



# WPI

## Particle Motion During Magnetron Sputter Deposition

A Major Qualifying Project Submitted to the Faculty of

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Bachelor of Science

By:

---

Benjamin R. Newmark

---

Kylie M. Sullivan

---

Blake Currier, Advisor

---

Joshua Cuneo, Advisor

## **Abstract**

Magnetron sputter deposition is integral in the creation of semiconductor devices like integrated computing chips and electron microscope slides. The goal of this project was to characterize target erosion during the magnetron sputtering process using both GEANT4 Monte Carlo simulations and experimental methods. This was completed using an electron motion simulation and sputtering experiments with copper and aluminum targets. These results showcased that target erosion is based on target material and magnet placement, not particle energy.

## **Acknowledgements**

We would like to thank Prof. Blake Currier and Prof. Joshua Cuneo for their support, without which this paper most likely wouldn't exist.

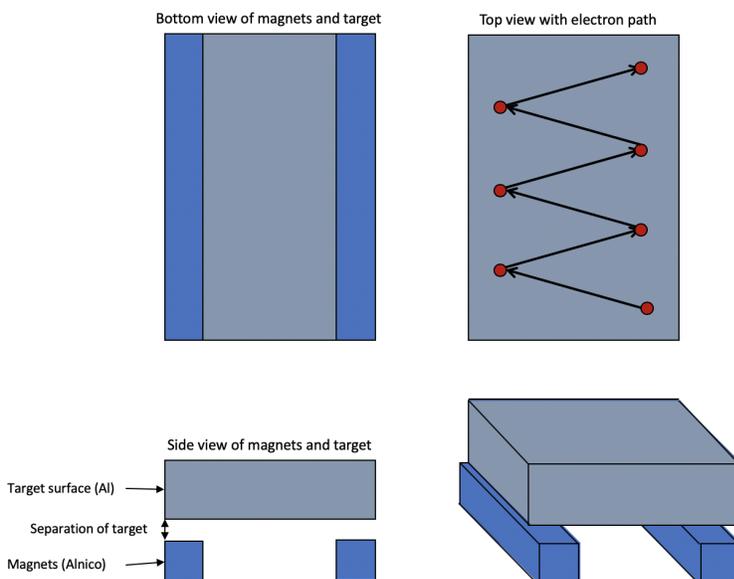
We would also like to thank Prof. Martin Gutsche for organizing this project and always being around to push us in the right direction.

Finally, we would like to thank our friends and family back home for supporting us from afar.

## Executive Summary

Magnetron sputter deposition is a plasma-assisted material transfer process. Plasma is confined to a small tank and used to remove small pieces of a target material which deposits onto a substrate. The plasma is confined and controlled using strong magnets and an electric field created within the tank. This work is interested in the motion of the electrons within the plasma, close to the target surface, and used GEANT4 simulations to evaluate how it could be altered based on magnetic field strength and location.

Real-world sputter deposition setups were investigated, and a model based on practical plasma motion was designed. Theoretical electron paths were described based on the design of simplified magnetron sputter deposition systems. The position of the electrons and velocity during the oscillations, due to the magnetic field between the two long magnetic poles and above the target surface, were observed, as illustrated below.



Simulations were programmed and run in Geant4, an object-oriented C++ Monte Carlo platform designed to simulate particles traveling through various types of matter. In this work, a target object was created over two magnets. Particles were fired from a particle gun across the target material (Al), and their interactions with the magnetic field were tracked using multiple “envelope” volumes that store information about the particle's position, velocity, and acceleration.

These simulations helped detect the amount of time spent by electrons in three even sections: near the north pole of the magnet, near the south pole of the magnet, and in between the two sections. The three detector volumes together cover the entire volume above the target. These borders are defined in even thirds across the surface of the target to help improve our understanding of the electron density profile. Sputtering experiments were also completed to compare the erosion pattern and amount of material deposited based off of various target materials. Copper and aluminum targets were sputtered until 1 gram of material was removed.

The goal of this project was to determine how and where the target material degrades. By tracking areas of high electron density, these simulations will show exactly what areas of the target will be worn down the fastest, as high electron density indicates a higher collision rate in that area. The sputter deposition geometry can then be adjusted by moving the magnets or adjusting the position of the target to more evenly distribute material removal, lengthening the life of the target and minimizing the cost of the overall process.

# Table of Contents

Abstract	2
Acknowledgements	3
Executive Summary	4
Table of Contents	5
Table of Tables	6
Table of Figures	7
Introduction	8
Background	9
Plasma Physics, Interactions, and Practical Applications	9
Plasma Behavior and Interactions	10
Individual Motion of a Single Charged Particle	10
Magnetron Sputter Deposition	12
Sputter Deposition Process	12
Computational Simulation of Particle Motion	14
Monte Carlo Methods	15
Geant4	15
Methods	17
1. Simulated (Geant4) Analysis	17
Electron Density Across the Target Material	18
Comparison of Geometries	20
2. Target Erosion of Varying Materials	21
Results	22
1. Simulated (Geant4) Results	22
Electron Density Across the Target Material	22
Benchmarking of Geometries	23
2. Experimental Target Erosion of Varying Materials	24
Discussion	29
Future Work	31
Simulate Entire Target	31
Simulate Ion Interaction	31
In-Lab Electron Monitoring	32
References	33
Appendix A: Simulation Data	37
Appendix B: Simulation Code Excerpts	39

## Table of Tables

<b>Table 1.</b> Percentage of steps taken by electrons in each of the three pre-defined volumes.	20
<b>Table 2.</b> Comparison of theoretical and simulated particle velocity.	22
<b>Table 3.</b> Copper target mass depletion.	23
<b>Table 4.</b> Aluminum target mass depletion.	24
<b>Table 5.</b> Copper target erosion rates.	25
<b>Table 6.</b> Aluminum target erosion rates.	26

# Table of Figures

<b>Figure 1.</b> Everyday plasma examples.	9
<b>Figure 2.</b> Plasma cyclotron gyration with guiding centers.	11
<b>Figure 3.</b> Parametric plot of cyclotron gyration.	11
<b>Figure 4.</b> Plasma particle motion in electric and magnetic fields.	12
<b>Figure 5.</b> Used aluminum sputtering target.	13
<b>Figure 6.</b> Diagram of the sputtering tank.	13
<b>Figure 7.</b> Diagram of ions bombarding the target surface.	14
<b>Figure 8.</b> Geant4 example visualization.	16
<b>Figure 9.</b> Geometry and dimensions of simulation.	18
<b>Figure 10.</b> Electron gun and electron direction.	18
<b>Figure 11.</b> Magnetic and electric field orientation.	19
<b>Figure 12.</b> Orientation of magnets and areas of observation.	20
<b>Figure 13.</b> Rectangular sputtering target.	20
<b>Figure 14.</b> Glowing plasma inside sputtering chamber.	21
<b>Figure 15.</b> Line graph depicting percentage of steps taken by electrons.	23
<b>Figure 16.</b> Copper target before and after sputtering.	25
<b>Figure 17.</b> Aluminum target before and after sputtering.	26
<b>Figure 18.</b> Copper target mass change over time.	27
<b>Figure 19.</b> Aluminum target mass change over time.	28
<b>Figure 20.</b> Simulated electron motion.	29
<b>Figure 21.</b> Electron energy Maxwell-Boltzmann distribution.	30
<b>Figure 22.</b> Whole target simulation setup with magnets.	31
<b>Figure 23.</b> Sputtering tank with plasma bulk.	32

## Introduction

Plasma is a state of matter in which an ionized gaseous substance becomes highly electrically conductive to the point that long-range electric and magnetic fields dominate the behavior of the matter (Chen, 2010). It is used in everyday objects such as neon signs and fluorescent light bulbs, as well as for research purposes (Kelley, M. C., Liley, B. S., 2019). It is typically studied for its interactions with other substances, as plasma conducts electricity and can be controlled using electric and magnetic fields unlike standard gases. Plasma beams, comprised of either negative electrons or positive ions, are used in particle accelerators to study the nature of matter (Princeton Plasma Physics Laboratory, 2019).

NTB Interstaatliche Hochschule für Technik, a University in Buchs, Switzerland, uses plasma to assist in the creation of semiconductors through a process called magnetron sputter deposition (NTB Interstaatliche Hochschule für Technik, n.d.). During this process, plasma electrons and ions are used to remove some amount of a target material, typically aluminum, gold, or another metal. The removed material is then deposited in layers onto a substrate that eventually becomes a semiconductor device.

The objective of this project is to characterize target erosion patterns. Using a combination of both Monte Carlo simulations and experimental results, areas of high particle density will be analyzed, as high particle density indicates high erosion rate (Geant4, 2019).

Beginning with a simplified simulation programmed in Geant4, a program specializing in physics-based particle simulations, electrons of various energies were analyzed. A dipole magnetic field was set up in conjunction with a constant electric field around a simulated aluminum target. Electrons were then injected into the system. The simulated electrons were tracked and timed as they moved through the system to better understand their behavior. Density profiles were compared for various electron energies. Sputtering experiments with varying target materials were conducted at NTB. From these experiments, data on erosion patterns of targets based on their material was taken.

The results of this project will help increase the longevity of target materials in magnetron sputter deposition techniques. As material is removed from the target, it is typically removed unevenly. Having a better understanding of the density of electrons near various parts of the target will allow researchers to better understand how ions are deposited on the substrate of semiconductor devices. Targets can be moved and rotated according to planned areas of high removal, thereby allowing the targets to be used for longer. Extending the lifetime of targets would lower the overall material cost of sputter deposition, making the process less expensive overall.

# Background

## Plasma Physics, Interactions, and Practical Applications

Plasma physics is the branch of physics that deals with the fourth state of matter. Plasma is defined as ionized gases where the atoms have broken down into their positive and negative components, ions and electrons (Chen, 2010). Neon signs, fluorescent light bulbs, lightning flashes, and Tesla coil arcs all contain plasma, examples of which can be seen in Figure 1 (Kelley, M. C., Liley, B. S., 2019). Neon signs and fluorescent light bulbs both operate on the same principle of trapping plasma in a small tube (Helmenstein, 2019). The plasma particles are then excited to the point where photons are emitted, creating the glow associated with such lights. Lightning and tesla coil arcs are very similar in that both involve the fast discharge of electricity from a negatively charged object, the clouds or a tesla coil, into a grounded object (Science Learning Hub, 2014). As the electricity is discharged through the air, gas particles become ionized allowing them to be visible to the naked eye.



**Figure 1.** Examples of plasma include (A) neon lights and (B) lightning strikes (Yellow Octopus, 2019, West, 2018).

Plasma is also used in labs to assist in various ventures, from cleaner energy and rocket propulsion to semiconductor creation. Plasma is used in the energy field as a heat source to convert water into steam, a possible replacement for fossil fuel burning in the future (Kelley, M. C., Liley, B. S., 2019). In rocketry, plasma propulsion is theorized to be more efficient than traditional solid or liquid fuels (University of Washington, 2012). Tests are being run in labs to see how different varieties of plasma-assisted propulsion compared to standard propulsion. Semiconductor devices can also be fabricated using a plasma-assisted deposition method (Hughes, 2014). Different layers of conductive substances are deposited onto a non-conductive surface to create integrated computing chips and transistors.

## Plasma Behavior and Interactions

Plasma interactions can be broken down into two categories, individual particle motion ([Background: Individual Motion of a Single Charged Particle](#)) and collective particle behavior and interactions (Chen, 2010). Individual plasma particle interactions are best explained in relation to standard gas particle interactions. Particles in gases are neutral and therefore move in straight lines, undisturbed, until they interact with another particle or force. Plasma particles are not neutral, however, and therefore interact differently. Plasma-neutral interactions can follow the same basic principles as neutral gas particle interactions, but plasma-plasma interactions cannot, and consequently fall under the umbrella of collective behavior.

The trait known as collective behavior is a defining characteristic of plasma and differentiates it from standard gases (Kelley, M. C., Liley, B. S., 2019). Collective behavior refers to motion that depends not only on the conditions immediately surrounding a particle but also on conditions in other parts of the plasma (Chen, 2010). Charged particles form local concentrations of positive or negative charge, which can interact at large distances. These interactions can alter the motion of other particles through induced electric fields, currents, or magnetic fields.

### Individual Motion of a Single Charged Particle

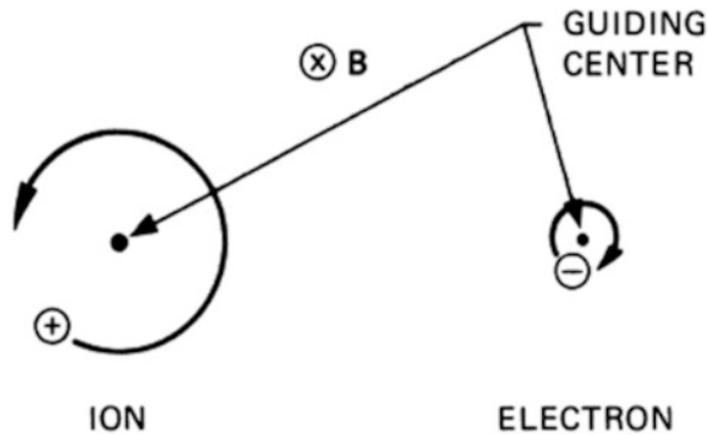
Individual plasma particle motion can be broken down into two main components: cyclotron gyration and guiding center drift (Chen, 2010). The first component, cyclotron gyration, is ruled by the magnetic field (Fox, 2018, Briesemeister, 2015). If looking at the particle from above, cyclotron gyration would appear as orbits around a central point, referred to as the guiding center, seen in Figure 2. The angular speed of these orbits is known as the *cyclotron frequency*, and is described by

$$\omega_c = \frac{|q|B}{m} \quad (1)$$

In Equation 1,  $\omega_c$  is the angular speed of the particle,  $q$  is the charge,  $m$  is the mass, and  $B$  is the magnetic field affecting the particle (Chen, 2010, Fox, 2018, Briesemeister, 2015). The angular speed is also used to characterize the size of the orbits, known as the *Larmor radius*. Dividing the relative velocity perpendicular to the magnetic field  $v_{\perp}$  by the angular speed  $\omega_c$  yields an orbit radius such that

$$r_L = \frac{v_{\perp}}{\omega_c} = \frac{v_{\perp}m}{|q|B} \quad (2)$$

It's important to note that, due to the presence of mass  $m$  in the denominator of Equation (1) and the numerator of Equation (2), lighter particles, such as electrons, would orbit both smaller and faster than heavier particles, such as ions (Briesemeister, 2015).



**Figure 2.** A depiction of plasma particle cyclotron gyration (Chen, 2010).

The complete motion of a particle in the X-Y plane, including both the Larmor radius and the cyclotron frequency, can be described using the force equation for the particle (Chen, 2010, Fox, 2018, Briesemeister, 2015). The force resulting in cyclotron motion is known as the *Lorentz Force*:

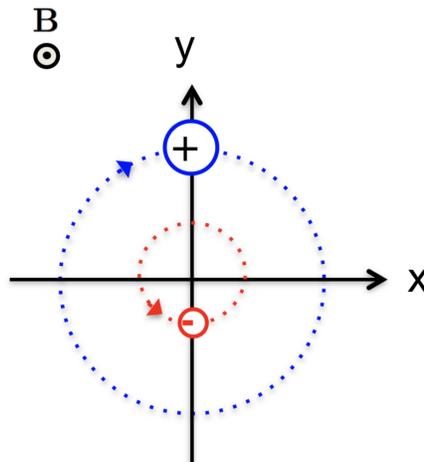
$$\underline{F} = m\underline{a} = q(\underline{v} \times \underline{B}) \quad (3)$$

Taking acceleration  $\underline{a}$  to be the second derivative of position, this equation can be integrated twice and solved, resulting in two equations, one for the X motion of the particle and one for the Y motion:

$$x - x_0 = r_L \sin(\omega_c t) \quad (4)$$

$$y - y_0 = \pm r_L \cos(\omega_c t) \quad (5)$$

The guiding center around which the particle orbits is located at  $(x_0, y_0)$ . The two equations can be graphed together using a parametric plot to produce paths like those in Figure 3.



**Figure 3.** Parametric plot of cyclotron gyration (Fox, 2018).

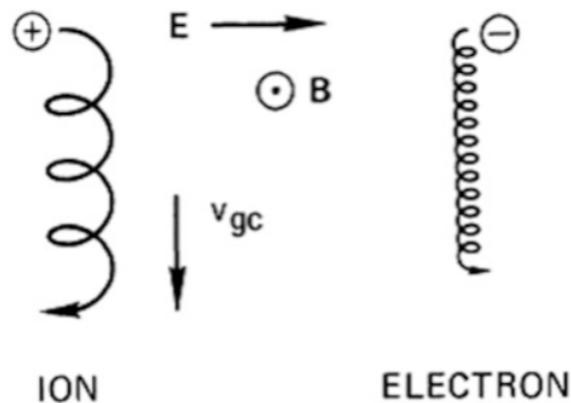
The second component of individual particle motion is guiding center drift. As mentioned previously, guiding centers are the points around which charged particles orbit in a magnetic field. However, in the presence of electric ( $\underline{E}$ ) and magnetic ( $\underline{B}$ ) fields, guiding centers also move at a constant velocity  $v_E$  such that

$$v_E = \frac{E \times B}{B^2} \quad (6)$$

(Chen, 2010, Fox, 2018, Briesemeister, 2015). If the electric and magnetic fields are perpendicular, Equation (5) simplifies to:

$$v_E = \frac{E}{B} \quad (7)$$

Guiding center drift and cyclotron gyration are two halves of individual plasma particle motion. Coupled together, they appear similar to the paths draw in Figure 4 and represent the motion seen in particles in many plasma-assisted lab techniques.



**Figure 4.** Particle movement in perpendicular electric and magnetic fields (Chen, 2010).

## Magnetron Sputter Deposition

Magnetron sputter deposition is a plasma-assisted material transfer technique used frequently in semiconductor device creation (Hughes, 2014). Plasma is used to remove material from a target, then the removed pieces build up in layers on the substrate that will eventually become a semiconductor. NTB Interstaatliche Hochschule für Technik uses this process to create integrated computing chips in particular (M. Gutsche, personal communication, 12 August 2019).

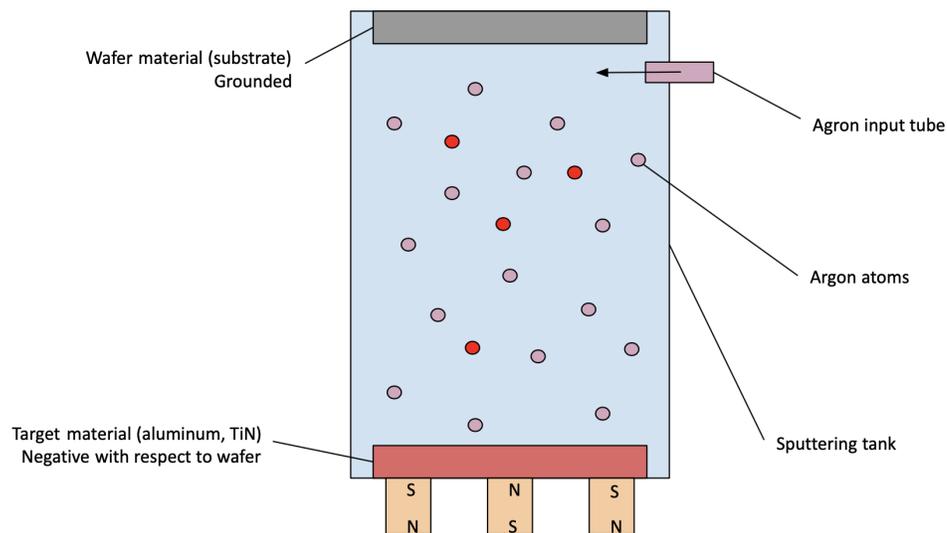
### **Sputter Deposition Process**

Magnetron sputtering is conducted in a closed chamber with vacuum pressure (Hughes, 2014). A target material, typically a metal disk or rectangular plate, and substrate, the object that will become a semiconductor device, are loaded into the chamber.



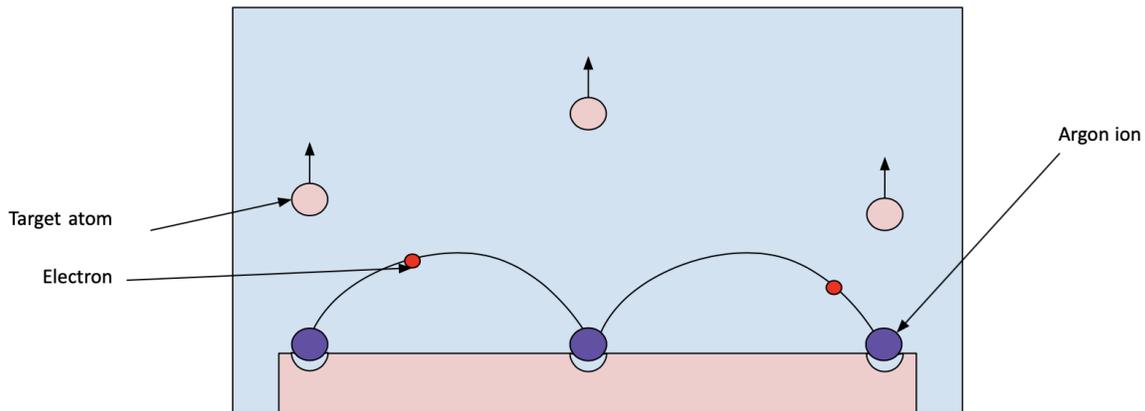
**Figure 5.** An example of a used aluminum sputtering target. The ring-shaped divots in the surface are referred to as racetracks. (Green, 2018)

The chamber is then filled with a noble gas. Argon is a popular choice due to the fact that it can be drawn from the air relatively easily (Hughes, 2014). A few free electrons are also present in the chamber. Extremely strong magnets are then placed along the bottom metal target in ring-like patterns. Typically, there is one pole in the center of the target and another around the outside edge of the target (M. Gutsche, personal communication, 22 August 2019). The target and the substrate are then charged, with the target being negative with respect to the grounded substrate (AJA International, 2019).



**Figure 6.** A diagram of the sputtering tank, prior to plasma ignition.

After the plasma is ignited, the electrons become much more energetic. The now-energetic electrons collide into the neutral argon atoms, ionizing them and creating more free electrons in the process (Hughes, 2014). The positive argon ions are attracted to the negative target material regardless of where the collision occurs and become trapped by the magnetic field. As the ions and electrons move through the electric and magnetic fields close to the target surface, both bombard the target. However, because the argon ions are so much heavier, they are the only particles that have any effect on the target.



**Figure 7.** A diagram of plasma bombarding the target surface.

As the ions bombard the target, they remove material. Due to the arrangement of the magnetic field, there are areas of higher material removal closer to the poles of the field (M. Gutsche, personal communication, 12 August 2019). These areas can be seen in Figure 7 and are called racetracks. The removal happens with such force that the target material is blown clear of the magnetic field. Due to the charge on the removed target atoms, they become attracted and stick to the grounded substrate. This process continues until a layer of target material of the desired thickness is achieved. Many substrates go through the sputtering process multiple times to create multiple layers of deposited target materials (EvaTec, 2019). Once all layers are complete, the wafers go through a secondary process to remove small sections of the deposited material, leaving behind a complete semiconductor, such as an integrated computing chip (M. Gutsche, personal communication, 27 August 2019).

## Computational Simulation of Particle Motion

Subatomic particles, such as electrons, are extremely small and are therefore unable to be seen by the naked eye (Balagopal, 2017). That does not mean that researchers are uninterested in how subatomic particles move and behave (Adelmann, 2013). Subatomic particles and their interactions are being studied globally, notably at institutes such as CERN (CERN, 2019). However, due to the fact that such particles cannot be easily observed during experiments, one of the most common methods to study their motion is computational motion simulations (Pease, 2018).

Computational physics analyses typically involve programs that use some degree of randomization to rule the behavior of particles. A specific example of this is the Monte Carlo method, which uses algorithms to make numerical estimates of unknown quantities (Pease, 2018). The Monte Carlo method has been used in this way to make predictions in particle accelerators (Adelmann, 2013, Pease, 2018). Using data from previous experiments and theory, algorithms are programmed to analyze the probability of thousands of separate occurrences, much faster than humans can.

### **Monte Carlo Methods**

Monte Carlo methods were originally devised to study nuclear fission during the Manhattan Project (Dizikes, 2010, Pease, 2018). This process has improved applicable in many areas of study including physics, finance, math, and data science. In general, Monte Carlo simulations are defined as methods that simulate realistic and probable outcomes of a problem using random inputs. The random inputs are modeled on a probability distribution (normal, log, discrete, etc.) with realistic caps (Dizikes, 2010).

All Monte Carlo methods, regardless of their defined purpose, begin the same way: with a series of known variables which can be calculated based on given data (Dizikes, 2010, Kenton, 2019). An example of this is using historical market data to predict daily price drift of a specific resource, like a stock (Kenton, 2019). By comparing the price of this resource from one day to the next day many times, it is possible to obtain a fairly accurate value for the daily price drift. The other major component of a Monte Carlo simulation is an unpredictable variable, or one that cannot be calculated by hand confidently. An example of this is market volatility; on any given day the market may behave differently, making finance predictions a common use for Monte Carlo methods. All unpredictable variables use a random input value, typically confined to a logical range based on what the variable represents (Dizikes, 2010). Once all predictable and unpredictable variables have been defined, calculations are carried out to assess whatever the original model was supposed to investigate (Kenton, 2019). In the case of the finance example, this would be the overall price of a resource including predicted daily drift and with unpredictable market volatility. A significant number of trials are then performed, each time with a new random input. The more trials performed, the more accurate the result. Once completed, Monte Carlo simulations can provide a good picture of the likely outcome of a problem.

It is important to understand that Monte Carlo methods cannot account for any unexpected event or anything that is not programmed into the calculation (Kenton, 2019). If, for example, the price of the stocks from the previous example were to unexpectedly skyrocket, the Monte Carlo simulation would not be able to predict that.

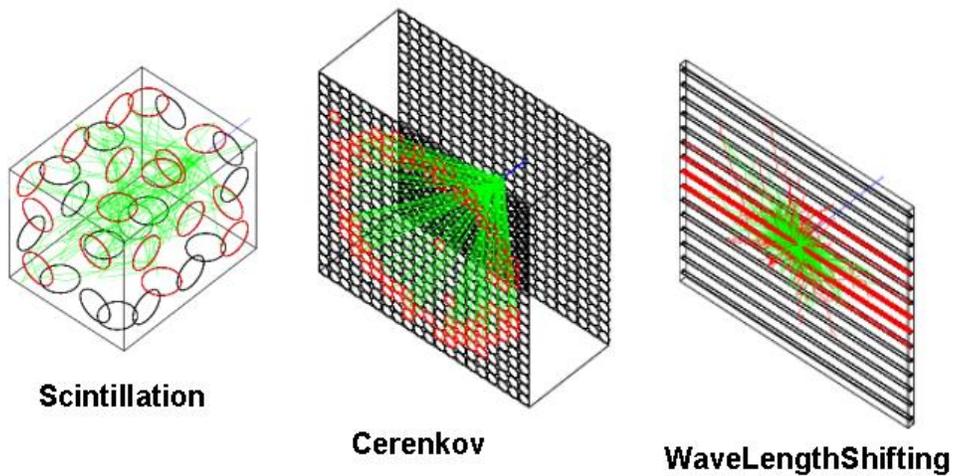
Geant4 uses Monte Carlo simulations to calculate the probability of certain events, such as particle creation, energy deposition, and particle movement (Geant4, 2019). As it relates to this project, Geant4 uses Monte Carlo methods to calculate the path of particles through a medium based on pre-defined factors, such as the presence of magnetic and electric field.

### **Geant4**

Geant4 is a program designed to simulate the “passage of particles through matter” (Geant4, 2019). The program was initially designed by CERN for use in high energy physics simulations, such as BaBar and the International Linear Collider project (Geant4, 2019). Geant4 uses Monte Carlo methods programmed in C++ to track detector response as well as run management, visualization, and user interface (Agostinelli, S. et al, 2003). Steady advancements have allowed Geant4 to also be used space science, astrophysics, medical physics, material science, and nuclear engineering. Geant4 models the path of particles through pre-specified mediums based on the density and cross section of materials as well as outside forces affecting the particles, as seen in Figure 8. Various Monte Carlo method algorithms can be

attributed to particles to alter how the particles behave in specific scenarios, such as in large particle colliders or small-scale magnetron sputter deposition tanks.

To create a Geant4 input file, a world volume and envelope volume must first be defined. Within a world, objects are defined as logical volumes that are given additional physical volumes. They need a shape, size, and material (CERN, n.d.). Objects and volumes can then be given field managers, such as magnetic or electric fields within them. Custom fields can be added and programmed to fit any specific requirements. Multiple objects can be added into a simulation's hierarchy, adding to the simulated world. When all objects are added, a step and run function must be created to collect data and execute the simulation. Once the program is completely established, it can be processed using visualization software. Geant4 is compatible with a variety of different visualization software, including HepRepFile, Ray Tracer, and OpenGL (Generowicz, 2002). Data can then be stored and analyzed based on the outputs of the programmed simulation.



**Figure 8.** An example of a Geant4 visualization involving photon paths (MacPhail, T., Gumplinger, P., n.d.)

# Methods

## 1. Simulated (Geant4) Analysis

The program is made up of multiple classes that are split into two main categories: those ran at initialization, and those used during a run. The main class of the program is `sputtering`, which initializes `sputteringDetectorConstruction`, `sputteringActionInitialization`, and `sputteringPhysicsList`. The first of these, `sputteringPhysicsList`, is a small class that runs at start-time. Its functions set up the native physics rules for the world in general, as well as specific rules for the magnetic and electric field and the electron particles. After `sputteringPhysicsList` finishes initializing, `sputteringDetectorConstruction` is ran. It contains all the data for each object within the simulation. It creates the logical volumes first, which contain the size, shape, and material for each object. It then creates the physical volumes for each object, which contains the location within the world and associated information such as rotation and mother volume.

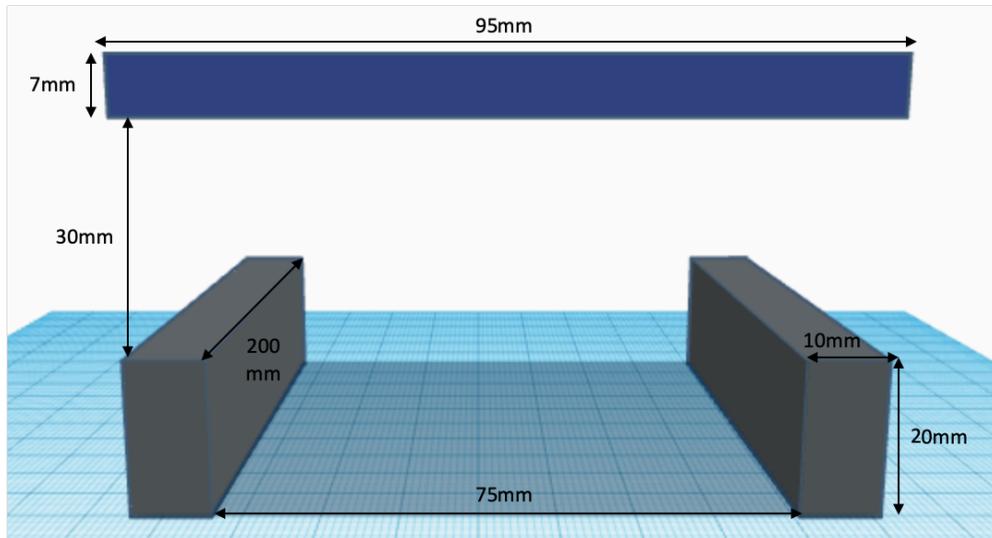
After creating the objects, `sputteringDetectorConstruction` creates the electric and magnetic field. The magnetic field code calls `dipoleField`, a custom-made non-uniform magnetic field. Geant4 only natively supports uniform magnetic fields, so it was necessary to create a custom class to create the desired field. There are two important functions, `getFieldValue` and `E.getFieldValue` is the main function and is required as a magnetic field class. It takes in a set of coordinates in (x, y, z) format and returns a vector in (vx, vy, vz) format. The second function, `E`, is a helper function that takes in the magnet's charge, as well as the coordinate location of the two magnets and the coordinates of the point requested. It calculates and outputs a partial version of the field strength for X and Y both as a Pair object ([Appendix B: Simulation Code Excerpts - E Helper Function](#)). `E` is called twice, returning a Pair each time. The two returned x-values are summed, and so are the two y-values. This process returns field lines of the correct shape, but of uniform strength.

Next, the field strength needs to be set. To do this, the x- and y-coordinates were taken into account to create a proper "modifier" value that would accurately set the field strength. The x- and y-based modifiers are calculated separately, and then averaged. Finally, the field value vectors were calculated (Mahdavi, Azadboni, Azadboni, 2011). The z-value vector would always be 0, but the x- and y-values were calculated using previously tested equations before being returned as a list of three values representing the field value vector ([Appendix B: Simulation Code Excerpts - Magnetic Field Calculation](#)).

The next class, `sputteringActionInitialization`, builds `sputteringRunAction` and `sputteringEventAction`, setting them as user actions. When a user triggers a run, these two classes, alongside `sputteringSteppingAction` and `SputteringPrimaryGeneratorAction`, will be called on. `sputteringRunAction` is used at the start and end of a run. At the start of a run, it resets the gathered data. At the end of a run, it takes and outputs all of the gathered data. `sputteringEventAction` is called by `sputteringSteppingAction`, which is used at the end of each "step", and checks if the particle has ended its step within one of the three scoring volumes - logical volumes created to track data. If it has, `sputteringSteppingAction` calls `sputteringEventAction` to save the data. `sputteringEventAction` will take in the information from `sputteringSteppingAction` - which will be a number representing which of the three scoring volumes the particle ended its step in (1, 2, or 3). `SputteringPrimaryGeneratorAction` is the final class used at start-of-run time. Within it, the particle gun is created and its properties are set. The gun is coded to fire electrons at a set energy, and has its momentum set by a directional vector. The gun's initial position is also set.

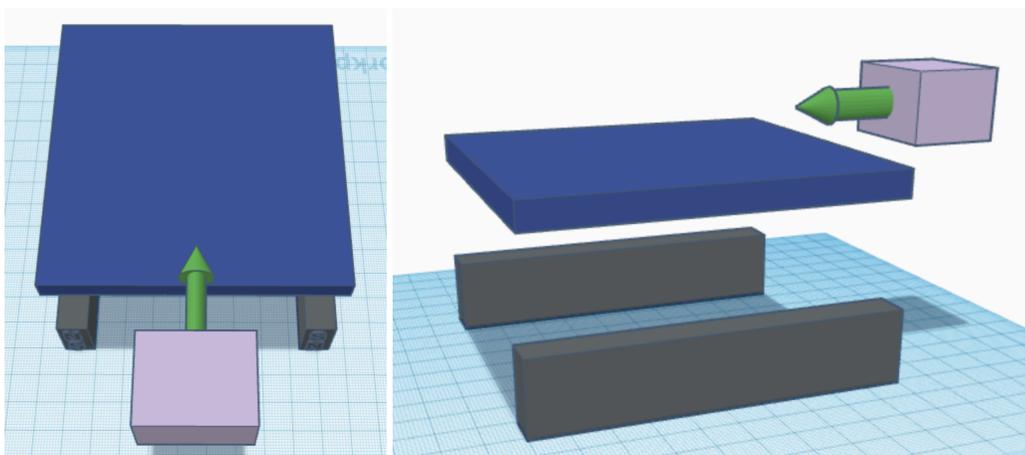
### **Electron Density Across the Target Material**

The simulation has two magnets, a target, an electron gun, and an electric field. The magnets are made of Alnico, a common iron alloy used in sputtering deposition magnets, and have dimensions of 10mm x 20mm x 200mm. The magnets are spaced 75mm apart. The aluminum target is 95mm x 7mm x 200mm and is centered 30mm above the surface of the magnets. The target and the area above it are contained in an envelope of argon gas. The geometry and dimensions of the target and magnets can be seen in Figure 9 ([Appendix B: Simulation Code Excerpts - Geometry Definition](#)).



**Figure 9.** The geometry and dimensions of the magnetron sputtering simulation, as programmed in Geant4.

The electron gun is located at one end of the target and fires a single electron across the surface of the target, allowing the simulation to analyze the position of the electron as it interacts with the system. The geometry of the electron gun can be seen in Figure 10.



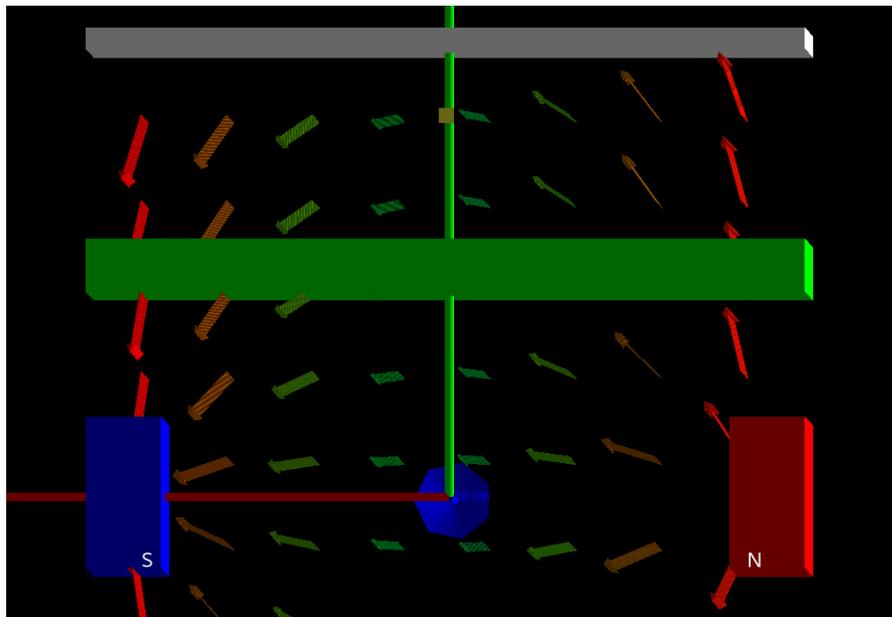
**Figure 10.** Depictions of the electron gun (lavender box) and the direction at which the electron is released (green arrow).

The electric field has a strength of 1000 Volts per centimeter up through the plane of the target. The modeled field is a dipole magnetic field, which causes the field to be stronger above the magnets and weaker towards the center of the target (Mahdavi, Azadboni, Azadboni, 2011). The equations used to describe the field are:

$$B_x = \frac{B_m d}{\pi} \frac{1}{\delta y} \delta \left( -\frac{\cosh\left(\frac{\pi x}{d}\right) \sin\left(\frac{\pi y}{d}\right)}{\cosh^2\left(\frac{\pi x}{d}\right) - \cos^2\left(\frac{\pi y}{d}\right)} \right) \quad (8)$$

$$B_y = -\frac{B_m d}{\pi} \frac{1}{\delta x} \delta \left( -\frac{\cosh\left(\frac{\pi x}{d}\right) \sin\left(\frac{\pi y}{d}\right)}{\cosh^2\left(\frac{\pi x}{d}\right) - \cos^2\left(\frac{\pi y}{d}\right)} \right) \quad (9)$$

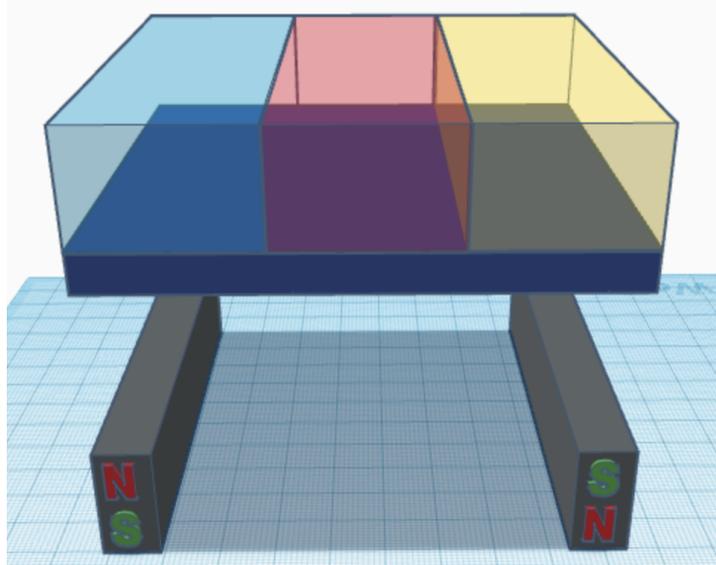
In these equations,  $B_m$  is the maximum strength of the magnets, 0.5 Tesla. The variables  $x$  and  $y$  are (x,y) coordinates, relative to the center of the system.  $d$  is the overall separation of the magnets, 75 millimeters. Collectively, these equations produce a two-dimensional, varying strength magnetic field that is used to trap electrons within the magnetic field ([Appendix B: Simulation Code Excerpts - Magnetic Field Calculation](#)). The varying strength magnetic field is depicted in Figure 11.



**Figure 11.** Magnetic field orientation, as it applies to the target surface. Field vectors vary in color based on their relative strength, with red being the strongest and green the weakest.

The simulation divides the target surface and the area above it into three even volumes. These volumes are defined as “north pole”, the third of the target closest to the north pole of the magnet indicated by the blue box, “south pole”, the third of the target closest to the south pole of the magnet indicated by the yellow box, and “center”, which covers the area in between the two indicated by the red box. The division of volumes can be seen by the transparent colored boxes in Figure 12. As the electron

travels across the surface of the target, it oscillates between the two poles while also drifting laterally across the surface of the target. The simulation tracks the average amount of time the electron spends in each section ([Appendix B: Simulation Code Excerpts - Detector Definition](#)). Different energy levels of electrons were tested to better understand the overall electron density characteristics close to the target. Comparisons of probable electron location based on energy can be found in [Results: Electron Density Across the Target Material](#).



**Figure 12.** The orientation of the magnets and the predefined sections of observation.

### **Comparison of Geometries**

The geometry simulated is a portion of the geometry used in the NTB sputter deposition process. In a lab setting the sputtering target is a rectangle with an electron path that winds across the surface in an oval shape, seen in Figure 13 as the divot in the target. The section simulated represents the straight portions of the electron path. Theoretical and simulated electron drift velocities were compared to better establish the accuracy of the simulation.



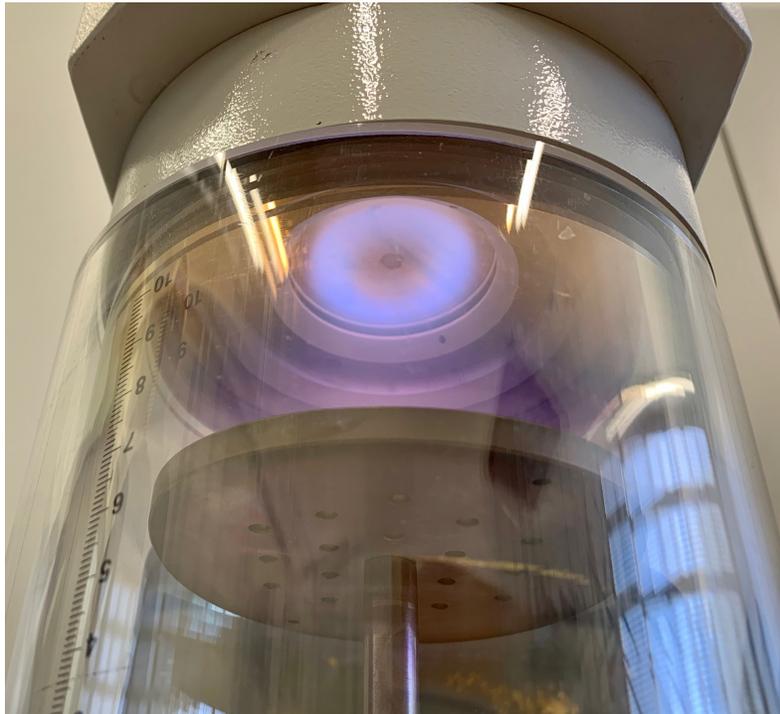
**Figure 13.** A used rectangular target, showcasing the oval-shaped electron path (Thin Film Consulting, 2015).

## 2. Target Erosion of Varying Materials

A series of experiments were completed at NTB to analyze the amount of material deposited and target erosion pattern based off of the target material. The sputtering chamber used was a Sputter Coater SCD005, designed to prepare slides for electron microscopes by depositing thin layers of conductive material onto the slide to limit static electricity during imaging. The process is used is functionally the same as discussed in [Background: Sputter Deposition Process](#), simply on a smaller scale.

To begin, the initial mass of the copper target was benchmarked. Then the target was loaded into the chamber. Following this, the sputtering process was initiated. For each trial, the target was sputtered until 0.1 gram of material was eroded. Targets were sputtered for 10 minutes, then 15 minutes, then a final 15 minutes. Mass measurements were taken after each sputtering event, resulting in 4 mass measurements per trial, including an initial measurement.

An interesting design feature of the smaller sputtering chamber is the inclusion of transparent chamber walls. These walls allow users to watch the sputtering process in real time, in particular, the location of the plasma based on its glow. This feature can be seen in Figure 14.



**Figure 14.** *Glowing plasma as seen through the sputtering chamber walls. The plasma racetrack on the surface of the target can be seen as the light purple ring.*

The copper target was then sputtered two more times, repeating the same procedure. The process was then completed with an aluminum target. Comparison between initial and final masses for each target and sputtering event, as well as qualitative descriptions of varying erosion patterns, can be found in [Results: Benchmarking of Geometries](#).

# Results

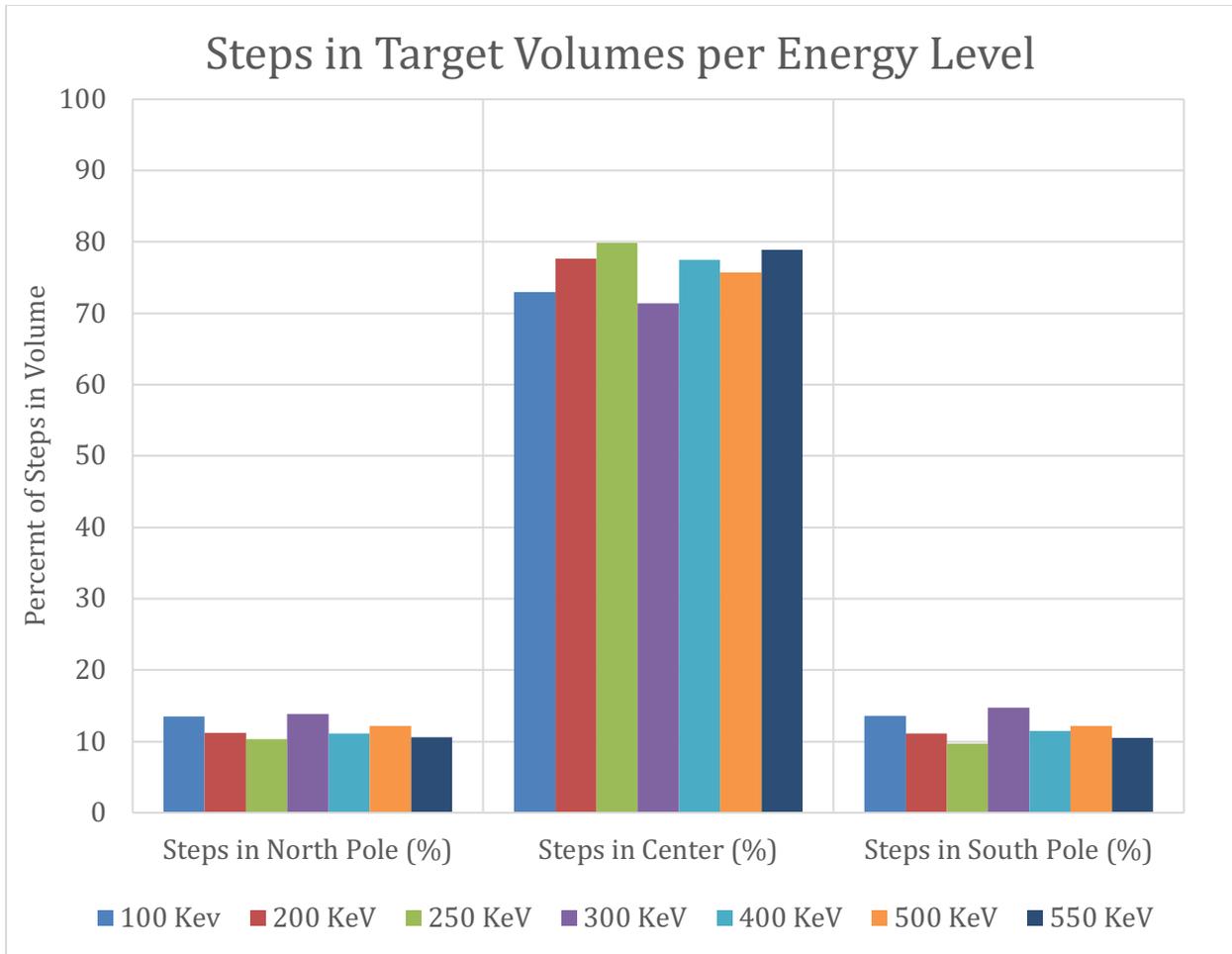
## 1. Simulated (Geant4) Results

### *Electron Density Across the Target Material*

Geant4 was used to complete Monte-Carlo simulations of electron movement across the surface of a magnetron sputter deposition target. The simulation tallied the number of steps taken by a particle within three equally defined volumes, as seen in Figure 12. Steps are defined as single particle movements that are limited to a very small distance. These steps were then summed and compared across the three volumes. Six runs were completed per energy level, and the resultant step-in-volume percentages were averaged. The complete raw data, including all of the runs, can be found in [Appendix A: Simulation Data](#). Table 1 shows the average percentage of steps taken by an electron in each of the three volumes. This data is again displayed in Figure 15 in the form of a line graph. Each of the lines on the chart represents a different energy level.

<b>Electron Energy (KeV)</b>	<b>Steps in North Pole (%)</b>	<b>Steps in Center (%)</b>	<b>Steps in South Pole (%)</b>
100	13.49	72.96	13.55
200	11.21	77.66	11.13
250	10.35	79.91	9.74
300	13.87	71.38	14.75
400	11.08	77.48	11.44
500	12.14	75.70	12.16
550	10.57	78.90	10.53

**Table 1.** Percentage of steps taken in each volume per energy level.



**Figure 15.** A bar chart representing the percentage of steps in each of the three pre-defined volumes for various energy levels.

### **Benchmarking of Geometries**

In order to properly benchmark the accuracy of the simulation, theoretical and simulated velocities were compared. As discussed in [Background: Individual Motion of a Single Charged Particle](#), the drift velocity of a particle is defined as  $v_E = \frac{E}{B}$ . In this simulation, this is shown as the velocity across the surface of the target. Using the simulated magnetic field strength and electric field strength at 15 points, the theoretical drift velocity was calculated. The 15 points used were split evenly between the three detector volumes. The simulated electric field was constant for all points at 1000 Volts per centimeter. The simulated drift velocity was calculated using the particle momentum across the surface of the target. Table 2 shows the simulated and theoretical drift velocities as well as the percent error in the simulation.

## TABLE TO BE COMPLETED

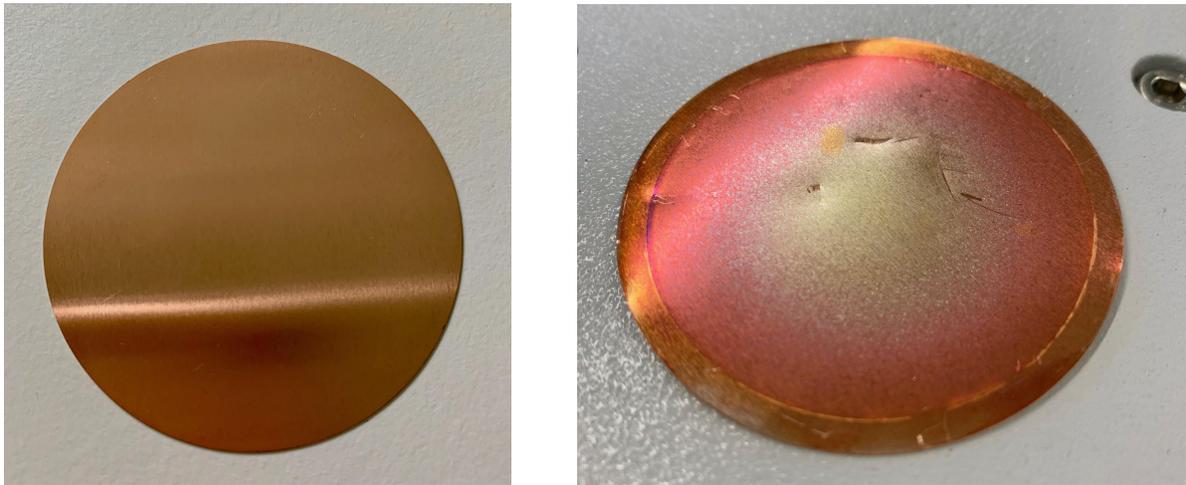
*Table 2. Comparison of theoretical and simulated particle velocity.*

### **2. Experimental Target Erosion of Varying Materials**

Two different targets were sputtered to investigate varying erosion patterns and rates based off material. First, a copper target was sputtered until 0.1 gram of material was eroded. This was done in separate, timed sputtering events. After every timed sputtering event, notes on erosion patterns were taken as well as the mass of the target. This procedure was repeated twice more, resulting in 3 trials and 12 total mass measurements, as seen in Table 3. Images of the copper target before and after the 12 sputtering events can be seen in Figure 16. An aluminum target was sputtered in exactly the same manner as the copper target. Aluminum target mass values and before and after images can be found in Table 4 and Figure 17, respectively. The slight increase in mass during the third aluminum trial is likely due to oil being deposited onto the target by touching it.

Trial	Time (s)	Mass (mg)	Percent Difference
<b>Pre-Sputtering</b>	0	4037.9	---
<b>1</b>	600	3991.57	1.15399896
	1500	3943.29	1.21690868
	2400	3926.67	0.42236555
<b>2</b>	3000	3870.6	1.43819568
	3900	3813.58	1.48408809
	4800	3769.47	1.16338413
<b>3</b>	5400	3737.95	0.83970259
	6300	3706.3	0.85032072
	7200	3646.63	1.62302647

**Table 3.** Copper target mass depletion for 3 trials.



**Figure 16.** Copper target (A) before and (B) after sputtering.

Trial	Time (s)	Mass (mg)	Percent Difference
Pre-Sputtering	0	1257.77	---
1	600	1257.63	0.01113143
	1500	1257.404	0.01797192
	2400	1257.136	0.02131603
2	3000	1256.97	0.01320549
	3900	1256.67	0.02386977
	4800	1256.68	-0.0007958
3	5400	1256.71	-0.0023872
	6300	1256.69	0.00159147
	7200	1256.68	0.00079574

**Table 4.** Aluminum target mass depletion for 3 trials.



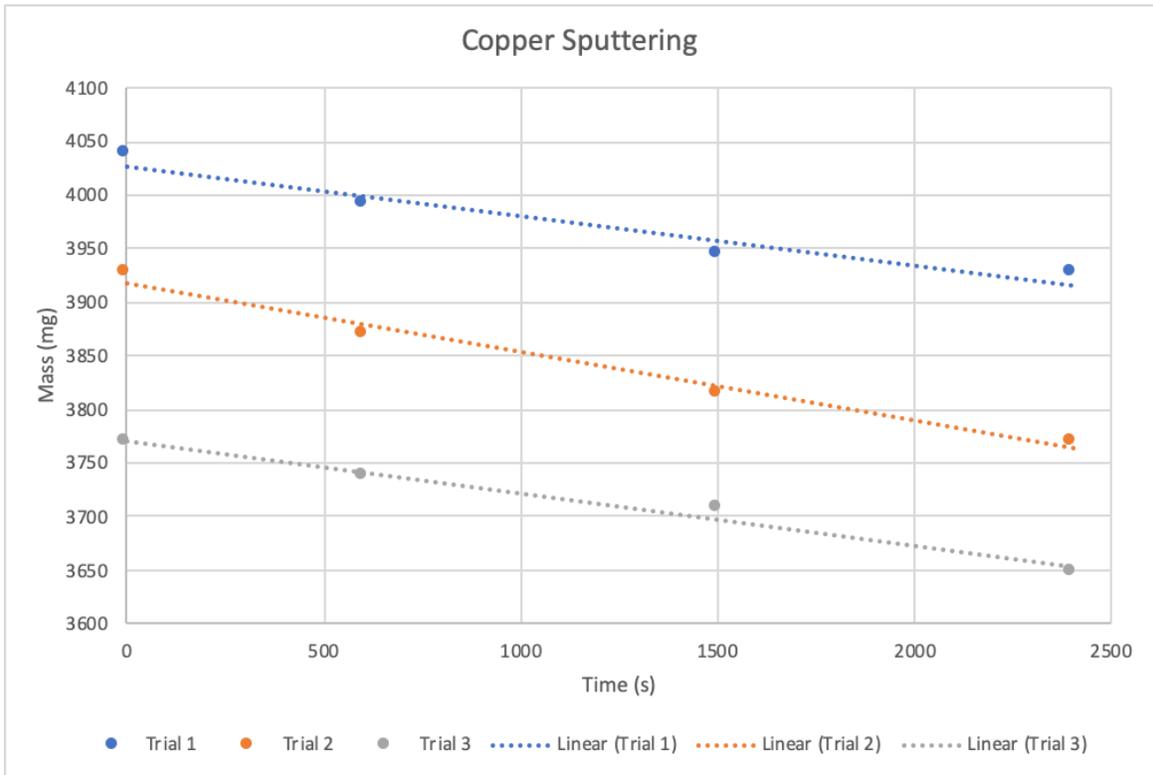
**Figure 17.** Aluminum target (A) before and (B) after sputtering.

Qualitative results for both target materials indicated that at the beginning of the targets lifetime, sputtering was slower and more erratic. As the target is used more, areas of higher erosion become increasingly more apparent. This is reflected in Figures 16 and 17, where the racetracks are clearly visible to the naked eye.

Using mass data, erosion rates were compared for copper and aluminum. Erosion rates were calculated using a linear trendline fit to each sputtering trial. Copper erosion rates can be found in Table 5 and Figure 18. Aluminum erosion rates can be found in Table 6 and Figure 19.

Copper Erosion Rates	
Trial	Mass Change Rate (mg/s)
Trial 1	$-4.63 \times 10^{-2}$
Trial 2	$-6.45 \times 10^{-2}$
Trial 3	$-4.95 \times 10^{-2}$

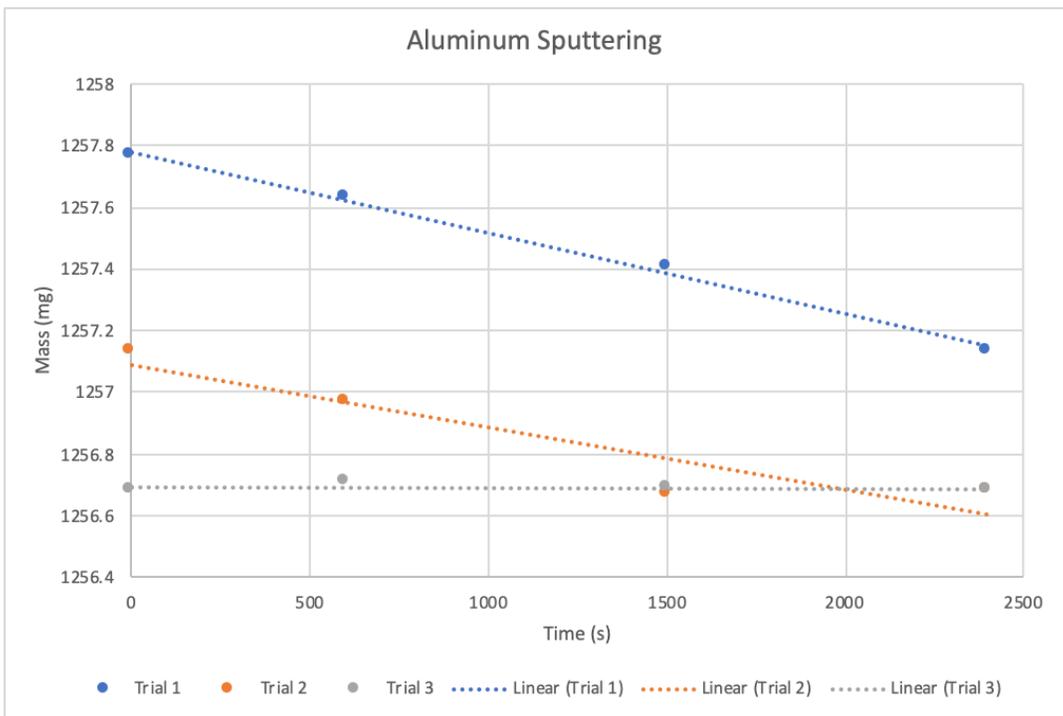
**Table 5.** Copper target erosion rates by trial.



**Figure 18.** Copper sputtering mass change over time by trial.

Aluminum Erosion Rates	
Trial	Mass Change Rate (mg/s)
Trial 1	$-3 \times 10^{-4}$
Trial 2	$-2 \times 10^{-4}$
Trial 3	$-6 \times 10^{-6}$

**Table 6.** Aluminum target erosion rates by trial.



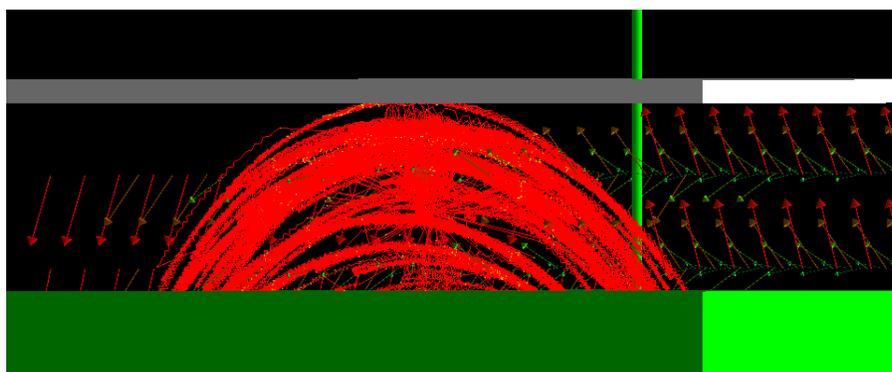
**Figure 19.** Aluminum sputtering mass change over time by trial.

## Discussion

Magnetron sputter devices are designed in a way that causes targets to wear unevenly. Magnets placed below the targets dictate the path of charged particles, such as ions and electrons, around the target in racetracks. These racetracks can be seen on the surface of targets, even after sputtering events are over, because they wear much faster than the rest of the target. When a target is declared no longer usable, the racetracks have worn very thin but there is usually a large amount of material left. In order to effectively combat uneven target erosion and the waste of useful target material, it is important to understand the process of target erosion. Using a combination of electron motion simulations and experimental controlled sputtering trials, areas of high target erosion were analyzed.

The Geant4 electron motion simulations tracked the number of steps, or movements limited to a very small distance, taken by electrons in each of three defined volumes. Comparing these results across multiple energy levels shows that electrons spend a majority of their time moving through the center of the racetrack, regardless of energy level. This is logical, as plasma used in magnetron sputtering has electrons of many different energy levels, but the racetrack remains the unchanged.

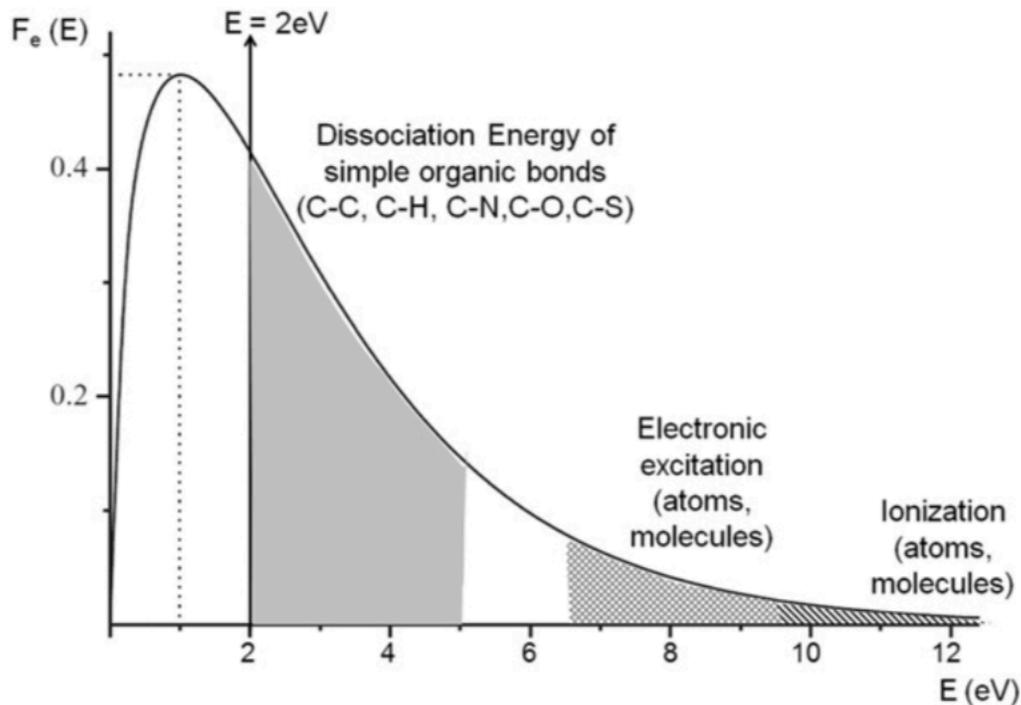
The largest discrepancy between the Geant4 simulation and the sputtering experiments was in the lateral movement across the target. As seen in the experiments, charged particles move across the target in ring like patterns. This is due to the presence of an electric field which gives the particles a drift. Despite the presence of a strong electric field in the simulation (1000 Volts per centimeter), particles did not drift across the target. Particles, instead, oscillated from magnet to magnet without drifting, as seen in Figure 20.



**Figure 20.** *The simulated particles oscillated in place, rather than oscillating while drifting across the target, despite the presence of a strong electric field.*

Another discrepancy between simulation and physical sputtering is the energy of the electrons used. In practice, the average electron energy during a sputtering event is 2 eV and energy varies with a Maxwell-Boltzmann distribution, as seen in Figure 21. In this figure, the average value of 2 eV is shown by the vertical arrow through the distribution and the upper limit of electron energy is shown to be close to 12 eV. When 2 eV runs were attempted using the simulation, electrons did not appear to move. In order to properly analyze electron movement, the energy was increased to a point far beyond typical sputtering event energies. However, varying the energy past that point did not appear to alter the path of electron

motion. Based on the lack of variance in electron motion, it is anticipated that at 2 eV electrons would behave in the same manner that they do at 100-550 KeV.



**Figure 21.** An example Maxwell-Boltzmann energy distribution where the average electron energy is 2eV (Thiry, Cornil, Konstantinidis, Syders, 2016).

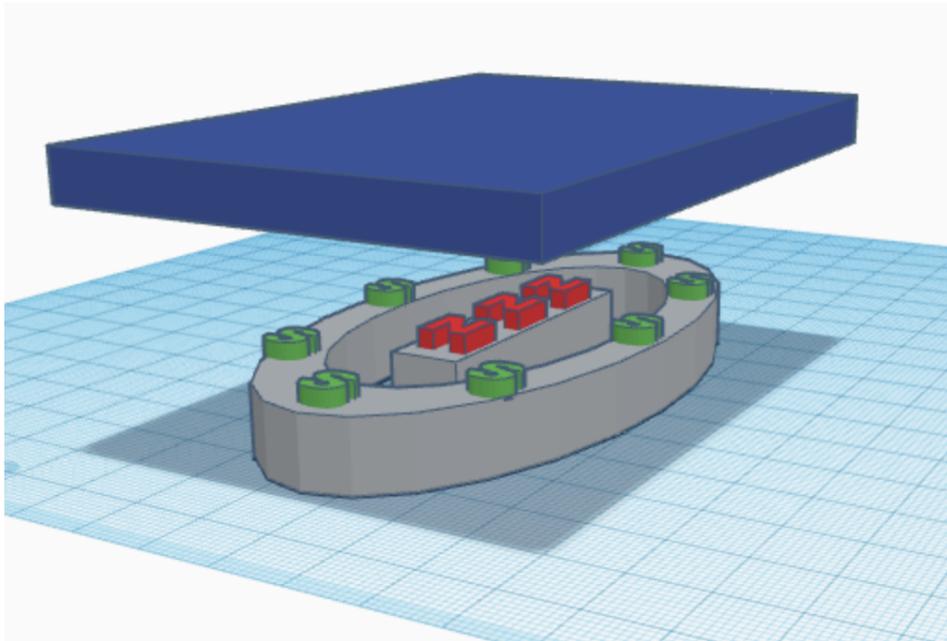
Despite the lack of electron drift in the simulations, the results obtained still help to better illustrate how targets erode. Regardless of particle energy, the target erosion pattern is similar. This indicated that target erosion is therefore based on magnet strength and placement, rather than particle energy. Experimental results indicated that target material also impacts erosion patterns.

Armed with the knowledge gained from this study, further explorations of magnet strength and position could be conducted to limit target erosion. The simulation in its current state could be utilized to predict areas of high particle density for various magnet orientations, which will assist in determining the most ideal orientation for even target erosion. Even target erosion would extend the lifetime of targets, reducing the overall cost of magnetron sputter deposition.

## Future Work

### Simulate Entire Target

In this project, a small section of the target was selected to be analyzed. The straight portions of the target were simulated, but they do not accurately demonstrate the entire sputtering process. In order to properly reflect the magnetron sputter system at NTB, an entire target should be simulated. This would require adjusting the magnets and magnetic field. The magnets would need to be expanded such that the north facing magnet runs through most of the middle of the rectangular target and the south facing magnet wraps around the edges of the target as seen in Figure 22. Currently, the magnets and electric field are arranged such that charged particles would leave the surface of the target, but this is not realistic. Particles are retained to decrease the amount of time a single sputtering event takes. Simulating the entire target would allow users to see how charged particles continue to behave, past the straight sections on either side of the target.

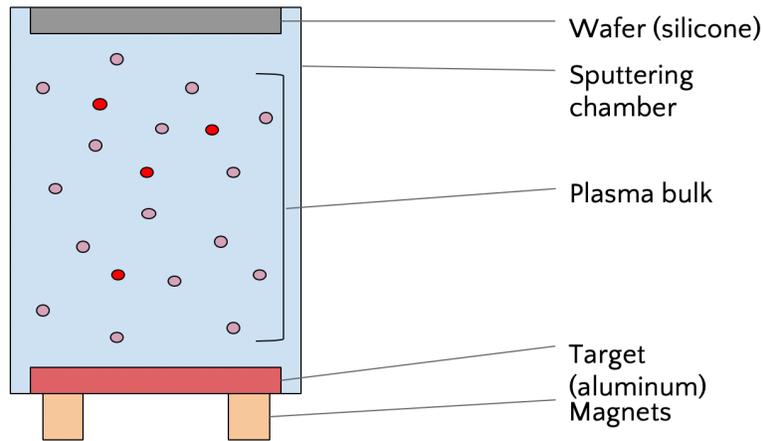


**Figure 22.** Diagram showing the entire target with magnets below it. The magnet with the south pole facing up wraps around the north pole facing magnet, creating inward-facing field lines.

### Simulate Ion Interaction

Characterizing electron movement close to the target during sputtering events was the first step in an overall optimization process. The next step in this process would be to simulate the ionization process, located in the plasma bulk. This project focused very heavily on the small volume immediately above the target, but that does not show the entire sputtering process. Much of the plasma involved in magnetron sputtering is in the plasma bulk, rather than close to the target or wafer. The plasma bulk occupies much of the area between space in the center of the tank, as seen in Figure 23. High energy

electrons within the plasma bulk ionize the argon gas in the chamber, resulting in argon ions. Some of the newly created argon ions then stay in the plasma bulk, while others travel to the surface of the target, moving similarly to the electrons studied in this process. Modeling the ionization process would help researchers learn more about how various electron energies impact the ionization process, as well as how best to increase the frequency of ionization.



**Figure 23.** Entire sputtering tank, including plasma bulk.

### **In-Lab Electron Monitoring**

To obtain results that accurately reflect sputtering processes at NTB, the University should install some degree of electron monitoring in their system. This electron monitoring could potentially be in the form of density readings at various locations within the tank, which would align with the further optimization of their system by increasing understanding of overall electron motion. Instituting electron monitoring within the NTB sputtering chambers would allow the University to benchmark any simulated data based on their systems, eventually allowing the University to optimize their sputtering systems specifically.

## References

- Adelmann, A. (2013, May 2). *Computational Particle Accelerator Physics Multidisciplinary Science for next generation Accelerators* [Lecture presentation]. Retrieved from <https://amas.web.psi.ch/people/aadelmann/talks/mit-talk-2013.pdf>
- AEM Deposition (2019). *Unused sputtering target* [Photograph]. Retrieved from <https://www.aemdeposition.com/sulfide-targets/tin-sulfide-sputtering-targets.html>
- Agostinelli, S. et al. (2003). *Geant4 - a simulation toolkit* [Program package and description]. Retrieved from <https://infoscience.epfl.ch/record/49909>
- Balagopal, P. (2017, March 18). *Why can't we see electrons with our naked eyes?* [Forum post]. Retrieved from <https://physics.stackexchange.com/questions/319650/why-cant-we-see-electrons-with-our-naked-eyes>
- Briesemeister, A. (2015, June 8). *Single Particle Motion* [Lecture presentation]. Retrieved from <https://suli.pppl.gov/2015/course/SingleParticleMotion.pdf>
- Böttcher, L. (2019, August 2). *Computational Statistical Physics* [Class notes]. Retrieved from [https://ethz.ch/content/dam/ethz/special-interest/phys/theoretical-physics/itp-dam/documents/compstatphys/CompStatPhysPartII\\_notes.pdf](https://ethz.ch/content/dam/ethz/special-interest/phys/theoretical-physics/itp-dam/documents/compstatphys/CompStatPhysPartII_notes.pdf)
- CERN (2019). *Subatomic particles* [Research overview]. Retrieved from <https://home.cern/science/physics/subatomic-particles>
- CERN (n.d.). *Detector Description: Basics* [Overview presentation]. Retrieved from [https://geant4.web.cern.ch/sites/geant4.web.cern.ch/files/geant4/collaboration/working\\_groups/geometry/training/D2-Basics.pdf](https://geant4.web.cern.ch/sites/geant4.web.cern.ch/files/geant4/collaboration/working_groups/geometry/training/D2-Basics.pdf)
- Chen, F. (2010). *Introduction to Plasma Physics and Controlled Fusion: Volume 1: Plasma Physics*. (2nd Edition). New York, New York: Plenum Press.
- Daudelin, A. J. (2014). *Fast Neutron Characterization*. Unpublished Major Qualifying Project. Worcester Polytechnic Institute.
- Dizikes, P. (2010, May 17). *Explained: Monte Carlo simulations* [Article]. Retrieved from <http://news.mit.edu/2010/exp-monte-carlo-0517>
- Fox, W. (2018, June 11). *Single Particle Motion* [Lecture presentation]. Retrieved from [https://suli.pppl.gov/2018/course/Fox\\_SULI\\_2018.pdf](https://suli.pppl.gov/2018/course/Fox_SULI_2018.pdf)

- flagstaffotos (2005). *Tesla Coil Arc* [Photograph]. Retrieved from [https://en.wikipedia.org/wiki/Tesla\\_coil#/media/File:Lightning\\_simulator\\_questacon02.jpg](https://en.wikipedia.org/wiki/Tesla_coil#/media/File:Lightning_simulator_questacon02.jpg)
- Gallagher, P. T. (n.d.) *Single-Particle Motion* [Lecture presentation]. Retrieved from [https://www.tcd.ie/Physics/people/Peter.Gallagher/lectures/PlasmaPhysics/Lecture3\\_single\\_particle.pdf](https://www.tcd.ie/Physics/people/Peter.Gallagher/lectures/PlasmaPhysics/Lecture3_single_particle.pdf)
- Generowicz, J. (2002, November 11). *Geant4 Visualisation* [Overview presentation]. Retrieved from <http://geant4.web.cern.ch/sites/geant4.web.cern.ch/files/geant4/collaboration/workshops/users2002/talks/lectures/FirstStepsVis.pdf>
- Green, J. (2018). *Used sputtering target* [Photograph]. Retrieved from <http://samsputtertargets.emyspot.com/blog/sputtering-target/how-to-replace-the-sputtering-target.html>
- Geant4 (n.d.) *Overview* [Software package overview]. Retrieved from <https://geant4.web.cern.ch>
- Gutsche, M. (2019, August 12). Personal communication.
- Gutsche, M. (2019, August 22). Personal communication.
- Gutsche, M. (2019, August 27). Personal communication.
- Gutsche, M. (2019, September 9). Personal communication.
- Gutsche, M. (2019, September 20). Personal communication.
- Harnett, E. (2012). *Electrical and Plasma Rocket Propulsion* [Lecture presentation]. Retrieved from [https://courses.ess.washington.edu/ess-102/FALL12/Lecture24\\_ElectricPropulsion\\_v2.pdf](https://courses.ess.washington.edu/ess-102/FALL12/Lecture24_ElectricPropulsion_v2.pdf)
- Helmenstein, A. M. (2019, January 25). *How Neon Lights Work (A Simple Explanation)* [Overview article]. Retrieved from <https://www.thoughtco.com/how-neon-lights-work-606167>
- Hughes, M. (2014, November 24). *What Is Sputtering? Magnetron Sputtering?* [Equipment description]. Retrieved from <http://www.semicore.com/what-is-sputtering>
- Int. J. Model. Simul. Sci. Comput. 1 (2010) 157–178. Retrieved from <https://www.worldscientific.com/doi/abs/10.1142/S1793962310000122>
- Kelley, M. C., Liley, B. S. (2019) *Applications of Plasma* [Encyclopedia entry]. Retrieved from <https://www.britannica.com/science/plasma-state-of-matter/Applications-of-plasmas>

- Kenton, W. (2019, June 10). *Monte Carlo Simulation Definition* [Article]. Retrieved from <https://www.investopedia.com/terms/m/montecarlosimulation.asp>
- Lawrence Berkley National Lab - Advanced Light Source Engineering Group (n.d.) *Plasma Physics: V. Wall Phenomena: Diffusion and Sheaths* [Derivation notes]. Retrieved from [http://www-eng.lbl.gov/~dleitner/USPAS\\_2016\\_Fundamental\\_Of\\_Ion\\_Sources/Reference%20Material/Plasma/Debye\\_length\\_sheath.pdf](http://www-eng.lbl.gov/~dleitner/USPAS_2016_Fundamental_Of_Ion_Sources/Reference%20Material/Plasma/Debye_length_sheath.pdf)
- MacPhail, T., Gumplinger, P. (n.d.). *Geant4 photon visualization* [Photograph]. Retrieved from <https://geant4.web.cern.ch/gallery>
- Mahdavi, M., Azadboni, F. K., Azadboni, R. K. (2011, 30 October). *Trapping of Free Electrons in Multipole System*. Retrieved from [https://www.researchgate.net/publication/257587946\\_Trapping\\_of\\_Free\\_Electrons\\_in\\_Multipole\\_System](https://www.researchgate.net/publication/257587946_Trapping_of_Free_Electrons_in_Multipole_System)
- NTB Interstaatliche Hochschule für Technik (n.d.). *PVD Sputtering Tool BAS 450* [Equipment description]. Retrieved from <https://www.ntb.ch/en/rd/institutes/mnt/infrastructure/clean-room-technologies/bas-450/>
- Pease, C. (2018, September 6). *An Overview of Monte Carlo Methods* [Overview article]. Retrieved from <https://towardsdatascience.com/an-overview-of-monte-carlo-methods-675384eb1694>
- Princeton Plasma Physics Laboratory (2019). *Basic Plasma Science* [Experiment descriptions]. Retrieved from <https://www.pppl.gov/research/basic-plasma-science>
- Sarkar, J. (2014). *Sputtering Materials for VLSI and Thin Film Devices*. Retrieved from <https://www.sciencedirect.com/topics/engineering/plasma-density>
- Schuengel, E. (2019, August 30). Personal communication.
- Science Learning Hub (2014, April 29). *Lightning explained* [Overview article]. Retrieved from <https://www.sciencelearn.org.nz/resources/239-lightning-explained>
- Thin Film Consulting (2015). *Rectangular sputtering target* [Photograph]. Retrieved from <http://thfc.de/rectangular-highvacuum-magnetrons/magnet-array-options>
- Thiry, D., Cornil, K., Konstantinidis, S., Syders, R. (2016, May). *Plasma diagnostics for the low-pressure plasma polymerization process: A critical review*. Retrieved from [https://www.researchgate.net/publication/299593562\\_Plasma\\_diagnostics\\_for\\_the\\_low-](https://www.researchgate.net/publication/299593562_Plasma_diagnostics_for_the_low-)

[pressure\\_plasma\\_polymerization\\_process\\_A\\_critical\\_review/link/5b2a1eb04585150c634000ae/download](https://doi.org/10.1039/C8PY00000A)

West, A. (2018). *Lightning strike in Florida* [Photograph]. Retrieved from <https://eu.naplesnews.com/story/news/environment/2019/07/29/florida-summers-bring-increased-lightning-dangers/1798261001/>

Yellow Octopus (2019). *Cactus Neon Lights* [Photograph]. Retrieved from <https://www.yellowoctopus.com.au/products/neon-cactus-light-sunnylife>

## Appendix A: Simulation Data

Below the raw data for simulation runs can be found. The yellow rows are calculated averages, as used in [Results: Electron Density Across the Target Material](#).

Electron Energy (KeV)	Total Points	North Points	Center Points	South Points	% North	% Center	% South
100	1218	219	882	117	17.98	72.41	9.61
100	179	6	167	6	3.35	93.30	3.35
100	440	72	292	76	16.36	66.36	17.27
100	407	41	321	45	10.07	78.87	11.06
100	176	40	84	52	22.73	47.73	29.55
100	2947	307	2331	309	10.42	79.10	10.49
100	895	114	680	101	13.49	72.96	13.55
200	2141	222	1700	219	10.37	79.40	10.23
200	1685	241	1192	252	14.30	70.74	14.96
200	2028	157	1708	163	7.74	84.22	8.04
200	2764	270	2248	246	9.77	81.33	8.90
200	1408	139	1143	126	9.87	81.18	8.95
200	961	146	664	151	15.19	69.09	15.71
200	1831	196	1443	193	11.21	77.66	11.13
250	364	46	282	36	12.64	77.47	9.89
250	2069	145	1798	126	7.01	86.90	6.09
250	2537	147	2241	149	5.79	88.33	5.87
250	3831	602	2683	546	15.71	70.03	14.25
250	1016	98	823	95	9.65	81.00	9.35
250	2542	287	1925	330	11.29	75.73	12.98
250	2060	221	1625	214	10.35	79.91	9.74
300	2015	265	1490	260	13.15	73.95	12.90
300	2228	319	1520	389	14.32	68.22	17.46
300	1515	202	1151	162	13.33	75.97	10.69
300	875	161	519	195	18.40	59.31	22.29
300	151	25	99	27	16.56	65.56	17.88

300	1458	109	1243	106	7.48	85.25	7.27
300	1374	180	1004	190	13.87	71.38	14.52
400	3878	421	3046	411	10.86	78.55	10.60
400	1525	205	1129	191	13.44	74.03	12.52
400	2214	165	1837	212	7.45	82.97	9.58
400	539	54	434	51	10.02	80.52	9.46
400	1720	288	1129	303	16.74	65.64	17.62
400	789	63	656	70	7.98	83.14	8.87
400	1778	199	1372	206	11.08	77.48	11.44
500	2941	432	2095	414	14.69	71.23	14.08
500	2346	277	1814	255	11.81	77.32	10.87
500	1270	132	985	153	10.39	77.56	12.05
500	2172	249	1696	227	11.46	78.08	10.45
500	2841	431	1949	461	15.17	68.60	16.23
500	3430	320	2791	319	9.33	81.37	9.30
500	2500	307	1888	305	12.14	75.70	12.16
550	1838	177	1487	174	9.63	80.90	9.47
550	1676	197	1257	222	11.75	75.00	13.25
550	1913	130	1652	131	6.80	86.36	6.85
550	1611	212	1216	183	13.16	75.48	11.36
550	3808	408	3000	400	10.71	78.78	10.50
550	1876	213	1442	221	11.35	76.87	11.78
550	2120	223	1676	222	10.57	78.90	10.53

## Appendix B: Simulation Code Excerpts

### *E Helper Function*

```
12  std::pair<G4double, G4double> E(const double charge, const int il[2], const G4double x, const G4double y)
13  {
14      G4double underPart = pow(pow((x-il[0]),2) + pow((y-il[1]),2), 1.5);
15      G4double retX = charge * (x-il[0]) / underPart;
16      G4double retY = charge * (y-il[1]) / underPart;
17
18      return std::make_pair(retX, retY);
19  }
```

### *Magnetic Field Calculation*

```
std::pair<G4double, G4double> p1 = E(force, initLoc1, coordX, coordY);
std::pair<G4double, G4double> p2 = E(-force, initLoc2, coordX, coordY);
G4double eX1 = p1.first;
G4double eY1 = p1.second;

G4double eX2 = p2.first;
G4double eY2 = p2.second;

rawFieldX = eX1 + eX2;
rawFieldY = eY1 + eY2;

////////////////////////////////////

// y-coordinate-based field strength modifier
double strengthModY = (coordY != 0) ? (strengthModY = 1 + 1/coordY) : (1);
// x-coordinate-based field strength modifier
double strengthModX = sqrt(abs(coordX));

//end modifier to field strength
double fieldStrengthModifier = sqrt(pow(strengthModX,2)+pow(strengthModY,2));

//calculate the field value as an x-vector and y-vector
G4double xFieldValue = rawFieldX/sqrt(pow(rawFieldX,2)+pow(rawFieldY,2)) * fieldStrengthModifier * tesla;
G4double yFieldValue = rawFieldY/sqrt(pow(rawFieldX,2)+pow(rawFieldY,2)) * fieldStrengthModifier * tesla;

//add field strength vectors for all vald points
if (point[3] == 0) {
    field[0] = xFieldValue;
    field[1] = yFieldValue;
    field[2] = 0 * tesla;
}
```

### **Geometry Definition**

```
G4double magnetSpacing = 85.0;
G4double magnetX = 10.0;
G4double magnetY = 20.0;
G4double magnetZ = 200.0;
G4double targetX = magnetSpacing + magnetX;
G4double targetY = 7.0;
G4double targetZ = magnetZ;

//relevant spacing between magnets and targets
G4double differenceY = 30.0*mm;
G4double magDistance = 85.0*mm;

//all object physical coordinates (centers)
G4ThreeVector mpos1 = G4ThreeVector( magDistance/2, 0, 0);
G4ThreeVector mpos2 = G4ThreeVector(-magDistance/2, 0, 0);
G4ThreeVector tpos = G4ThreeVector(0, differenceY/2, 0);
G4ThreeVector cpos = G4ThreeVector(0, differenceY+30, 0);
G4ThreeVector gpos = G4ThreeVector(0, differenceY+20, -70);
G4ThreeVector dpos = G4ThreeVector(0, differenceY/2, 0);
G4ThreeVector ipos1 = G4ThreeVector(magDistance/2.7, differenceY+1, 0);
G4ThreeVector ipos2 = G4ThreeVector(0, differenceY+1, 0);
G4ThreeVector ipos3 = G4ThreeVector(-magDistance/2.7, differenceY+1, 0);
```

## Detector Definition

```
// get volume of the current step
G4LogicalVolume* volume
  = step->GetPreStepPoint()->GetTouchableHandle()
    ->GetVolume()->GetLogicalVolume();

// check if we are in the volumes worth tracking
bool inChecker = false;
if ( (volume->GetName()) == "logicShapeTrack1"
    || (volume->GetName()) == "logicShapeTrack2"
    || (volume->GetName()) == "logicShapeTrack3"
){
  inChecker = true;
}

if (!inChecker) return;

auto Pos = step->GetPostStepPoint()->GetPosition();
dipoleField* dataField = new dipoleField();
double coords[4] = {Pos[0],Pos[1],Pos[2],-1.0};
double *back;
auto thing = step->GetTrack()->GetVertexMomentumDirection();
dataField->GetFieldValue(coords, back);
int volNum = (volume->GetName()).back()-'0';

fEventAction->addHit(volNum);
```

```
void sputteringEventAction::addHit (int volume) {  
    //add electron hit for north/south/center scoring volume  
    switch (volume){  
        case 1:  
            sv1++;  
            break;  
        case 2:  
            sv2++;  
            break;  
        case 3:  
            sv3++;  
            break;  
    }  
}
```

```

int vol1 = sv1.GetValue();
int vol2 = sv2.GetValue();
int vol3 = sv3.GetValue();

// Print
//
if (IsMaster()) {
    G4cout
    << G4endl
    << "-----End of Global Run-----";
}
else {
    G4cout
    << G4endl
    << "-----End of Local Run-----";
}

G4cout
<< G4endl
<< " The run consists of " << nofEvents << " " << runCondition
<< G4endl
<< " Hits within volume : "
<< G4endl
<< "\tNorth: " << vol1
<< G4endl
<< "\tCenter: " << vol2
<< G4endl
<< "\tSouth: " << vol3
<< G4endl
<< "\tTotal: " << (vol1 + vol2 + vol3)
<< G4endl
<< "-----"
<< G4endl
<< G4endl;
}

```