross-sectional area at the end of the duct. The sampling probability p'_2 then describes the fraction of particles that pass all the way through the duct.

4.8 Anisotropic Sources

In reality, neutrons entering the guide are not isotropic. Accounting for this in analog simulations is simply a matter of adjusting the number of emitted particles accordingly, but this is not suitable for the duct source. Instead the weight calculated previously in Section 4.6 needs to be supplemented with a new factor.

The source data is stored in pairs of sorted one- and two-dimensional vectors. The angular distribution is given by two vectors of size—in this case—12 elements. The first one contains polar angles (relative to the z axis, i.e. the direction of the beamline) given as cosine values ranging from 1 to -1. Its companion vector gives the flux in the corresponding direction, and after normalization, the relative probability of emission in that angular range. The values in the vectors are paired by their index.

Also tied to each angular range are a vector of energies and a vector of corresponding fluxes arranged in the same manner as the angular data. In other words, each emission angle has its own energy spectrum.

Thus, all relevant particle data can be deduced from the cosine μ of its polar angle θ , i.e. $\mu = \cos\theta$. This section will describe how to calculate the angular weight factor. Energy selection will be dealt with in Section 4.9.

The cosine of the emission angle is calculated with the help of Equation (4.3) by taking the scalar product of the normalized momentum vector and the z unit vector $\hat{\mathbf{z}}$:

$$\mu = \hat{\mathbf{z}} \cdot \frac{\mathbf{u}}{\|\mathbf{u}\|} \quad (4.16)$$

The vector index i is then found by performing a linear search in the μ vector. A linear search is appropriate in this case for two reasons. First, the vector is small, only 12 elements. And second, due to the geometry of the problem, most emission angles will fall into the first range, meaning that the first comparison yields a match. Thus the linear search's best case scenario complexity of $\mathcal{O}(1)$ (constant time) is also the most common in this case, and the worst case of $\mathcal{O}(n)$ only occurs exceptionally. A linear search might no longer be appropriate or optimal should the number of elements in the vector be increased substantially in a future application.

The next step is to calculate the angular weight factor w_μ . The two factors that contribute to w_μ are

- the fractional size $\Delta\mu$ of the interval, and
- the flux ϕ_i (or rather the probability of emission p_i) in the interval.

A large flux emanating from a narrow angle yields large statistical weights for associated particles, and conversely, a small flux from a wide angle results in small statistical weights. For a found index i

$$\Delta\mu = \frac{\mu_{i-1} - \mu_i}{\mu_{max} - \mu_{min}}, \quad \mu_i < \mu \leq \mu_{i-1} \quad (4.17)$$

$$p_i = \frac{\phi_i}{\sum_k \phi_k} \quad (4.18)$$

where μ_{i-1} and μ_i are the upper and lower interval limits, respectively. Then

$$w_\mu = \frac{p_i}{\Delta\mu}. \quad (4.19)$$

The new weight is then obtained as

$$w_{new} = w_\mu \cdot w_{old}. \quad (4.20)$$

Unlike the polar angle, the azimuthal angle is assumed to be isotropic. A similar procedure could be applied to account for azimuthal anisotropy should it be required in some other scenario.

4.9 Energy Selection

As noted above, each angular range has its own unique energy distribution. In general, directions with narrower angles with respect to the proton beamline

show larger high energy contributions, as discussed in Section 2.5. The energy vectors and corresponding flux vectors are stored as vectors-of-vectors, where the index i found in Section 4.8 enables retrieval of the appropriate energy and flux vectors for the current μ . A second index, j , identifies the energy range and corresponding flux in the vectors at position i in the vectors-of-vectors.

Energy selection can be done in an analog fashion, i.e. in a way that corresponds precisely to the physical situation. This was done as part of the work for this thesis and the results are presented in Chapter 5. This kind of energy selection is identical whether the simulation as whole is analog or if angular source biasing is used and requires no modification of the weight of the Monte Carlo particles.

The implementation uses a C++11 standard library distribution called `std::piecewise_constant_distribution`, and random engine `std::mt19937`, an implementation of the Mersenne-Twister [54] pseudo-random number generator (PRNG), to create a weighted random number generator that selects neutron energies in 1:1 correspondence with their physical probability. GEANT4 has a selection of built-in distributions, but unfortunately lacks a piecewise constant one. Hence the necessity use standard C++ alternatives.

Using this analog energy selection comes with two drawbacks. The first problem is technical and is simply that a PRNG is a relatively heavy object and each worker thread needs its own instance for thread safety reasons (using external random number generators is discouraged in GEANT4).

Manually implementing a version of `std::piecewise_constant_distribution` using a standard GEANT4 random number generator would solve the first problem, but would do nothing to address the second problem, which is conceptual in nature. The spectra represented in the input vectors are moderated and consequently the bulk of the neutrons are cold or thermal, with only a small fraction displaying energies approaching the GeV range. But these contribute disproportionately to the flux in deep-penetration problems and are therefore of great interest for the results.

The introduction of energy biasing neatly solves both these problems. Although originally conceived as a method to programmatically streamline the source code, the potential for advances in variance reduction for shielding problems soon became apparent. A similar procedure has been successfully tried before by Liang et al. using the general-purpose Monte Carlo transport code FLUKA in an application for the Chinese Spallation Neutron Source (CSNS) [34].

To implement the idea, an energy between the low and high extremes is chosen at random, without consideration of the real probabilities. The particle weight is then adjusted in accordance with Equation (3.7). The energy randomization takes place in the logarithmic domain since the energy spectrum is logarithmic in nature. If E_{min} and E_{max} are the lowest and highest possible energies, respectively, then a random number $q \in \mathbb{R}$ is generated linearly on the interval $[\log E_{min}, \log E_{max}]$ and the particle energy E is given by

$$E = 10^q. \tag{4.21}$$

Once an energy E is chosen, the second vector index j must be found to complete the i, j pair of indices. The reasons given above for using a linear search to find i do not apply to the search for j , because

1. the energy data is much more fine-grained than the angular data so the vectors to be searched are substantially larger in size, and
2. there is no obvious pattern as to which index is most likely.

Thus a binary search is used instead. The search can never terminate early because it is searching for a range rather than an exact value, meaning that it always encounters its worst case scenario complexity of $\mathcal{O}(\log n)$. This is not a problem since the number of energy bins n ranges in the hundreds, so the binary search still terminates after just a couple of iterations. The binary search is also more frugal in terms of memory usage than some alternative fast(er) searches.

The new weight is then calculated in analogy with Equations (4.17)–(4.20) as

$$\Delta E = \frac{\log E_{j-1} - \log E_j}{\log E_{max} - \log E_{min}}, \quad E_j < E \leq E_{j-1} \quad (4.22)$$

$$p_j = \frac{\phi_j}{\sum_k \phi_k} \quad (4.23)$$

$$w_E = \frac{p_j}{\Delta E} \quad (4.24)$$

$$w_{new} = w_E \cdot w_{old} \quad (4.25)$$

where E_{j-1} and E_j are the upper and lower interval limits, respectively, and w_E is the weight factor arising from the energy biasing.

The energies in the source spectrum used in this thesis range from 1 μeV to 2 GeV, i.e. $1 \cdot 10^{-6}$ to $2 \cdot 10^9$ eV. Thus the random number q falls in the range $[\log(10^{-6}), \log(2 \cdot 10^9)] \simeq [-6, 9.3]$. The percentage of simulated Monte Carlo neutrons with energies in excess of 10 MeV is therefore boosted to over 15 %, and more than 8 % fall in the 100+ MeV category.

4.10 Problem Geometry

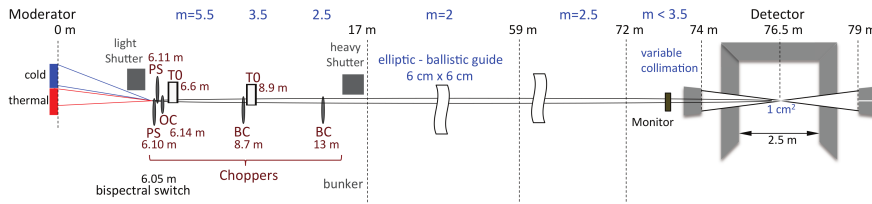


Figure 4.2: The DREAM beamline at ESS as pictured by Schweika et al. in the instrument’s proposal [46]. Reprinted under the terms of the CC-BY-3.0 [55].

The geometry used is a simplified model of the DREAM beamline at ESS, a straight beamline dedicated to feeding neutrons to a powder diffractometer [45]. It is a medium-length beamline by ESS standards; in the model it measures 77.0 m from the beamline entrance (two meters from the moderator surface) to the back of the beamstop. A detailed technical overview from the instrument’s

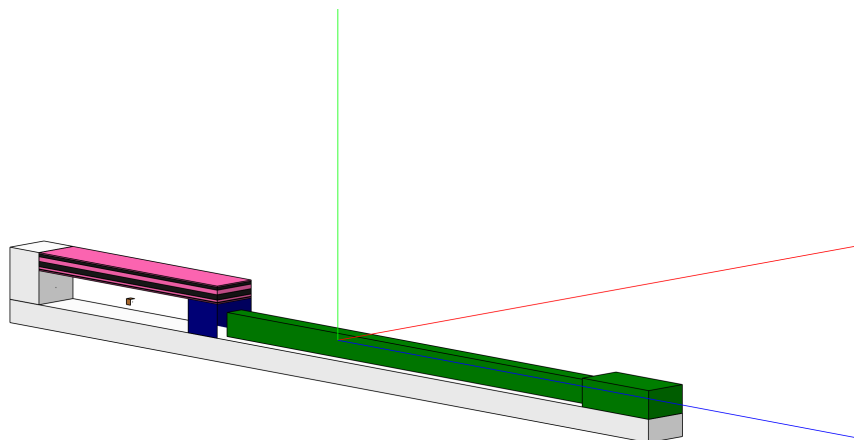


Figure 4.3: The simplified model of the ESS DREAM beamline used as a test case for the duct source generator. The coordinate axes are pictured as $x, y, z =$ red, green, blue. Each color in the figure represents a material. The iron floor and monolith is gray, the copper collimators (only one is visible from this angle) are brown, the heavy concrete bunker wall is blue, and the concrete beamline shielding and instrument cave is green. The bunker roof alternates between polyethylene (pink) and steel (black).

proposal is shown in Figure 4.2. The simplified GEANT4 model is pictured in Figure 4.3.

The entire model measures $X \times Y \times Z = 4.03 \times 6.48 \times 77.0$ m. It can be regarded as being divided into four major sections: the monolith, the bunker, the beamline shielding, and the instrument cave. The structure rests on a 2.0 m thick iron floor that occupies the entire lower part of the model. Covering the bunker section is the bunker roof, a massive, layered structure that measures 1.55 m in total thickness. The seven layers alternate between polyethylene and steel to ensure maximum shielding effectiveness across the neutron spectrum.

The monolith is a 3.5 m thick slab of iron completely covering the xy plane at the beginning of the beamline. The only opening in it is the 3×3 cm² beamline port.

Behind the monolith is the bunker area, an 18 m open area followed by a 3.5 m thick wall made of heavy (magnetite) concrete. In reality this space is quite crowded by various equipment, such as choppers, but in this model it is empty save for two identical copper collimator blocks measuring $0.5 \times 0.5 \times 0.5$ m³. One is placed in the middle and the other is placed flush with the bunker wall. Like the monolith, both have beamline openings to allow the neutrons to pass through. The bunker is completely covered by the above mentioned bunker roof.

Following the bunker is a 44 m long “corridor”, 0.5 m wide and 1.48 m high. It is surrounded by standard concrete shielding that is 60 cm thick. At its end is the instrument cave, a room measuring $2.0 \times 1.48 \times 6.0$ m³. It is shielded by 1 m thick concrete shielding.

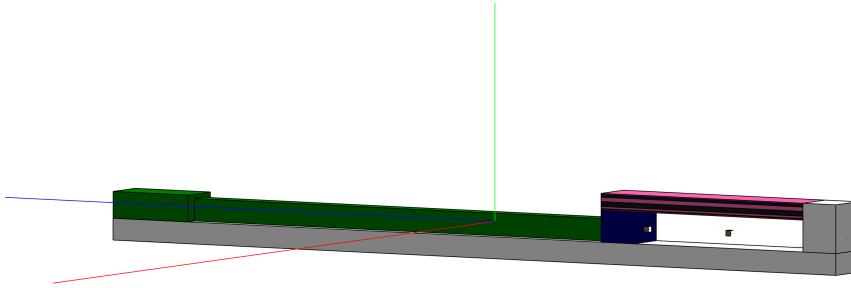


Figure 4.4: The model pictured from the left side. Both collimator blocks in the bunker are visible, as is the layered roof structure above them.

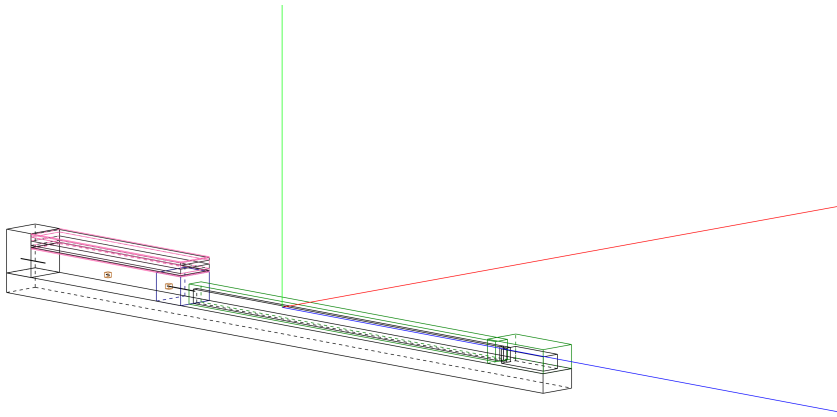


Figure 4.5: Same view as Figure 4.3 but with wireframes to reveal the inner structures. The instrument cave is visible on the righthand side.

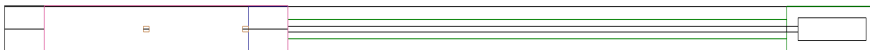


Figure 4.6: Top wireframe view.



Figure 4.7: Right side wireframe view.

4.11 Weight Window Generator

The weight window generator algorithm was implemented previously by John Stenander and Douglas DiJulio, and used largely as-is in this thesis. See Section 3.9 for an overview. For further details on its implementation, refer to John’s master’s thesis [44] or the related article [38].

It is worth mentioning the figures-of-merit (FOM) used previously to judge the efficacy of the weight window generator. They are defined as [38]

$$\text{FOM}_1 = \left(\sum_{i=1}^N \frac{Re_i^2}{N} \cdot T \right)^{-1} \quad (4.26)$$

$$\text{FOM}_2 = \sqrt{\frac{1}{N} \left(\sum_{i=1}^N Re_i^2 - \frac{1}{N} \left(\sum_{i=1}^N Re_i \right)^2 \right)} \quad (4.27)$$

where N is the number of (non-zero) cells, T is the simulation wall clock time, and Re_i are the relative errors of the cells. An effective variance reduction method should result in an increased FOM_1 and a decreased FOM_2 . These definitions are not without problems and care should be taken when the results are judged. One problem mentioned by van Wijk et al. [39] is that there is no straightforward way to take into account cells that receive zero fluxes at the end of the simulation, and these are simply discarded. Another is that employing a variance reduction may well lead to increased average relative errors since analog simulations typically show extremely small errors in the vicinity of the source.

Chapter 5

Results and Discussion

5.1 Proof of Concept

The duct source was first tested on a simple “toy problem” to verify its validity. The geometry consisted of a straight beamline measuring $5\text{ cm} \times 5\text{ cm} \times 10\text{ m}$, fully surrounded by a GEANT4 sensitive detector to register the wall current. A total of 98 detectors were placed in the beam path to register the current. The toy geometry is depicted in Figure 5.1.

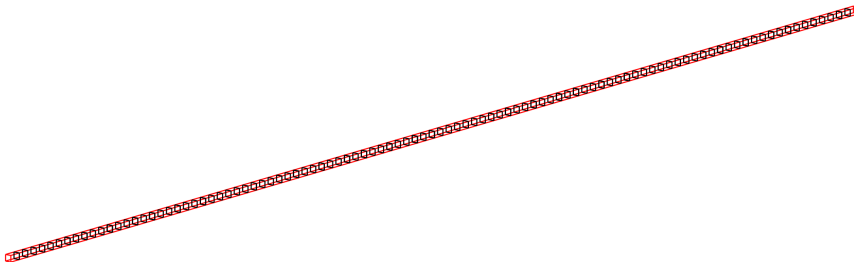


Figure 5.1: Graphical representation of the short beamline section with detector tiles in the beamline.

Ten million primary particles were simulated for every test case with the toy problem. The GEANT4 output data was processed using the ROOT data analysis framework [56].

Figure 5.2 shows the resulting wall current for the analog isotropic source, with arbitrary units on the y axis due to normalization. The red curve is proportional to $1/r^3$, with the proportionality constant fitted to match the histogram. The number of histogram bins is set to 10,000. This result serves as a control; the result of any variance reduction method must match this one qualitatively. The precision is high for the first 0.5 m, but quickly becomes unacceptable at greater distances from the source. Very few particles make it to the second half of the guide and many bins show zero hits.

Figure 5.3 shows the normalized current registered by the in-guide detectors placed 10 cm apart for the length of the guide. The plotted green curve is

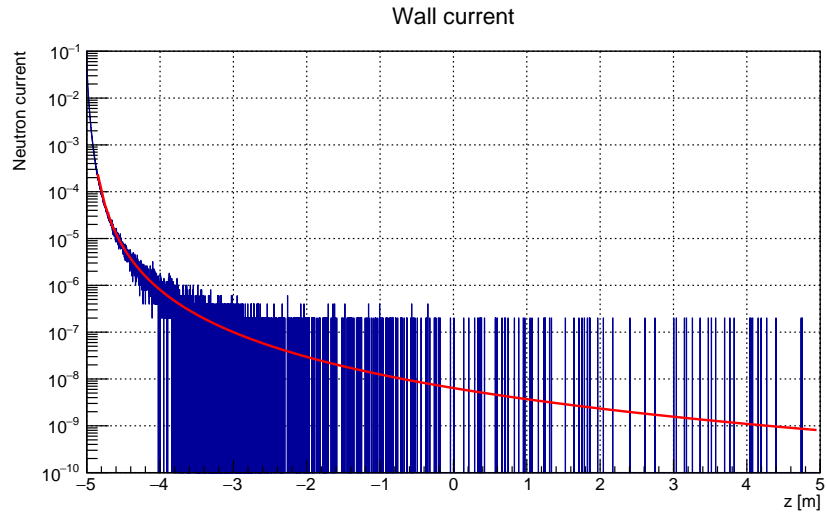


Figure 5.2: Histogram (normalized) of the wall current with the analog isotropic source. The function shown in red is proportional to $1/r^3$.

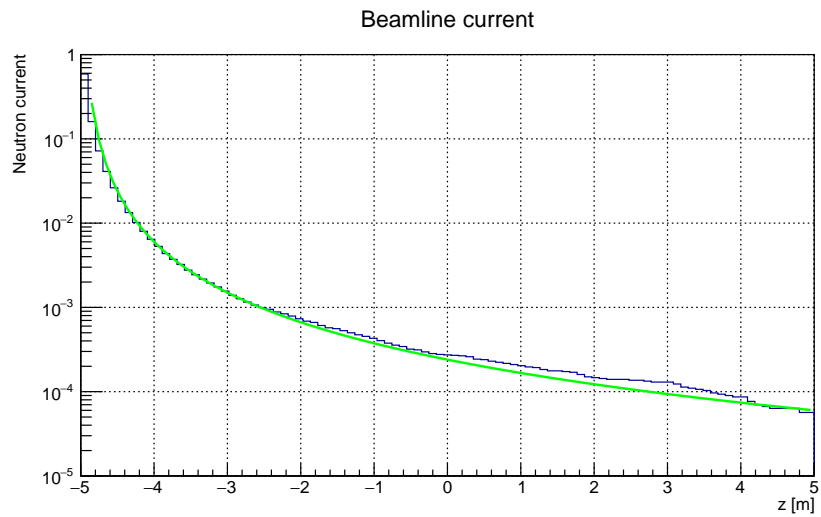


Figure 5.3: The beamline current (normalized) for the analog isotropic source. The function shown in green is proportional to $1/r^2$.

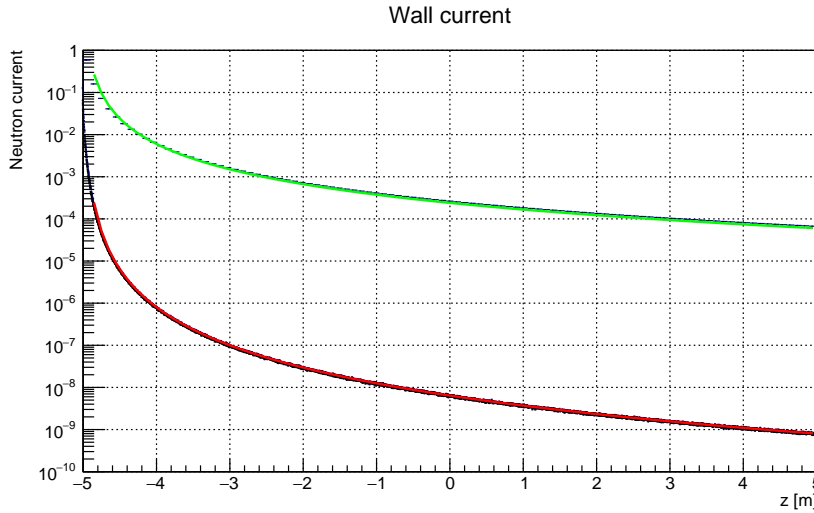


Figure 5.4: Duct source, isotropic case. Measured wall and beamline currents with functions proportional to $1/r^3$ (red) and $1/r^2$ (green) superimposed.

proportional to $1/r^2$ and fitted to the histogram. There are no zero count bins here due to the fact that a single particle traversing the whole beamline is enough to populate every bin. Still, the histogram data deviates noticeably from the theoretical value toward the end of the guide.

Figure 5.4 shows both the wall and beamline currents for the isotropic duct source, i.e. the duct source equivalents of Figures 5.2 and 5.3. The histograms reveal a dramatic improvement in the statistics throughout the guide at a slight cost of precision near the very beginning. Every wall current bin is now satisfactorily populated, and a strict adherence to the theoretical value is observed for both the wall and beamline currents. The green and red $1/r^2$ and $1/r^3$ fitted plots are unchanged from the analog Figures. The duct source can be extended indefinitely without loss of precision as long as the number of primary particles is increased by the same factor as the guide length, in stark contrast with the analog simulation, where the number of primaries has to increase exponentially in order to avoid loss of precision at the end of the guide.

Figure 5.5 shows the unnormalized wall currents for isotropic analog and duct sources. The matching histograms demonstrate that there are no factors missing in the calculation of the duct source particle weights.

Figure 5.6 showcases the passthrough functionality described in Section 4.7. Particles are generated uniformly on the interval $[-5 \text{ m}, 0 \text{ m}]$, and the passthrough parameter is set to $p'_2 = 0.25$ (the previous value was 0.01.) In other words, 25 % of the particles travel past the $z = 0 \text{ m}$ point. The precision drops slightly toward the end of the guide but the data still shows excellent adherence to the qualitative trend dictated by the analog case.

Switching from the isotropic source to the anisotropic source yields Figure 5.7, which shows

- the anisotropic analog source in blue,

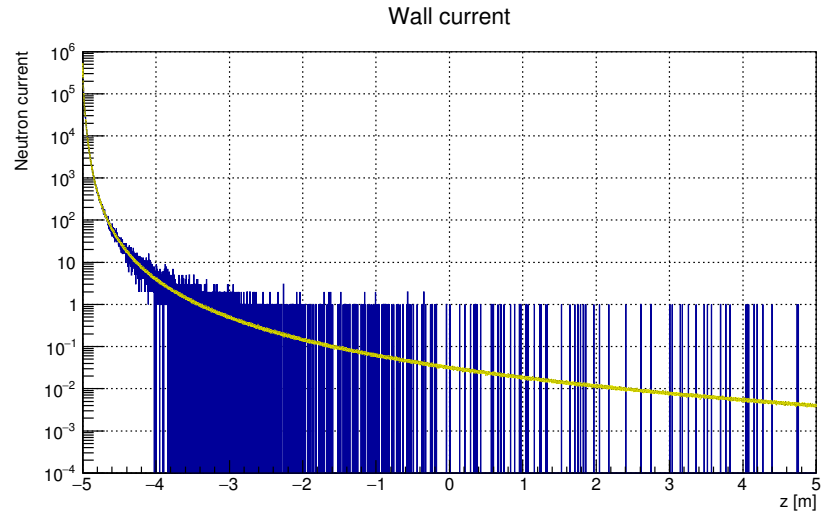


Figure 5.5: Wall current for the isotropic analog source (blue) and the isotropic duct source (yellow), unnormalized.

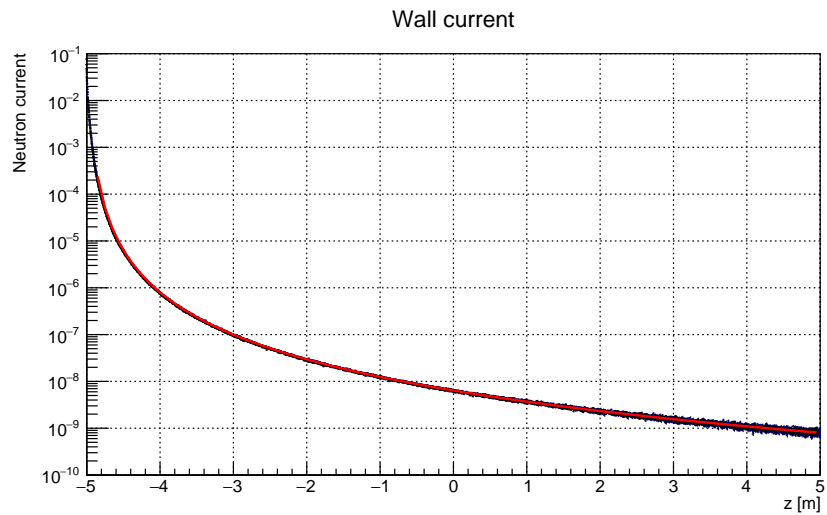


Figure 5.6: Wall current with the duct source extending from -5 to 0 m and 25 % of the particles passing through the cross section of the guide at $z = 0$ m.

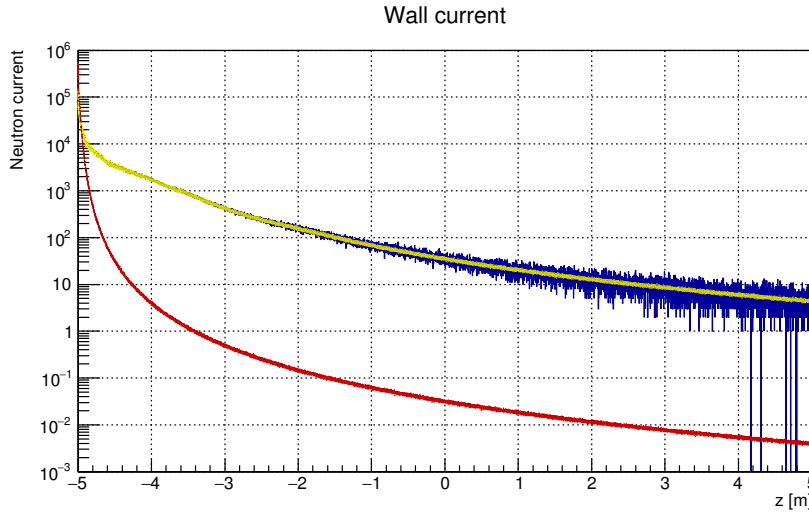


Figure 5.7: Wall current for the anisotropic analog source (blue), the anisotropic duct source (yellow), and the isotropic duct source (red). Unnormalized.

- the anisotropic duct source in yellow, and
- the isotropic duct source in red, for comparison.

The figure demonstrates that the anisotropic weighting in the duct source, detailed in Section 4.8, yields the expected result. Another conclusion is that the anisotropy substantially increases the flux down the guide; the increase is roughly three orders of magnitude beyond the 2 m point without losing accuracy near the source.

5.2 Energy Biasing

The analog simulation in the energy domain is represented by Figure 2.7. The uniform distribution of energies as generated using the energy biasing technique is shown in Figure 5.8. All particles, regardless of emission angle, are represented in this plot. The gaps in the spectrum arise because no particles are emitted with these energies.

Applying the particle weights yields Figure 5.9. The spectrum exhibits the correct general structure, but is ragged due to a few very high-weight particles that affect the convergence of the simulation. As it turns out, the offending particles appear when the angle of emission relative to the z axis is large. Filtering out particles that cross the beamline wall within 10 cm of the source yields the spectrum shown in Figure 5.10.

Two strategies for alleviating or eliminating this problem present themselves. In order of preferability they are

1. use a source spectrum calculated for placement a small distance from the beginning of the duct, and
2. increase the number of primaries.

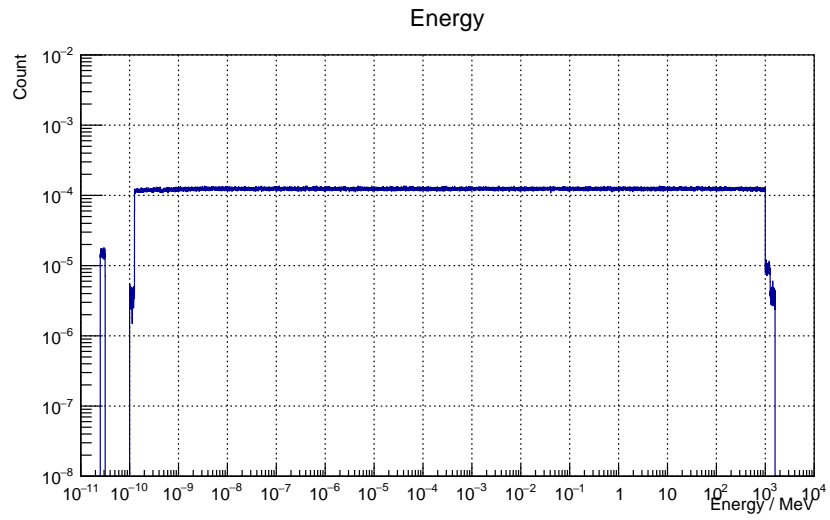


Figure 5.8: Unweighted energy spectrum as used in the energy biasing technique.

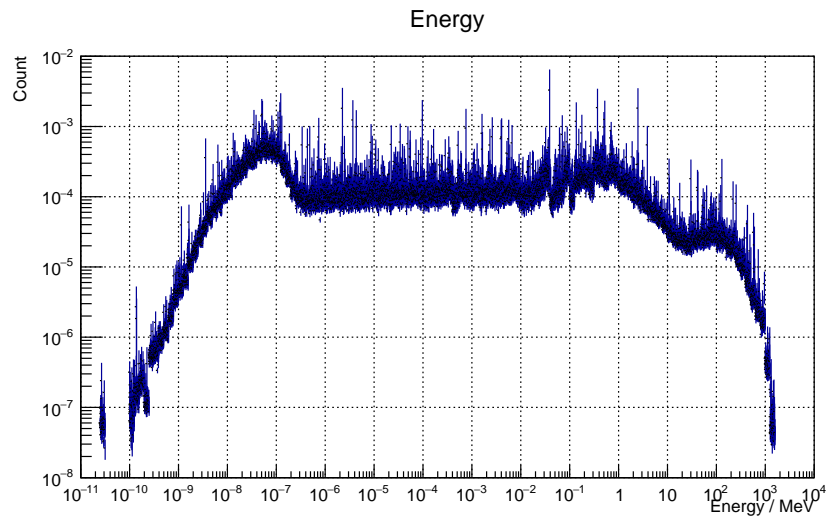


Figure 5.9: The energy spectrum from Figure 5.8 with weights applied.

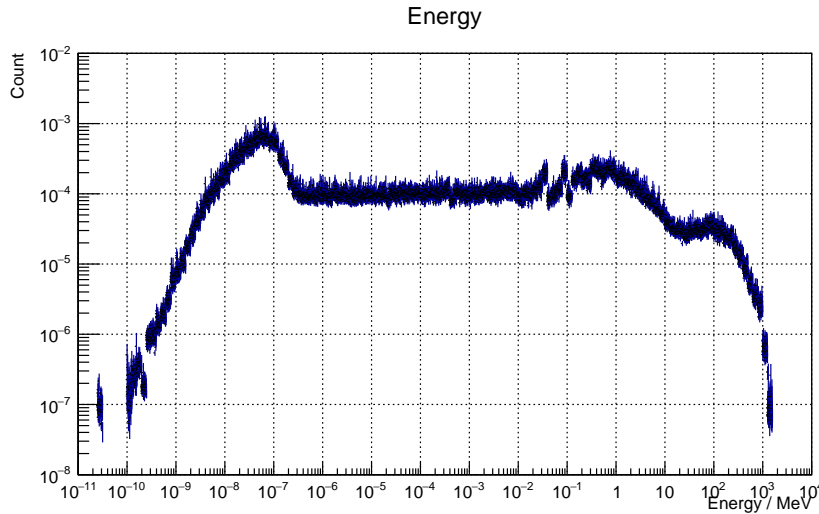


Figure 5.10: The spectrum from Figure 5.9 after filtering out particles that cross the duct wall within 10 cm of the source.

5.3 Beamline Results

All results in this Section were obtained using the anisotropic sources. A fully analog simulation of the DREAM-based geometry is shown in Figures 5.11 and 5.12. The flux maps are calculated at the beamline level in the horizontal (xz) and vertical (yz) planes. Outlines of the geometry are overlaid.

The analog simulation populates the monolith and the empty bunker space, and parts of the iron floor near the source. The bunker wall and roof remain essentially unpopulated, as does most of the rest of the geometry. Some particles travel down the length of the beamline and interact with the shielding surrounding the cave at its end, populating the empty space within the cave.

Turning on the duct source (with analog energy selection)—see Figures 5.13 and 5.14—leads to significant improvement particularly in the area in proximity with the beamline: the beamline shielding, the cave shielding, the cave floor, and the bunker wall. Some roof penetration can also be seen, as well as greater penetration in the bunker floor.

Flux maps produced with the duct source in combination with energy biasing are shown in Figures 5.15 and 5.16. The result is virtually complete population of the bunker wall and the cave shielding. The area outside the beamline shielding is also starting to become populated. One of the largest improvements is seen in the bunker roof, where the penetration reaches through several of the shielding layers.

Figures 5.17 and 5.18 show the flux maps resulting from a simulation combining the duct source with energy biasing and the weight window technique. The simulation whose results are shown in Figures 5.15 and 5.16 were used as a pre-run in order to set up the weight windows for this simulation. The pre-run results do not count toward these flux maps and they should therefore be judged with the necessity of spending time on the pre-run in mind.

The weight windows do not improve statistics in the region extending from

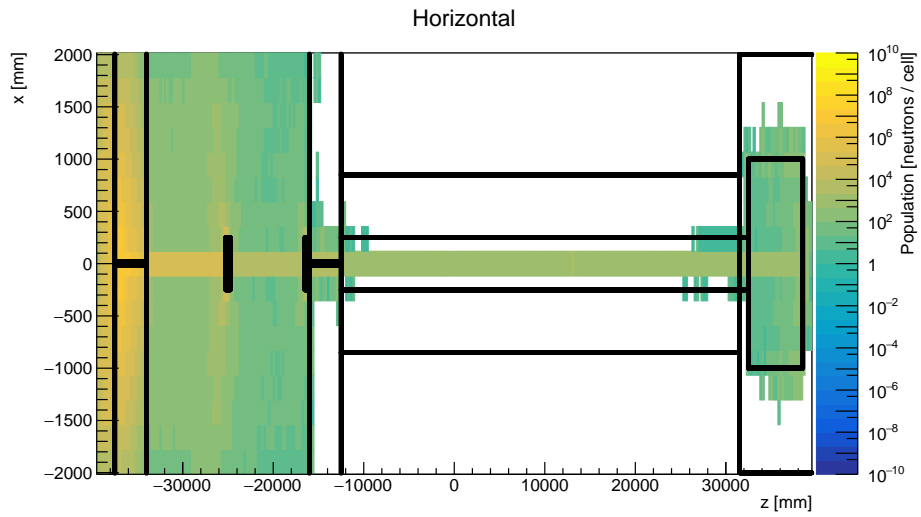


Figure 5.11: Flux map in the horizontal plane, analog simulation, anisotropic source, ~ 16 CPU hours.

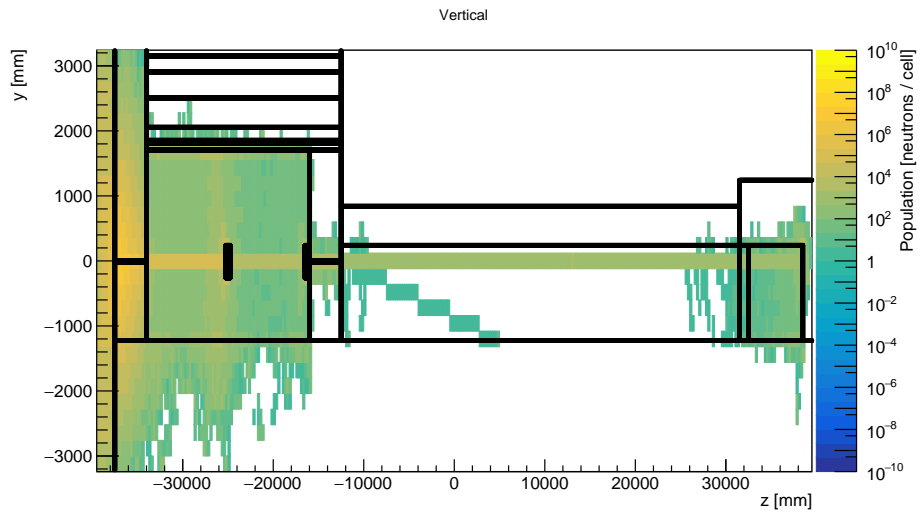


Figure 5.12: Vertical plane flux map from same data as Figure 5.11.

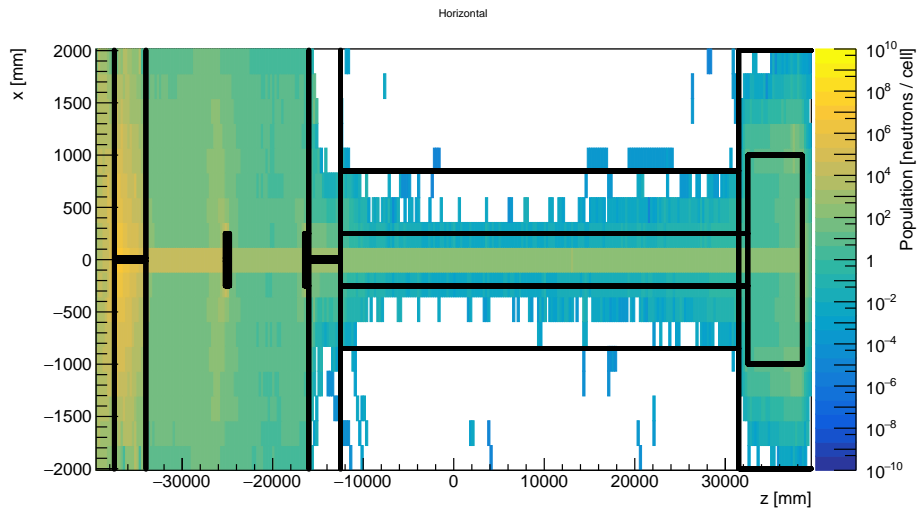


Figure 5.13: Horizontal plane flux map produced using the anisotropic duct source without energy biasing, ~ 16 CPU hours.

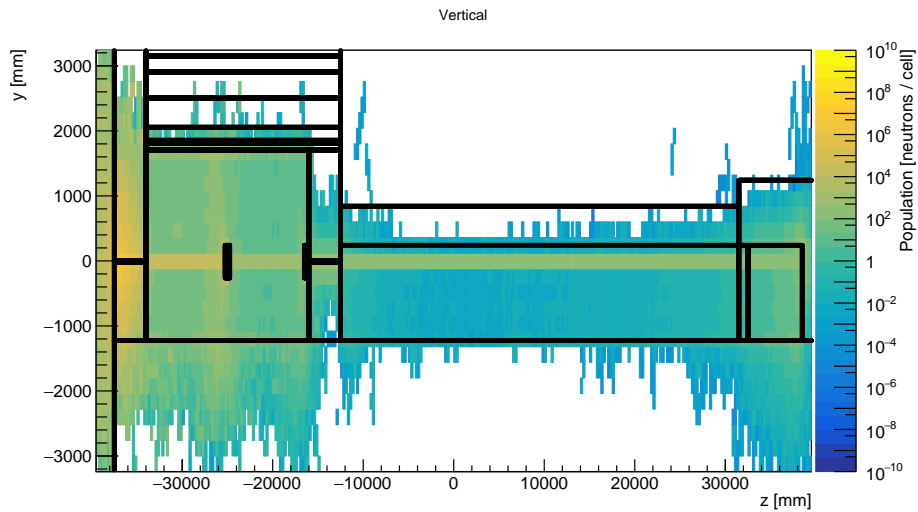


Figure 5.14: Vertical plane flux map from the same data as Figure 5.13.

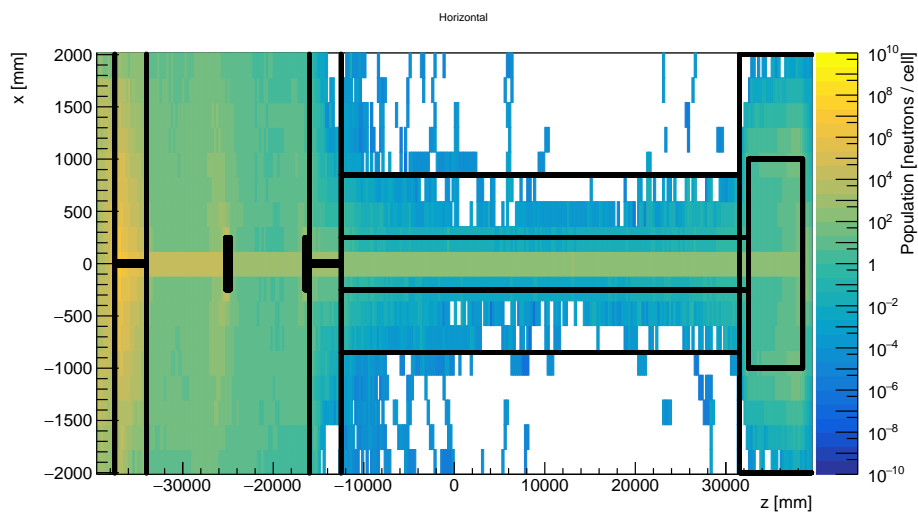


Figure 5.15: Horizontal plane flux map produced using the anisotropic duct source with energy biasing, ~ 16 CPU hours.

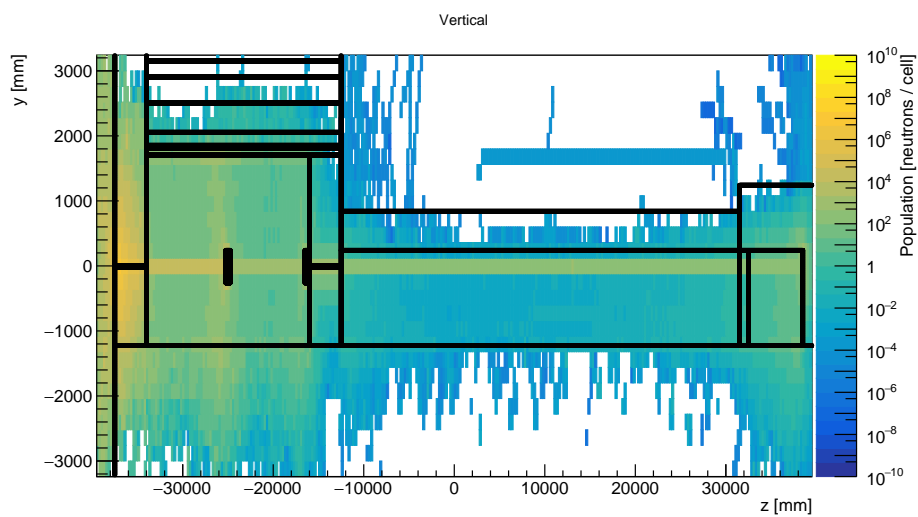


Figure 5.16: Vertical plane flux map from the same data as Figure 5.15.

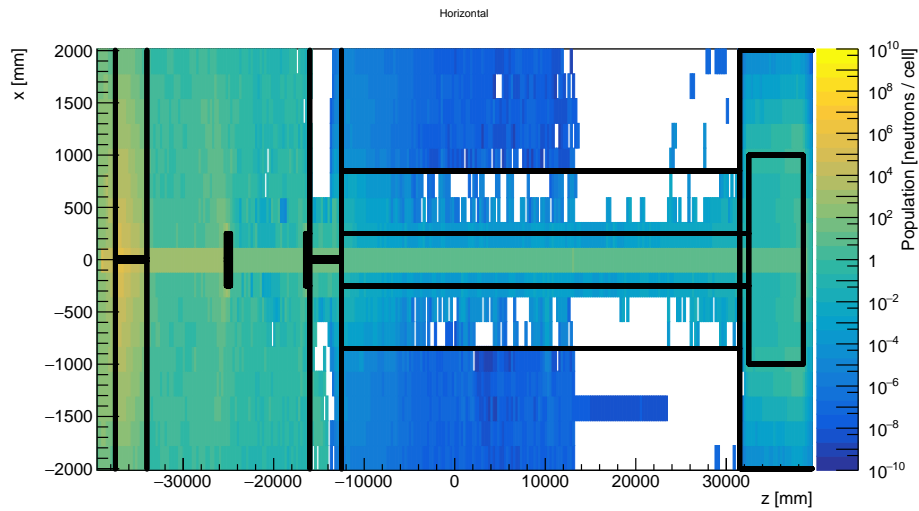


Figure 5.17: Horizontal plane flux map produced using the anisotropic duct source with energy biasing and weight windows, ~ 16 CPU hours.

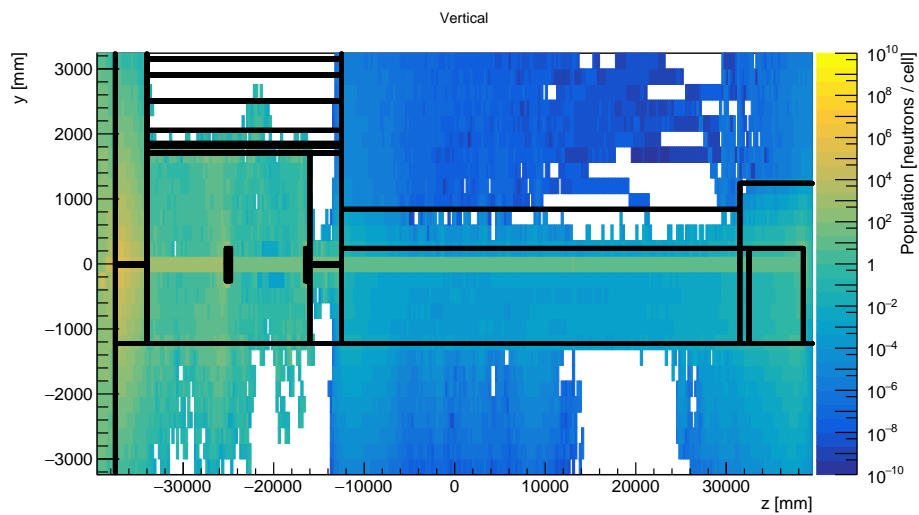


Figure 5.18: Vertical plane flux map from the same data as Figure 5.17.

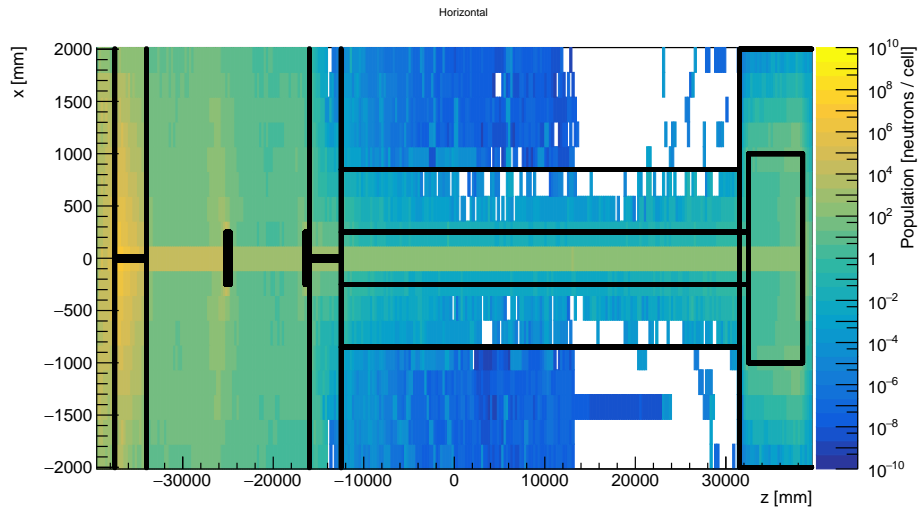


Figure 5.19: The combined fluxes of Figures 5.15 and 5.17, ~ 32 CPU hours total.

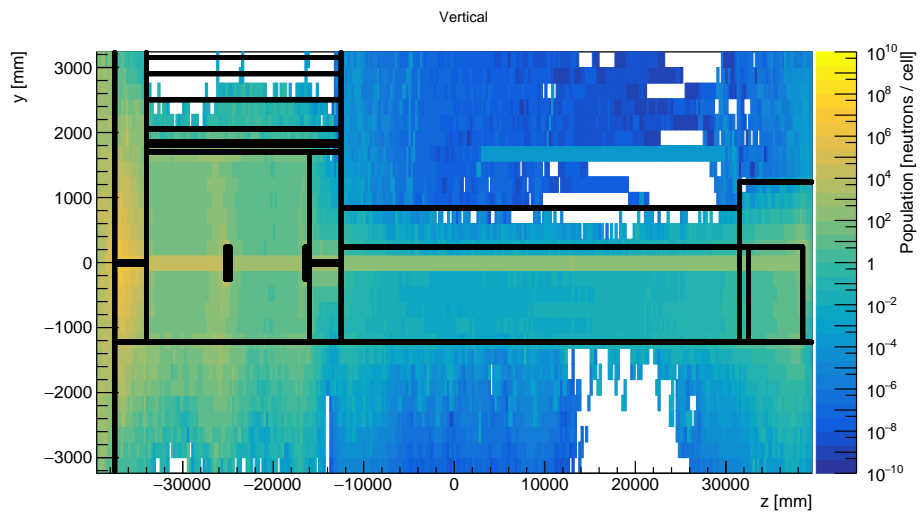


Figure 5.20: The combined fluxes of Figures 5.16 and 5.18, ~ 32 CPU hours total.

the source to the end of the bunker wall. The benefit is instead to be found in the area outside the beamline shielding. This large section of the geometry is now to a large extent populated, with gaps mostly in the iron floor.

The combined fluxes of Figures 5.15 and 5.17 are shown in Figure 5.19. The corresponding vertical plane flux map is shown in Figure 5.20. The result is a simulation requiring 32 CPU hours in total to complete. The Figures demonstrate the strength of combining different variance reduction techniques to efficiently acquire results in all areas of the geometry: angular source biasing to push particles down the beamline, source energy biasing to penetrate thick shielding, and weight windows to push particles laterally.

5.4 Figures-of-Merit and Empty Cells

Table 5.1 show the FOMs (see Section 4.11) resulting from various configurations and runtimes. The runs are independent from each other and consequently show some internal variation. For example, the 4 hour analog simulation managed to populate a larger fraction of cells than its 8 hour counterpart. All figures were produced using a mesh consisting of 312 cells.

To simplify discussion, the configurations are named using the shorthands

1. Case 1 – analog source
2. Case 2 – duct source without energy biasing
3. Case 3 – duct source with energy biasing
4. Case 4 – duct source with energy biasing and weight windows.

The Case 1 simulation shows the largest FOM_1 value for every runtime with the exception of 16 hour Case 4 simulation using the weight windows. It should be understood, however, that the necessary (16 hour) pre-run is omitted from this result, yielding a somewhat unfair advantage for the weight window technique.

That the Case 1 simulation shows the best values for FOM_1 does not mean that it is performing better than the other cases. The fraction of populated cells ranges from 44 % for the 8 hour run to just 50 % for the 120 hour run. The empty cells are discarded from the FOM calculations, and consequently the result is as if more than half of the geometry didn't exist. Figures 5.11–5.12 reflect this; the bunker area is well populated, but the rest of the geometry is largely empty.

Another factor that improves the analog simulation's FOM_1 is that more particles are simulated per unit time due to the particles generally travelling short distances and undergoing few shielding interactions before stopping. In particular, the energy biasing technique provokes a lot of interactions with the shielding due to the higher energy of the particles. This takes a lot of CPU effort to simulate leading to fewer primaries being simulated per unit time.

The duct source fares better with regards to FOM_2 , but the problem of discarded zero flux cells still unfairly favours the analog simulation. Even at 120 CPU hours, Case 1 is still showing a slightly better figure than Case 2.

One possible conclusion is that the FOMs are of at best marginal value for this type of simulation and a more suited alternative is desired. Taking

Source type	FOM ₁	FOM ₂	Zero cells	CPU hours
Case 1	0.0022098	0.309335	53 %	4
Case 2	0.0009009	0.309193	33 %	4
Case 3	0.0008033	0.290219	6 %	4
Case 4	n.p.	n.p.	n.p.	4
Case 1	0.0043443	0.161668	56 %	8
Case 2	0.0005678	0.282158	31 %	8
Case 3	0.0005330	0.251019	3 %	8
Case 4	n.p.	n.p.	n.p.	8
Case 1	0.0015345	0.222598	54 %	12
Case 2	0.0003116	0.329249	19 %	12
Case 3	0.0003540	0.261680	0 %	12
Case 4	n.p.	n.p.	n.p.	12
Case 1	0.0006741	0.289913	52 %	16
Case 2	0.0002687	0.316079	22 %	16
Case 3	0.0003386	0.231660	0 %	16
Case 4	0.0008167	0.145681	0 %	16
Case 1	0.0003028	0.252053	52 %	48
Case 2	0.0000881	0.323634	7 %	48
Case 3	0.0001725	0.205794	1 %	48
Case 4	n.p.	n.p.	n.p.	48
Case 1	0.0001643	0.301386	50 %	120
Case 2	0.0000812	0.313091	11 %	120
Case 3	0.0001212	0.226918	0 %	120
Case 4	n.p.	n.p.	n.p.	120

Table 5.1: Figures-of-Merit and empty cells for different runtimes.

the number of primary particles into account could be a way to go here. The fraction of unpopulated cells also needs to be taken into account. FOM_1 could for example be multiplied by the fraction of populated cells.

An alternative way to look at it would be that the duct source shouldn't be considered a true GVR method, which is what the FOMs are intended to judge. Instead it could be regarded as an LVR. This viewpoint is supported by the results for the 16 hour runs, wherein Case 4 shows the highest FOM_1 and, by a significant margin, the lowest FOM_2 of all four cases. The weight window method is definitely a GVR method, and as such it is unsurprising that it performs well in tests intended to judge GVR efficacy.

Disregarding Case 1 and looking only at cases 2–4 supports this view. Case 3 displays a better FOM_2 value for every runtime, and the difference gets more pronounced the longer the runtime. A similar story holds true for FOM_1 . Case 4 further improves both figures over Case 3. Thus the duct source becomes more and more “GVR-like” with the inclusion of first energy biasing and then weight windows, and this is reflected by the figures-of-merit.

The fraction of populated cells is the only figure of the three that gives a straightforward indication as to the efficacy of the various VR techniques. At 4 CPU hours, the duct source is showing 67 % population, and the duct source with energy biasing is already at 94 %. At 48 hours, Case 3 shows 93 % population, Case 4 is at 99 %, while the analog simulation is at a mere 48 %. Every cell is populated after 16 CPU hours with the weight windows activated.

Chapter 6

Conclusions and Outlook

The duct source has proven adept to the point of perfection at reducing variance inside the neutron guide. A large number of particles are effortlessly transported to distances that are effectively out of reach for analog simulations. Altering the nominally uniform sampling allows for increased statistics in areas of particular interest, if desired. Including energy biasing significantly increases populations in areas of the geometry that are sheltered behind thick shielding and improves prospects for studies of high energy backgrounds. A small penalty is paid in the form of high-weight particles that may cause lack of convergence when the angle of emission is large.

The duct source is nicely complemented by the automated weight window procedure, with the two techniques doing most of their work in different areas of the geometry.

The two figures-of-merit seem to be of at best marginal usefulness (and at worst misleading) for judging the efficacy of the duct source. A possible reason for this is known weaknesses in their definitions. Another is that it might be unfair to judge the duct source as a GVR method, and it should instead be considered an LVR method. The number of zero flux cells does give a direct indication of how well the chosen variance reduction methods perform, but does not take the relative error into account.

An attractive feature of the the duct source is that the execution times are highly predictable: if N primary particles takes T units of time to simulate, then $10N$ particles will take about $10T$ units of time. The duct source works well in a multithreaded environment and future use of it is likely to be conducted using MPI (see Reference [57]), a Message Passing Interface used in parallelization and supported by GEANT4.

A straightforward extension to the duct source would be to allow the duct to curve, creating uniform statistics on e.g. a line-of-sight guide. Whether this is of any significant benefit is to be determined considering that the straight duct source can be seen as a reasonable approximation of a guide with large radius of curvature.

A coupling of the duct source with logic for neutron optics has been implemented in PHITS [58] and should be investigated in GEANT4, too. The duct source efficiently delivers precisely the type of neutrons that interact with an optical guide, enabling studies of shielding concerns related to neutron optics, or even virtual neutron experiments.

An idea that was investigated during the course of this work was to account for material attenuation and fire more particles in directions of thick shielding. The idea did not pan out due to the scattering characteristics of low energy neutrons, but with the introduction of energy biasing it deserves a second look, as high energy neutrons are much more directional in terms of shielding penetration. The application of such a technique would likely only be appropriate for high energy neutrons.

A long-term outlook would be to study the effects of sky- and groundshine on neutron instruments.

Bibliography

- [1] W. J. L. Buyers, “Neutron and other stories from Chalk River”, *Physics in Canada*, vol. 56, no. 2, 2000.
- [2] H. Z. Bilheux, R. McGreevy, and I. S. Anderson, *Neutron Imaging and Applications: A Reference for the Imaging Community*, ser. Neutron Scattering Applications and Techniques. Boston: Springer, 2009.
- [3] J. Chadwick, “The existence of a neutron”, *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 136, no. 830, pp. 692–708, 1932. DOI: 10.1098/rspa.1932.0112.
- [4] A. K. Heller and J. S. Brenizer, “Neutron radiography”, in *Neutron Imaging and Applications: A Reference for the Imaging Community*, H. Z. Bilheux, R. McGreevy, and I. S. Anderson, Eds., Boston: Springer, 2009, pp. 67–80.
- [5] S. Peggs, R. Kreier, C. Carlile, et al., *ESS Technical Design Report*. Lund: European Spallation Source (ESS), 2013.
- [6] ESS. (2018). Building ESS, [Online]. Available: <https://europeanspallationsource.se/building-ess> [Accessed: 03/28/2018].
- [7] O. Hallonsten, “The politics of european collaboration in big science”, in *The Global Politics of Science and Technology - Vol. 2*, M. Mayer, M. Carpes, and R. Knoblich, Eds., Heidelberg: Springer, 2014, pp. 31–46.
- [8] D. Clery, “European neutron source finally finds a home”, *Science*, vol. 324, p. 1247, 2009.
- [9] Skånetrafiken. (Jan. 12, 2018). Spårväg Lund C – ESS, [Online]. Available: <https://www.skanetrafiken.se/aktuellt/sparvag-lund-c---ess/> [Accessed: 03/21/2018].
- [10] Lunds Kommun. (Jan. 12, 2018). Spårväg Lund C – ESS, [Online]. Available: <https://www.lund.se/sparvaglund> [Accessed: 03/21/2018].
- [11] ESS. (2018). Instruments, [Online]. Available: <https://europeanspallationsource.se/instruments> [Accessed: 03/25/2018].
- [12] G. J. Russell, M. M. Meier, and T. A. Broome, “Summary of discussion session on beamline shielding considerations for spallation neutron sources”, in *Proceedings of the ICANS-VI Conference*, 1982, p. 493.
- [13] N. Cherkashyna, K. Kanaki, T. Kittelmann, et al., “High energy particle background at neutron spallation sources and possible solutions”, *Journal of Physics: Conference Series*, vol. 528, no. 1, 2014. DOI: 10.1088/1742-6596/528/1/012013.

- [14] N. Cherkashyna, R. J. Hall-Wilton, D. D. DiJulio, et al., “Overcoming high energy backgrounds at pulsed spallation sources”, in *Proceedings of the ICANS-XXI Conference*, 2015, pp. 479–490.
- [15] K. Niita, H. Iwase, T. Sato, et al., “Recent developments of the PHITS code”, *Progress in Nuclear Science and Technology*, vol. 1, pp. 1–6, 2011.
- [16] H. Iwase, T. Nakamura, and K. Niita, “Development of general-purpose particle and heavy ion transport Monte Carlo code”, *Journal of Nuclear Science and Technology*, vol. 39, no. 11, pp. 1142–1151, 2002.
- [17] K. S. Krane, *Introductory Nuclear Physics*. Hoboken, NJ: John Wiley & Sons, 1988.
- [18] M. F. L’Annunziata, *Radioactivity: Introduction and History*. Amsterdam: Elsevier, 2007.
- [19] L. Pauling, “Atomic radii and interatomic distances in metals”, *Journal of the American Chemical Society*, vol. 69, no. 3, pp. 542–553, 1947. DOI: 10.1021/ja01195a024.
- [20] F. H. Allen, D. G. Watson, L. Brammer, et al., “Typical interatomic distances: Organic compounds”, in *International Tables for Crystallography vol. C: Mathematical, Physical and Chemical Tables*, E. Prince, Ed., Wiley InterScience, 2006, pp. 790–811.
- [21] P. M. Bentley, C. P. Cooper-Jensen, and K. H. Andersen, “High intensity neutron beamlines”, *Reviews of Accelerator Science and Technology*, vol. 6, pp. 259–274, 2013.
- [22] G. S. Bauer, “Physics and technology of spallation neutron sources”, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 463, no. 3, pp. 505–543, 2001.
- [23] G. J. Russell, “Spallation physics – an overview”, in *Proceedings of the ICANS-XI Conference*, 1990, pp. 291–299.
- [24] N. Watanabe, “Neutronics of pulsed spallation neutron sources”, *Reports on Progress in Physics*, vol. 66, no. 3, pp. 339–381, Feb. 21, 2003. DOI: 10.1088/0034-4885/66/3/202.
- [25] M. J. Berger, J. S. Coursey, M. A. Zucker, et al. (2017). Stopping-power & range tables for electrons, protons, and helium ions, [Online]. Available: <https://www.nist.gov/pml/stopping-power-range-tables-electrons-protons-and-helium-ions> [Accessed: 04/18/2018].
- [26] D. Filges and F. Goldenbaum, *Handbook of Spallation Research: Theory, Experiments and Applications*. Weinheim: Wiley, 2009.
- [27] K. Batkov, A. Takibayev, L. Zanini, et al., “Unperturbed moderator brightness in pulsed neutron sources”, *Nuclear Instruments and Methods in Physics Research A*, vol. 779, pp. 500–505, 2013.
- [28] F. Mezei, “The raison d’être of long pulse spallation sources”, *Journal of Neutron Research*, vol. 6, pp. 3–32, 1997.
- [29] W. T. Weng, “SNS accumulator ring design and space charge considerations”, in *AIP Conference Proceedings*, vol. 448, 1998, pp. 152–167. DOI: 10.1063/1.56768.

- [30] N. V. Shetty, “Study of particle transport in a high power spallation target for an accelerator-driven transmutation system”, PhD thesis, RWTH Aachen University, 2013.
- [31] Los Alamos National Laboratory. (2018). A general Monte Carlo N-Particle (MCNP) transport code, [Online]. Available: <https://mcnp.lanl.gov/> [Accessed: 04/09/2018].
- [32] V. G. Veselago, “The electrodynamics of substances with simultaneously negative values of ϵ and μ ”, *Soviet Physics Uspekhi*, vol. 10, no. 4, pp. 509–514, 1968.
- [33] A. Ghatak and K. Thyagarajan, *Introduction to Fiber Optics*. Cambridge: Cambridge University Press, 1998.
- [34] T.-r. Liang, F. Shen, T.-j. Liang, et al., “Validation of source biasing method for its use in CSNS beamline shielding calculation”, *Radiation Protection Dosimetry*, vol. 162, no. 3, pp. 208–214, 2014.
- [35] I. Lux and L. Koblinger, *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations*. Boca Raton: CRC Press, 1991.
- [36] J. K. Shultis and R. E. Faw, *An MCNP primer*, Department of Mechanical and Nuclear Engineering, Kansas State University, 2011.
- [37] J. F. Briesmeister, *MCNP – a general Monte Carlo N-Particle transport code*, ver. 4C, Los Alamos National Laboratory, 2000.
- [38] J. Stenander and D. D. DiJulio, “Application of automated weight windows to spallation neutron source shielding calculations using GEANT4”, *Nuclear Instruments and Methods in Physics Research A*, vol. 798, pp. 167–171, 2015.
- [39] A. J. van Wijk, G. Van den Eynde, and J. E. Hoogenboom, “An easy to implement global variance reduction procedure for MCNP”, *Annals of Nuclear Energy*, vol. 38, pp. 2496–2503, 2011.
- [40] T. E. Booth, “Genesis of the weight window and the weight window generator in MCNP – a personal history”, Los Alamos National Laboratory, Tech. Rep. Los Alamos Report LA-UR-06-5807, 2006.
- [41] A. Turner and A. Davis, “Improving computational efficiency of Monte-Carlo simulations with variance reduction”, in *International Conference on Mathematics and Computational Methods Applied to Nuclear Science & Engineering*, 2013.
- [42] F. Wasastjerna, “Using MCNP for fusion neutronics”, PhD thesis, Helsinki University of Technology, 2008.
- [43] M. A. Cooper and E. W. Larsen, “Automated weight windows for global Monte Carlo particle transport calculations”, *Nuclear Science and Engineering*, vol. 137, pp. 1–13, 2001.
- [44] J. Stenander, “GEANT4 simulations of neutron flux using automated weight windows with applications to ESS”, Master’s thesis, Lunds Tekniska Högskola, 2015.
- [45] ESS. (2018). DREAM – a versatile powder diffractometer, [Online]. Available: <https://europeanspallationsource.se/instruments/dream> [Accessed: 05/19/2018].

- [46] W. Schweika, N. Violini, K. Lieutenant, et al., “DREAM — a versatile powder diffractometer at the ESS”, *Journal of Physics: Conference Series*, vol. 746, no. 1, p. 012013, 2016. DOI: 10.1088/1742-6596/746/1/012013.
- [47] Copyright Holders of the GEANT4 Collaboration. (2006). GEANT4 Software License, [Online]. Available: <https://geant4.web.cern.ch/license/LICENSE.html> [Accessed: 05/04/2018].
- [48] The GEANT4 Collaboration. (2018). GEANT4 – a simulation toolkit, [Online]. Available: <http://geant4.web.cern.ch/> [Accessed: 05/04/2018].
- [49] The GEANT4 Collaboration, *Book for application developers release 10.4*, Rev1.0: Dec 8th, 2017, GEANT4 Collaboration, 2017.
- [50] D. D. DiJulio, C. P. Cooper-Jensen, H. Björgvinsdóttir, et al., “High-energy in-beam neutron measurements of metal-based shielding for accelerator-driven spallation neutron sources”, *Physical Review Accelerators and Beams*, vol. 19, pp. 053501-1–7, 5 2016. DOI: 10.1103/PhysRevAccelBeams.19.053501.
- [51] INCL. (2014). INCL the Liège Intranuclear Cascade model, [Online]. Available: <http://irfu.cea.fr/dphn/Spallation/incl.html> [Accessed: 05/14/2018].
- [52] INCL. (2014). INCL GEANT4 physics list, [Online]. Available: <http://irfu.cea.fr/dphn/Spallation/physlist.html> [Accessed: 05/14/2018].
- [53] G. F. Knoll, *Radiation detection and measurement*. New York: John Wiley & Sons, 1979.
- [54] M. Matsumoto and T. Nishimura, “Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator”, *ACM Transactions on Modeling and Computer Simulation (TOMACS) – Special issue on uniform random number generation*, vol. 8, pp. 3–30, 1 1998. DOI: DOI:10.1145/272991.272995.
- [55] *Creative Commons Attribution 3.0 Unported*, version 3.0, Creative Commons Corporation. [Online]. Available: <https://creativecommons.org/licenses/by/3.0/legalcode> [Accessed: 05/21/2018].
- [56] I. Antcheva, M. Ballintijn, B. Bellenot, et al., “Root - A C++ framework for petabyte data storage, statistical analysis and visualization”, *Computer Physics Communications*, vol. 180, no. 12, pp. 2499–2512, 2009. DOI: 10.1016/j.cpc.2009.08.005.
- [57] Message Passing Interface Forum, *MPI: A Message-Passing Interface Standard, Version 3.1*. Knoxville, Tennessee: University of Tennessee, 2015.
- [58] M. Hino, T. Oda, Y. Kawabata, et al., “Supermirror neutron guide system for neutron resonance spin echo spectrometers at a pulsed neutron source”, *Journal of Nuclear Science and Technology*, vol. 54, no. 11, pp. 1223–1232, 2017. DOI: 10.1080/00223131.2017.1359699.