A Review of Monte Carlo Methods and Their Application in Medical Physics for Simulating Radiation Transport

Joe Shields

University of New Hampshire

Follow this and additional works at: https://scholars.unh.edu/honors

Part of the Analytical, Diagnostic and Therapeutic Techniques and Equipment Commons, Computer Sciences Commons, and the Physics Commons

Recommended Citation


https://scholars.unh.edu/honors/657

This Capstone is brought to you for free and open access by the Student Scholarship at University of New Hampshire Scholars' Repository. It has been accepted for inclusion in Honors Theses and Capstones by an authorized administrator of University of New Hampshire Scholars' Repository. For more information, please contact Scholarly.Communication@unh.edu.
A Review of Monte Carlo Methods and Their Application in Medical Physics for Simulating Radiation Transport

Joseph Shields

Advisor: Dr. Karl Slifer
Co-Advisor: Dr. Dawn Meredith
Abstract

Monte Carlo methods are used to calculate statistical behavior through the use of random number generators and probability density functions. They have been used extensively in medical physics for research in radiotherapy, designing technology, dosimetry, and advanced clinical applications. This paper provides a background on Monte Carlo methods and a review of radiation therapy physics and dosimetry. Additionally, there is a discussion of the different ways Monte Carlo methods are used in medical physics as well as a review of current research related to Monte Carlo methods. The final portion of this paper contains my own Monte Carlo simulation using the EGSnrc software toolkit to carry out two different simulations. One simulation serves as a basic introduction to using the software and demonstrates some of its capabilities, while the other is a more complex simulation that models a realistic scenario in medical physics.
Contents

1 Background 3
  1.1 Review of Monte Carlo Methods 3
  1.2 Review of Radiation Therapy Physics and Dosimetry 6

2 Monte Carlo Methods in Medical Physics 12
  2.1 Applications in Dosimetry 12
  2.2 Applications in Modeling External Photon Beams 13
  2.3 Applications in Modeling External Electron Beams 15
  2.4 Applications in Quality Assurance 17
  2.5 Other Current Research 18

3 Monte Carlo Simulation Using EGSnrc Software 21
  3.1 Simple Simulation with Tantalum Slab 21
  3.2 LINAC and Beam Simulation 29

4 Conclusion 32

5 References 34
1 Background

1.1 Review of Monte Carlo Methods

The Monte Carlo (MC) method is a widely used technique with a variety of applications. It can be difficult to provide an exact definition for MC method due their diversity in application, therefore for our purposes it will be defined in the general context of medical physics as the following: Monte Carlo is a numerical method to solve equations or to calculate integrals based on random number sampling [28].

Random number generators (RNG) are required for a MC simulation in order to produce a large set of uncorrelated numbers. Since computer program outputs are inherently predictable, they must appear random. Therefore the result of these RNGs must be “pseudorandom”. A useful RNG for applications in radiation therapy must have the following attributes.

- Have a period long enough such that it is not used several times, making the results of the MC simulation correlated.
- The numbers must be uniformly distributed in multiple dimensions. More specifically, random vectors of $n$-tuple random numbers must be uniformly distributed in $n$-dimensional space.

One class of simple RNGs is the linear congruential generators which produces a sequence of integers $I_n$ by the recurrence relationship

$$I_{j+1} = (aI_j + c) \mod m$$

where $a$ is a multiplier, $c$ is an increment, and $m$ is a modulus. The CERN program library is a good resource for finding high-quality RNGs and test programs to observe the created sequences in multiple dimensions [22].

As a simple demonstration, we will use the MC method to calculate pi. Consider a box with an area $A_{\text{box}} = R^2 = 1$ containing $N_{\text{box}}$ sample points that are uniformly distributed. If the box contains a circle of area $A_{\text{cir}} = \frac{1}{4}\pi R^2$ that contains $N_{\text{cir}}$ sample points, then we can assume the following ratio relationships.

$$\frac{N_{\text{cir}}}{N_{\text{box}}} \approx \frac{A_{\text{cir}}}{A_{\text{box}}} = \frac{\pi}{4}$$

(2)
From this, we obtain the estimation for pi that can be calculated with the MC method [24].

\[ \pi \approx 4 \frac{N_{\text{cir}}}{N_{\text{box}}} \]  

(3)

The accuracy of this estimation increases as the total number of samples is increased. To demonstrate this example using the Python random module, I created the following script which provides a visualization for the MC method used, shown in Figure (1).

```python
import random
import matplotlib.pyplot as plt
import numpy as np

N = 1000
circle_samples = 0
square_samples = 0
x1 = []
y1 = []
angle = np.linspace(0, 2*np.pi, 150)
radius = 1
x2 = radius * np.cos(angle)
y2 = radius * np.sin(angle)
fig = plt.figure()
ax = fig.add_subplot(111)
for i in range(N):
    random_x = random.uniform(-1, 1)
    random_y = random.uniform(-1, 1)
    dist_from_origin = random_x**2 + random_y**2
    if dist_from_origin <= 1:
        circle_samples += 1
        square_samples += 1
x1.append(random_x)
y1.append(random_y)
pi = 4*(circle_samples/square_samples)
plt.title('N=1000')
ax.scatter(x1, y1, s = 1)
ax.plot(x2, y2, color = 'r')
ax.set_aspect(1)
print("Estimation of Pi=", pi)
```

The calculation was repeated for \(N=100\), \(N=1,000\), \(N=10,000\). The results of the estimation for pi was 3.2401, 3.0840, and 3.1416 respectively. The estimation could be repeated further with higher values of N to produce more accurate results.
Figure 1: MC estimations of pi with the total number of samples \( N = 100, 1000, 10000 \).

As the total number of samples increases, the accuracy of the estimation increases. The increase in accuracy also requires an increase in computing time. These are two important characteristics of MC simulations.

The MC method can be used as a stochastic method for numerical integration which is capable of solving equations that would otherwise be impossible analytically [28]. Consider an area \( A \) enclosed by the function \( f(x) \) on the interval \([a, b]\).

\[
A = \int_{a}^{b} f(x) \, dx
\]  

(4)

Then a randomly generated number \( \eta_i \) uniformly distributed in the range \([0, 1]\) can be scaled to the range \([a, b]\) by

\[
\xi_i = (b - a) \eta_i + a
\]  

(5)

thus \( \xi_i \) is uniformly distributed in the range \([a, b]\). Then we can give a rough estimate of the area \( A \) by

\[
A_1 = (b - a) f(\xi_1)
\]  

(6)

then repeating and improving this estimate by averaging the areas from two runs

\[
A_2 = \frac{1}{2} \left[(b - a) f(\xi_2) + A_1\right]
\]  

(7)

\[
= \frac{b - a}{2} [f(\xi_1) + f(\xi_2)]
\]

therefore the generalization becomes obvious after continuously repeating this process.

\[
A_N = \frac{b - a}{N} \sum_{i=1}^{N} f(\xi_i)
\]  

(8)

In the limit \( N \to \infty \), the estimated area \( A_N \) converges to the real integral \( A \). This method can be expanded to multiple dimensions. Assume \( f(x) \) is now a function that shall be
integrated in a volume $V$ with $D$ dimensions. Now instead of random numbers for the MC integration, we need a set of random points (or vectors) uniformly distributed in the multidimensional volume $V$. If we have our randomly generated set $\xi_1...\xi_N$, then this leads us to the basic theorem of MC integration which includes the uncertainty of the estimation

$$\int dV f(x) \approx V \langle f(x) \rangle \pm (b-a) \sqrt{\frac{\langle f^2(x) \rangle - \langle f(x) \rangle^2}{N}}$$

where Eq.(10) is the average function value and average function value squared respectively [28].

$$\langle f(x) \rangle = \frac{1}{N} \sum_{i=1}^{N} f(\xi_i) \quad \text{and} \quad \langle f^2(x) \rangle = \frac{1}{N} \sum_{i=1}^{N} f^2(\xi_i)$$

1.2 Review of Radiation Therapy Physics and Dosimetry

Radiation therapy uses ionizing radiation to harm and destroy cancer cells. Ionization is the process of a neutral atom either gaining or losing an electron, becoming a negatively or positively charged ion respectively. When charged particles such as protons, electrons, and $\alpha$ particles have a sufficient amount of kinetic energy capable of ionizing a neutral atom through collisions, they are known as directly ionizing radiation. Alternatively, neutral particles such as neutrons and photons capable of ionization are known as indirectly ionizing radiation [8].

The four primary processes responsible for ionization in radiotherapy are the Compton effect, photoelectric effect, coherent scattering, and pair production. Compton scattering occurs when an incident photon collides with a charged particle such as an electron. During the collision the energy and momentum are transferred to the charged particle while the photon deflects with reduced energy and a change in momentum [14].

The photoelectric effect occurs when an incident photon has a collision and transfers its entire energy $h\nu$ to the atom, where $h$ is Planck’s constant and $\nu$ is the frequency. This process ejects an orbital electron from the atom with energy equal to $h\nu - E_B$ where $E_B$ is the binding energy of the particular atom.

Coherent scattering, also known as Rayleigh scattering, can be described by considering the wave nature of electromagnetic radiation. This interaction occurs when an incident wave passes by an electron and sets it into oscillation, causing the electron to re-radiate the energy at the same frequency. This emits a wave with the same wavelength as the incident wave, thus no energy is transferred into the medium [8]. The intensity ($I$)
Figure 2: Diagram illustrating the Compton effect. In this interaction, the free electron receives energy from the incident photon and is emitted at an angle $\theta$. The incident photon, with reduced energy, is scattered at an angle $\phi$. Reproduced from [8].

Figure 3: Diagram illustrating the photoelectric effect. In this interaction, the entire energy of the photon is transferred to the atom which ejects an electron known as the photo-electron. The vacancy from the emitted photo-electron is filled by outer orbital electrons which emit characteristic x-rays. There is also the possibility of Auger electrons which are produced by the internal absorption of characteristic x-rays. Reproduced from [8].

of the scattered wave has the relationship $I \propto \frac{1}{\lambda^4}$, therefore this interaction is most dependent on short incident wavelengths.

Similar to the photoelectric effect, pair production results in total attenuation of the incident photon. Pair production occurs at higher energies of at least 1.022 MeV ($2m_e$). As the photon interacts with the strong electromagnetic field of the atom it undergoes a change of state, transforming into an electron and positron.
It is important to define the characteristics of the beam emitting x-rays or γ-rays from a radioactive source. These beams contain a large number of photons in a variety of energies [8]. The fluence (Φ) of a beam, which has units of m$^{-2}$, is analogous to flux in electromagnetism. It is defined as the number of photons ($dN$) that enter a cross-sectional area ($da$) [8].

$$\Phi = \frac{dN}{da} \quad (11)$$

Then naturally the fluence rate (ϕ), also known as flux density, is defined as the fluence per unit time.

$$\phi = \frac{d\Phi}{dt} \quad (12)$$

Similarly we can define the energy fluence (Ψ) and the energy fluence rate (ψ), also known as energy flux density or intensity.

$$\Psi = \frac{dE_n}{da} \quad (13)$$
\( \psi = \frac{d\Psi}{dt} \) \hspace{1cm} (14)

For a monoenergetic beam, the sum of the photon energies \( (dE_{\text{fl}}) \) is the number of photons \( (dN) \) times the energy of each photon \( (h\nu) \).

\[ dE_{\text{fl}} = dNh\nu \] \hspace{1cm} (15)

It’s also important to discuss how photons are attenuated when passing through material. This can be characterized by the intensity of the photons as well as the material it is traveling through [8].

\[ I(x) = I_0 e^{-\mu x} \] \hspace{1cm} (16)

Eq. (16) shows how the attenuation of a photon beam is described by an exponential function, where \( I(x) \) is the intensity as a function of distance, \( I_0 \) is the initial intensity, and \( \mu \) is the attenuation coefficient that is material dependent.

Radiation dosimetry deals with methods for quantitative determination of energy deposited in a medium through direct or indirect ionizing radiation [15]. A variety of quantities and key calculations will be defined in this section. Kerma, which stands for kinetic energy released per unit mass, is a non-stochastic quantity that is applicable for indirectly ionizing radiation such as photons and neutrons. It represents the energy transferred from indirectly ionizing radiation to the charged particles. It can be defined quantitatively as

\[ K = \Psi \left( \frac{\bar{\mu}_{\text{en}}}{\rho} \right) / (1 - \bar{g}) \] \hspace{1cm} (17)

where \( \bar{\mu}_{\text{en}}/\rho \) is the averaged mass-energy absorption coefficient and \( \bar{g} \) is the average fraction of an electron energy lost to radiative processes [8]. Kerma has units of J/kg or, in SI units, Gray (Gy). Kerma can also be divided into two components; inelastic collisions with atomic electrons \( (K^{\text{col}}) \) and radiative collisions with atomic nuclei \( (K^{\text{rad}}) \). Therefore Kerma can be written as a sum.

\[ K = K^{\text{col}} + K^{\text{rad}} \] \hspace{1cm} (18)

Similarly cema, which stands for converted energy per unit mass, is a non-stochastic quantity that is applicable for directly ionizing radiation such as electrons and protons. It represents the energy lost by charged particles \( (dE_c) \) in a unit of mass \( (dm) \) of material. It can be described quantitatively by Eq. (19) and it has the same units as Kerma.

\[ C = \frac{dE_c}{dm} \] \hspace{1cm} (19)
The linear stopping power ($S$) is defined as the expectation value of the rate of energy lost per unit path length of the charged particle ($dE/dx$). For electrons and positrons, Bethe theory is used to calculate stopping powers [15]. Additionally, the mass stopping power is defined as the linear stopping power divided by the density of the absorbing medium. The typical units for linear stopping power and mass stopping power respectively are MeV/cm and MeV·cm$^2$/g.

Absorbed dose ($D$) is applicable to both indirectly and directly ionizing radiation and can be defined as the mean energy ($\bar{\epsilon}$) imparted by ionizing radiation to matter of mass $m$. The absorbed dose is often usually derived from energy loss along a particle path-length segment, thus it’s related to particle fluence. It’s common to calculate fluence differential in energy ($\Phi_E$), which has units cm$^{-2}$MeV$^{-1}$ [1]. When calculated through MC simulations, for either charged or uncharged particles, the absorbed dose in the medium ($D_{med}$) can be calculated by the following equations.

$$D_{med}^{\text{CPE}} = \int_0^{E_{\text{max}}} \left[ \Phi_E \right]_{\text{med}} \left[ S_{el}(E)/\rho \right]_{\text{med}} dE$$  \hspace{1cm} \text{for charged particles} \hspace{1cm} (20)$$

$$D_{med}^{\text{TCPE}} = \int_0^{k_{\text{max}}} k \left[ \Phi_k \right]_{\text{med}} \left[ \mu_{en}(k)/\rho \right]_{\text{med}} dk$$ \hspace{1cm} \text{for photons} \hspace{1cm} (21)$$

Eq.(20) and Eq.(21) can also be used to calculate the absorbed dose in a detector ($D_{det}$). The acronyms over the equal signs CPE and TCPE stand for charge-particle equilibrium and transient charged-particle equilibrium respectively. Charged-particle equilibrium exists for a volume $V$ if each charged particle of a given type and energy leaving the volume is replaced by an identical particle entering [12]. This is for lower energy particles, typically below 500 keV, since attenuation can be neglected. If there is no CPE then the quantity calculated in Eq. (20) is not absorbed dose, instead it is cema ($C$). Transient charged-particle equilibrium occurs for higher energy particles since attenuation causes there to be less charged particles produced with increased depth [7]. If there is no TCPE then the quantity calculated is kerma ($K$). Additionally it is important to note $k$ as the photon energy and $S_{el}/\rho$ as the mass electronic stopping power (additional info can be found in ICRU Report 85 [1]).

Using Eq.(20) and Eq.(21), the Bragg-Gray stopping power ratio can be defined as

$$s_{BG}^{\text{med, det}} = \frac{\int_0^{E_{\text{max}}} \left[ \Phi_E \right]_{\text{prim}} \left[ S_{el}(E)/\rho \right]_{\text{med}} dE}{\int_0^{E_{\text{max}}} \left[ \Phi_E \right]_{\text{prim}} \left[ S_{el}(E)/\rho \right]_{\text{det}} dE}$$  \hspace{1cm} (22)$$
where we use the Bragg-Gray assumption which says that the cavity (detector) is so small that it does not disturb the fluence of the charged particles when inserted into the medium [5]. This gives us the condition $\Phi_{\text{med}} \approx \Phi_{\text{det}}$. It is also assumed that the primary charged-particle fluence does not include secondary or higher-order particles produced by collisions with the primary particles.
2 Monte Carlo Methods in Medical Physics

Monte Carlo methods help both clinical physicists and researchers in better understanding dose calculations and modeling a variety of radiation sources. This section explores the use of Monte Carlo methods in dosimetry, external beam source modeling, and for advanced treatment planning. Additionally, current research being done to improve Monte Carlo simulations and their efficiency for clinical use is also discussed in this section.

2.1 Applications in Dosimetry

When determining dose calculations experimentally, there are often quantities that are difficult or even impossible to calculate analytically. Therefore these quantities need to be determined numerically through the use of MC methods. In radiation dosimetry, detectors are typically composed of several different components. The materials of each of these components differ substantially from the medium where the absorbed dose is to be determined [28]. This leads to a well established problem that is characterized in terms of perturbation factors. Inserting a detector in a medium causes a change in the electron spectrum within the detector radiation sensitive volume relative to that in the homogeneous medium. This effect is known as perturbation [1]. Now taking into account these perturbation factors, and assuming the Bragg-Gray assumption is valid, the absorbed dose of the medium becomes

\[ D_{\text{med}} = \bar{D}_{\text{det}} \cdot s_{\text{BG}}^{\text{med,det}}(Q) \cdot p_{\text{det}}(Q) \]  

(23)

where \( p_{\text{det}}(Q) \) is the perturbation factors of the detector and \( Q \) is a given radiation beam quality. A major constraint imposed on the perturbation factors is that they are small and independent of each other. Under these assumptions, the perturbation factors can be described as the following by the different detector components or effects

\[ p_{\text{det}}(Q) = \prod_{i} p_{\text{det},i} \]  

(24)

\[ = p_{\text{dis}} \cdot p_{\text{wall}} \cdot p_{\text{fl}} \cdot p_{\text{cel}} \cdot p_{\text{stem}} \cdots \]

where \( p_{\text{dis}} \) accounts for the effect of replacing a volume of water by that of the detector, \( p_{\text{wall}} \) accounts for the presence of non-water-equivalent materials in the detector body and walls, \( p_{\text{fl}} \) corrects for the intrinsic difference in fluence between water and the detector volumes, and \( p_{\text{cel}} \) and \( p_{\text{stem}} \) correct for the presence of a central electrode and stem,
respectively, if they are relevant to the type of detector involved [1]. Other factors may be included as well depending on the case being considered.

\[
P_{\text{cal}} = \frac{D_2}{D_1} \quad P_{\text{stem}} = \frac{D_3}{D_2} \quad P_{\text{wall}}P_{\text{sleeve}} = \frac{D_4}{D_3} \quad P_{\text{dis}}P_{\text{cal}}S_{\text{w,air}} = \frac{D_5}{D_4}
\]

![Figure 6: Chain of dose ratios to calculate ionization chamber perturbation factors [1, 29]. Reproduced from [29].](image)

Difficulties in electron simulations has led to special consideration into the MC calculations of perturbation factors. The first MC calculations for ionization chamber perturbation factors were made for $^{60}$Co in-air measurements [3, 13, 4]. In a separate development by Wulff et al. [29], the perturbation factors from Eq.(24) are calculated using a chain of dose ratios that include the effects from different chamber components. This process is illustrated in Figure (6).

### 2.2 Applications in Modeling External Photon Beams

Monte Carlo methods are an incredibly powerful tool that can be used to accurately model radiation transport for applications in radiotherapy. A common use of MC modeling in external beam therapy is creating a virtual model of the radiation source. External beam therapy can be performed using lower-energy photon beams, electron beams, or hadron beams. Applications of these MC models include the design and optimization of the source, studying radiation detector response, and for treatment planning [28]. MC methods have provided a simple approach in deriving photon spectra from radiation sources. They were also used extensively in the design of linear accelerators (LINAC) for photon beam optimization. These MC models used a modular approach that are a popular practice today.

Clinically used photon beams produced by a LINAC typically have an energy range of 4-25 MeV. A generic design of a LINAC is shown in Figure (7). The order of the compo-
In a photon LINAC, the photons are produced by electrons colliding with a metal target. After that the beam is shaped by primary and secondary collimators. The flattening filter and wedge both act as attenuators to create a uniform field. Some LINACs have multileaf collimators to create beams that conform to the shape of the tumor. Reproduced from [28].

Components shown may vary depending on the manufacturer. In order to build a truly accurate model of the LINAC for a MC simulation, the components must be known in great detail. This includes; their mass densities; position, dimensions, and shape of defining surfaces; and their motion. In order to calculate uncertainties in the calculations of the simulation, it helps to determine the uncertainties of the physical qualities just listed. However, this has proven not to be a trivial task. To combat miscalculations, the simulations are validated against known dose calculations.

The BEAM/EGSnrc user interface is currently the most widely used LINAC simulation package. It allows the user to easily assemble a LINAC model and building blocks with a variety geometries. The code also allows for extensive beam analysis through the use of particle tagging and according to interaction types, interaction sites, etc. for photons, electrons, and positrons. The most important components in this model are the target,
flattening filter, secondary collimators, and wedges. Target and flattening filters are the most important sources of contaminant electrons, unless a wedge is present. For clinical applications, the MC method requires extensive information on the beam characteristics such as energy, angular, and spatial distributions of the particles in the beam. Efforts to improve these models are continuously being made, such as modeling a multiple-source photon beam based on the characterization of clinical photon beams used in radiotherapy [6].

2.3 Applications in Modeling External Electron Beams

Photon beams have alternative methods for calculating patient dose that rival the accuracy and performance of Monte Carlo simulations. For electron beams, this is not the case as MC methods are necessary for proper dose calculations. It is for this reason that MC methods have gained popularity in electron beam radiotherapy. Electron beam radiotherapy is administered less often than photon therapy. Typical electron beam energies for clinical use range from 4-20 MeV. At these energies, electron beams can be used for superficial tumors (5 cm or less) with a characteristically sharp drop-off in dose after the tumor [8].

A schematic for a Monte Carlo model of an electron beam LINAC is shown in Figure (9). This model differs from the photon beam LINAC in several key ways; the absence
Figure 9: Schematic of LINAC components in a Monte Carlo model of an electron beam LINAC. Unlike the photon LINAC, the electron LINAC has no metal target. It also has a scattering foil which broadens the beam and enhances uniformity. Reproduced from [28].

of the photon target; the presence of a scattering foil to broaden the beam; and a multistage collimator to help shape the beam close to the irradiated volume. The latter is necessary since electrons scatter much more in air than photons, therefore it is required that the beam is collimated close to the surface of the patient [28].

Similarly to photon beam LINACs, BEAM/EGSnr construction software is the primary tool for Monte Carlo simulations of the electron beam LINAC. It is known that electron beam models are particularly sensitive to the details of scattering foils, especially for fields of high energy. One study considered four different parameters of the scattering foils and found that the distance between the foils is critical [2]. Additional research is being carried out to improve computational time required for simulating electrons using variance reduction techniques.
2.4 Applications in Quality Assurance

In advanced radiotherapy, more complex techniques are employed in order to allow for precise dose distributions. 3D conformal radiotherapy (CFRT) has the high-dose volume match the target volume and avoids normal tissues. This allows for safer and more effective radiotherapy and it has been a major development in the field. Intensity modulated radiation therapy (IMRT) is the most advanced form of CFRT. It has the ability to automatically geometrically shape radiation fields, modulate the intensity of the radiation via computer control, verify that the radiation is being delivered accurately, and eliminate or quantify uncertainties [27].

To provide quality conformal dose distributions, these advanced forms of radiation therapy require complex treatment planning and a more complex beam delivery system. Furthermore, the patient anatomy, heterogeneity, organ motion, and deformation cause additional complexities. The long list of uncertainties and complexity of the treatment process, in addition to the serious consequences of any errors, requires there to be a comprehensive quality assurance (QA) to ensure that the patient will receive the proper dose distributions [28].

Monte Carlo simulations can be used as a comprehensive tool for QA since it can perform numerical experiments in the patient geometry and the treatment head. The kinematics of the radiation beam can be represented by a phase-space data set which allows for the retrieval of properties of the particles, such as type, energy, direction, and location. The data set can be generated from a set of parameters that analytically describe the beam, and its accuracy is critical for the performance of the MC simulation [23, 26]. In addition to the phase space data and patient specific data as specified by a CT or other imaging data, the MC QA analysis process requires treatment delivery information. This is depicted in Figure (10).

Through the use of MC simulations, a patients treatment plan can be verified with confidence. The MC calculated dose distribution incorporated into the patient’s geometry, setup, organ movement/deformation information can be used for treatment assessment and outcome analysis. These capabilities allow for advanced radiotherapy treatment to be performed reliably and with accuracy, making MC simulations a very valuable tool throughout the entire process.
Figure 10: Treatment beam information used for a Monte Carlo based dose verification. If the patient’s three-dimensional (3D) image data is available in real time, the actual dose received by the patient can be determined using Monte Carlo simulation and compared with that of the original treatment plan. This will verify the entire radiotherapy process for a particular treatment, which will be the most complete patient-specific treatment QA procedure. Reproduced from [28].

2.5 Other Current Research

Due to the nature of MC methods, the computation time that is required can be exhaustive and often impractical for clinical applications. A great deal of research is focused on solving these problems and progress has certainly been made throughout the years. These techniques that are used to speed up simulation times are known as variance reduction techniques (VRT) [10]. These VRTs can give more emphasis on the quantities of interest, therefore producing more relevant information for a given CPU time [16]. They operate either by manipulating the numbers and weights of the transported particles or by modifying the mean free paths for the relevant interaction processes. In a paper by Garcia-Pareja et al. [9], multiple VRTs were compared and assessed on their performance. Figure (11) shows the results of combining two VRTs; interaction forcing and splitting. Interaction forcing essentially shortens the mean free path to increase the frequency of interactions while the technique of splitting exploits local symmetries such as a particles that are symmetric under rotations around an axis.
Figure 11: X-ray emission spectra from a tungsten target being bombarded by 100 keV electrons at normal incidence. The left plot is from a 30 minute simulation. The right plot was generated in 15 minutes with the same code but also using the VRTs of interaction forcing and emission splitting bremsstrahlung photons and x-rays. Additionally, the error bars that represent statistical uncertainty have been significantly reduced from the left plot to the right plot. Reproduced from [9].

Another paper by Sarrut et al. [21] explores the use of artificial intelligence (or neural networks) being used for applications in Monte Carlo simulations. A neural network is composed of connected neurons that are typically organized in layers. A neuron in a neural network is similar to a biological neuron. It receives input from other neurons, performs some processing, and produces an output. The connections between the neurons have associated weights and each neuron has an associated activation function that generates the neuron’s output, for example a non-linear function which maps from an open domain to a closed domain. The input to the neuron’s activation function is the weighted sum over the outputs of all the connected neurons from the previous layer.

\[ x^{(i+1)} = f \left( W^{(i)} x^{(i)} + b^{(i)} \right) \]  

(25)

In Eq. (25), \( f \) is the activation function where \( x^{(i)} \) represents the output, \( W^{(i)} \) represents the weights matrix, and \( b^{(i)} \) represents the bias for the layer \( i \). The weights’ values are determined during the training phase, meaning each weight has a value that is optimized to adapt the neural network to handle a particular task. This is carried out with a training data set.
A Generative Adversial Network (GAN) is a special deep neural network that, once trained, can be used to generate data with similar statistics as the training set. Recent works in medical physics have explored the used of GANs to model particle source distributions and potentially speed up MC simulations [19, 20]. For these methods, the training data set for the GAN is a phase space file generated by a standard MC simulation which contains particle information (energy, position, direction) when they reach a specific surface. Once the GAN is trained, its network can act as a compact and fast phase space generator for MC simulations that replaces files that were several gigabytes in size with a neural network that is several megabytes in size. This neural network has the ability to generate a large number of particles which allows the user to speed-up the MC simulations significantly.

Simulations performed with the GAN as a phase space generator showed a very good dosimetric accuracy in comparison to the real phase space generator [21]. However, despite a large number of successful results, GANs have been shown to be notoriously difficult to train. They tend to suffer from issues such as instability which has led to attempts of various formulations based on different methods. An in-depth study on the most suitable variants for a MC simulation is yet to be undertaken.
3 Monte Carlo Simulation Using EGSnrc Software

3.1 Simple Simulation with Tantalum Slab

This portion of the paper will provide a brief overview of EGSnrc software and a few examples of simulations I performed using it. EGSnrc is a collection of generic routines that simulate the transport and interactions of ionizing radiation such as electrons, photons, and positrons in matter [25]. It is free to use and it has a wide range of applications utilizing radiation transport physics. Most applications are written in MORTRAN3 (native EGSnrc language) or C++, although they can also be written in FORTRAN and C.

![Schematic of EGSnrc applications and their corresponding languages. Reproduced from [25].](image)

In order to install and run EGSnrc on a Windows computer, it is required to have compilers for FORTRAN, C++, and C. Additionally, the GNU make utility is necessary and features such as the git utility, Tcl/Tk interpreter, and Grace plotting tool are strongly recommended [11].

For a simple demonstration of some of the capabilities of the EGSnrc software, I simulated radiation that is incident on a 1mm thick tantalum slab. This was carried out using a egs++ application, where egs++ is a C++ class library that allows for C++ applications to interface directly with the MORTRAN3 EGSnrc core [25]. The egs++ input file, titled slab.egsinp, is an input file that defines the conditions of the simulation. It contained the following contents (note that the syntax of the text editor is set to ruby since it resembles the syntax used by EGSnrc).
### RUN CONTROL

```
: start run control:
    ncase = 1e3 # The number of histories to simulate
: stop run control:
```

### GEOMETRY

```
: start geometry definition: # Only 1 geometry definition block allowed

### Define a plate of tantalum
    : start geometry: # Many geometry blocks can be defined
        name = slab # This name can be anything you like
        library = egs.ndgeometry
        type = EGS_XYZGeometry
        x-planes = -5, 5 # in cm
        y-planes = -5, 5 # in cm
        z-planes = -10, 0, 0.1, 10 # in cm
    : start media input:
        media = vacuum tantalum # Indexed as 0 1
        set medium = 1 1 # Set region 1 to medium 1 (tantalum)
        # Other regions default to medium 0 (vacuum)
    : stop media input:
    : stop geometry:

### Use the geometry with this name for the simulation
    simulation geometry = slab

: stop geometry definition:
```

### MEDIA

```
: start media definition: # Only 1 media definition block allowed
```

# Defining media in the input file like this is called "pegsless" mode

### Cutoff energies, in MeV

- $ae = 0.521$ # lowest energy for electrons (kinetic + 0.511)
- $ap = 0.01$ # lowest energy for photons (kinetic)
- $ue = 50.511$ # maximum energy for electrons (kinetic + 0.511)
- $up = 50$ # maximum energy for photons (kinetic)

### Tantalum

: start tantalum: # this name can be anything

  - density correction file = tantalum # name the file (no ext.)

: stop tantalum:

### Lead

: start lead:

  - density correction file = lead

: stop lead:

### Water

: start water:

  - density correction file = water, liquid

: stop water:

: stop media definition:

####################################

### SOURCE

: start source definition: # Only 1 source definition block allowed

### Pencil beam

: start source:

  - name = pencil_beam
  - library = egas, parallel, beam
  - charge = -1
  - direction = 0 0 1

: start spectrum:

  - type = monoenergetic
  - energy = 20 # in MeV

: stop spectrum:

: start shape:

  - type = point
position = 0 0 –10 # in cm

:stop shape:

:stop source:

simulation source = pencil_beam

:stop source definition:

#############################################################
### VIEWER CONTROL
#############################################################

:start view control:

# Here we are just assigning some colors for nice viewing in the egs_view application
set color = lead 120 120 200 200
set color = tantalum 120 255 120 255
set color = water 0 220 255 200

:stop view control:

#############################################################
### AUSGAB OBJECTS
#############################################################

:start ausgab object definition: # Only 1 ausgab definition block allowed

### Particle tracks
:start ausgab object:
name = tracks
library = egs_track_scoring
:stop ausgab object:

:start ausgab object:
name = dose
library = egs_dose_scoring
dose regions = 1
volume = 10
:stop ausgab object:

:stop ausgab object definition:

The key portions of this input file that are worth noting is the geometry, media, source, and ausgab objects. The geometry is where we define the object that the radiation will
be incident upon. This is where we create a tantalum slab with our desired dimensions.
Next, the media sections is where we define the range of allowed energies of the incident particles as well as the possible materials for the slab. Notice how we also define media for lead and water, which allows us to switch the material of our slab in the geometry section. The source section is where we define the shape and properties of our radiation source. We defined a source that will emit electrons (charge -1) in the shape of the tip of a pencil. Additionally, we gave the electrons an energy of 20 MeV and a starting position of (0,0,-10cm). Finally we have the ausgab section which allows us to add features such as particle tracking and dose calculations. The particle tracks allow for a visualization of the path of the particles before and after interacting with the tantalum slab. The dose calculations provides information on the dose deposited in every region of the geometry.

The simulation was ran for a couple different scenarios in order to demonstrate the physics capabilities of the EGSnrc code. First, the simulation was ran for 20 MeV incident electrons. The visual portion of the simulation output, produced by egs_view, is shown in Figure (14). The electron beam originates on the left side of the slab (negative z) in the pencil beam shape we had previously defined. The red lines represent electrons, the yellow lines represent photons, and the blue lines represent positrons. Additionally, the black regions before and after the green tantalum slab are a vacuum. Clearly there are a lot more particles on the right side of the slab than there are on the left. These particles

![Figure 14: Particle tracks for 20 MeV electrons incident on a tantalum slab.](image-url)
originate from the variety of mechanisms that were covered in 1.2. Although there are only a few, positrons were produced during the collisions. This is expected since the incident electrons are carrying 20 MeV of energy, well above the 1.022 MeV threshold necessary for pair production. The simulation was repeated with water and lead slabs.

<table>
<thead>
<tr>
<th>Medium</th>
<th>$\rho$ [g/cm$^3$]</th>
<th>$V$ [cm$^3$]</th>
<th>$E_{\text{dep}}$ [MeV]</th>
<th>$D$ [Gy]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ta</td>
<td>16.654</td>
<td>10.0</td>
<td>2.5699e0 ± 1.578%</td>
<td>2.4720e-12 ± 1.578%</td>
</tr>
<tr>
<td>Pb</td>
<td>11.350</td>
<td>10.0</td>
<td>1.6328e0 ± 1.901%</td>
<td>2.304e-12 ± 1.901%</td>
</tr>
<tr>
<td>$H_2O$</td>
<td>1.0</td>
<td>10.0</td>
<td>1.7423e-1 ± 1.140%</td>
<td>2.7912e-12 ± 1.140%</td>
</tr>
</tbody>
</table>

Table 1: Dose calculations for 20 MeV incident electrons.

The dose calculation, which was outputted in the command line, is shown in Table (1). As discussed before, the dose ($D$) is defined as the energy deposited per unit mass. The volume ($V$) and density ($\rho$) of the tantalum plate respectively are 10 cm$^3$ and 16.654 g/cm$^3$. Therefore the mass is 166.54 g and the dose can be calculated to be $2.472 \times 10^{-12}$ Gyu.

Next, I changed the incident particle to photons to compare its dose and tracks to that of the electrons. This adjustment is made by simply changing the charge to zero in the source section of the egs input file. The visual output produced by egs_view is shown in Figure (15). The simulation was carried out twice with two different energies, 1 MeV and 20 MeV. It is clear that more particles are produced by the 20 MeV photons than the 1 MeV photons. Additionally, there are no positrons produced by the 1 MeV incident photons since it did not reach the threshold for pair production. There is clearly pair production for the 20 MeV incident photons.

Shown in Table (2) and Table (3) are the dose calculations for 1 MeV and 20 MeV incident photons respectively. Notice that the dose is lower than that of the electrons by a factor of roughly 10 and 100 for the 1 MeV and 20 MeV incident photons respectively. As expected, this indicates that photons interact with matter less than electrons do. This decrease in interactions also led to a higher uncertainty since there were fewer events sampled in the slab. Additionally, the computation time was about 10 times faster for the photons since there were fewer events. The 1 MeV incident photon simulation was faster.
Figure 15: Particle tracks for 1 MeV incident photons (top) and 20 MeV incident photons (bottom), each incident on a tantalum slab.
than the 20 MeV incident photon simulation as well, the general rule of thumb being that lower energy simulations require less computational time.

<table>
<thead>
<tr>
<th>Medium</th>
<th>$\rho$ [g/cm$^3$]</th>
<th>$V$ [cm$^3$]</th>
<th>$E_{dep}$ [MeV]</th>
<th>$D$ [Gy]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ta</td>
<td>16.654</td>
<td>10.0</td>
<td>5.3022e-2 ± 11.755%</td>
<td>5.1003e-14 ± 11.755%</td>
</tr>
<tr>
<td>Pb</td>
<td>11.350</td>
<td>10.0</td>
<td>4.4328e-2 ± 13.426%</td>
<td>6.2566e-14 ± 13.426%</td>
</tr>
<tr>
<td>H$_2$O</td>
<td>1.0</td>
<td>10.0</td>
<td>1.8279e-3 ± 45.882%</td>
<td>2.9283e-14 ± 45.882%</td>
</tr>
</tbody>
</table>

Table 2: Dose calculations for 1 MeV incident photons.

<table>
<thead>
<tr>
<th>Medium</th>
<th>$\rho$ [g/cm$^3$]</th>
<th>$V$ [cm$^3$]</th>
<th>$E_{dep}$ [MeV]</th>
<th>$D$ [Gy]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ta</td>
<td>16.654</td>
<td>10.0</td>
<td>3.3692e-1 ± 12.104%</td>
<td>3.2409e-13 ± 12.104%</td>
</tr>
<tr>
<td>Pb</td>
<td>11.350</td>
<td>10.0</td>
<td>1.2804e-1 ± 15.469%</td>
<td>1.8072e-13 ± 15.469%</td>
</tr>
<tr>
<td>H$_2$O</td>
<td>1.0</td>
<td>10.0</td>
<td>4.6386e-4 ± 53.390%</td>
<td>7.4310e-15 ± 53.390%</td>
</tr>
</tbody>
</table>

Table 3: Dose calculations for 20 MeV incident photons.

This simple example provides a good introduction into how to use the egs++ code and for learning some of its basic, yet powerful, capabilities. We saw how electrons could be backscattered and how they produced secondary particles. We also saw how photons had less interactions and how varying their energies influences the production of secondary particles. Lastly, we also saw how dose calculations are made so that we can interpret our results with greater detail. The next simulation will use some more advanced features of EGSnrc software.
3.2 LINAC and Beam Simulation

The BEAMnrc application uses EGSnrc to accurately produce LINAC simulations. It has been used by many clinical groups to build radiotherapy modelling systems. Additionally, it is typically paired with a second application, such as DOSXYZnrc to perform voxelized transport (e.g. patient modeling) [25]. BEAMnrc uses a powerful graphical user interface (GUI) that allows users to create new accelerators and run simulations. It requires the installation of the Tcl/Tk packages that were mentioned prior.

The new accelerator was created so that the component module was set for dynamic mode, meaning the motion of the jaws of the LINAC can be synchronized with other components. The dynamic motion of the jaws was provided by the following text file.

```
Ex: 2 static fields, 1 dynamic, for 2 jaws
6
0.0
40, 50, -1, -1, -2, -2
51, 61, -1, -1, -2, -2
0.3
40, 50, -1, -1, -2, -2
51, 61, -1, -1, -2, -2
0.3
40, 50, 2, 2, 1, 1
51, 61, 2, 2, 1, 1
0.6
40, 50, 2, 2, 1, 1
51, 61, 2, 2, 1, 1
0.6
40, 50, 1, 1, -1, -1
51, 61, 0.05, 0.05, -0.05, -0.05
1.0
40, 50, 1, 1, -1, -1
51, 61, 1, 1, -1, -1
```

The format of the sequence file is described in section 15.3.8 of the BEAMnrc User Manual [18]. The sequence defines two static fields and one dynamic field in six steps. In units of cm, the upper y-jaws are at $40 \leq z \leq 50$ and the x-jaws are at $51 \leq z \leq 61$. From index 0.0 to 0.3 (30% of the simulation), the jaws are in a static position. The repeated indices, 0.3 and 0.6, result in simulated collimator shifts while the beam is off. Then from
index 0.6 to 1.0 there is motion of the \( x \)-jaws from a nearly closed position to a \( 2 \times 2 \) opening. The view of the first position of the jaws before any motion begins is shown in Figure (16).

![Figure 16](image_url)

**Figure 16:** Jaw positions of simulated LINAC before motion. The blue region represents air while the purple region represents media from the 700icru data file [17].

Once all parameters were properly set and the LINAC was compiled, the simulation was ran with 100,000 histories. The simulation ran fairly quickly and outputs a phase space file. This phase space file was then analyzed using the Grace plotting software. A scatter plot containing the phase space data was produced and is shown in Figure (17). The plot shows how the jaws collimated the source and as expected, there are two off-axis fields and one field in the center. It is easy to confirm that the field matches the \( 2 \times 2 \) collimator openings that were specified but the jaws sequence. It is also easy to see the other smaller openings on either side.

Next, to create an MC simulation on a patient phantom, measured images such as a CT image need to be converted into a egsphant format. A patient phantom acts as a stand-in for human tissue to ensure that systems and equipment will be working properly for the patient. The egsphant is a rectilinear voxel format in a plain text file. This essentially takes data structures that store geometric information in a continuous domain and converts them into a rasterized image (a discrete grid). CT and MRI images are acquired in a special digital format known as DICOM which ensures that the high quality of the images is retained. There is a tool distributed with EGSnrc called ctrechtate that can be used to convert DICOM files to the egsphant format.
Figure 17: X-Y scatter plot of phase space data from LINAC simulation. This plot demonstrates that the beam is primarily concentrated in the $2 \times 2$ opening specified by the jaws sequence.

Figure 18: Dose distribution created from the LINAC simulation and CT DICOM data. As expected, the primary concentration of the dose is in the center of the $2 \times 2$ opening as well as the center of the two other smaller openings.

For this simulation, a sample CT DICOM file was used from the EGSnrc github release page. The next stage of the radiotherapy simulation process, a dose calculation, can be performed now using this CT DICOM file. This was done using DOSXYZnrc and the BEAMnrc accelerator that was just created. The simulation was ran and it produced a 3D map of the dose. The distribution is shown in Figure (18).
4 Conclusion

This review provided a brief background on the physics of radiotherapy and dosimetry in order to give a more complete review of how Monte Carlo methods are used in medical physics. Monte Carlo methods are a great tool to accurately simulate radiation transport which allows physicists to simulate linear accelerators, perform dose calculations, and more. This makes Monte Carlo simulations a great tool for academic research and engineering in the field of medical physics. However, the extensive computational time required for complex Monte Carlo simulations make them impractical for routine clinical use. This has led to modern research that explores new ways to improve the computational time needed for the simulations.

Research by Sarrut et al. [21] demonstrates how the use of artificial intelligence shows promise, but some of the neural networks being used are proving to be notoriously difficult to train. Future research in the use of AI in Monte Carlo simulations is still needed to try and find more effective methods to train the neural networks that would make the simulations practical. Additionally, since neural networks in Monte Carlo simulations learn from properties of data sets, there is no guarantee that they are providing a plausible representation of the physics it is simulating. This is another challenge that will have to be overcome with further research.

Other research by Garcia-Pareja et al. [9] explores variance reduction techniques that also aim to reduce the computation time of Monte Carlo simulations. These techniques operate either by manipulating the numbers and weights of the transported particles or by modifying the mean free paths for the relevant interaction processes. Multiple techniques were found to be effective in achieving lower computation time if used in proper scenarios. Their work provided evidence of the usefulness of VRTs to speed up simulations of radiation transport.

The EGSnrc software is widely used throughout the medical physics community. It has vast capabilities that are used to simulate radiation transport and to model linear accelerators. This review provided a brief background into the EGSnrc code and explored some of the capabilities it has. This included an analysis of the dose on a slab of different materials when exposed to electrons or photons. We also saw how the EGSnrc code can be used to construct and compile a model for a linear accelerator. Using this model, we could simulate how radiation is delivered and received by a patient phantom provided by
CT DICOM data. These simulations provided an introduction to EGSnrc software, but future work can further explore the vast capabilities it has.
5 References


[12] C-m Charlie Ma and D Ph. Measurement of Radiation Table of Contents.


