Spring 5-2022

Ion Production and Mitigation in DC High-Voltage Photo-Guns

Joshua T. Yoskowitz
Old Dominion University, yojo333@yahoo.com

ION PRODUCTION AND MITIGATION IN DC HIGH-VOLTAGE PHOTO-GUNS

by

J. T. Yoskowitz
B.S. June 2016, Union College
M.S. May 2018, Old Dominion University

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

PHYSICS

OLD DOMINION UNIVERSITY
May 2022

Approved by:

Geoffrey Krafft (Director)

Joseph Grames (Member)

Moskov Amaryan (Member)

Helmut Baumgart (Member)

Jean Delayen (Member)

ABSTRACT

ION PRODUCTION AND MITIGATION IN DC HIGH-VOLTAGE PHOTO-GUNS

J. T. Yoskowitz
Old Dominion University, 2022
Director: Dr. Geoffrey Krafft

One of the biggest obstacles to operating a GaAs polarized electron source with a long charge lifetime is the mitigation of ion back-bombardment. Several techniques exist to either clear ions from the accelerator or to mitigate ion damage of the photocathode. Predicting the effectiveness of these techniques requires sophisticated simulation models of electron impact ionization within the photo-gun. In this work, the effectiveness of applying a positive anode bias voltage to mitigate ion damage and increase the charge lifetime of the GaAs photocathode was studied over three run periods at the Continuous Electron Beam Accelerator Facility (CEBAF) at Jefferson Lab (JLab). The charge lifetime with the biased anode configuration was 1.80 ± 0.51 times longer than the lifetime of the usual grounded anode configuration. Simulations of ionization within the CEBAF photo-gun and adjacent beamline were made using General Particle Tracer (GPT) and a new C++ custom element to predict and explain the substantial improvement in charge lifetime. The experimental results and the development of the ionization custom element, with its use in simulations of ion back-bombardment with the biased anode, are described in detail.

"Imagination is more important than knowledge. For knowledge is limited, whereas imagination embraces the entire world, stimulating progress, giving birth to evolution."
– Albert Einstein

ACKNOWLEDGMENTS

First and foremost, I would like to thank my parents for their endless love and support throughout my grad school career and throughout my life in general. My mom has been a role model for me and has encouraged me to keep working towards my goals, no matter how challenging they are. My dad has helped me throughout my career as a physicist, especially in computer skills. In particular, he helped me write and debug my GPT ionization custom element. He also wrote a VBA macro for PowerPoint that has helped me to organize literally thousands of plots and tables from the GPT systematic studies into neat plot grids and tables and has saved me countless hours (or perhaps *weeks*) of tedious work.

I would like to thank my advisors Dr. Joseph Grames and Dr. Geoffrey Krafft for their guidance and support throughout my thesis research. They have helped ensure the success of my research by providing opportunities to collaborate with other scientists and present my work at conferences, and helping me whenever I ran into problems.

I have had the incredible opportunity to collaborate with many scientists at JLab and at other laboratories around the world and they have each made an impact on my thesis research. First, I would like to thank the members of the Center for Injector Studies (CIS) group. They have all helped me with my experiments. In particular, I'd like to thank Dr. Marcy Stutzman for showing me the vacuum systems at CEBAF and providing me with RGA scans of the CEBAF photo-gun and beamline, and Dr. Matt Poelker, Dr. Carlos Hernandez-Garcia, Dr. Riad Suleiman, Dr. Shukui Zhang, Dr. Md. Abdullah Mamun, Dr. Yan Wang, Bubba Bullard, and Phil Adderley who have each contributed to my understanding of ionization and emittance and their importance in electron accelerators. I would also like to thank Dr. Cristhian Alfonso Valerio Lizarraga and Gaspar Ricardo Montoya Soto, whom I have collaborated with to understand the effect of the laser spot size on the charge lifetime of the photocathode. They also helped me benchmark my ionization custom element using the simulation code IBSimu. A special thanks goes to Dr. S. B. van der Geer, a co-developer of General Particle Tracer (GPT), and Dr. Alicia Hofler, who have helped me numerous times with my GPT simulations and have helped me develop the ionization custom element. Lastly, I'd like to thank Dr. Gabriel Palacios-Serrano for creating the CST models of the CEBAF photo-gun used in my GPT simulations of the biased anode experiments.

I would like to thank my thesis committee members Dr. Moskov Amaryan, Dr. Helmut Baumgart, and Dr. Jean Delayen, who have all helped guide my thesis towards conclusion and ensure its success.

In the physics departments at Old Dominion University and Union College, I would like to thank my physics professors who have contributed to my knowledge of physics and have given me the tools to face any challenge and overcome any obstacle in my physics research. Finally, a special thanks to all of my fellow graduate students, including Lindsay Thornton, Tyler Viducic, Dr. Sajini Wijethunga, and Dr. Mark Stefani, and to all of my friends for their help and encouragement throughout my graduate school career.

This work was supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics under contract DE-AC05-06OR23177.

TABLE OF CONTENTS

Page

LIST OF TABLES ix

LIST OF FIGURES xi

Chapter

1. INTRODUCTION 1

2. IONIZATION THEORY 8
   - 2.1 OVERVIEW 8
   - 2.2 ELECTRON IMPACT IONIZATION 8
   - 2.3 ION PRODUCTION RATE 8
   - 2.4 SECONDARY ELECTRON ENERGY DISTRIBUTION 13
   - 2.5 ION ENERGY DISTRIBUTION 15

3. EXPERIMENTAL SETUP 17
   - 3.1 CEBAF POLARIZED ELECTRON SOURCE 18
   - 3.2 LASER SETUP AND PROFILE MEASUREMENTS 27
   - 3.3 CEBAF INJECTOR BEAMLINE 31
   - 3.4 QE MEASUREMENT AND QE SCAN 37

4. BIASED ANODE EXPERIMENTS 40
   - 4.1 BIASED ANODE TESTS 40
   - 4.2 EXPERIMENTAL PARAMETERS 45
   - 4.3 BEAM CURRENT, VACUUM, ION PRODUCTION RATE 45
   - 4.4 QE MEASUREMENT AND CHARGE LIFETIME RESULTS 47
   - 4.5 QE SCAN RESULTS 51

5. GPT SIMULATIONS 55
   - 5.1 OVERVIEW 55
   - 5.2 SIMULATION SETUP 56
   - 5.3 GPT ALGORITHM 58
   - 5.4 IONIZATION CUSTOM ELEMENT ALGORITHM 61
   - 5.5 SIMULATION RESULTS 72
   - 5.6 SYSTEMATIC STUDIES 77

6. DISCUSSION 83

7. CONCLUSIONS 88

REFERENCES 90

Page

APPENDIX 99

VITA 132

LIST OF TABLES

Table Page

1. Calculated Values of the Ionization Cross Section and Ion Production Rate per Unit Length for H_2 Gas 11

2. Table of the Most Probable Ion Speed and the Mean Ion Speed for Various Ion Species 16

3. Partial Pressures and Densities of Residual Gases in the CEBAF Photo-gun, as Measured in the RGA Spectrum 27

4. x- and y-Diameters for Each Laser Using the D4-Sigma Method 30

5. Peak Current Phase for Each Anode Voltage 43

6. Experimental Parameters for Each Run Period 45

7. Estimates of the Average Partial Pressure and Ion Production Rate per Unit Length of H_2, CO, CO_2 and CH_4 in the Photo-gun Vacuum Chamber 47

8. Charge Lifetime Values for Each Laser Used in Each Run Period 51

9. Improvement Ratios Between Charge Lifetimes τ_i for Biased Anode Regions and Grounded Anode Regions in Run Periods 1 and 2 51

10. Table of Relevant Parameters for All QE Scans 54

11. Global Parameters for All Simulations 56

12. Simulation-Specific Parameters for Each Run Period 56

13. Cash-Karp Constants for a_i, b_{ij}, c_i, and c_i^* 60

14. $f(x)$, $F(E)$ and $g(E)$, for Linear, Quadratic, and Cubic Distributions Assuming $E_{max} = 1000$ 65

15. Table of Kinematic Parameters for Each Particle Involved in Ionization 67

16. GPT Benchmarking Simulation Results 69

17. Comparison of GPT and IBSimu Simulation Results 72

18. Ion Ratios Between Grounded and Biased Anode Simulations for Run Periods 1 and 2 76

Table Page

19. Photocathode Ion Ratio Table for H_2 Gas in the Laser Spot Position Systematic Study 79

20. List of Ionization Parameters to Be Written to IPR Info File 113

21. List of Supported Gas and Orbital Names With Their Respective Ionization Energies 117

LIST OF FIGURES

Figure Page

1. Diagram of Ion Back-Bombardment Within the CEBAF Photo-gun 3

2. Diagram of the Ionization Process 9

3. Log-Log Plot of the Ionization Cross Section of Several Target Gas Particles Using Eq. 2 10

4. Log-Log Plot of the Ion Production Rate per Unit Length of Several Target Gas Particles Using Eq. 4 12

5. Log Plots of the Differential Cross Section for Secondary Electron Energies of Several Target Gas Particles Assuming a Primary Electron Energy of 130 keV 14

6. Maxwellian Distribution for Various Ion Species With the Most Probable Speed Corresponding to an Energy of 4 eV 16

7. Diagram of the CEBAF Accelerator 17

8. Layout of CEBAF Injector Extending from the Photo-gun to the Dipole Magnet 19

9. Layout of CEBAF Injector Extending from the Dipole Magnet to the Faraday Cup ... 19

10. Diagram of the CEBAF Interleaved Electron Beam Using Four 248.5 MHz Lasers 20

11. Schematic (Left) and Photo (Right) of the CEBAF Photo-gun 21

12. Plots of the Electric Potential Along the Central Axis of the Beamline for 0 V and 1000 V Anode Configurations 22

13. Diagram of the Structure and Polarization of a Strained-Superlattice Phototocathode 24

14. RGA Scan of the CEBAF Photo-gun Vacuum With the Electron Beam Off 26

15. Schematic of the Physical Layout of the Four Fiber Lasers 28

16. Diagram of the Setup for the Laser Profile Measurements 29

17. Relative Intensity Profiles for Lasers A, B, C, and D 30

18. Photo of a Haimson Steering Coil Pair on the Beamline 31

Figure Page

19. Photo of a Solenoid on the Beamline 32

20. Photo of a BPM at the Location of a Steering Coil Pair 33

21. Photo of a Viewer on the Beamline 34

22. Photo of the Dipole Magnet on the Beamline 35

23. Diagram of the Chopper System Containing Two RF Chopper Cavities, Two Solenoid Lenses, and the Master Slit 36

24. Photo of the RF Chopper System and Faraday Cup 37

25. Example Raw QE Scan (Top), the Interpolated QE Scan (Center), and the Interpolated QE Scan After Calibration (Bottom) 39

26. EPICS Screenshot of a Beam Size Measurement Using Viewer #1 41

27. Transverse Beam Size Measurements for Each Viewer as a Function of Anode Voltage 42

28. Chopper Scans Across Slit A for Each Anode Voltage 43

29. Beam Position vs. Anode Voltage as Measured by Each BPM Pair 44

30. Plots of the Beam Current, Cumulative Charge Extracted From the Photocathode, and Photo-gun Vacuum For Run Periods 1, 2, and 3 46

31. Plot of the QE Measurements for Run Period 1 as a Function of the Cumulative Charge Extracted 48

32. Plots of the QE Measurements for Run Period 2 as a Function of the Cumulative Charge Extracted for Laser Spot #1 (Top) and Laser Spot #2 (Bottom) 49

33. Plot of the QE Measurements for Run Period 3 as a Function of the Cumulative Charge Extracted 50

34. QE Scans Taken Before (Left) and After (Center) Run Period 1 and the Corresponding QE Difference Scan (Right) 52

35. QE Scans (Left) and QE Difference Scans Between Successive QE Scans (Right) for Run Period 2 53

36. QE Scans Taken Before (Left) and After (Center) Run Period 3 and the Corresponding QE Difference Scan (Right) 54

Figure Page

37. Cross Section of the Initial CEBAF Injector Beamline 57

38. Plots of the Transverse Distribution (Left) and Temporal Distribution (Right) of the Electron Bunch 58

39. Monte Carlo Algorithm for Ion Production 63

40. Histograms of Linear, Quadratic, and Cubic Distributions for $N = 10^1, 10^2, 10^3$ and 10^4 66

41. Snapshot of a GPT Simulation After the Primary Electron Bunch Travels 400 mm ... 68

42. Normalized Histogram of Secondary Electron Energies From All 50 CH_4 Simulations With the Normalized Differential Cross Section Curve Overlaid 70

43. Normalized Histogram of Ion Energies From All 50 CH_4 Simulations With the Maxwellian Distribution Overlaid 71

44. Plots of the Ion Distribution at the Photocathode for Each Simulation for Run Period 1 74

45. Plots of the Ion Distribution at the Photocathode for Each Simulation for Run Period 2 75

46. Plots of the Ion Distribution at the Photocathode for Each Simulation for Run Period 3 76

47. Plot of Number of Back-Bombarding Ions on the Photocathode vs Laser Spot Position for a 0 V Anode 79

48. Plot of Number of Back-Bombarding Ions on the Photocathode vs Laser Spot Position for a 961 V Anode 80

49. Plot of Number of Back-Bombarding Ions on the Active Area vs Laser Spot Position for a 0 V Anode 80

50. Plot of Number of Back-Bombarding Ions on the Active Area vs Laser Spot Position for a 961 V Anode 81

51. Plot of Number of Back-Bombarding Ions on the Laser Spot vs Laser Spot Position for a 0 V Anode 81

52. Plot of Number of Back-Bombarding Ions on the Laser Spot vs Laser Spot Position for a 961 V Anode 82

CHAPTER 1

INTRODUCTION

The ability of electron accelerators to efficiently produce polarized electron beams has facilitated experiments with extensive applications in nuclear and high energy physics. Among these applications are the spin dependence of certain fundamental interactions, the search for physics beyond the standard model, the measurement of nucleon form factors, and the measurement of the neutron skin thickness of lead [1, 2, 3, 4]. These experiments require polarized electron beams with high average current, high spin polarization, and long operating lifetimes. The design of a photo-gun that meets these requirements has been the goal of several decades of research.

The earliest polarized electron sources that use photoemission from GaAs photocathodes were proposed in 1974 at both the Stanford Linear Accelerator Center (SLAC) [5], and at the Laboratory of Condensed Matter Physics and Nanostructures (LPMCN) [6]. The proposed GaAs photocathodes were to be coated with Cs and O_2 to create a negative electron affinity (NEA) condition. The emission of electrons from the photocathode can be described as a three-step process [7]:

1. Photons incident on the photocathode are absorbed and excite electrons below the surface into the conduction band.

2. These electrons diffuse through the photocathode, some of which reach the surface.

3. Electrons with sufficient energy overcome the electron affinity and escape into vacuum.

Because of the NEA condition, incident photons with energies near the band-gap energy allow electrons in the conduction band to escape directly into the vacuum [8]. If the incident light is circularly polarized, these photo-emitted electrons are preferentially polarized in the opposite direction of the photon polarization, a process known as the Fano effect [9, 10]. Although 50% is the theoretical maximum polarization for bulk GaAs photocathodes due to valance-band degeneracy, Pierce *et al.* achieved electron polarization around 60% due to a slightly positive electron affinity [11, 12].

In 1988, it was first suggested that depolarization effects in the diffusion of electrons to the photocathode surface could be mitigated using a "blocking layer" of GaAlAs on top of

the GaAs substrate [13]. Later, in 1991, Maruyama *et al.* showed that the polarization can be increased over 70% by layering GaAsP, GaAlAs, or InGaAs upon a GaAs substrate in a process known as straining [14]. Straining breaks the valance-band degeneracy, allowing the preferential excitation of a single transition to the conduction band, thus increasing polarization. Following successful experiments using a photocathode composed of InGaAs or GaAsP grown on a GaAs substrate at SLAC [15, 16], other experiments using strained GaAs photocathodes soon followed at accelerators including Mainz-Microtron (MAMI) [17], NIKHEF [18], MIT-Bates [19], ELSA [20], and CEBAF [21].

While the measured polarization at each of these accelerators was between 75-88%, well above the maximum for bulk GaAs, long operating lifetimes, defined as the amount of time an accelerator can maintain a given beam current, could only be achieved at the expense of high current. The reason for short operating lifetimes is due to the sharp decay of photocathode quantum efficiency (QE), which is defined as the ratio between the number of incident photons to the number of photo-emitted electrons. The operating lifetimes were on the order of tens to hundreds of hours at best when running at high current. Since the operating lifetime can vary between accelerators with different beam currents, the charge lifetime, defined as the amount of charge extracted from the photocathode before the QE drops to 1/e of its original value, is more useful for comparison between accelerators and will be the metric used in this thesis herein.

It was first suggested in 1993 during the initial polarized electron experiments at the MAMI source that the rapid QE decay might be due to the degradation of the activation material in the vicinity of the laser spot by a "local effect" [22]. Experiments were performed to study this effect with three note-worthy observations:

1. With the laser incident on a grounded photocathode (i.e., with no photocurrent), there was no observable QE decay.

2. With the same laser incident on the photocathode with a cathode voltage of $-100\,\mathrm{kV}$ to produce photocurrent, a 20-30% reduction in QE across the entire active area was observed.

3. Longer charge lifetimes were observed with the laser spot near the edge of the active area, as opposed to the center.

The results of these experiments led to the hypothesis that QE decay might be due to a process known as ion back-bombardment [23]. Figure 1 shows the ion back-bombardment

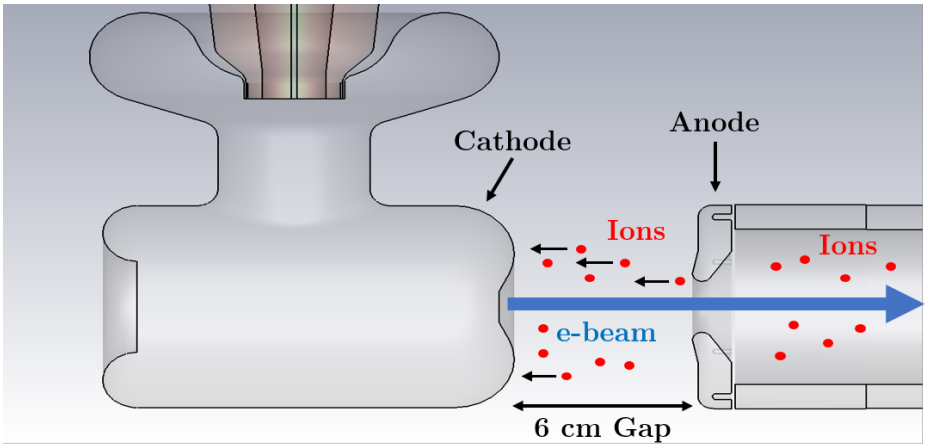

FIG. 1: Diagram of ion back-bombardment within the CEBAF photo-gun.

process within the CEBAF photo-gun. While a photo-gun vacuum is typically kept as low as possible during beam operation, no photo-gun can achieve perfect vacuum. Thus, residual gas will always be present within the photo-gun. While the electron beam is on, residual gas molecules can come into contact with the beam and become ionized. Where the ion travels depends on its origin:

1. Ions created within the cathode-anode gap will accelerate towards and strike the photocathode. The energy of the ions at impact depends on their origin.

2. Ions created downstream of the anode can still experience the electric field from the cathode. In this case, there are several possibilities:

   (a) Ions created close to the anode can pass through the anode aperture and strike the photocathode. The energy of these ions at impact is equal to the full cathode-anode voltage (i.e., if the cathode voltage were $-130\,\mathrm{kV}$, the ions would have $130\,\mathrm{keV}$ at impact).

   (b) Ions can accelerate towards the photocathode, but strike the anode electrode instead.

   (c) Ions created far from the anode can drift towards the anode, pass through its aperture, and strike the photocathode.

Several experiments have shown a correlation between the laser position on the active area and a reduction in QE from that location over the course of a run period (see, e.g., Fig. 6 in [24] and Fig. 7 in [25]). However, the specific processes by which ions reduce QE are still unknown and are the subject of speculation. Several mechanisms have been suggested, which depend on the properties of the atoms in the material and the energy of the ion. Here are several possibilities:

1. The ion can interact with an atom in the material. Depending on the energy of the ion, there are several possible outcomes:

   (a) The ion may back-scatter off the atom without damaging the lattice structure of the material. This would occur especially for ions with low incident energies (of a few eV or less) and ions with much lower mass than the atom. The atom recoils and then returns to its original position with its kinetic energy converted to phonons.

   (b) The ion may back-scatter from the atom with sufficient energy to displace the atom a distance greater than the spacing between atoms in the material. The displaced atom can displace further atoms in a cascade. If the cascade reaches the surface of the material, a surface atom can be ejected from the material. This process is known as ion sputtering [26].

   (c) The ion may penetrate the material. Every atom in the material that the ion interacts with slows the ion. The lost energy is converted into phonons. If the ion loses sufficient kinetic energy before leaving the material, it remains implanted in the material. This process is known as ion implantation [27].

2. The ion can interact with an electron in an atom. Depending on the energy of the ion, it can excite the electron into a higher energy state. The electron returns to its original energy state and releases energy in the form of a photon or another electron (called an Auger electron). On the other hand, if the ion has a kinetic energy greater than the ionization energy, it can eject it from the atom. If the electron is ejected from one of the inner shells, an outer shell electron takes its place and loses energy in the form of an X-ray, in a process known as particle-induced X-ray emission (PIXE) [28]. To reach the inner shell electron of most atoms, the ion must have kinetic energies on the order of a few MeV.

3. The ion can ionize several different atoms as it penetrates the surface of the material. The ejected secondary electrons can then travel through the material and either ionize more atoms, recombine with ions and emit photons, or reach the surface of the material and escape. Each secondary electron has the potential to escape the material provided it can overcome the surface binding energy by the time it reaches the surface. The total number of escaped electrons emitted in this fashion is known as secondary electron yield (SEY). [29, 30].

In addition to ion back-bombardment, ions can also cause several other deleterious effects on the electron beam such as fast ion instability [31, 32], charge neutralization [33], and beam halo [34]. Thus, ion production is generally undesirable in electron accelerators.

Ion mitigation techniques typically seek to either reduce the ion production rate, clear ions from the accelerator, or to reduce their negative effects. One strategy to directly reduce the ion production rate is to improve the vacuum levels: the lower the vacuum level, the lower the residual gas density and hence, lower ion production rate. Several methods have been studied to improve the vacuum within the CEBAF photo-gun, including improving the vacuum pump speed [35] and using heat treatments to reduce the outgassing rate [36]. Substantial improvement in charge lifetime was seen at the CEBAF photo-gun after lowering the photo-gun vacuum [37].

Alongside improving vacuum, another strategy to mitigate the effects of ions is to improve the resilience of the photocathode by choosing different activation materials. While GaAs photocathodes are typically activated with Cs, O, and/or NF_3 [38], the use of elements such as Li [39], Sb [40, 41], or Te [42], have been shown to increase the robustness of the active layer against ion damage. Also, using photocathode substrates such as molybdenum instead of GaAs have been shown to have a near-infinite charge lifetime [43]. However, molybdenum photocathodes do not produce polarized electron beams.

Ion back-bombardment can also be mitigated by directly influencing the trajectories of the ions themselves, either by clearing the ions from the accelerator or by reducing the damaging effects when they reach the photocathode. Several techniques currently exist to clear ions from the accelerator. However, any clearing technique is complicated by the fact that ions tend to be created within the electron beam and thus remain trapped by its potential [44]. One clearing technique is to introduce a clearing gap [45], which allows trapped ions to drift into the gap and exit the beam. The gap must be frequent enough to ensure that the rate of ion clearing is greater than or equal to the ion production rate. Another clearing technique is to use a clearing electrode [46]. The bias of the electrode is such that

trapped ions overcome the beam potential and exit the beam at a rate that is greater than the charge neutralization time. The minimum necessary bias voltage to meet this condition is typically on the order of 10 volts [45, 46, 47]. Lastly, one other clearing technique is beam shaking. Beam shaking takes advantage of the transverse oscillation of trapped ions. The oscillation frequency of the ions depends on the mass of the ion and the current density of the electron beam [48]. By deliberately introducing an RF electric field near the resonant ion frequency, its oscillation amplitude can be driven to the point where the ion is removed from the beam [49, 50].

While each of these three clearing techniques may be effective at clearing ions from the electron beam throughout the accelerator, they would not be effective at clearing ions produced within the photo-gun because the motion of these ions is dominated by the photo-gun potential, which is typically 100-300 kilovolts. Thus, mitigation strategies for ion back-bombardment typically seek to reduce the damaging effects of bombarding ions. One such strategy is to increase the laser size to cover a wider region of the active area [51]. Increasing the laser size increases the transverse beam size, which in turn increases the area over which back-bombarding ions hit the active area. If the beam current is kept the same, then increasing the laser size would reduce the ion damage per unit area and thus reduce the rate of QE decay and increase the charge lifetime. This method was previously tested at the CEBAF injector at JLab and was shown to increase the lifetime of a GaAs photocathode up to a factor of four [52].

In this thesis, the mitigation strategy studied was to apply a positive bias voltage to the anode of the CEBAF photo-gun. The positive bias repels ions originating downstream of the anode potential maximum, which might otherwise reach the photocathode with the highest kinetic energy and thus could be the most damaging. The bias should not affect any ions created within the cathode-anode gap, though these ions would have less kinetic energy at the photocathode and would be less damaging.

While the biased anode technique has previously been studied at JLab [53] and Brookhaven National Lab (BNL) [54], detailed simulation models of ion back-bombardment have not yet been made prior to this thesis to predict and quantify the benefit of biasing the anode to increase the charge lifetime. A sophisticated simulation model was developed using a new custom element designed within the framework of the simulation package General Particle Tracer (GPT) [55] to simulate electron impact ionization. Simulations were made using an electrostatic model of the CEBAF photo-gun created using CST Studio Suite software [56]. By applying the simulation model to the results of experiments performed at

CEBAF, the improvement in charge lifetime using a biased anode configuration over the usual grounded anode configuration was explored.

This thesis is organized as follows: the theory of ionization is discussed in Chapter 2 and the biased anode experiment and results are described in Chapters 3 and 4. The GPT simulations of the biased anode experiments are described in Chapter 5 as well as a detailed description of the ionization custom element. The experimental and simulation results are discussed in Chapter 6 and the conclusions and future work are discussed in Chapter 7.

CHAPTER 2

IONIZATION THEORY

2.1 OVERVIEW

To understand how and where ions originate in the CEBAF injector, the theory of electron impact ionization is described in this chapter. The ionization probabilities and the energy distributions of each particle following ionization are discussed. The individual calculations discussed are used directly in the GPT ionization custom element described in Chapter 5 to create realistic simulations for the biased anode experiments described in Chapter 4.

2.2 ELECTRON IMPACT IONIZATION

Electron impact ionization occurs when an electron interacts with an atom or molecule resulting in the ejection of an electron. Figure 2 shows a diagram of the electron impact ionization process. The incident electron is called the primary electron and the atom or molecule is called the target gas particle. After ionization, the target gas particle becomes a positive ion and the incident electron scatters away and is called a scattered electron. The ejected electron is called the secondary electron. Whichever electron has *lesser* energy is called the secondary electron and the other is the scattered electron. Ionization can occur provided the primary electron overcomes the minimum energy, the ionization energy, required to remove an electron from the target gas particle. Both the secondary and scattered electrons may continue to ionize provided they have sufficient kinetic energy to do so.

2.3 ION PRODUCTION RATE

The number of ions produced, N_{ion}, by a number of electrons, N_e, in a given timestep is calculated using the formula:

$$N_{ion} = \rho \sigma d N_e \tag{1}$$

where ρ is the gas density in ions/m^3, σ is the ionization cross section in m^2, and d is the distance travelled in meters by the electron in the timestep. The ionization cross section may be calculated using Reiser's formula [57], originally derived by Bethe [58]:

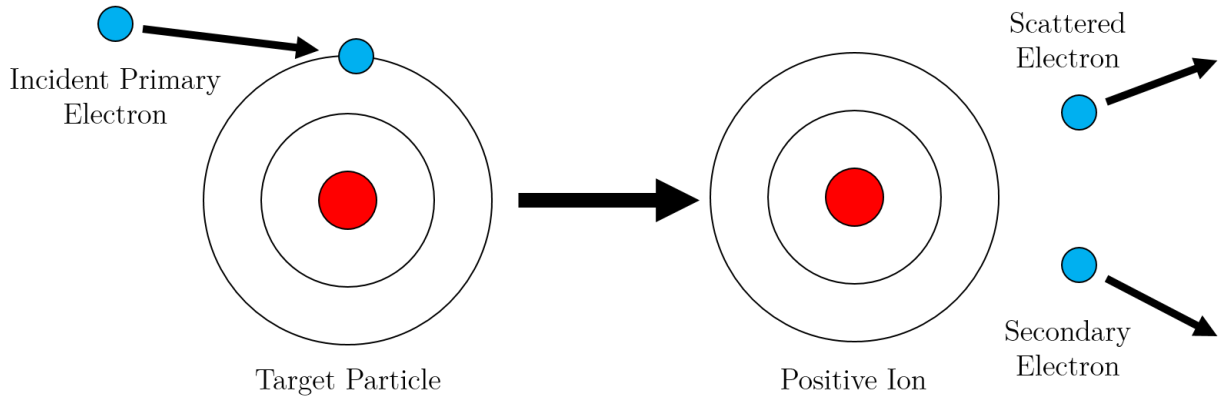

FIG. 2: Diagram of the ionization process.

$$\sigma\left[\mathrm{m}^2\right] = \frac{1.872 \times 10^{-24} A_1}{\beta_e^2} f\left(T\right)\left[\ln\left(7.515 \times 10^4 A_2 \beta_e^2 \gamma^2\right) - \beta_e^2\right] \tag{2}$$
$$f\left(T\right) = \frac{B}{T}\left(\frac{T}{B} - 1\right)$$

where β_e and γ are the relativistic factors for the primary electron, A_1 and A_2 are derived from empirical constants given by Rieke and Prepejchal [59] that depend on the gas species, and $f\left(T\right)$ is a function for fitting σ at low energies, i.e., when the primary electron kinetic energy T is close to the ionization energy B of the target gas particle. Figure 3 shows a log-log plot of the ionization cross section for typical gas species found in a photo-gun vacuum. Low energy electrons are orders of magnitude more likely to ionize than high energy electrons regardless of the target gas particle. The rise in ionization cross section at electron kinetic energies higher than 1 GeV is due to the γ^2 factor being high at these energies.

Dividing Eq. 1 by the timestep Δt and the distance travelled d gives the ion production rate per unit length:

$$\frac{N_{ion}}{\Delta t \cdot d} = \rho\sigma\frac{N_e}{\Delta t} \tag{3}$$

Note that Eq. 3 denotes the *average* ion production rate per unit length and assumes that the gas density is relatively uniform throughout the distance travelled d. In the limit of

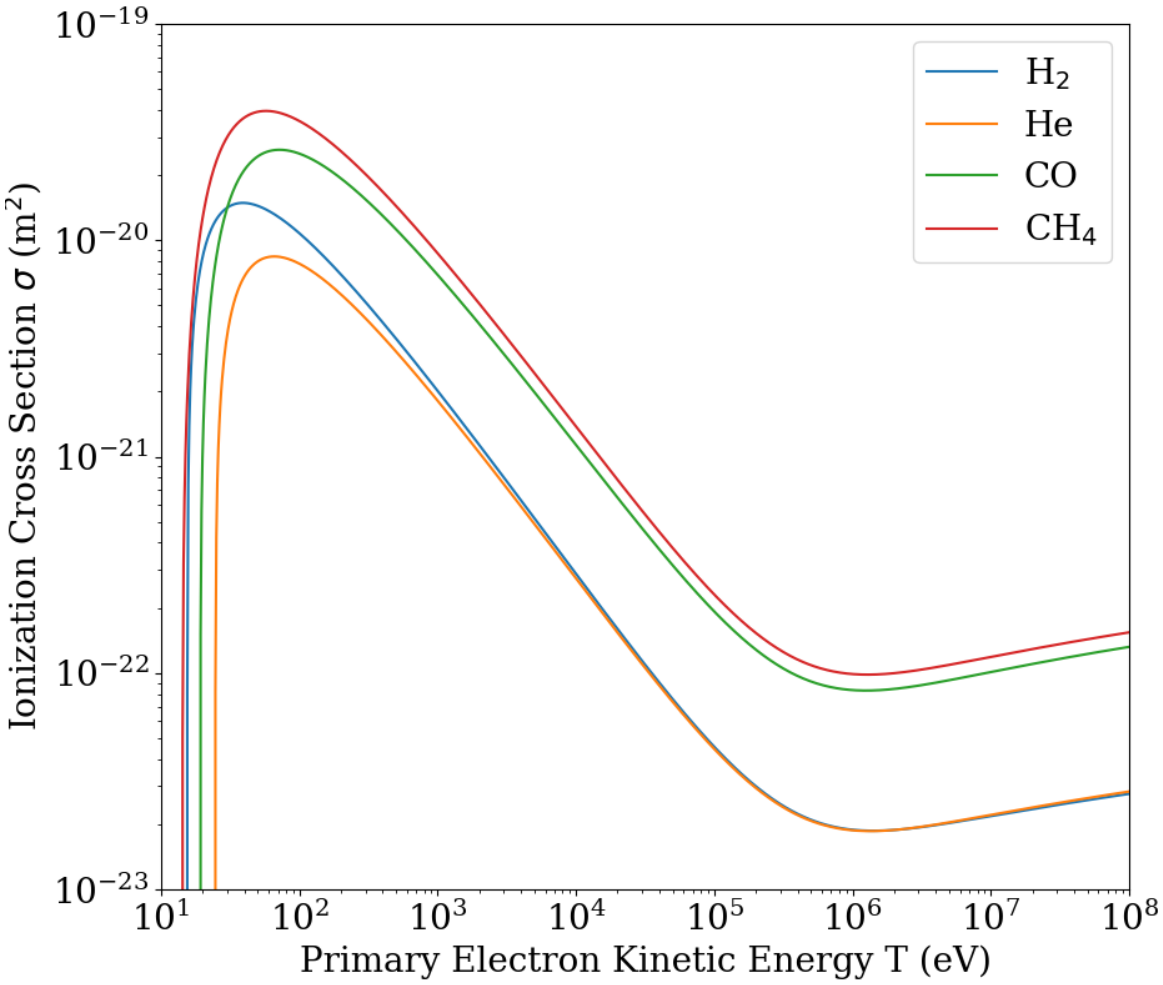

FIG. 3: Log-log plot of the ionization cross section of several target gas particles using Eq. 2.

Primary Electron Energy (keV)	$\sigma_{H_2^+}\,(\mathrm{m}^2)$	$\frac{dN_{\mathrm{H}_2^+}}{dt}\left(\frac{1}{d}\right)\left(\frac{\mathrm{H}_2^+}{\mathrm{s{\cdot}m}}\right)$
0.1	1.1×10^{-20}	2.4×10^5
1	2.0×10^{-21}	4.4×10^4
10	2.9×10^{-22}	6.3×10^3
100	4.6×10^{-23}	1.0×10^3

TABLE 1: Calculated values of the ionization cross section and ion production rate per unit length for H_2 gas.

small timesteps, the instantaneous ion production rate per unit length is given by:

$$\frac{dN_{ion}}{dt}\left(\frac{1}{d}\right) = \rho\sigma\frac{I}{e} \tag{4}$$

where I is the electron current and e is the elementary charge. Figure 4 shows plots of the ion production rate per unit length for various gas species. The ion production rate curves are dominated by the ionization cross section. Table 1 shows values for the ionization cross section and ion production rate per unit length for H_2 gas for various primary electron energies, assuming a typical CEBAF residual gas density of $\rho = 3.56 \times 10^{10}\,\mathrm{m}^{-3}$ and a typical beam current of $I = 100$ µA ($\approx 6.24 \times 10^{14}$ electrons/s). For a beam energy of 130 keV, which is the beam energy of the CEBAF polarized electron source, the current required to produce an ion production rate per unit length of 1 H_2^+ ion per second per meter is approximately 0.1 µA, which is equivalent to 7.4×10^{11} electrons/s. Thus, it is likely that a large number of ions are produced every second within the CEBAF photo-gun.

To give a rough estimate of the number of back-bombarding ions that hit the CEBAF photocathode during a typical run period, we can use Eq. 4 with typical CEBAF conditions: $\rho = 3.56 \times 10^{10}\,\mathrm{m}^{-3}$ (10^{-12} Torr), $\sigma(130\,\mathrm{keV}) = 3.84 \times 10^{-23}\,\mathrm{m}^2$, and $I_e = 100$ µA. In this case, the H_2^+ production rate per unit length along the beamline downstream of the anode is about 844 H_2^+ ions per second per meter. Between the anode and the dipole magnet ($d = 1.61$ m), the beam energy is roughly constant, so the ion production rate in this region is about 1360 H_2^+ ions per second.

Within the cathode-anode gap ($d = 6$ cm), electrons undergo acceleration. In this case,

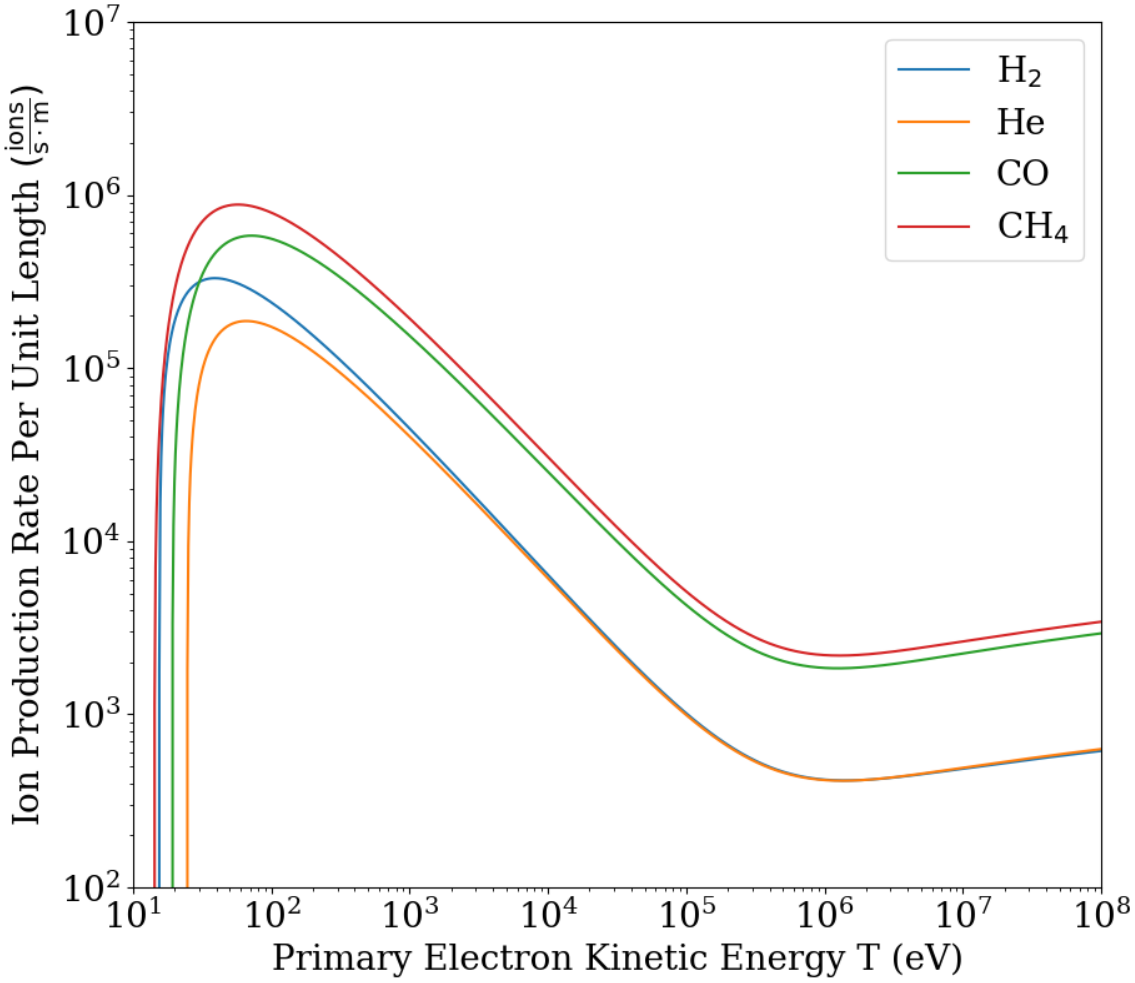

FIG. 4: Log-log plot of the ion production rate per unit length of several target gas particles using Eq. 4. A typical CEBAF residual gas density of $3.56 \times 10^{10}\,\mathrm{m}^{-3}$ is assumed.

the ion production rate can be integrated along the beam trajectory from the cathode to the anode. With the approximation that the electric potential (and hence the kinetic energy of the beam) increases linearly with distance between the cathode and anode, the average ion production rate per unit length in this region is about 3320 H_2^+ ions per second per meter while the ion production rate is about 266 H_2^+ ions per second. The higher ion production rate per unit length in the cathode-anode gap is due to lower energy electrons having higher ionization cross sections. The lower ion production rate is due to the shorter distance.

2.4 SECONDARY ELECTRON ENERGY DISTRIBUTION

The distribution of possible secondary electron energies W for a given primary electron energy T is given by the differential cross section derived in the Binary Encounter Dipole Model [60]:

$$\begin{aligned}\frac{d\sigma(W,T)}{dW} &= \frac{S}{B(t+u+1)}\left[\frac{\left(\frac{N_i}{N}\right)-2}{t+1}\left(\frac{1}{w+1}+\frac{1}{t-w}\right)\right. \\ &+ \left.\left(2-\frac{N_i}{N}\right)\left(\frac{1}{(w+1)^2}+\frac{1}{(t-w)^2}\right)+\frac{\ln t}{N(w+1)}\frac{df(w)}{dw}\right] \qquad (5)\\ S &= 4\pi a_0^2 N R^2/B^2 \\ t &= T/B \\ u &= U/B \\ w &= W/B \\ N_i &= \int_0^\infty \frac{df}{dw}dw\end{aligned}$$

where a_0 is the Bohr radius, R is the Rydberg energy, N is the number of electrons in the subshell of the gas molecule prior to ionization, U is the average kinetic energy of electrons in the subshell, and df/dw is the differential oscillator strength. Because the differential oscillator strengths for CO and CH_4 are not well-known, Kim and Rudd give an approximation for the differential cross section [60]:

$$\frac{d\sigma(W,T)}{dw} = \frac{S}{t+u+1}\left[\frac{1}{(t-w)^2}+\frac{1}{(w+1)^2}-\frac{1}{t+1}\left(\frac{1}{t-w}+\frac{1}{w+1}\right)\right.$$
$$\left.+\ \ln t\left(\frac{1}{(t-w)^3}+\frac{1}{(w+1)^3}\right)\right] \tag{6}$$

Figure 5 shows differential cross-section curves of each supported gas species assuming a primary electron energy of 130 keV. The majority of secondary electrons produced are likely to have kinetic energies substantially smaller than primary electron kinetic energies.

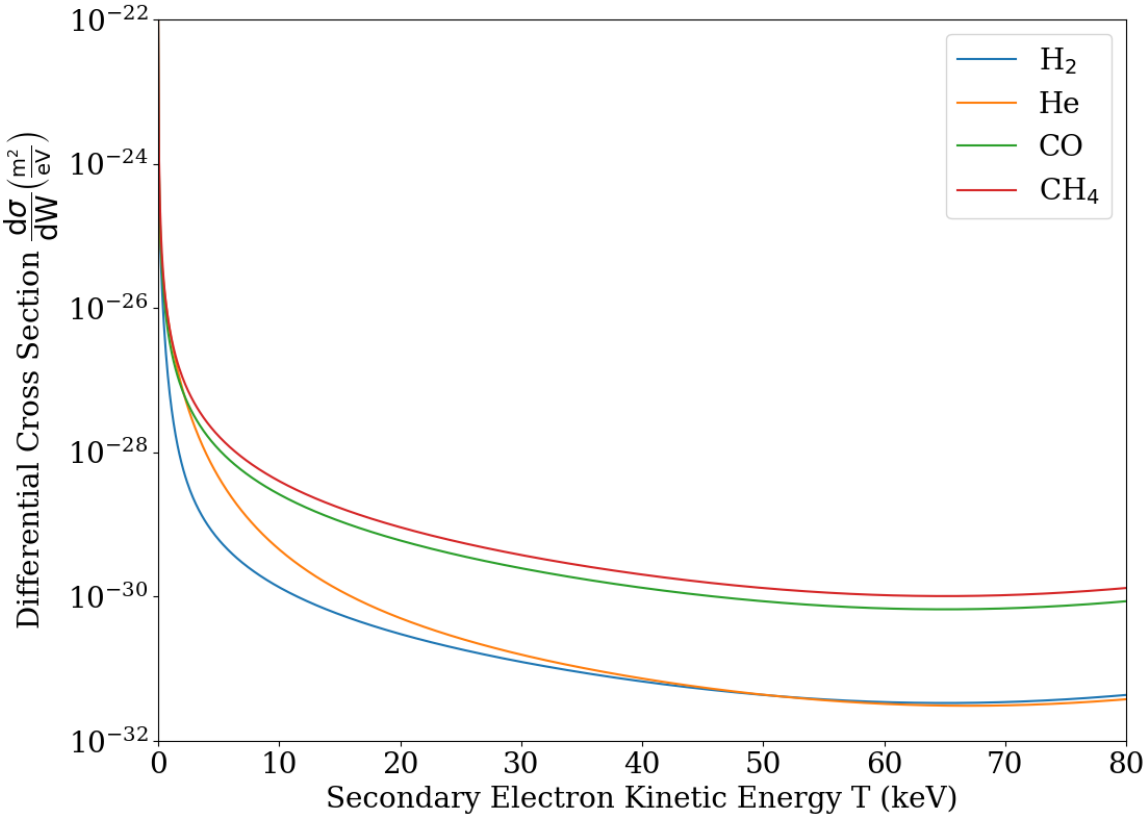

FIG. 5: Log plots of the differential cross section for secondary electron energies of several target gas particles assuming a primary electron energy of 130 keV.

2.5 ION ENERGY DISTRIBUTION

Because the primary electron mass is orders of magnitude less than any target gas particle, the kinetic energy of the ion depends largely on the energy of the target gas particle. The CEBAF photo-gun and injector beamline are assumed to be at room temperature (293.15 K) and have an average kinetic energy of 38 meV. Provided the target gas particle follows the ideal gas law, the distribution of ion energies can be modelled as a Maxwellian distribution, which is often defined in terms of speed v:

$$F(v)dv = \sqrt{\frac{2}{\pi}} \frac{v^2 \exp -\frac{v^2}{2a^2}}{a^3} dv \tag{7}$$
$$a = \left(\frac{kT}{m}\right)^{\frac{1}{2}}$$

where k is the Boltzmann constant, T is the gas temperature, and m is the mass of the ion.

It is difficult to estimate precisely the amount of energy transferred to the ion in an ionization event. Because eV-scale energies must be transferred to a neutral atom to ionize it, and in order to allow specific computations to be made, it will be assumed in subsequent calculations that the most probable ion speed, $v_{\text{prob}} = \sqrt{2}a$, corresponds to a kinetic energy 4 eV. Figure 6 shows the Maxwellian distributions of ion speeds and Table 2 lists the most probable speeds and the mean speeds ($\overline{v} = 2v_{\text{prob}}/\sqrt{\pi}$) of various ion species.

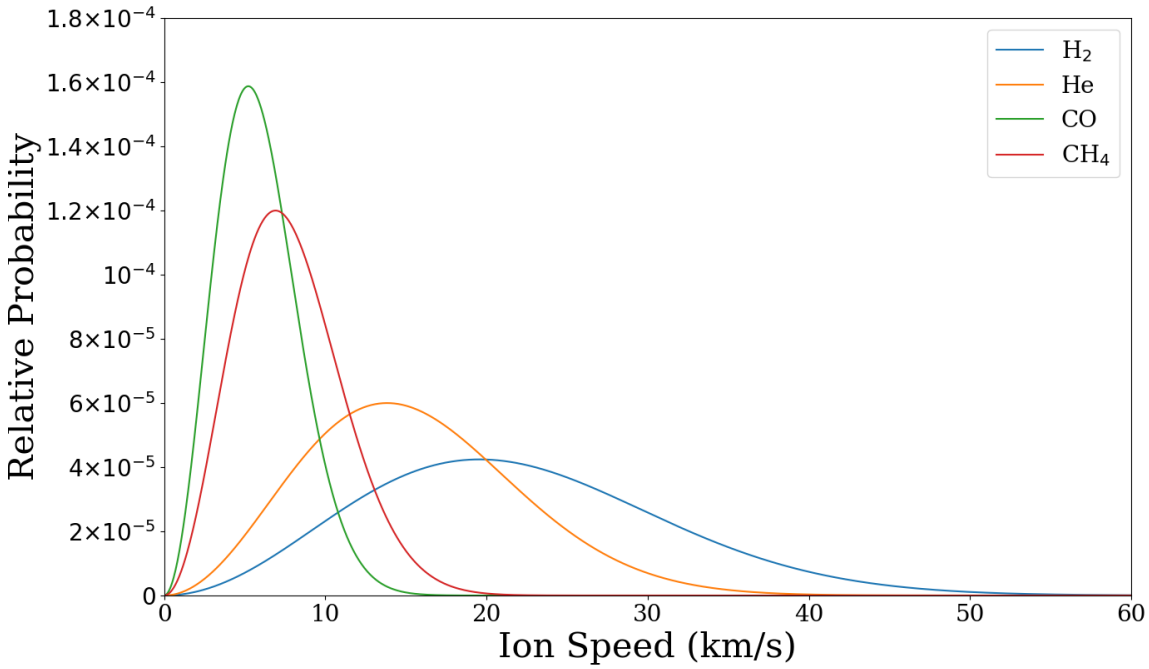

FIG. 6: Maxwellian distribution for various ion species with the most probable speed corresponding to an energy of 4 eV.

Ion Species	v_{prob} (m/s)	$\overline{v}$ (m/s)
H_2	19572	22084
He	13840	15617
CO	5230	5902
CH_4	6919	7808

TABLE 2: Table of the most probable ion speed and the mean ion speed for various ion species.

CHAPTER 3

EXPERIMENTAL SETUP

Experiments took place at the CEBAF accelerator to study the efficacy of biasing the anode to limit ion back-bombardment. While the entire accelerator was in operation throughout each experiment, the experiments focused on the initial section of the injector, specifically from the DC high-voltage photo-gun to the first Faraday cup. Figure 7 shows a diagram of the CEBAF accelerator denoting its main components and the location of the injector.

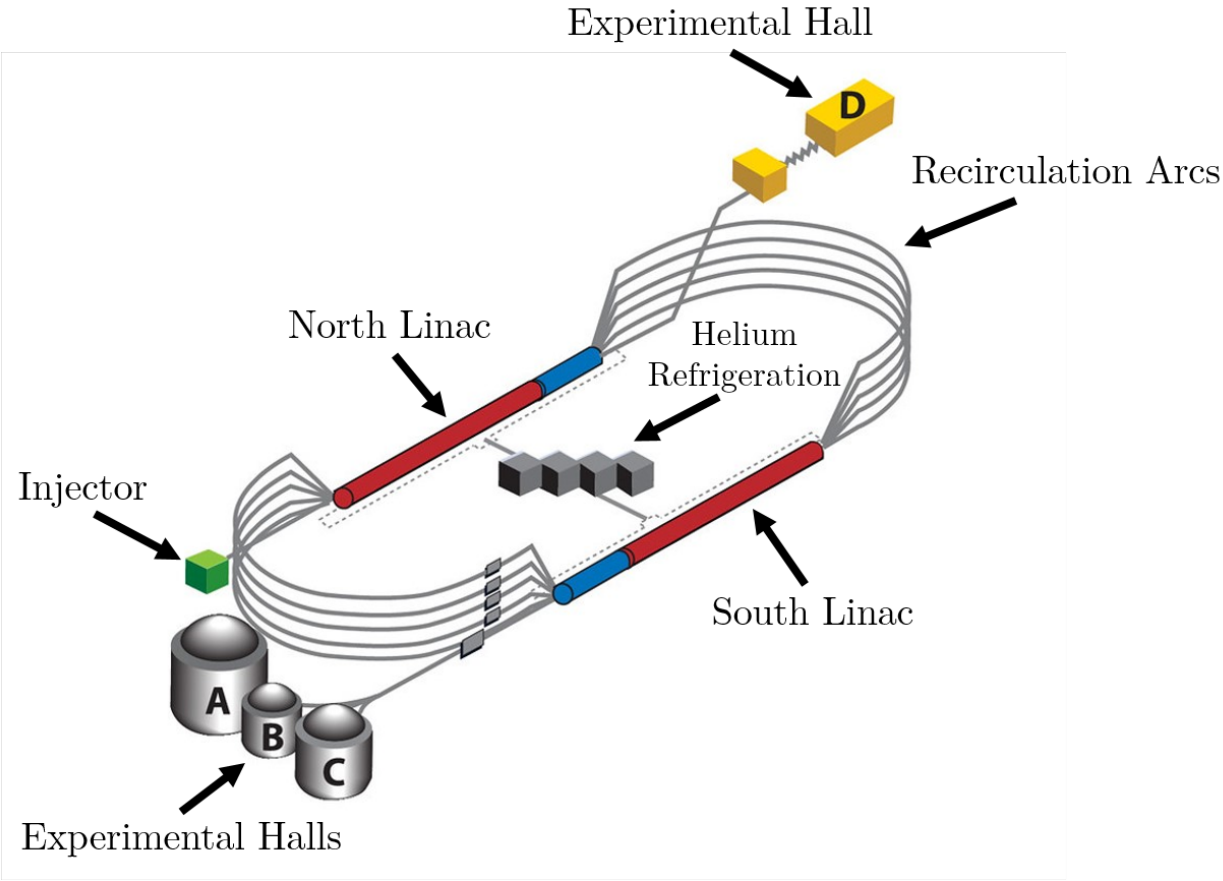

FIG. 7: Diagram of the CEBAF accelerator.

In this chapter, the experimental setup and experimental methods are described. First, the photo-gun, its constituents, and the photocathode preparation are described. Then the setup of the four lasers, the elements along the beamline of the polarized electron source, and the two main measurement techniques, QE measurement and QE scan, are described in detail.

3.1 CEBAF POLARIZED ELECTRON SOURCE

The CEBAF polarized electron source creates a spin-polarized electron beam using a DC high-voltage photo-gun. Figures 8 and 9 show layouts of the CEBAF polarized electron source extending from the photo-gun to the Faraday cup with beamline elements relevant to the biased anode experiments labelled. Note that the location of viewers #2 and #3 were installed following the biased anode experiments and were used in the biased anode tests described in Ch. 4. Approximate locations of these viewers are shown in Fig. 9. The electron beam originates within the photo-gun and then passes through the beamline. The electron beam itself is composed of four interleaved electron beams using four lasers, allowing four experimental halls to receive electron beams simultaneously. Figure 10 shows a diagram of the CEBAF interleaved electron beam.

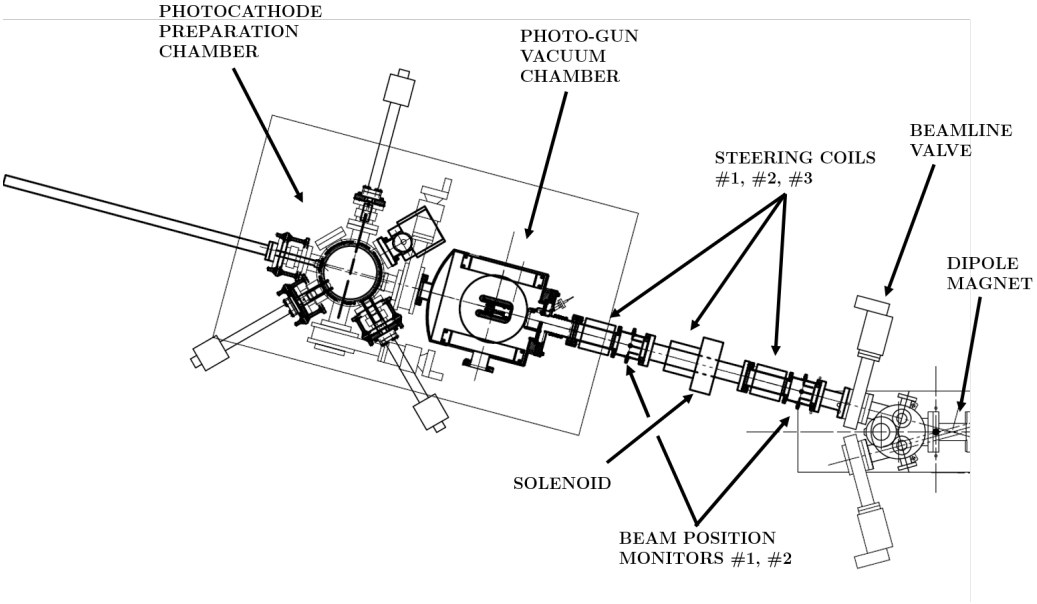

FIG. 8: Layout of CEBAF Injector extending from the photo-gun to the dipole magnet. The steering coils, beam position monitors, and viewer are numbered in order of appearance along the beamline.

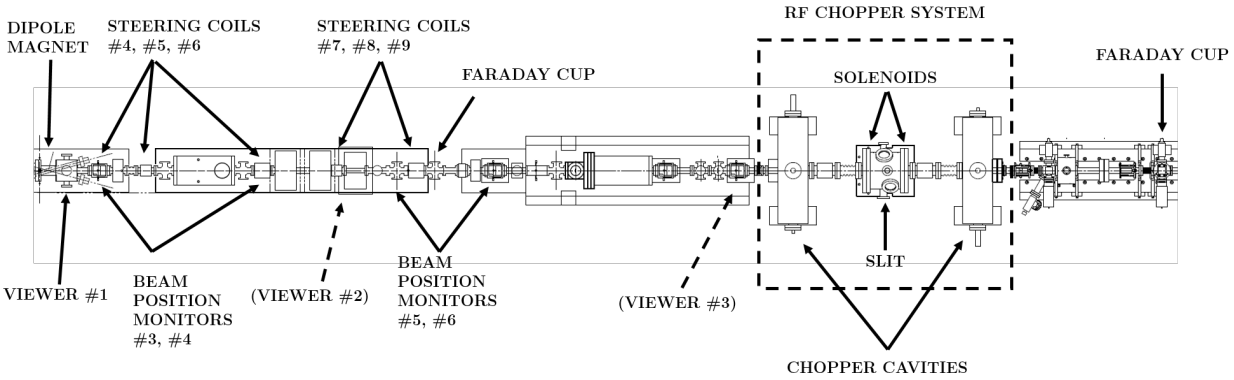

FIG. 9: Layout of CEBAF Injector extending from the dipole magnet to the Faraday cup. The steering coils, beam position monitors, and viewers are numbered in order of appearance along the beamline.

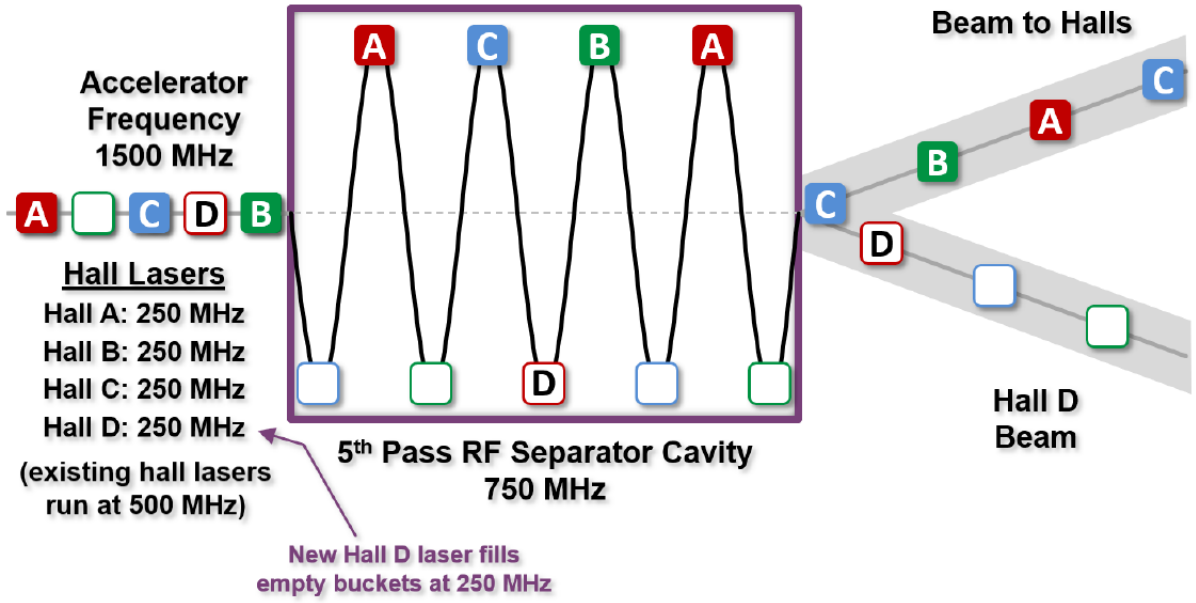

FIG. 10: Diagram of the CEBAF interleaved electron beam using four 248.5 MHz lasers [61, 62].[1]

3.1.1 PHOTO-GUN

Figure 11 shows a schematic and photo of the DC high-voltage photo-gun vacuum chamber and its main components. The cathode is a tee-shaped electrode made from 316LN stainless steel. It is located at the center of the photo-gun and is biased via a high-voltage cable connected to a high-voltage power supply. An inverted ceramic insulator insulates the HV cable from the gun chamber vacuum. The area where the cathode and insulator meet vacuum is known as the "triple-point," and has been shown to have sufficiently high electric field when the cathode is energized to lead to high-voltage breakdown [63, 64]. To reduce the electric field at this point, the design of the cathode was modified to include an electrostatic shield surrounding the triple point [65]. The Pierce geometry of the front edge of the cathode focuses electrons leaving the photocathode through the anode aperture [66].

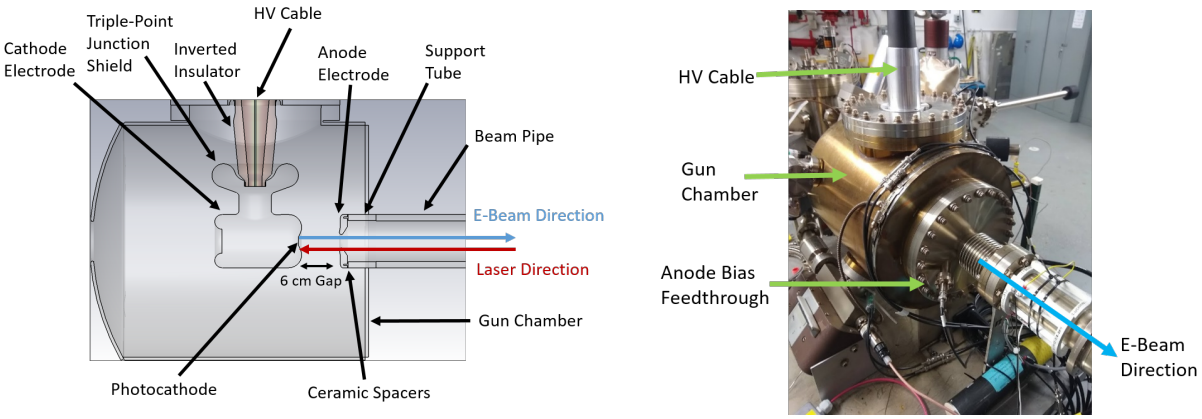

FIG. 11: Schematic (left) and photo (right) of the CEBAF photo-gun.

3.1.2 BIASED ANODE CONFIGURATION

Previously, the anode electrode was grounded. To bias the anode, the support assembly was modified to electrically isolate the anode from the support tube using a ceramic spacer. The downstream flange of the photo-gun chamber was then modified to include a UHV compatible 10 kV high voltage feed-through to allow the anode to be independently biased with a local power supply. A JLab ion pump power supply was used during to bias the anode during run periods 1 and 2 a Bertan (BA-315) power supply during run period 3. An advantage of the former power supply is the ability to control the voltage remotely, whereas the latter power supply must be manually adjusted.

Electrostatic models of the grounded and biased anode configurations of the photo-gun were made using CST Studio software. Figure 12 shows plots of the electric potential along the central axis for a −130 kV cathode and either 0 V or 1000 V anode. A 1000 V anode bias voltage creates a potential peak that is predicted to be sufficient to repel ions downstream of the anode. Due to the geometry of the anode electrode, this peak occurs approximately 3 cm downstream of the anode aperture. Because ions created downstream of the anode peak may still experience significant effects of the electric potential due to their relatively small initial kinetic energies, the electrostatic model extends from the photocathode to 0.935 m downstream of the photocathode, after which the potential is negligible.

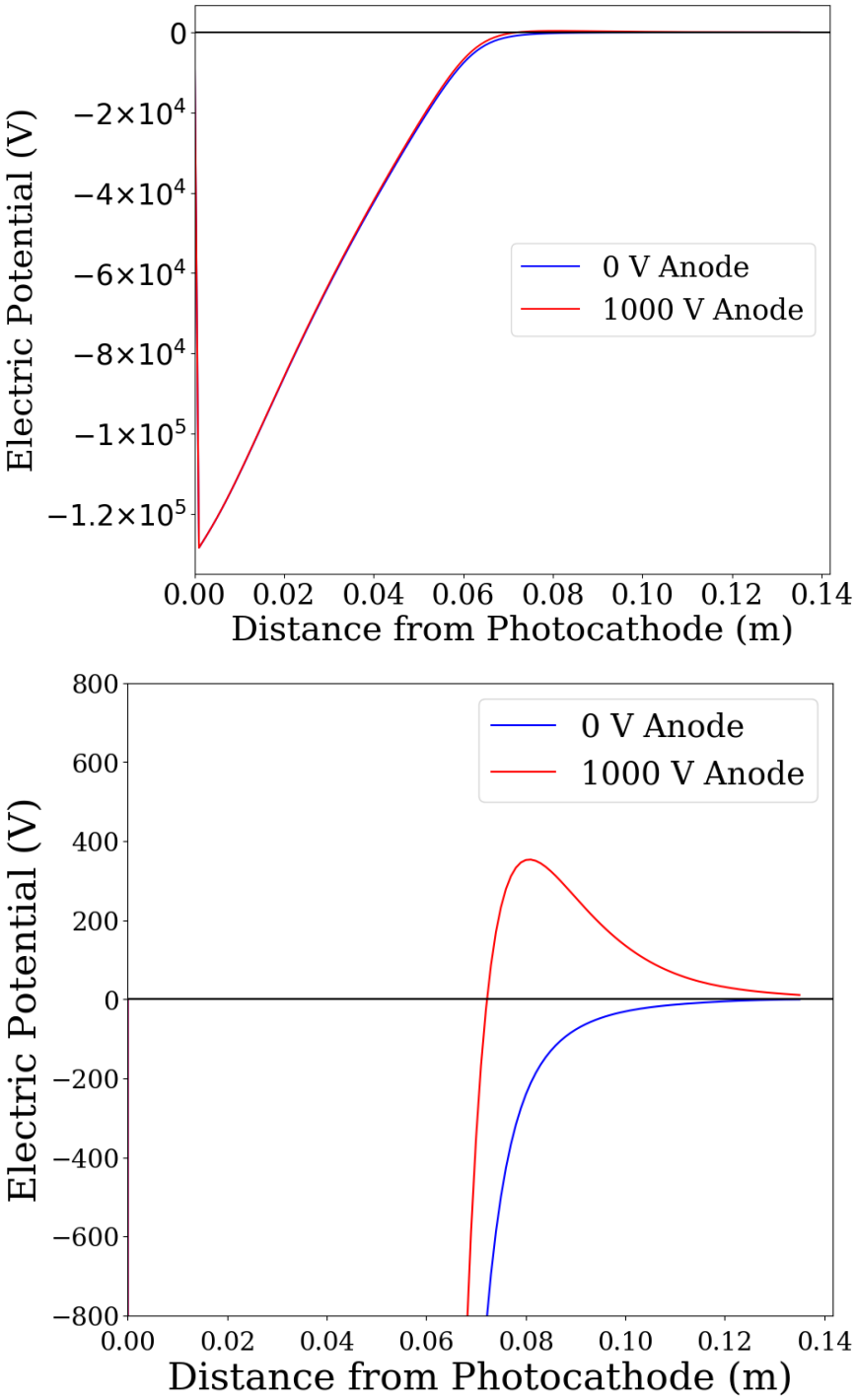

FIG. 12: Plots of the electric potential along the central axis of the beamline for 0 V and 1000 V anode configurations. The top plot shows the full range of the photo-gun potential, while the bottom plot is zoomed into $|V| \leq 800$ V to better see the positive potential peak when the anode is biased.

3.1.3 POLARIZED PHOTOCATHODE PREPARATION

The photocathode used was a strained superlattice GaAs/GaAsP photocathode. It is composed of GaAs grown atop a GaAsP substrate. The mismatch between the two epitaxial layers leads to a strain in the crystal structure that allows incident light to excite electrons with a preferential spin state. Unlike in a strained-layer photocathode, the GaAsP substrate in a superlattice photocathode is composed of 10-20 alternating thin layers of GaAs and GaAsP. These layers substantially increase the quantum efficiency of the photocathode by allowing excited electrons to efficiently pass through the layers [67]. Figure 13 shows the structure of the strained superlattice photocathode and polarization as a function of incident photon wavelength.

The photocathode was cut from a 2-inch wafer using a diamond-tip scribe. To prepare the photocathode for use in the photo-gun, the photocathode is indium soldered to a 3 cm diameter puck made of molybdenum, as molybdenum has a small thermal expansion coefficient. Next the puck is placed into a loading chamber, which is brought under vacuum to 10^{-10} Torr and then heated to 150° C for eight hours to remove contaminants from the photocathode.

The puck is then brought back to room temperature and then moved into a preparation chamber, where the photocathode is activated to create the NEA condition necessary for photoemission. To do this, the "yo-yo" method [68, 69] was used, in which caesium, by evaporation from a SAES caesium metal dispenser [70], and NF_3 from an external source, were applied to the surface in alternation until the optimal quantum efficiency was reached. Typically only a small area of the photocathode is activated instead of the entire photocathode surface to eliminate the possibility of unwanted emission from the edges of the photocathode. For the biased anode experiments, the active area was limited to a 5 mm diameter area at the center of the photocathode using an activation mask. After activation, the puck is moved into the gun chamber. The puck passes through the back of the cathode electrode and is held in place using spring-loaded rollers within the cathode such that the activated surface of the photocathode faces the anode.

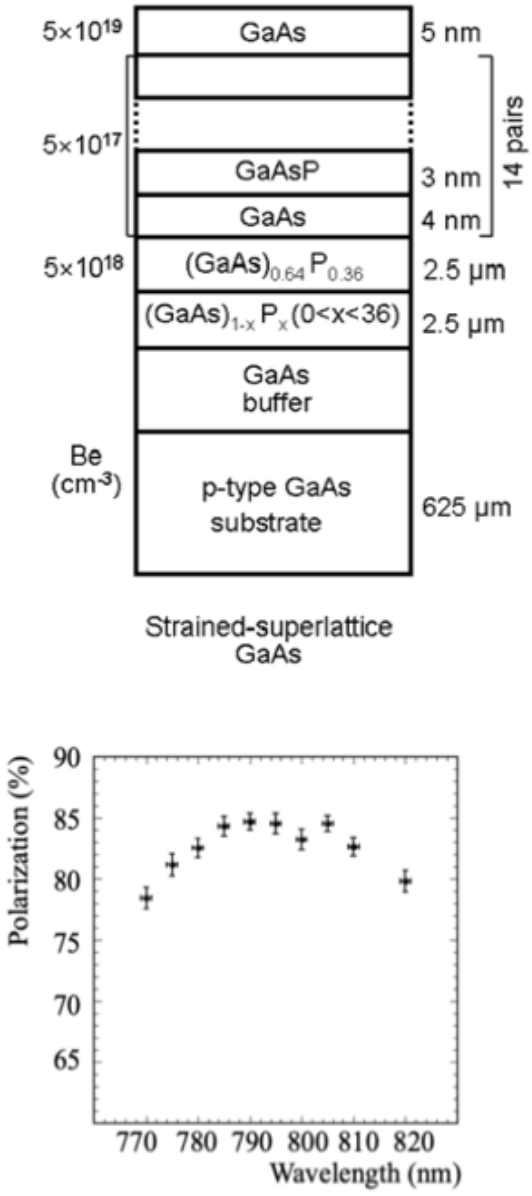

FIG. 13: Diagram of the structure and polarization of a strained-superlattice photocathode [71].[2]

3.1.4 PHOTO-GUN VACUUM CONDITIONS

The photo-gun chamber is maintained at ultrahigh vacuum ($\sim 10^{-12}$ Torr) using ten non-evaporable getter (NEG) pumps at the bottom of the chamber [72, 73]. The NEG pumps are effective at removing residual gases, such as H_2 gas, by absorbing the gas molecules into the NEG material [74]. NEG material also coats the inner lining of the beam pipe to maintain vacuum within the beam pipe. A 40 L/s differential ion pump is attached to the chamber via a right-angle valve to help remove residual gases that are not efficiently removed by the NEG's, such as noble gases [35].

An SRS-RGA residual gas analyzer (RGA) in conjunction with a Leybold extractor gauge was used to determine the partial pressures of residual gases within the photo-gun. An RGA scan of the photo-gun prior to operation is shown in Fig. 14. The photo-gun vacuum is clearly dominated by H_2 gas, though other gas species such as CO, CH_4 and CO_2 exist in smaller quantities. Because the RGA is calibrated for N_2 gas, the peak pressures must be divided by the respective gas correction factors (given by MKS [75]) to determine the partial pressure of the gas. Note that there are multiple peaks for CH_4 due to ion fragmentation; the highest of these peaks ($m = 16$) corresponds to CH_4.

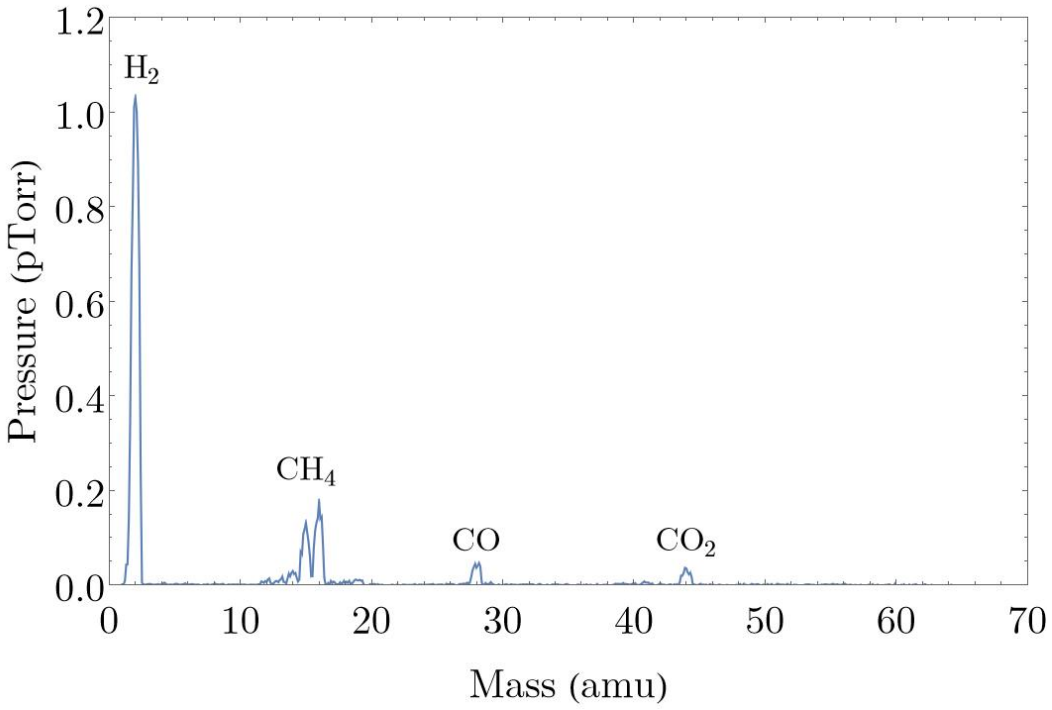

FIG. 14: RGA scan of the CEBAF photo-gun vacuum with the electron beam off.

Assuming the residual gases follow the ideal gas law, the individual gas densities can be calculated from the partial pressures using the formula:

$$n[\mathrm{m}^{-3}] = 2.65 \times 10^{20} P(\mathrm{Pa}) \approx 10^{-5} n_0 P(\mathrm{Pa}) \approx 3.54 \times 10^{22} P(\mathrm{torr}) \tag{8}$$

where $n_0 \approx 2.69 \times 10^{25}\,\mathrm{m}^{-3}$ is the Loschmidt constant. Table 3 shows the partial pressures and densities of each residual gas species in the RGA spectrum. The gases in the RGA spectrum are assumed to be prevalent within photo-gun and beamline during the experiments.

Gas Species	Partial Pressure (Torr)	Density (m^{-3})
H_2	9.4×10^{-13}	3.3×10^{10}
CH_4	3.1×10^{-14}	1.1×10^{9}
CO	1.7×10^{-14}	6.2×10^{8}
CO_2	9.4×10^{-15}	3.3×10^{8}

TABLE 3: Partial pressures and densities of residual gases in the CEBAF photo-gun, as measured in the RGA spectrum.

3.2 LASER SETUP AND PROFILE MEASUREMENTS

Four lasers are used to create electron beams that are delivered to Halls A, B, C, and D. Figure 15 shows a schematic of the physical layout. Each laser produces 780 nm light that illuminates the photocathode. During CEBAF operations, the intensities of the four lasers are adjusted to produce the requested beam current. The light from the four lasers is combined by beam splitters and a prism cube. The combined laser beam pass through a lens controlled by a stepper motor, which can deflect the beam horizontally or vertically, thereby shifting the laser position on the photocathode. An insertable power meter following the lens measures the laser power prior to incidence on the photocathode.

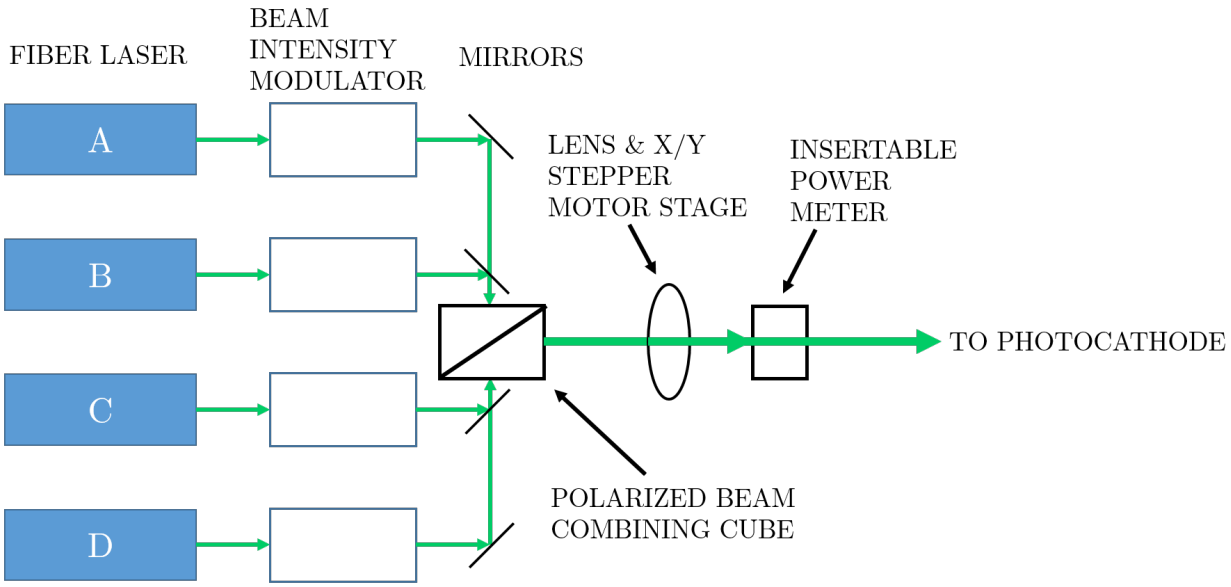

FIG. 15: Schematic of the physical layout of the four fiber lasers.

The transverse intensity profiles of the lasers were measured using a Ophir Spiricon silicon CCD camera [76]. To do this, a mirror is inserted after the lens such that one laser path leads to the CCD camera and the other path leads to the photocathode. Both path lengths are kept equal using a series of mirrors. Figure 16 shows the setup used to take laser profile measurements.

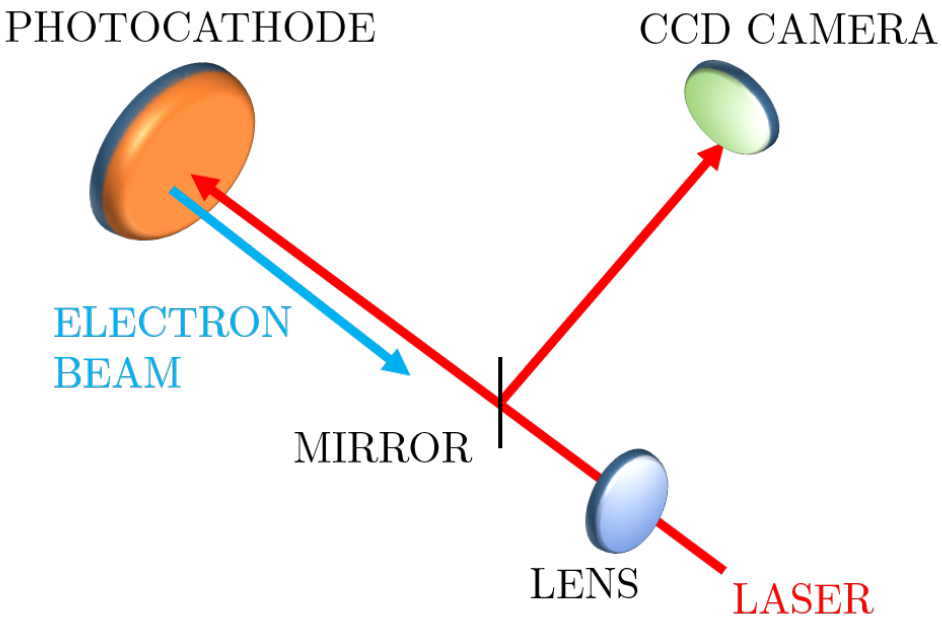

FIG. 16: Diagram of the setup for the laser profile measurements. The coordinate systems of the lasers, (x_L, y_L), and the electron beam (x_B, y_B) are shown for comparison.

The transverse intensity profiles of each laser were fit using BeamGage software [77] to determine the transverse sizes of the lasers. The transverse sizes were calculated using the D4-sigma method, which defines the x- and y-diameters as four times the standard deviation of the intensity profile projected in the x- and y- directions. Figure 17 shows the intensity profiles for each laser and Table 4 shows the x- and y- diameters, assumed to have an uncertainty of $\pm 2\%$ [76]. The x- and y- diameters of each laser are not equal, implying that each transverse intensity profile is slightly elliptical.

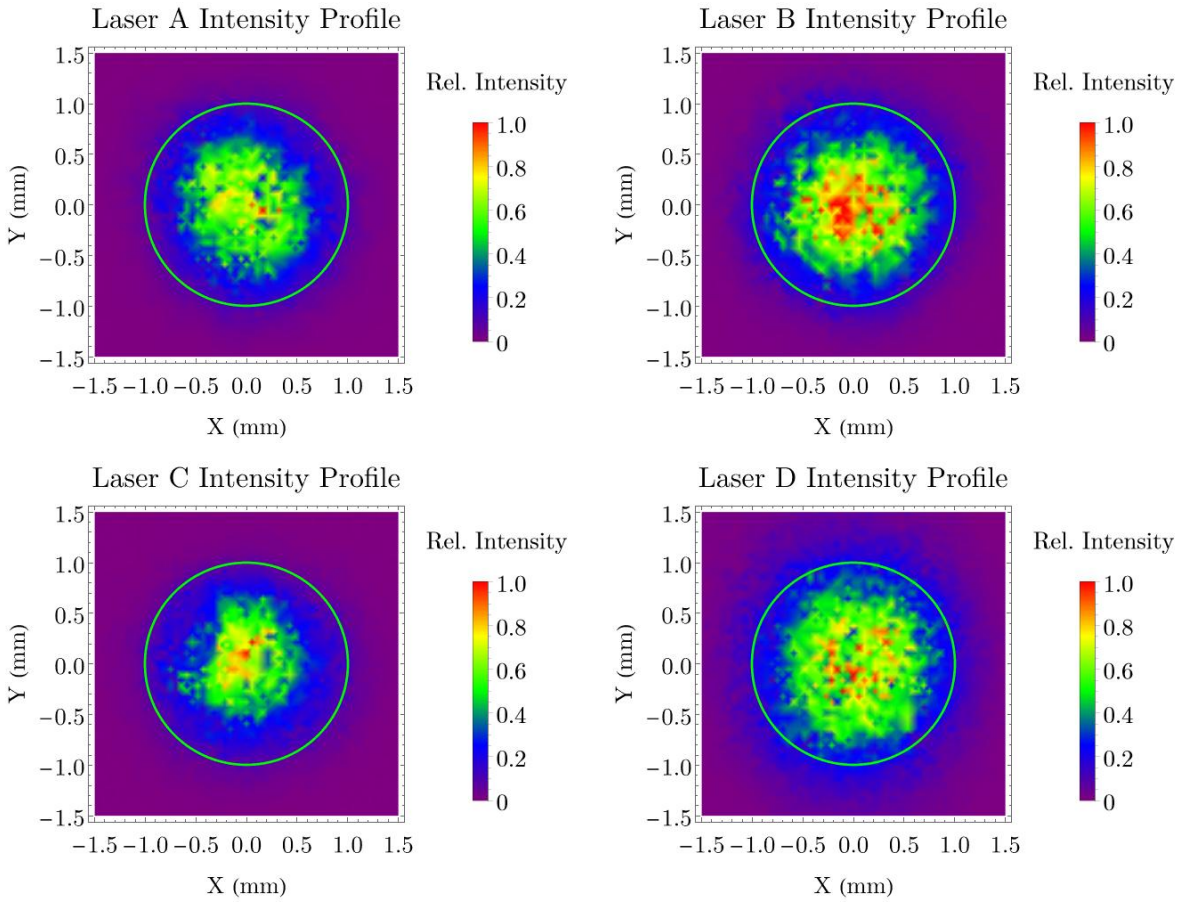

FIG. 17: Relative intensity profiles for lasers A, B, C, and D. The average transverse laser size is denoted by the 2 mm diameter green circle.

Laser	A	B	C	D
x-diameter (mm)	2.14	2.23	2.05	2.84
y-diameter (mm)	1.97	2.15	2.02	2.38

TABLE 4: x- and y-diameters for each laser using the D4-Sigma method.

3.3 CEBAF INJECTOR BEAMLINE

After leaving the photo-gun, the electron beam passes through a beamline where the beam is steered and shaped using steering coils and solenoids. Beam position monitors (BPM's) and viewers are used to track the transverse location and profile of the beam. An RF chopper cavity system controls the beam currents delivered to each experimental hall. The individual beamline elements used are described in the following subsections.

3.3.1 STEERING COILS

The beam is kept centered along the central axis of the beamline using Haimson steering coil pairs (horizontal and vertical) placed along the beamline. Since the beam initially receives a downward kick due to the electrostatic shield above the cathode electrode, two steering coil pairs immediately following the anode re-center the beam with the beamline. Subsequent steering coils correct for beam orbit irregularities and for Earth's magnetic field. Figure 18 shows a photo of a steering coil pair.

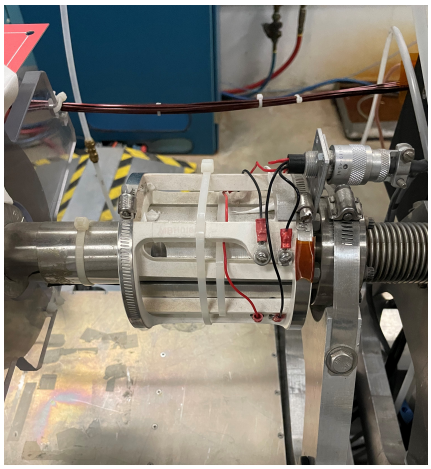

FIG. 18: Photo of a Haimson steering coil pair on the beamline.

3.3.2 SOLENOID

Solenoids provide transverse focusing along the beamline and to ensure proper steering at the steering coils and dipole magnet. The strength of the first solenoid field after the photo-gun is tuned such that the beam size reaches a minimum at the center of the dipole magnet. Figure 19 shows a photo of the first solenoid after the photo-gun.

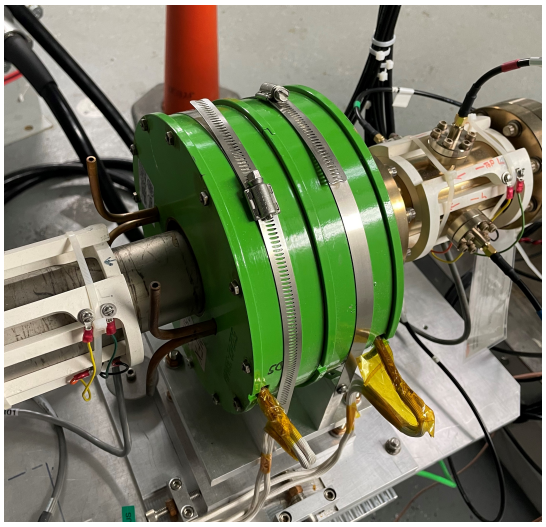

FIG. 19: Photo of a solenoid on the beamline.

3.3.3 BEAM POSITION MONITOR

Beam position monitors (BPM) measure the transverse position of the beam along the beamline. The BPM consists of two pairs of antennae that are parallel to the beamline axis. The transverse position of the beam in the x- or y-directions can be determined based on measured voltage signal induced by the beam. BPM's between the photo-gun and the RF chopping system were used in the initial biased anode tests to measure beam deflection as a result of anode bias. Figure 20 shows a BPM-steering coil pair.

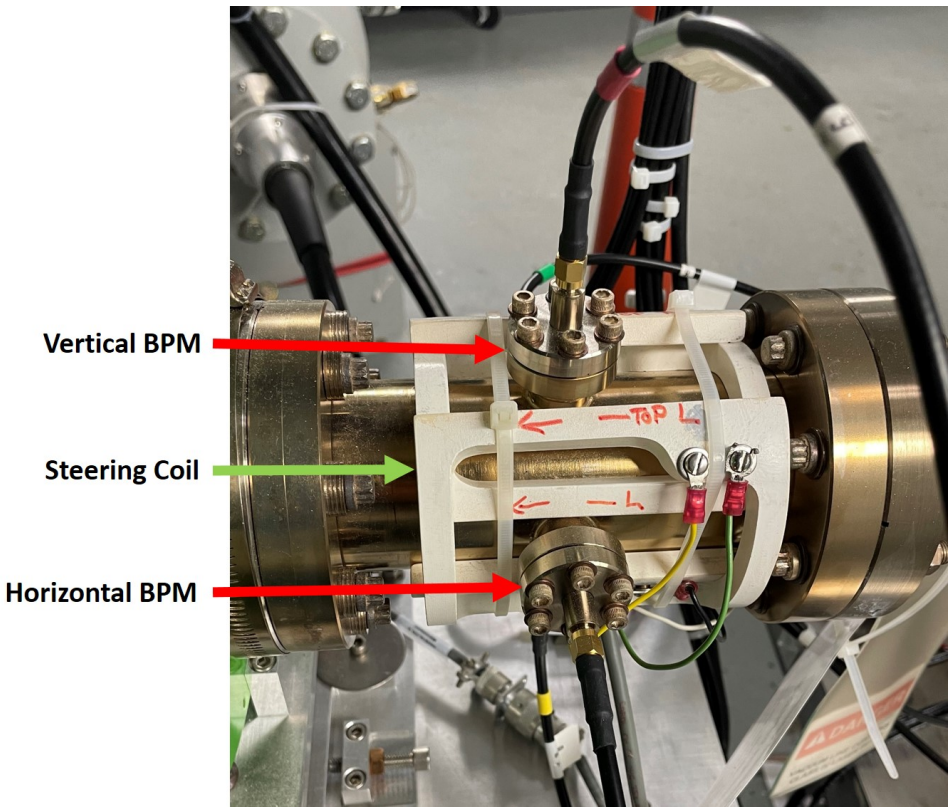

FIG. 20: Photo of a BPM at the location of a steering coil pair.

3.3.4 VIEWER

Viewers consist of an insertable screen made of yttrium aluminum garnet (YAG) that fluoresces when excited by the beam. The screen is oriented 45° with respect to the beam so that a CCD camera captures the fluorescent light passing through a view port. The intensity distribution of the light is proportional to the transverse current distribution of the beam. Since the transverse current distribution of the beam is a 2-D Gaussian, projections of the distribution in the x- and y-directions were measured to determine the transverse sizes of the beam (i.e., σ_x and σ_y) as a function of anode voltage. Figure 21 shows a photo of the first viewer after the photo-gun.

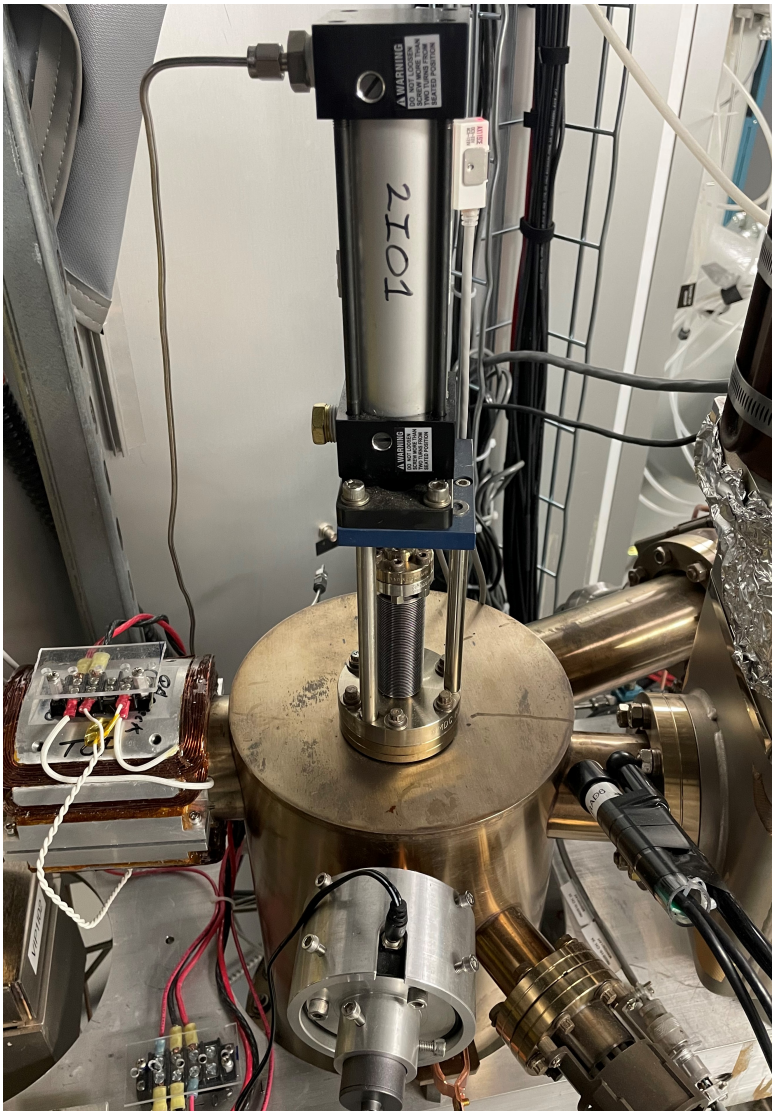

FIG. 21: Photo of a viewer on the beamline.

3.3.5 DIPOLE MAGNET

Following steering coil #3, the beam passes through a dipole magnet where the beam bends 15° to the left. The purpose of the dipole magnet is to allow for normal incidence of the lasers on the photocathode. Figure 22 shows a photo of the dipole magnet.

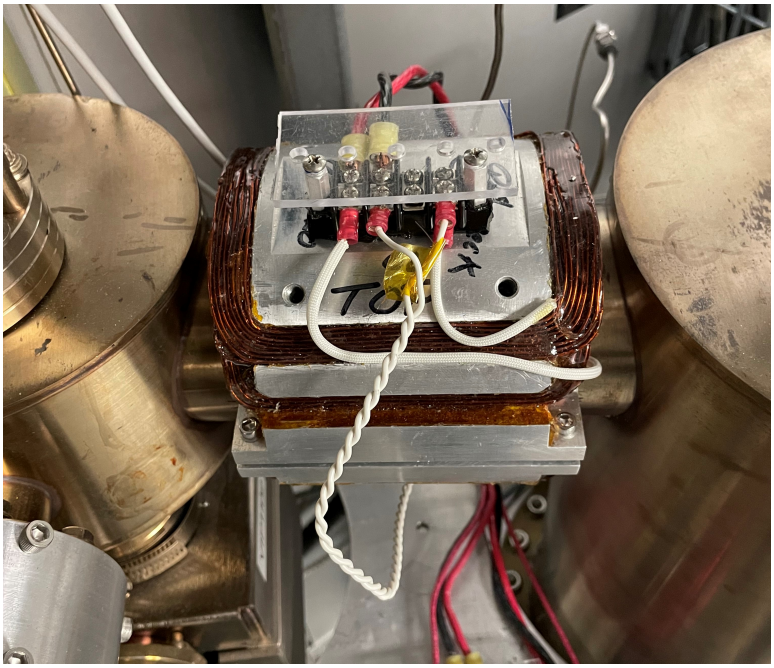

FIG. 22: Photo of the dipole magnet on the beamline.

3.3.6 RF CHOPPER SYSTEM AND FARADAY CUP

After passing through the dipole magnet, the beam passes through two 499 MHz RF chopper cavities with two solenoid lenses and a master slit in between. Figure 23 shows a diagram of the chopping system [78]. The beam enters the first RF chopper cavity and is deflected along a cone whose apex is at the center of the cavity. The deflection axis depends on the phase of the cavity at entry. The beam passes through a solenoid lens to make the beam parallel to the central axis before passing through master slit. The master slit contains three slits of variable aperture size, each corresponding to a different experimental hall. By adjusting the aperture size, one can select the beam current going to each hall. The beam is then re-centered with the beamline axis after passing through a second solenoid and chopper cavity.

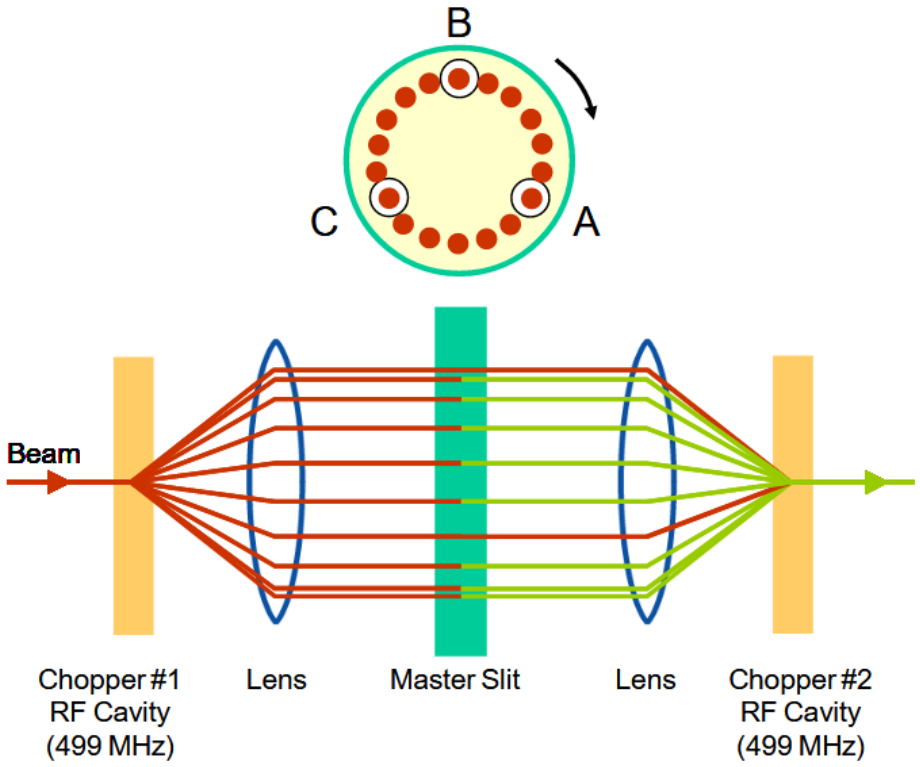

FIG. 23: Diagram of the chopper system containing two RF chopper cavities, two solenoid lenses, and the master slit.

After the chopping system, an insertable Faraday cup measures the beam current. The Faraday cup is made of an isolated electrode in a circuit with an ammeter. When electrons hit the electrode, they induce a current in the circuit that is measured by the ammeter. The Faraday cup can be used in conjunction with the chopping system to measure the longitudinal size of an electron bunch. By tuning the phase of the chopper cavities, the deflection axis of the bunch can be scanned across one of the apertures of the master slit. The current passing through the slit can then be measured by the Faraday cup. Figure 24 shows a photo of the chopper system and the Faraday cup.

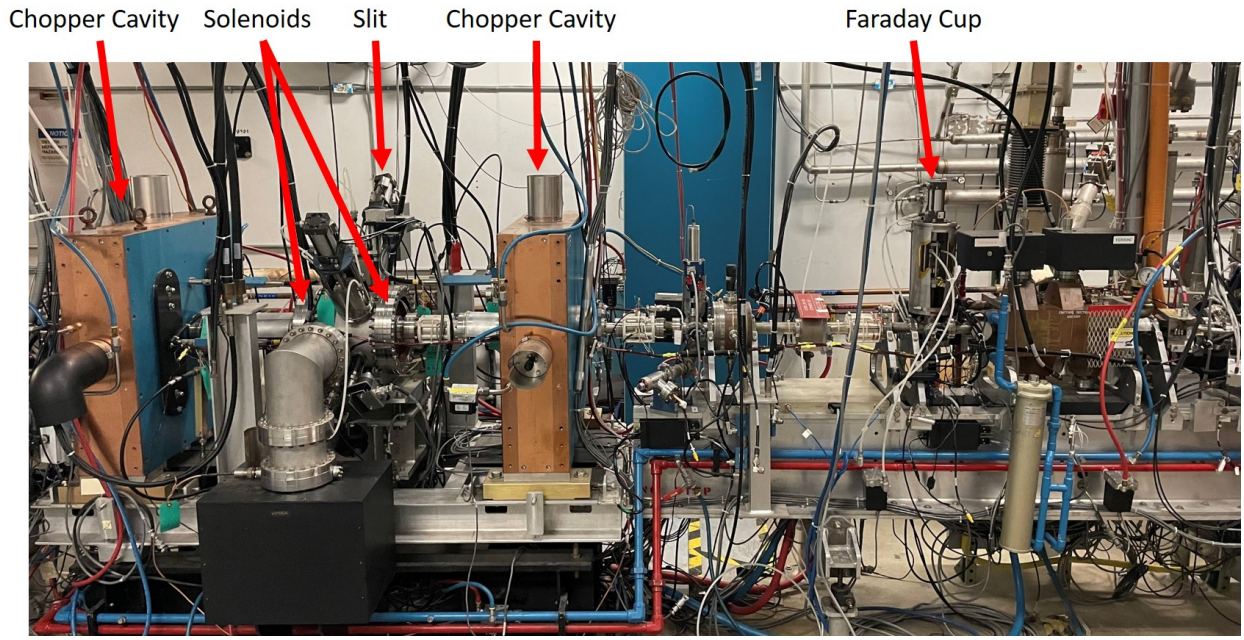

FIG. 24: Photo of the RF chopper system and the Faraday cup.

3.4 QE MEASUREMENT AND QE SCAN

QE measurements were used to track photocathode degradation throughout a run period. In a QE measurement, the electron beam is halted briefly and each laser illuminates the photocathode to produce current, which is collected on the Faraday cup. The power meter is then inserted to measure the laser power. From the laser power P, laser wavelength λ, and the measured current I, the QE can be calculated using the formula:

$$\begin{aligned} \mathrm{QE} &= \frac{hc}{e}\frac{I}{\lambda P} \\ \mathrm{QE}(\%) &= \frac{124 * I\,(\mathrm{mA})}{\lambda\,(\mathrm{nm})\,P\,(W)} \end{aligned} \tag{9}$$

where h is the Planck constant, c is the speed of light, and e is the elementary charge. QE measurements only measure the QE at the laser position on the photocathode active area.

QE scans, on the other hand, were used to measure photocathode degradation across the

entire active area, not just the laser position. In a QE scan, QE measurements are taken using a single laser in a gridded pattern across the active area. To do this, a lens that focuses the laser to a spot on the photocathode is moved by a stepper motor, which ensures that the measurement locations are equally spaced. These measurements are then interpolated to create smooth contour plots of QE across the active area. The units of the QE scan coordinates are originally in stepper motor units (i.e., steps) and the units of the QE values are originally electron currents measured on the anode, which is biased at 250 V. These must be calibrated to mm and QE respectively to create meaningful QE scans. To do this, the QE can be calculated from the current values using the parameters of the laser. To calibrate from steps to mm, the scan is first sliced horizontally and vertically in equal partitions so that plots of the QE across rows and columns can be made. The edge of the active area is at the maximum QE *gradient* along the edge of the QE distribution. The greatest distance (in steps) between two opposite edges is defined to be 5 mm, the diameter of the active area. The locations of the greatest distance in the horizontal and vertical directions are used to define the center of the active area. Because the active area is not always uniform, only the outermost "edges" are used in the calibration. Figure 25 shows an example raw QE scan from run period 2 and its interpolation pre- and post-calibration. By subtracting successive QE scans from each other, QE difference scans can be made to visualize the pattern of QE degradation throughout a given run period.

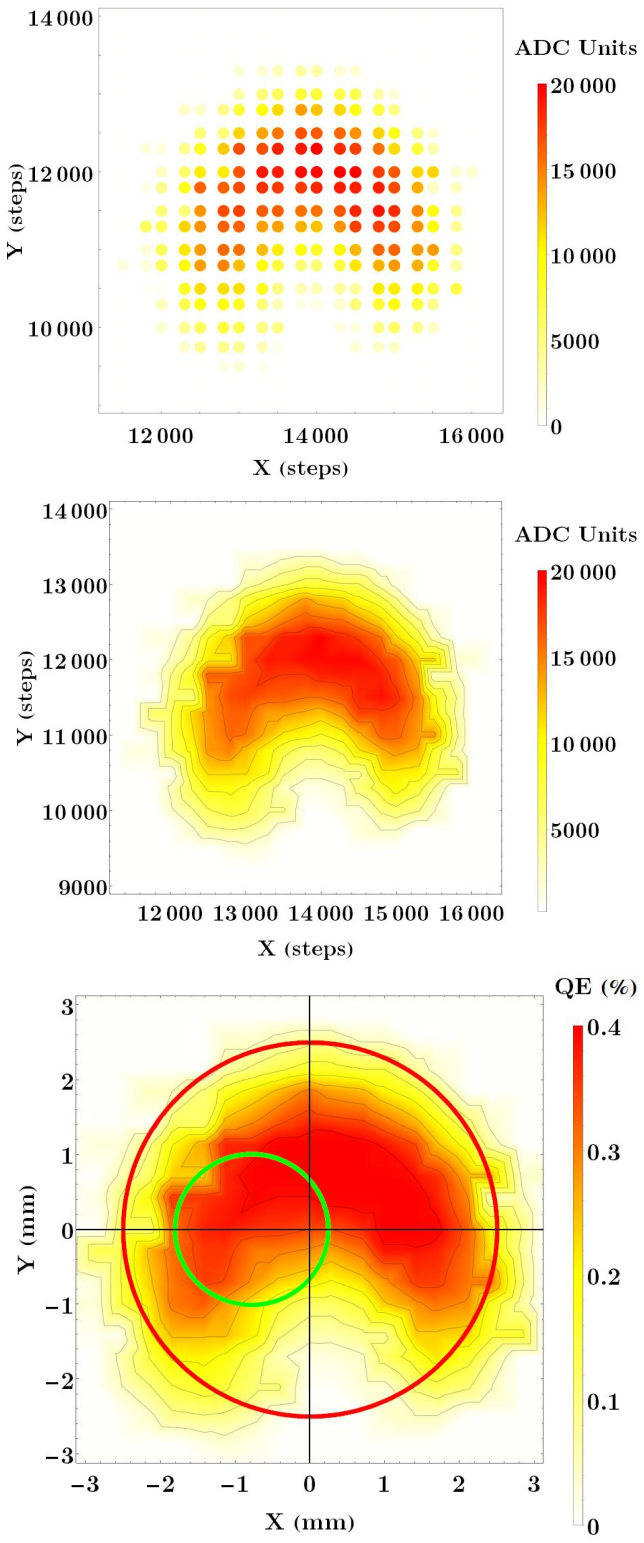

FIG. 25: Example raw QE scan (top), the interpolated QE scan (center), and the interpolated QE scan after calibration (bottom). The red circle denotes the 5 mm diameter active area and the green circle denotes the 2 mm diameter laser spot.

CHAPTER 4

BIASED ANODE EXPERIMENTS

The effectiveness of biasing the anode at increasing the charge lifetime was studied parasitically to three run periods at the CEBAF accelerator. The anode voltage alternated between grounded and biased in the first two run periods and remained biased throughout the third run period. During each run period, QE measurements were made to determine the charge lifetimes for grounded and biased anode configurations and periodic QE scans were made to track the degradation of the active area.

In this chapter, the biased anode experiments and results are described. First, three tests of the effects of biasing the anode on the beam quality are described. Then the results for each run period are described. Finally, the QE measurement and QE scan results are discussed.

4.1 BIASED ANODE TESTS

Biasing the anode is not expected to significantly affect beam quality. While the anode voltage does change the electrostatic potential of the photo-gun, it is expected that this effect is small. Three tests were performed to study the effect of biasing the anode on the beam size, deflection, and timing. The results of the tests are described in the following subsections.

4.1.1 BEAM SIZE VERSUS ANODE VOLTAGE

The anode electrode behaves as an electrostatic lens due to its axial symmetry and defocuses the beam. However, because the cathode potential is much greater than the anode potential, the increase in beam size is expected to be small. To test this, the transverse intensity profile of the beam was measured for anode voltages between 0 V and 1000 V in increments of 250 V using viewers #1-3. Figure 26 shows an EPICS screen of a beam size measurement at viewer #1 [79]. The transverse projections of the beam intensity were fit with Gaussian functions to calculate the beam size to one standard deviation in the $x-$ and $y-$directions at each viewer.

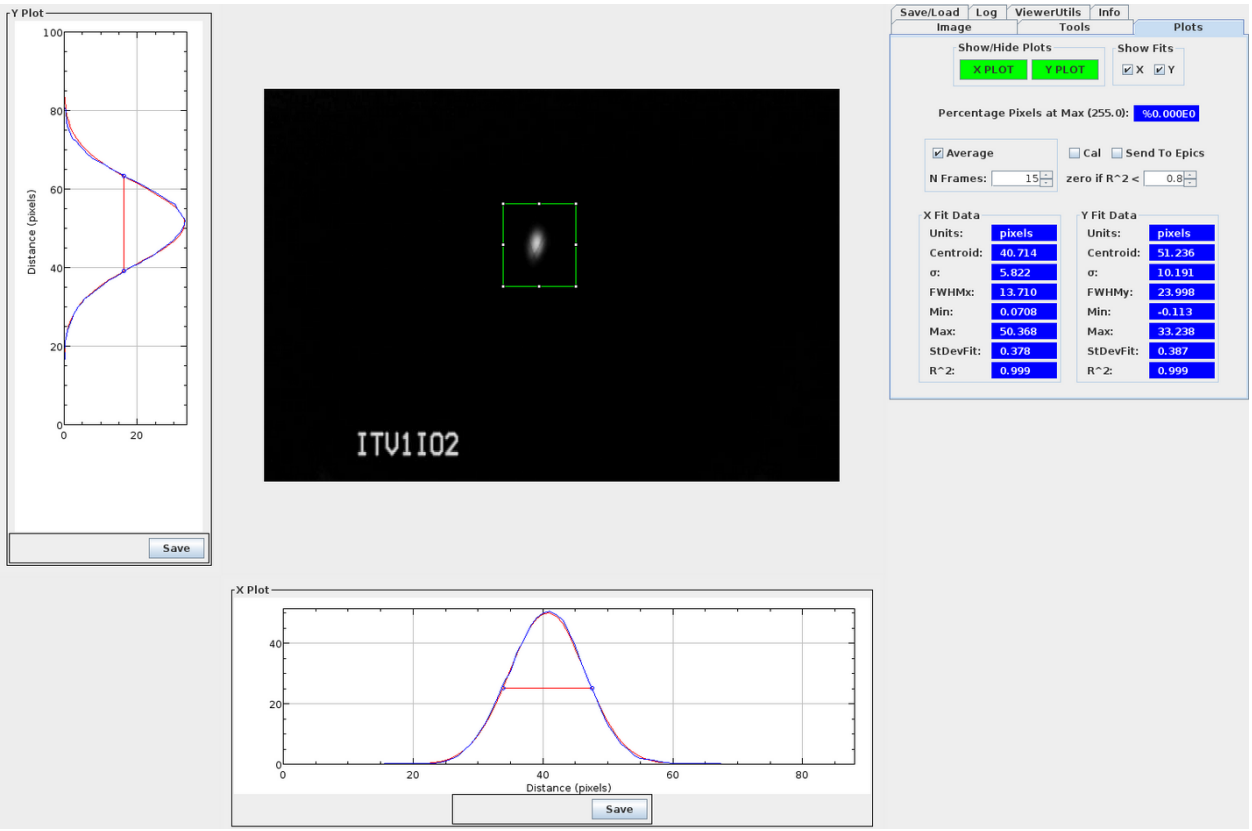

FIG. 26: EPICS screenshot of a beam size measurement using viewer #1.

The beam size was measured five times for each anode voltage at each viewer. The beam sizes were then averaged and subtracted from the beam size at 0 V to get the relative beam sizes for each anode voltage. Figure 27 shows the relative beam size at each viewer for each anode voltage. The error bars correspond to the standard deviation of the five beam size measurements. The beam sizes were fit with lines to determine the extent to which the size changes with anode voltage. While the beam size is seen to decrease on viewer #2, the difference in size between 0 V and 1000 V anode voltages is about 1.33 ± 0.46 pixels, or about 5.80 µm. The resolution of the CCD camera is 4.4 µm per pixel, so the change in beam size is small.

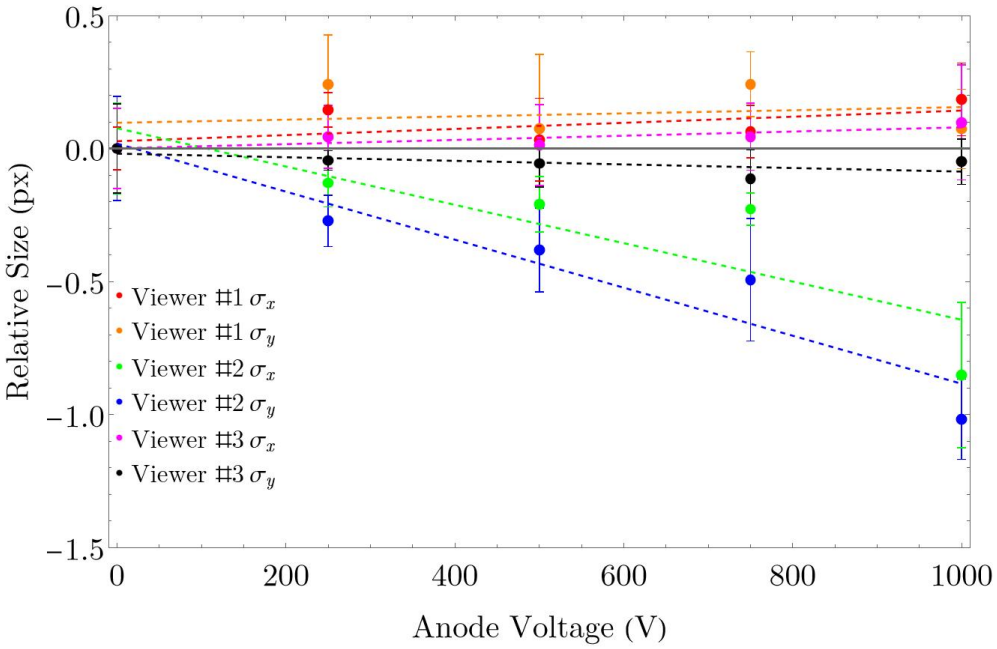

FIG. 27: Transverse beam size measurements for each viewer as a function of anode voltage.

4.1.2 TIME OF FLIGHT VERSUS ANODE VOLTAGE

When biasing the anode to 1000 V, the beam energy increases by roughly 400 V (see Fig. 12). This increase in beam energy will decrease the time of flight. However, this decrease is expected to be small. To test this, chopper scans were performed for anode voltages between 0 V and 1000 V in increments of 250 V. In each chopper scan, the electron beam is scanned across slit A (see Fig. 23) by adjusting the phase of the RF cavities between 90° and 120° in one-degree increments. Electron current that passes through the slit is collected at the Faraday cup and measured using a picoammeter. The phase values in all chopper scans were shifted by −105° to center the data at 0°. Each data set was then fit with a Gaussian function to determine the phase of the peak current. Figure 28 shows the chopper scans for each anode bias and Table 5 shows the phase of the peak current for each anode bias. The uncertainty values correspond to fit errors.

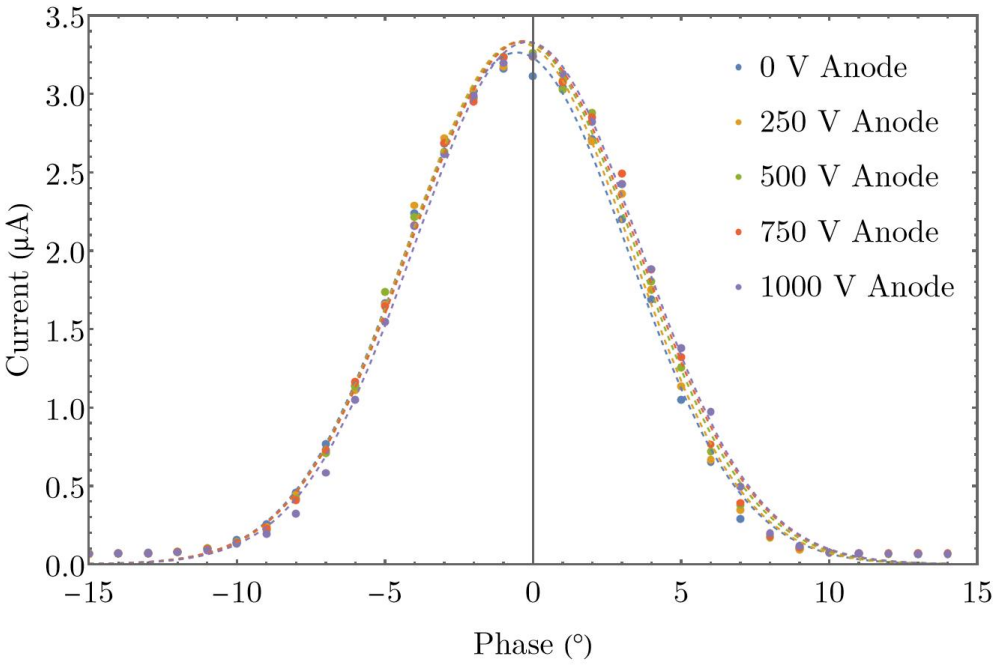

FIG. 28: Chopper scans across slit A for each anode voltage. The dashed lines represent Gaussian fits through each scan.

There is a shift in peak current phase to the right (i.e., positive phase) with increasing anode voltage. The difference in peak current phase between a 0 V anode and 1000 V is $0.35° \pm 0.06°$, which corresponds to a small difference in time of flight.

Anode Voltage (V)	Peak Current Phase (°)
0	-0.55 ± 0.04
250	-0.48 ± 0.05
500	-0.37 ± 0.05
750	-0.32 ± 0.05
1000	-0.20 ± 0.04

TABLE 5: Peak current phase for each anode voltage.

4.1.3 BEAM DEFLECTION VERSUS ANODE VOLTAGE

Because of the vertical asymmetry in the cathode geometry, it is expected that biasing the anode will cause a small vertical deflection of the beam. To test this, the transverse beam position was monitored using BPM #1 and #2 while raising the anode voltage between 0 V and 1000 V in 250 V increments. The anode bias was held for approximately one minute at each voltage to compute the average and standard deviation in beam position. Figure 29 shows the relative beam position at both BPM pairs as a function of anode voltage. The error bars correspond to the standard deviation in beam position. Each data set was fit with a linear trendline to better see the deflection with increasing anode bias.

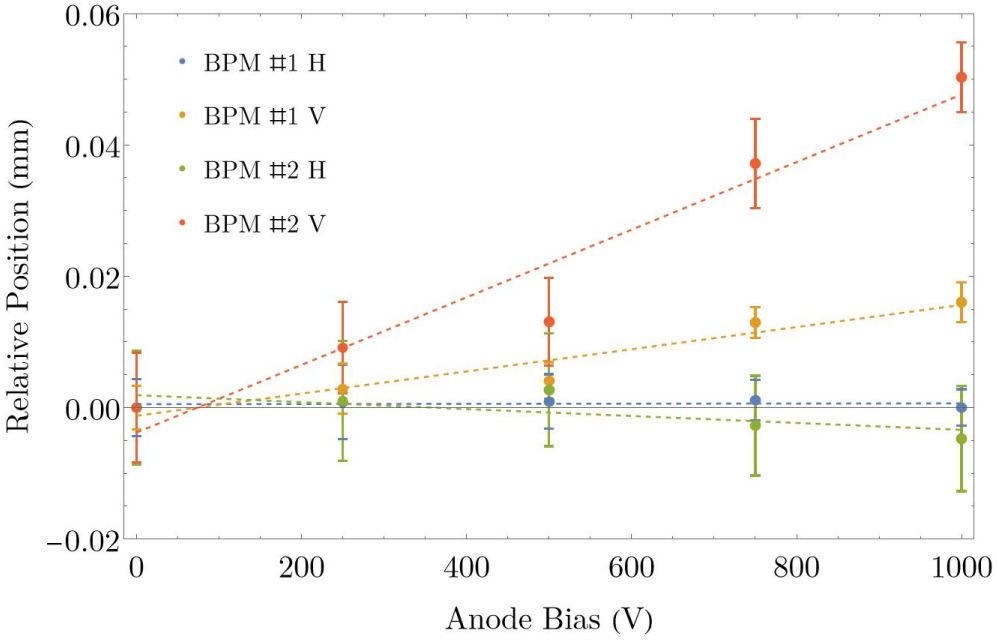

FIG. 29: Beam position vs. anode voltage as measured by each BPM pair. The error bars denote the standard deviation in measured beam position for each anode bias. The dashed lines are linear trendlines for each BPM data set.

It is expected that when biasing the anode from 0 V to 1000 V, the beam center would be

displaced by 103 ± 78 µm on average at BPM #2 (see section 5.6.1). Comparing the beam position at BPM #2 for 1000 V and 0 V, the average horizontal displacement is -4.71 ± 8.01 µm and the average vertical displacement is 50.3 ± 5.3 µm. The average total displacement is 50.5 ± 5.4 µm, which is consistent with the expected displacement. This displacement is small and can be easily compensated for using steering coils #1 and #2.

4.2 EXPERIMENTAL PARAMETERS

Table 6 shows relevant experimental parameters specific to each of the three run periods. During each run period, CEBAF delivered electron beams using 780 nm lasers. The cathode voltage remained at -130 kV for all three run periods.

Run Period	1	2	3
Dates	June 2019 - Sept 2019	Jan 2020 - Mar 2020	July 2020 - Sept 2020
Lasers/Halls used	A, B, C	A, B, C, D	A, B, C, D
Total charge extracted	245 C	428 C	254 C
Avg toal beam current	50 - 100 µA	150 - 225 µA	150 - 225 µA

TABLE 6: Experimental parameters for each run period.

4.3 BEAM CURRENT, VACUUM, ION PRODUCTION RATE

The total beam current and photo-gun vacuum levels were measured throughout each run period. Figure 30 shows plots of the beam current, cumulative charge extracted from the photocathode, and photo-gun vacuum during each run period. The higher beam currents in run periods 2 and 3 are due to operating all four experimental halls. During run periods 2 and 3, the vacuum pressure in the photo-gun stayed between $1 - 2 \times 10^{-12}$ Torr. The vacuum levels steadily rose from 2×10^{-12} Torr to 8×10^{-12} Torr during the course of run period 1 due to beam loss at downstream apertures.

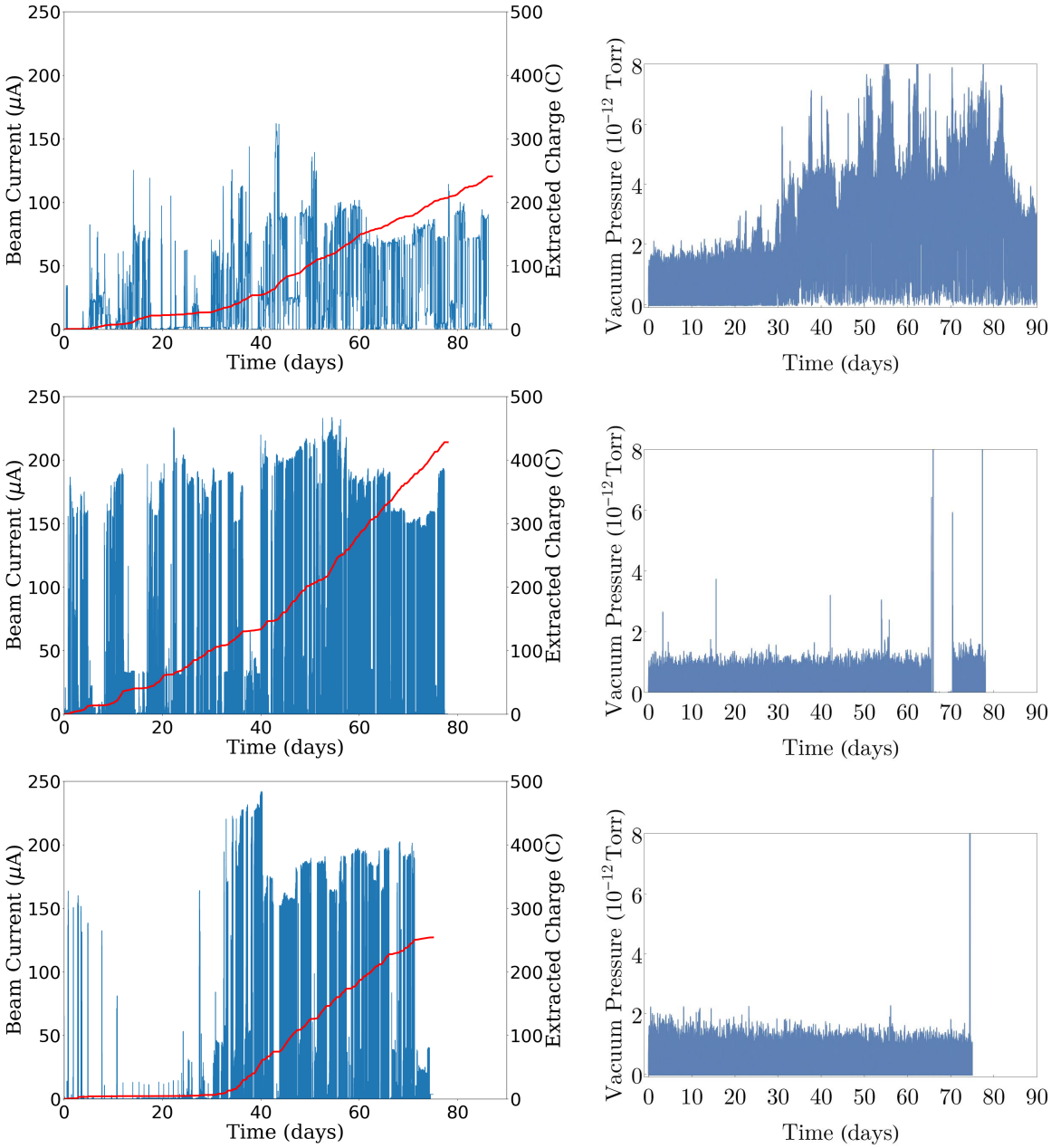

FIG. 30: Plots of the beam current (left, blue line), cumulative charge extracted from the photocathode (left, red line), and photo-gun vacuum (right, blue line) for run periods 1 (top), 2 (center), and 3 (bottom).

Using the average vacuum levels, the relative partial pressures from Table 3, and the average beam current, the ion production rate per unit length of H_2, CO, CO_2, and CH_4

within the photo-gun were computed using Eq. 4. Table 7 shows calculations of the ion production rate per unit length for each run period. The uncertainty values for the average beam current and average partial pressures correspond to standard deviations, which are relatively large due to the high variation in beam current and pressure throughout each run period.

Gas Species	Run Period	Avg. Beam Current (μA)	Avg. Partial Pressure (pTorr)	Ion Production Rate per Unit Length (ions/(s·m))
H_2	1	57 ± 31	2.6 ± 1.4	1300 ± 900
	2	123 ± 70	0.67 ± 0.47	700 ± 630
	3	119 ± 72	0.72 ± 0.35	720 ± 570
CO	1	57 ± 31	0.047 ± 0.024	98 ± 74
	2	123 ± 70	0.012 ± 0.008	54 ± 48
	3	119 ± 72	0.013 ± 0.006	56 ± 43
CO_2	1	57 ± 31	0.026 ± 0.014	100 ± 75
	2	123 ± 70	0.0067 ± 0.0047	55 ± 49
	3	119 ± 72	0.0072 ± 0.0035	57 ± 44
CH_4	1	57 ± 31	0.086 ± 0.045	220 ± 160
	2	123 ± 70	0.022 ± 0.015	120 ± 100
	3	119 ± 72	0.024 ± 0.012	120 ± 95

TABLE 7: Estimates of the average partial pressure and ion production rate per unit length of H_2, CO, CO_2 and CH_4 in the photo-gun vacuum chamber. The uncertainty values in the beam current and partial pressures correspond to one standard deviation in the respective measurements.

4.4 QE MEASUREMENT AND CHARGE LIFETIME RESULTS

Figures 31-33 show QE measurements for each laser used in each run period as a function of the cumulative charge extracted from the photocathode. The charge lifetimes for each anode bias in each run period (referred to as "anode bias regions") were calculated by fitting the QE measurements with an exponential decay function of the form:

$$\mathrm{QE} = \mathrm{QE}_0 e^{-\frac{t}{\tau}} \tag{10}$$

where QE_0 is the initial QE and τ is the charge lifetime – the amount of charge extracted before the QE falls to $1/e$ of its initial value. Table 8 shows charge lifetime values for each anode bias region in each run period. To quantify the improvement in charge lifetime between anode bias regions, Table 9 shows improvement ratios between charge lifetimes for biased and grounded anode configurations for the first two run periods.

Using the charge lifetimes from all run periods for laser A, the average charge lifetime for a grounded anode configuration is 191 ± 40 C and the average charge lifetime for a biased anode configuration is 343 ± 66 C. Thus, the average charge lifetime improvement is 1.80 ± 0.51.

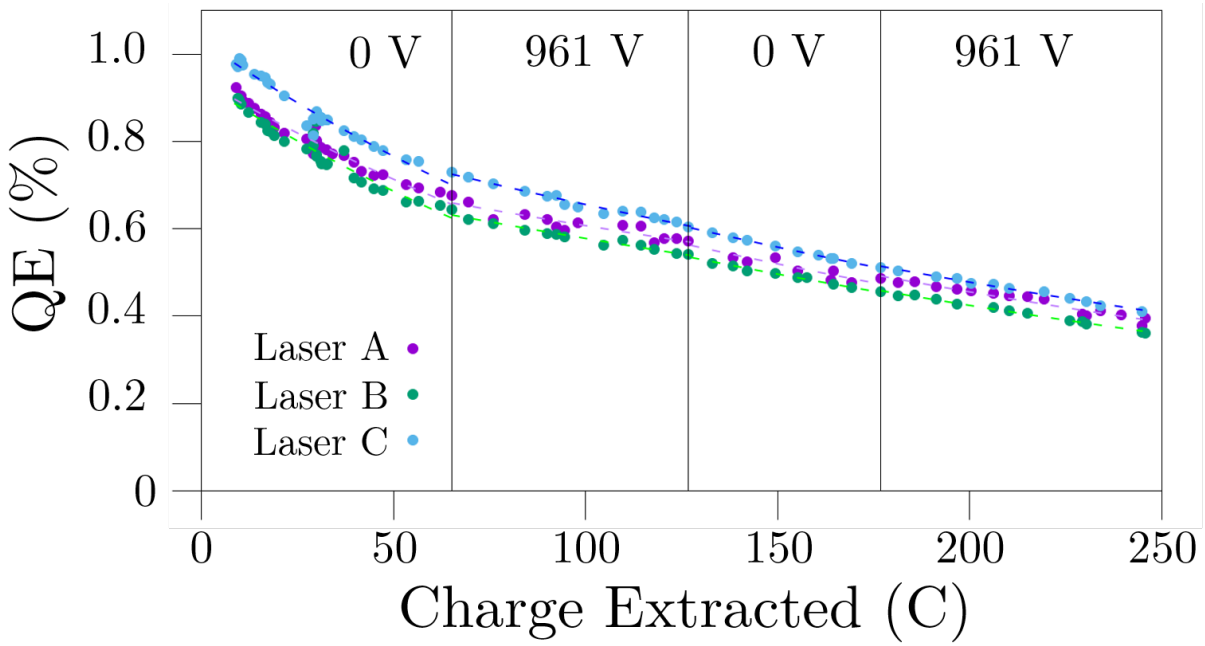

FIG. 31: Plot of the QE measurements for run period 1 as a function of the cumulative charge extracted. The dashed lines represent the exponential fits to calculate the charge lifetime for each anode bias.

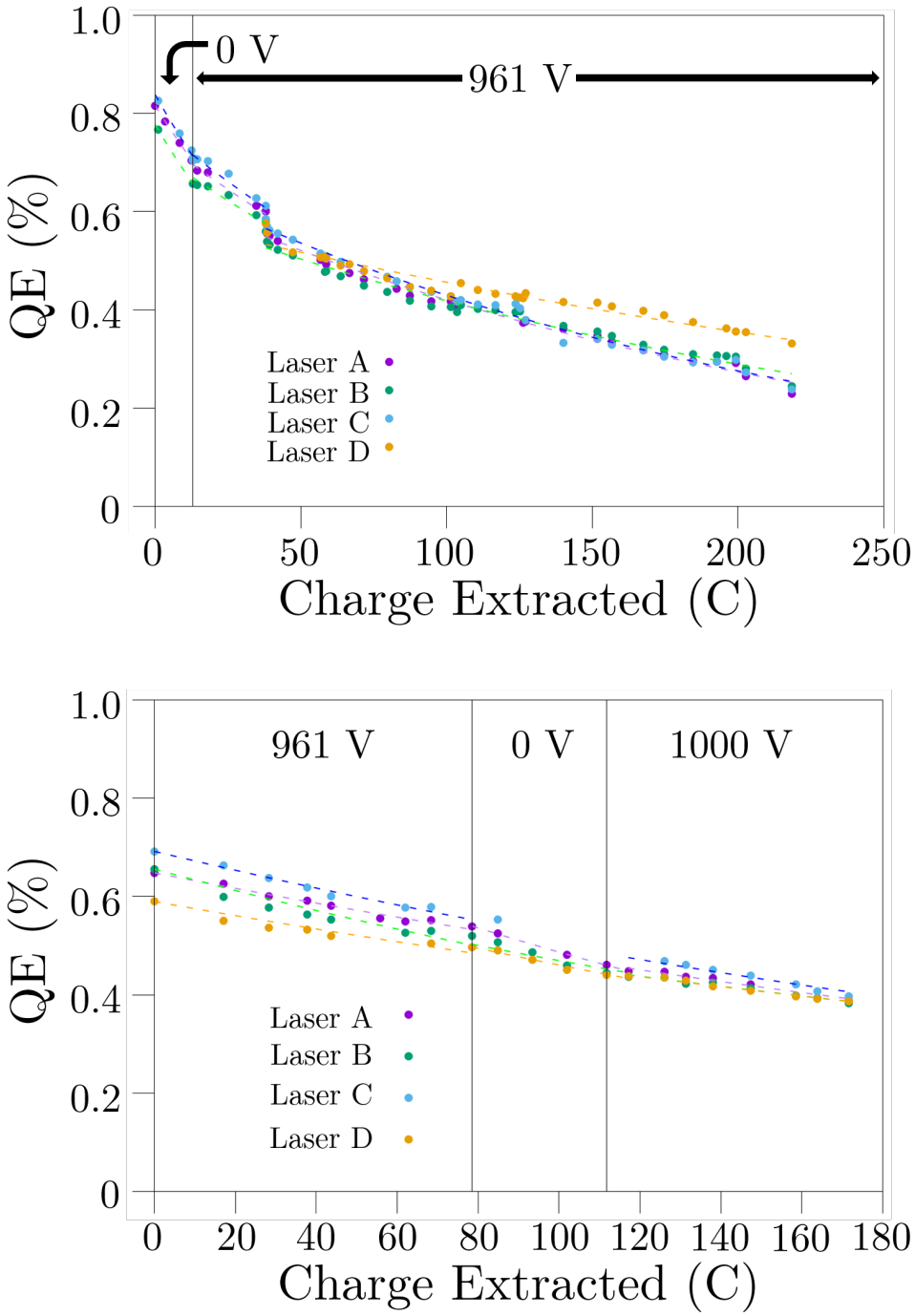

FIG. 32: Plots of the QE measurements for run period 2 as a function of the cumulative charge extracted for laser spot #1 (top) and laser spot #2 (bottom). The dashed lines represent the exponential fits to calculate the charge lifetime for each anode bias. Because the QE at the initial laser spot was too low during run period 2, the laser spot was shifted to an area on the photocathode with higher QE.

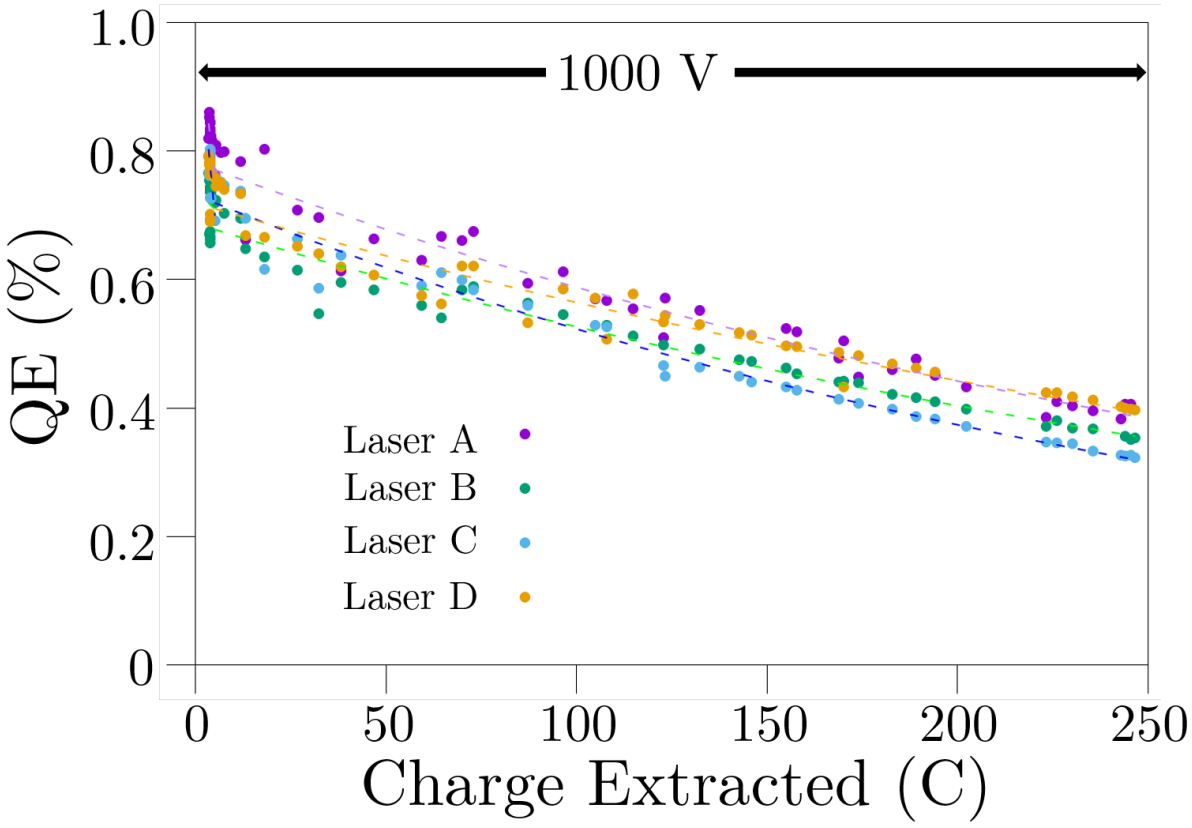

FIG. 33: Plot of the QE measurements for run period 3 as a function of the cumulative charge extracted. The dashed lines represent the exponential fits to calculate the charge lifetime.

Run Period	Anode Bias (V)	Anode Bias Region (C)	Charge Lifetime (C)			
			Laser A	Laser B	Laser C	Laser D
1	0	0-65	181 ± 8	161 ± 9	169 ± 7	N/A
	961	65-127	424 ± 53	392 ± 23	346 ± 16	N/A
	0	127-177	288 ± 39	304 ± 14	297 ± 5	N/A
	961	177-245	303 ± 18	302 ± 9	314 ± 11	N/A
2	0	0-13	85.8 ± 0.1	77.1*	80 ± 6	**
	961	13-219	196 ± 19	218 ± 25	189 ± 17	398 ± 19
	961	219-298	401 ± 10	291 ± 7	350 ± 11	402 ± 28
	0	298-331	208 ± 4	317 ± 19	**	279 ± 23
	1000	331-391	370 ± 29	416 ± 27	344 ± 52	421 ± 23
3	1000	0-247	350 ± 14	376 ± 14	298 ± 9	412 ± 17

TABLE 8: Charge lifetime values for each laser used in each run period. The uncertainty values in the charge lifetime values correspond to fit errors. * indicates there were insufficient QE measurements to get an uncertainty estimate. ** indicates the QE measurements were of insufficient quality to calculate the charge lifetime.

Run Period	Improvement Ratio	Ratio value			
		Laser A	Laser B	Laser C	Laser D
1	τ_2/τ_1	2.34 ± 0.31	2.43 ± 0.20	2.05 ± 0.13	N/A
	τ_4/τ_3	1.05 ± 0.16	0.993 ± 0.054	1.06 ± 0.04	N/A
2	τ_2/τ_1	2.28 ± 0.22	N/A	2.36 ± 0.28	N/A
	τ_3/τ_4	1.93 ± 0.06	0.918 ± 0.059	N/A	1.44 ± 0.16
	τ_5/τ_4	1.78 ± 0.14	1.31 ± 0.12	N/A	1.51 ± 0.15

TABLE 9: Improvement ratios between charge lifetimes τ_i for biased anode regions and grounded anode regions in run periods 1 and 2.

4.5 QE SCAN RESULTS

Figures 34-36 show calibrated QE scans for each run period. In run periods 1 and 3, QE scans were taken before and after the respective run periods and were taken once every two to three weeks in run period 2. QE difference scans were made using successive scans in each run period. These were made with a different color scheme to distinguish them from the QE scans. In each QE scan and difference scan, the red circle denotes the 5 mm active area and the green circle denotes the location of the 2 mm diameter laser spot. In run period 2, the second laser spot is denoted by a cyan circle. Table 10 shows the parameters for each QE scan in each run period, namely the laser spot position, the time elapsed, and the charge extracted since the first QE scan. Note that the QE scans in each run period are *not* coincident with a change in anode bias configuration, so the QE difference scans may not correspond to a single anode bias.

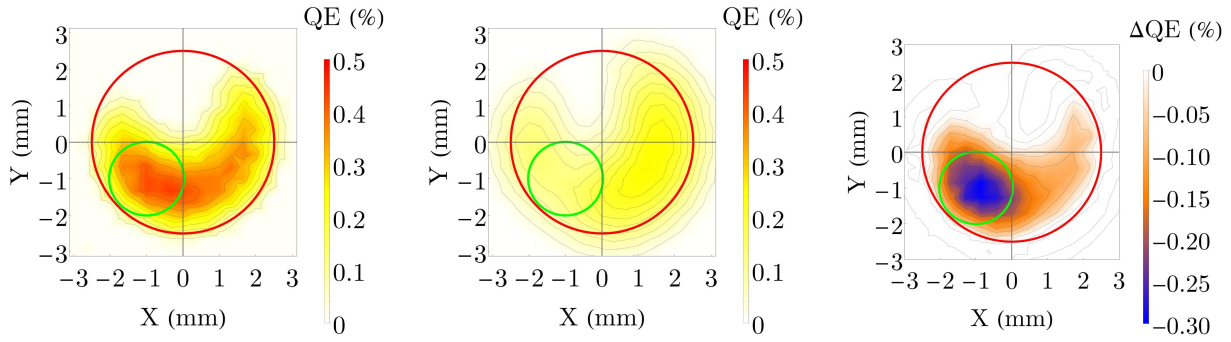

FIG. 34: QE scans taken before (left) and after (center) run period 1 and the corresponding QE difference scan (right).

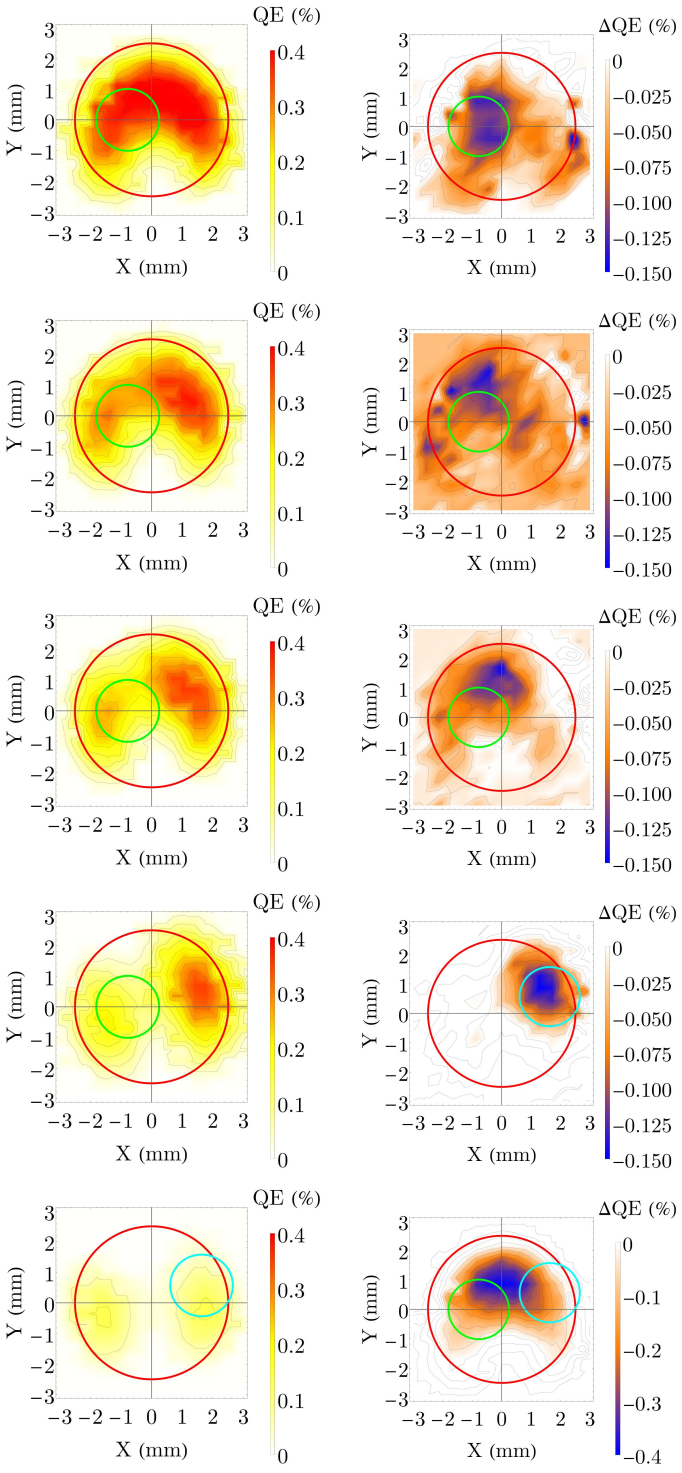

FIG. 35: QE scans (left) and QE difference scans between successive QE scans (right) for run period 2. The last QE difference scan (right, furthest down) is the difference between the first and last QE scan.

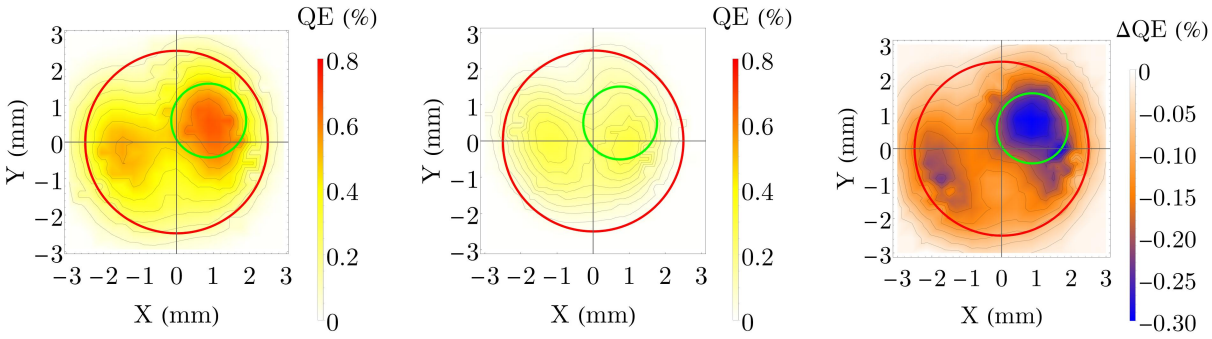

FIG. 36: QE scans taken before (left) and after (center) run period 3 and the corresponding QE difference scan (right).

Run Period	QE Scan #	Laser Spot (mm)	Time Since 1st QE Scan (days)	Charge Extracted Since 1st QE Scan (C)
1	1	(-1.0, -1.0)	0	0
	2	(-1.0, -1.0)	111	245.7
2	1	(-0.78, 0.00)	0	0
	2	(-0.78, 0.00)	10.7	13.0
	3	(-0.78, 0.00)	17.7	37.9
	4	(-0.78, 0.00)*	52.8	193
	5	(1.63, 0.57)*	80.8	404
3	1	(0.88, 0.59)	0	0
	2	(0.88, 0.59)	81	241

TABLE 10: Table of relevant parameters for all QE scans. *The laser spot shifted between QE scans 4 and 5 in run period 2. Between these two scans, the laser spot remained at (-0.78, 0.00) mm for the first 40 C, then was shifted to (1.63, 0.57) mm for the remaining 171 C.

CHAPTER 5

GPT SIMULATIONS

5.1 OVERVIEW

GPT simulations were made to help explain the experimental results. Each run period was partitioned so that simulations involve either a grounded or biased anode configuration. In each simulation, an electron bunch is tracked from the photocathode to viewer #1, located 1.54 m away from the photocathode. The total charge of the bunch is set at 10^{-4} times the total charge extracted during a given anode bias region, which reduces the number of ions produced in the simulation so that it can run in a reasonable amount of time. The number of ions can be scaled later to represent the actual number of ions produced because the ion production rate is proportional to the bunch charge (see Eq. 1). The bunch is then tracked over timesteps chosen by a fifth-order Runge-Kutta algorithm with adaptive stepsize control using the Cash-Karp Method to maintain the user-defined accuracy while minimizing computation time.

Using a C++ custom element, the electron bunch ionizes residual gas molecules along its trajectory. Only H_2 gas is ionized in the simulations, as it is the predominant residual gas species. Table 11 shows simulation parameters common to all simulations and Table 12 shows parameters specific to simulations for the respective experimental conditions.

The main goals of these simulations are:

1. Track ions that hit the photocathode and note where they originate, where they hit the photocathode, and with how much kinetic energy,

2. Compare ion density distributions for grounded and biased anode configurations to compare with QE scan measurements, and

3. Calculate ratios of the number of back-bombarding ions between grounded and biased anode configurations to compare with the charge lifetime improvement ratios in Table 9.

Simulation Parameter	Value
Cathode Voltage	-130 kV
Transverse Bunch Diameter (4σ)	2 mm
Bunch Length	50 ps (FWHM)
H_2 Density	$3.56 \times 10^{10}\,\mathrm{m}^{-3}$
Total Simulation Time	0.1 ms

TABLE 11: Global parameters for all simulations.

Run Period	Simulation	Anode Voltage (V)	Bunch Charge (mC)	Laser Position (mm)
1	1	0	6.51	(-1.00, -1.00)
	2	961	6.15	(-1.00, -1.00)
	3	0	5.02	(-1.00, -1.00)
	4	961	6.89	(-1.00, -1.00)
2	1	0	1.30	(-0.78, 0.00)
	2	961	23.79	(-0.78, 0.00)
	3	961	7.85	(1.63, 0.57)
	4	0	3.32	(1.63, 0.57)
	5	1000	6.00	(1.63, 0.57)
3	1	1000	24.14	(0.88, 0.59)

TABLE 12: Simulation-specific parameters for each run period.

5.2 SIMULATION SETUP

Figure 37 shows a layout of the electric and magnetic field maps used in the GPT simulations. By default, a coordinate system is used such that particles initially start in the $z = 0$ plane (photocathode plane) and travel in the $+z$-direction. The origin is defined to be at the center of the photocathode and the $+z$-axis is along the center of the photo-gun exit

beamline. All beamline elements are represented in the simulation using 2D and 3D field maps, either user-defined or via built-in GPT functions. The simulation region extends from the photocathode to viewer #1 and is bounded by the inner surface of the beam pipe, which is 6 cm in diameter. Any particle that hits the beam pipe, anode, photocathode, or viewer, is removed from the simulation.

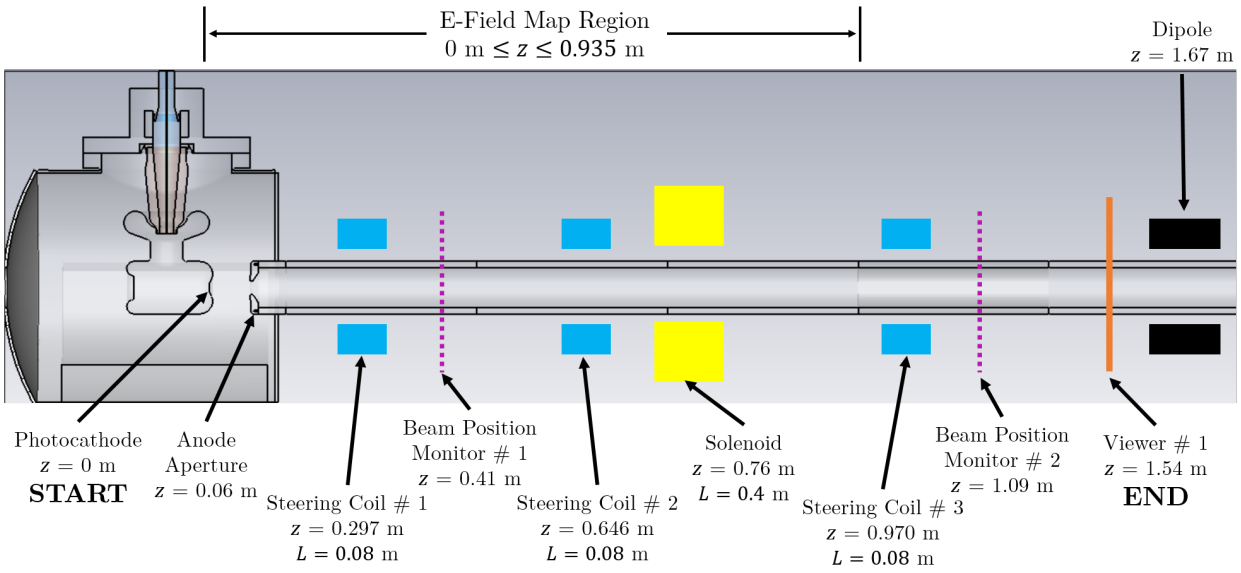

FIG. 37: Cross section of the initial CEBAF Injector beamline denoting the locations z of the beamline elements and the field map lengths L of steering coils and solenoid (not to scale).

5.2.1 BEAM PARAMETERS

The parameters of the electron bunch were chosen to reflect the real electron bunch parameters used in the experiments. Because it is impractical to simulate quadrillions of electrons, the bunch is instead made up of 10^4 macro-particles, each representing a fixed number of electrons such that the total charge of all macro-particles equals the specified bunch charge. The transverse profile of the bunch is Gaussian with $\sigma = 0.5\,\text{mm}$ and a cutoff

at 4σ, matching the 4σ laser diameter of 2 mm. To set the longitudinal distribution, one can set either longitudinal (z) or time (t) coordinates. Because the simulation is setup such that particles are removed if they are behind the photocathode, the latter option of setting the temporal distribution was used. All electrons start at $z = 0$ with negative t-values and are only placed in the simulation when $t = 0$. The distribution of t-values is Gaussian with FWHM = 50 ps, matching the CEBAF bunch length [80]. Figure 38 shows the transverse and temporal distributions of the electron bunch.

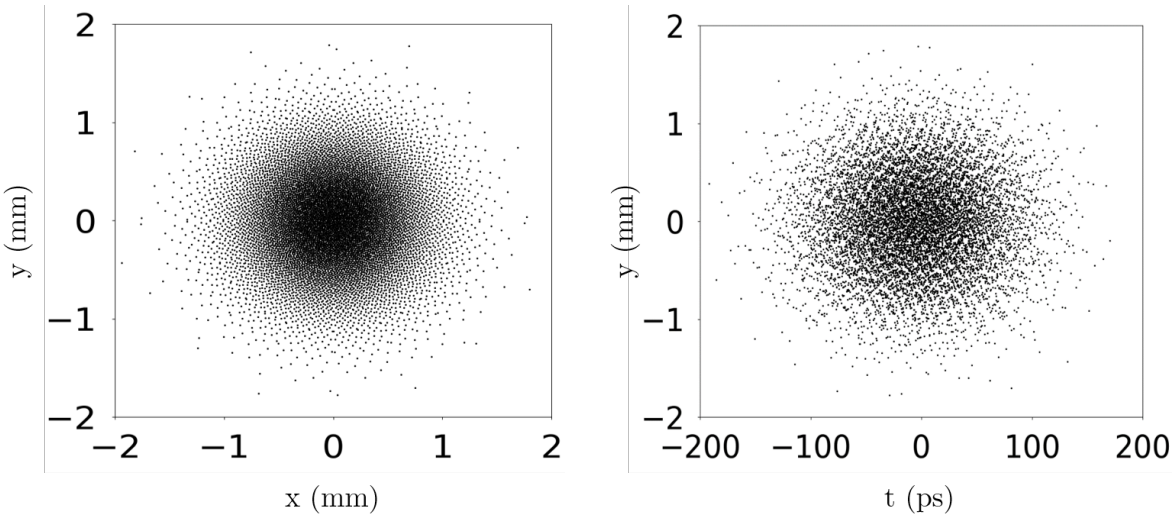

FIG. 38: Plots of the transverse distribution (left) and temporal distribution (right) of the electron bunch.

5.3 GPT ALGORITHM

The motion of a macro-particle i in a simulation with charge q_i and moving with velocity v_i is governed by the Lorentz force:

$$\mathbf{F}_i = q_i \left(\mathbf{E} + \mathbf{v}_i \times \mathbf{B}\right) \tag{11}$$

where $\mathbf{E}$ and $\mathbf{B}$ represent the total electric and magnetic fields experienced by the particle, whether due to beamline elements or due to other charged particles (neglecting radiation and retardation effects). The coordinates of the particle can be represented by a vector $\mathbf{y}(t)$ containing position and momentum coordinates of the particle:

$$\mathbf{y}(t) = \langle \mathbf{x}(t), \mathbf{p}(t) \rangle = \langle x(t), y(t), z(t), p_x(t), p_y(t), p_z(t) \rangle \tag{12}$$

The time-dependence of the position and momentum coordinates can then be described using the relativistic equations of motion:

$$\frac{d\mathbf{p}}{dt} = \mathbf{F} \tag{13}$$

$$\frac{d\mathbf{x}}{dt} = \frac{\mathbf{p}c}{\sqrt{\mathbf{p}^2 + m_i^2 c^2}} \tag{14}$$

Since $\mathbf{y}(t)$ contains the position and momentum of the particle, Eqs. 13 and 14 can be combined into a single differential equation:

$$\begin{aligned} \frac{d\mathbf{y}(t)}{dt} &= f(t, y(t)) \\ y(t_0) &= y_0 \\ y(t_1) &= y_1 \\ &\vdots \\ y(t_i) &= y_i \end{aligned} \tag{15}$$

where the initial conditions are given by the user-defined initial position and momentum distributions.

5.3.1 RUNGE-KUTTA AND THE CASH-KARP METHOD

Due to the complexity of any non-trivial particle simulation, especially ones involving space charge, Eq. 15 often cannot be solved analytically. Thus, it is solved numerically in GPT using the fifth-order Runge-Kutta method. In this method, the coordinates at time $t + h$ are calculated using Eq. 16 given the coordinates $y(t)$ at time t. The constants a_i, b_{ij}, and c_i are given by Cash and Karp in the Butcher tableau in Table 13 [81]. The accuracy for

a given timestep h is calculated by comparing $\mathbf{y}(t+h)$ to the fourth-order solution $\mathbf{y}^*(t+h)$, whose constants c_i^* are given in the last column of Table 13. If the error is greater than the user-defined accuracy in the input file, the timestep is failed and retried with a smaller timestep. In this way, the timesteps h are adapted to the changing complexity of a simulation while maintaining accuracy.

$$
\begin{aligned}
k_1 &= hf\left(t, \mathbf{y}\right) \\
k_2 &= hf\left(t + a_2 h, \mathbf{y} + b_{21}\mathbf{k}_1\right) \\
k_3 &= hf\left(t + a_3 h, \mathbf{y} + b_{31}\mathbf{k}_2 + b_{32}\mathbf{k}_2\right) \\
k_4 &= hf\left(t + a_4 h, \mathbf{y} + b_{41}\mathbf{k}_3 + b_{42}\mathbf{k}_3 + b_{43}\mathbf{k}_3\right) \\
k_5 &= hf\left(t + a_5 h, \mathbf{y} + b_{51}\mathbf{k}_4 + b_{52}\mathbf{k}_4 + b_{53}\mathbf{k}_4 + b_{54}\mathbf{k}_4\right) \\
k_6 &= hf\left(t + a_6 h, \mathbf{y} + b_{61}\mathbf{k}_5 + b_{62}\mathbf{k}_5 + b_{63}\mathbf{k}_5 + b_{64}\mathbf{k}_5 + b_{65}\mathbf{k}_5\right) \\
\mathbf{y}(t+h) &= \mathbf{y}(t) + c_1 k_1 + c_2 k_2 + c_3 k_3 + c_4 k_4 + c_5 k_5 + c_6 k_6 + \mathcal{O}\left(h^6\right) \\
\mathbf{y}^*(t+h) &= \mathbf{y}(t) + c_1^* k_1 + c_2^* k_2 + c_3^* k_3 + c_4^* k_4 + c_5^* k_5 + c_6^* k_6 + \mathcal{O}\left(h^5\right)
\end{aligned}
\tag{16}
$$

i	a_i	b_{i1}	b_{i2}	b_{i3}	b_{i4}	b_{i5}	c_i	c_i^*
1	0						$\frac{37}{378}$	$\frac{2825}{27648}$
2	$\frac{1}{5}$	$\frac{1}{5}$					0	0
3	$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$				$\frac{250}{621}$	$\frac{18575}{48384}$
4	$\frac{3}{5}$	$\frac{3}{10}$	$-\frac{9}{10}$	$\frac{6}{5}$			$\frac{125}{594}$	$\frac{13525}{55296}$
5	1	$-\frac{11}{54}$	$\frac{5}{2}$	$-\frac{70}{27}$	$\frac{35}{27}$		0	$\frac{277}{14336}$
6	$\frac{7}{8}$	$\frac{1631}{55296}$	$\frac{175}{512}$	$\frac{575}{13824}$	$\frac{44275}{110592}$	$\frac{253}{4096}$	$\frac{512}{1771}$	$\frac{1}{4}$

TABLE 13: Cash-Karp constants for a_i, b_{ij}, c_i, and c_i^*.

5.4 IONIZATION CUSTOM ELEMENT ALGORITHM

Electron impact ionization is not a built-in function in GPT. However, one can create custom elements within the framework of GPT to extend its functionality. A C++ custom element was developed and implemented to simulate electron impact ionization of gas molecules in real-time using Monte-Carlo routines. The ion, secondary electron, and scattered electron are then tracked using relativistic kinematics. The custom element algorithm is described in the following subsections. The custom element code itself is listed and its technical details are described in Appendix 7.

5.4.1 INPUT PARAMETERS AND ELIGIBILITY

In a given ionization simulation, the custom element is called in the GPT input file with the following parameters:

1. The coordinate system and boundaries of the ionization region, defined by the center and length of the region. For example, one can set the ionization region to be centered at $z = 1\,\mathrm{m}$ with length $1\,\mathrm{m}$, meaning the ionization region exists within $0.5\,\mathrm{m} \leq z \leq 1.5\,\mathrm{m}$ with no radial limit.

2. The gas species and density in cubic meters. The gas density can be constant or defined in file containing 1-D density data. Presently supported target gas particles are H_2, He, CO, and CH_4.

3. Secondary electron production - this can be turned on or off. If turned off, the custom element will still calculate the kinematic parameters of secondary electrons *as if* they were present for the purposes of calculating the kinematic parameters of other particles. (See section 2.4).

4. The subshell from which the secondary electron originates. By default, the subshell with the lowest ionization energy is chosen.

5. Option to output ionization parameters to a separate output file.

During each simulation timestep[3], the custom element loops through all particles present in the simulation. A particle is eligible to ionize if:

[3]It is important to note here that "simulation timestep" refers to the timestep h in the Runge-Kutta algorithm, *not* the timesteps written to the output file.

1. The particle is an electron or electron macro-particle.

2. The particle is in the coordinate system of the ionization region.

3. The particle is not removed (using one of the GPT particle removal elements) during in this timestep.

4. The midpoint of the particle's *displacement* vector in this timestep (using coordinates from $\mathbf{y}(t)$ and $\mathbf{y}(t + h)$) is within the ionization region. This condition allows for ionization of neutral target gas particles that exit and enter the ionization region within a timestep.

5. The kinetic energy of the particle at the end of the timestep is greater than the ionization (binding) energy of the gas molecule and the ionization cross section is positive.

If any of these conditions is false, the particle is ignored and the custom element continues to the next macro-particle.

5.4.2 MONTE CARLO ALGORITHM FOR ION PRODUCTION

The number of ions each eligible macro-particle creates in a timestep is calculated using Eq. 1. In the unlikely event that N_{ion} is larger than unity for any macro-particle, the timestep is retried with a smaller timestep until $0 \leq N_{ion} \leq 1$ for all electron macro-particles. N_{ion} is then interpreted as the probability that the particle will ionize during the timestep. The custom element then uses a Monte Carlo algorithm: if N_{ion} is less than a random number between 0 and 1, the particle ionizes and an ion is placed in a random location along the electron's trajectory during the timestep. Figure 39 shows a flow chart depicting the steps used in the Monte Carlo algorithm.

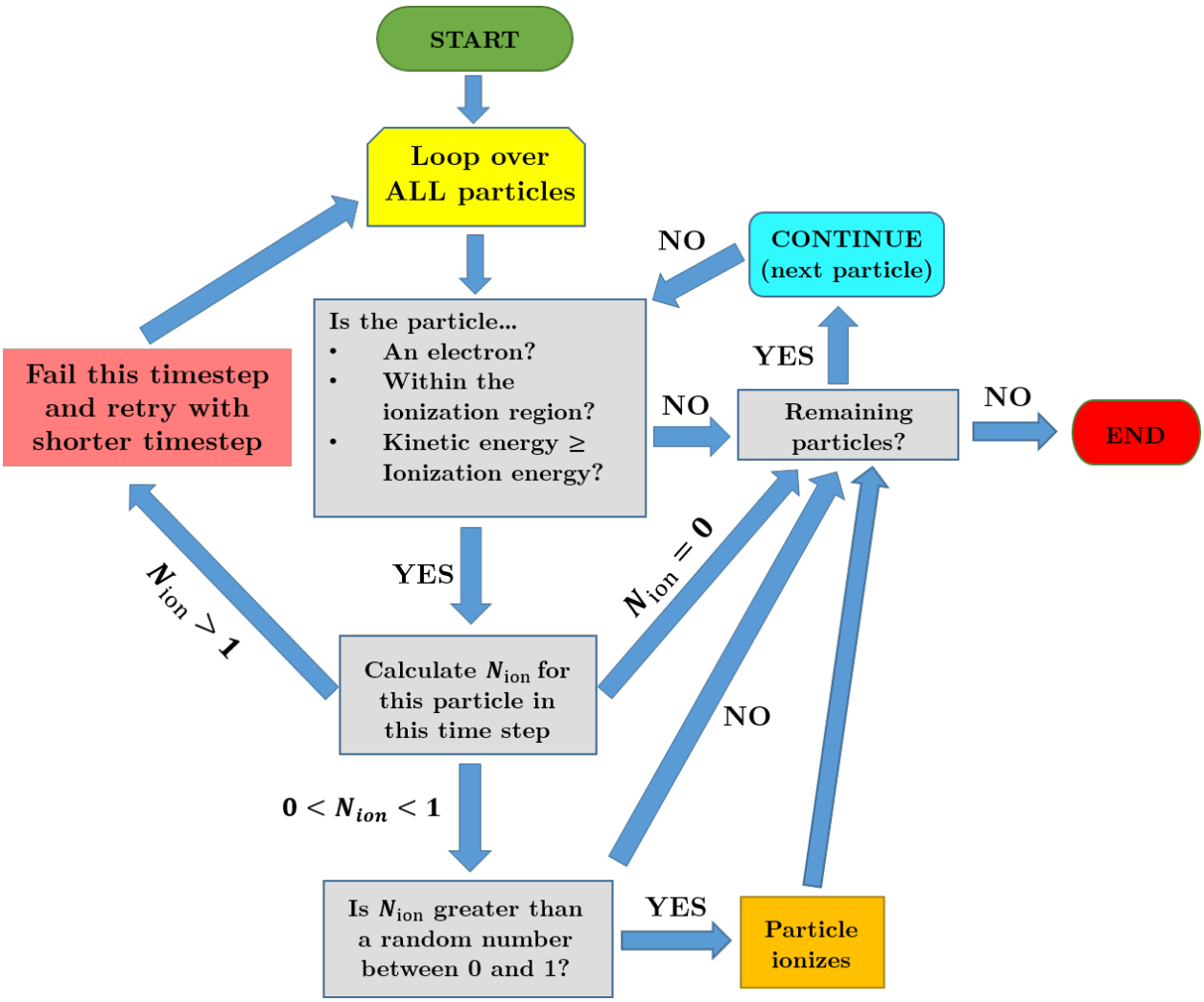

FIG. 39: Monte Carlo algorithm for ion production.

5.4.3 MONTE CARLO ALGORITHM FOR SECONDARY ELECTRON ENERGIES AND ION SPEEDS

To ensure a probabilistic selection of secondary electron and ion speeds, a Monte Carlo algorithm (different from the ionization algorithm) maps a random number between 0 and 1 onto a unique energy value such that the distributions of all energies/speeds are consistent with the theoretical distributions. The distribution of secondary electron energies is given by Eqs. 5 and 6 and the distribution of ion speeds is given by Eq. 43. These distributions

are *non-uniform*, as some energies/speeds are more likely than others. The random number generator produces a *uniform* distribution of numbers between 0 and 1. To map this uniform distribution to the non-uniform energy distributions, the Monte Carlo routines use the following algorithm.

First, the energy distributions must be integrated. Because the integral of the distribution functions cannot be expressed in closed-form, they are integrated numerically using the trapezoidal rule:

$$F\left(E_N\right) = \frac{1}{2}\sum_{i=1}^{N}\left(E_i - E_{i-1}\right)\left(f(E_i) + f(E_{i-1})\right) \tag{17}$$

where N is the number of integration steps and E is the energy or speed. The higher the number of steps N, the more accurate the integral is. However, a higher N requires more CPU time to compute the integral. This added CPU time can be substantial, especially for simulations with a very large number of macro-particles.

Let $g(E)$ be the ratio between the partial and total integrals of the distribution:

$$g(E) = \frac{F(E)}{F(E_{max})} \tag{18}$$

The values of $g(E)$ are strictly between 0 and 1 in steps of E_{max}/N. Thus, one can choose a random number between 0 and 1 and select the value of E for which $g(E)$ is closest to the random number. Because successive values of $g(E_N)$ are strictly *increasing*, the Monte Carlo routine checks if the random number is less than $g((i+1)E_i/N)$ for successive integers i, where $0 < i < (N-1)$. If so, then the energy/speed is assigned the value $(i+1)E_i/N$. If not, then the routine continues to the next integer i. In this way, the Monte Carlo routine does not have to calculate all values of $g(E)$ prior to choosing the random number, saving CPU time and memory usage.

To show that this Monte Carlo algorithm reproduces the integral of the distribution functions, consider the linear, quadratic, and cubic functions. In this case, the integrals $F(E)$ and the ratios $g(E)$ have closed-form solutions, which are listed in Table 14. Figures 40 show histograms of linear, quadratic, and cubic distributions using the MC algorithm for $N = 10^1, 10^2, 10^3$ and 10^4 assuming a precision of 500 steps and $E_{max} = 1000$. The deviation from the theoretical distribution decreases with higher numbers of integration steps, as expected.

Distribution	$f(x)$	$F(E)$	$g(E)$
Linear	x	$\frac{1}{2}E^2$	$10^{-6}E^2$
Quadratic	x^2	$\frac{1}{3}E^3$	$10^{-9}E^3$
Cubic	x^3	$\frac{1}{4}E^4$	$10^{-12}E^4$

TABLE 14: $f(x)$, $F(E)$ and $g(E)$, for linear, quadratic, and cubic distributions assuming $E_{max} = 1000$.

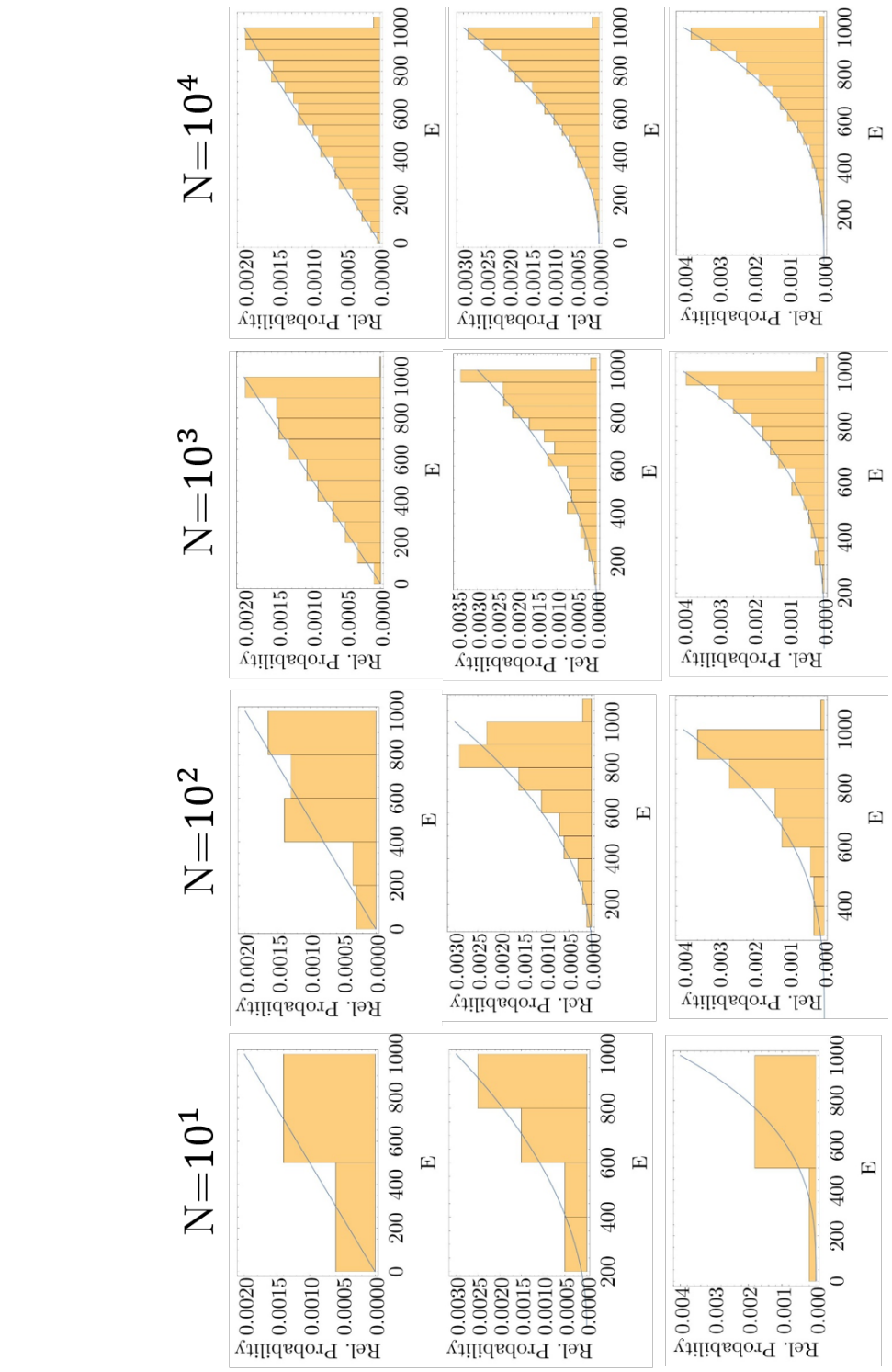

FIG. 40: Histograms of linear, quadratic, and cubic distributions for $N = 10^1, 10^2, 10^3$ and 10^4. The graph of each distribution function is overlaid for comparison.

5.4.4 THE SCATTERED ELECTRON AND KINEMATIC PARAMETERS

Given the energy and momenta of the primary electron, the secondary electron, the gas molecule, and the ion, the energy and momentum of the scattered electron can be calculated from energy and momentum conservation laws. From the mass m and kinetic energy T, the magnitude of the momentum $|\vec{p}|$ of a particle can be calculated as:

$$|\vec{p}| = mc\sqrt{\gamma^2 - 1} \tag{19}$$
$$\gamma = \frac{T}{mc^2} + 1$$

Table 15 summarizes the kinematic parameters of all particles involved in ionization. Unless otherwise known or derived, the direction of each particle is assigned to be random. It should also be noted that gas molecules are not explicitly present in the simulation. Rather, they are assumed to exist solely for calculation purposes.

Particle	Kinematic Parameter	Symbol	Calculated/Determined From
Primary Electron	Energy	T_e	Known/Given
	Momentum Magnitude	$\|\vec{p}_e\|$	
	Momentum Direction	$\hat{p}_e$	
Ion	Energy	T_{ion}	Eq. 7
	Momentum Magnitude	$\|\vec{p}_{ion}\|$	Eq. 19
	Momentum Direction	$\hat{p}_{ion}$	Random
Secondary Electron	Energy	W	MC routine (Eq. 5 or 6)
	Momentum Magnitude	$\|\vec{p}_{sec}\|$	Eq. 19
	Momentum Direction	$\hat{p}_{sec}$	Random
Target Gas Particle	Energy	T_{gas}	38 meV
	Momentum Magnitude	$\|\vec{p}_{gas}\|$	Eq. 19
	Momentum Direction	$\hat{p}_{gas}$	Random
Scattered Electron	Energy	T_{scat}	Energy Conservation
	Momentum Magnitude	$\|\vec{p}_{scat}\|$	Eq. 19
	Momentum Direction	$\hat{p}_{scat}$	Momentum Conservation

TABLE 15: Table of kinematic parameters for each particle involved in ionization.

5.4.5 CUSTOM ELEMENT BENCHMARKING

To ensure accuracy, the custom element was benchmarked against theoretical calculations and an established particle tracking code IBSimu [82], which includes ion production. To benchmark the Monte Carlo algorithm for ion production, the following GPT simulation was used: an electron bunch with a 0.5 mm rms transverse bunch size, 50 ps FWHM bunch length, travels along the z-axis between $z = 0$ m and $z = 0.5$ m. The bunch has a kinetic energy of 1 keV, since more ions are generated at 1 keV than the CEBAF beam energy of 130 keV. The bunch is composed of 10^4 electron macro-particles with a total charge of 10 µC of charge distributed uniformly with space charge forces neglected. The bunch ionizes H_2 gas, which has a constant pressure of 10^{-12} Torr. Figure 41 shows a snapshot of the GPT simulation after the primary electron bunch travels 400 mm.

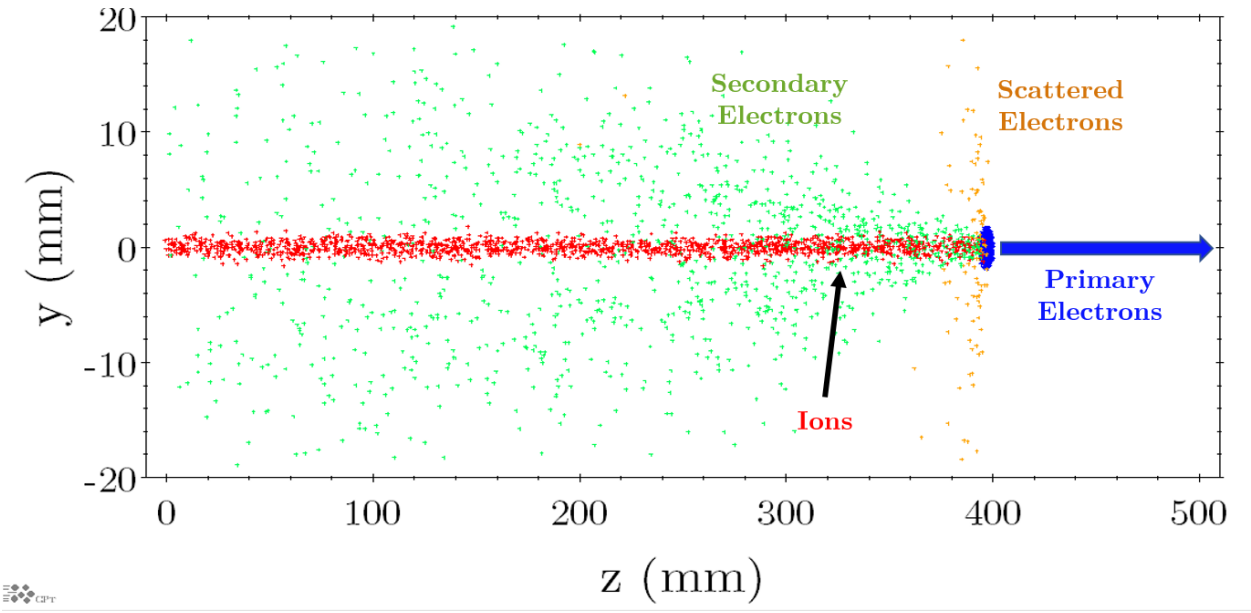

FIG. 41: Snapshot of a GPT simulation after the primary electron bunch travels 400 mm.

The GPT simulation was repeated 50 times each for H_2, CO, and CH_4 gas, assuming the

same gas density for each gas species. Each time the simulation was repeated, a different seed value for the random number generator was used in the Monte-Carlo algorithm. Table 16 tallies the average number of ions produced in each simulation and compares it with the predicted number of ions based on Eq. 1. The average number of ions produced for each gas species agrees with the theoretical number of ions to within 1.5%.

Gas Species	H_2	CO	CH_4
I (eV)	15.4	14.0	14.3
σ (10^{-21} m^2)	2.0	7.0	8.7
Theoretical # Ions	2028	6971	8770
Avg. Sim. # Ions	2002	6864	8638
Std. Dev.	48	92	105
% Diff.	1.3	1.5	1.5

TABLE 16: GPT benchmarking simulation results.

To benchmark the secondary electron energy algorithm, histograms of the energies of all secondary electrons produced in the simulations were created and compared with the differential cross section curves (Eqs. 5 and 6). Figure 42 shows the histogram of secondary electron energies for the CH_4 simulations with the corresponding differential cross section curve overlaid. The histogram was normalized such that the heights of each bar represent the relative distribution of secondary electron energies. While the shape of the normalized histogram follows the (normalized) differential cross section curve, the slight discrepancy is partly due to finite precision in the calculation of the secondary electron energies.

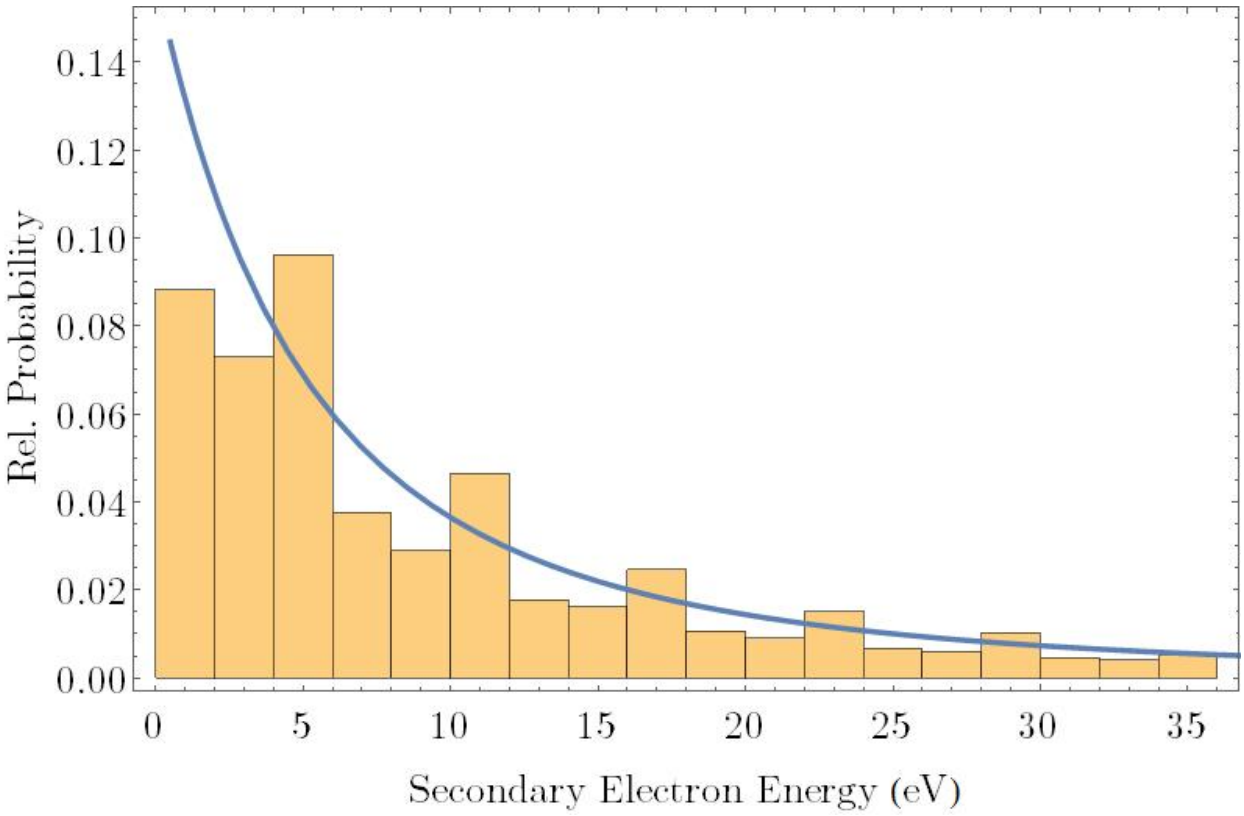

FIG. 42: Normalized histogram of secondary electron energies from all 50 CH_4 simulations with the normalized differential cross section curve overlaid.

The distribution of ion energies was benchmarked in a similar way. Figure 43 compares the normalized CH_4^+ ion speed histogram to the corresponding Maxwellian distribution curve from Eq. 7. The shape of the histogram closely resembles the Maxwellian distribution curve, indicating the accuracy of the Monte Carlo routines.

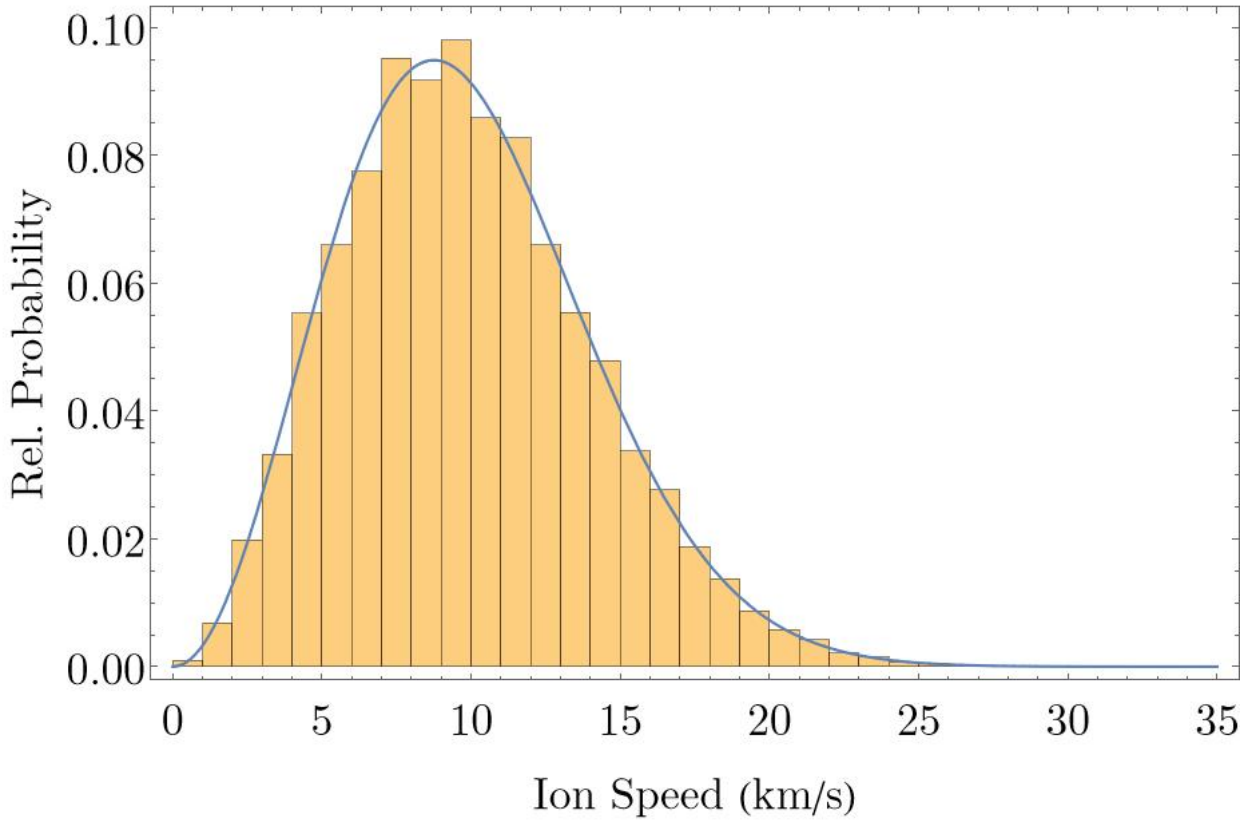

FIG. 43: Normalized histogram of ion energies from all 50 CH_4 simulations with the Maxwellian distribution overlaid.

To benchmark the custom element against IBSimu, the following simulation was performed in both codes: an electron bunch containing 10^4 electrons with uniform kinetic energy T travels along the z-axis through 1 m of H_2 gas. The gas density was tuned such that, on average, the electron bunch ionizes 100 H_2 gas molecules for $T = 1\,\mathrm{keV}$, $T = 130\,\mathrm{keV}$. Using Eq. 1, the required gas densities are $\rho\,(1\,\mathrm{keV}) \approx 4.97 \times 10^{18}\,\mathrm{m}^{-3}$ and $\rho\,(130\,\mathrm{keV}) \approx 2.60 \times 10^{20}\,\mathrm{m}^{-3}$. Both codes repeated the simulation 20 times for both kinetic energies with different seed values used in each simulation. The simulation results for GPT and IBSimu are summarized in Table 17. The number of ions produced in both codes are within 1.1% of the predicted number of ions, again demonstrating the accuracy of the GPT custom element and its consistency with established codes.

Simulation Code	GPT	IBSimu
Avg. Sim. Ions (1 keV)	98.9	101
% Diff (1 keV)	1.1	1.0
Avg. Sim. Ions (130 keV)	99.95	100.8
% Diff (130 keV)	0.05	0.8

TABLE 17: Comparison of GPT and IBSimu simulation results. The percent difference values compare the average number of ions produced in each simulations to the predicted 100 ions.

5.5 SIMULATION RESULTS

In order to determine where back-bombarding ions originate, each simulation used two virtual screens: one placed at the photocathode ($z = 0$) and one placed at the peak anode potential ($z = 0.08\,\mathrm{m}$). During each simulation, the parameters of all particles that hit or pass through either screen was recorded in the output. The screen data was then processed using a Python script. The script first removed any particle that was not an ion. Then the data was separated into two data sets: ions originating upstream and ions originating downstream of the peak anode potential. Each simulation particle had a unique ID. The ID of ions originating upstream of the anode were recorded at the photocathode screen, but not at the peak anode potential screen, whereas ions originating downstream of the anode were recorded in both screens.

Three types of plots of back-bombarding ion distributions were made for each simulation:

1. Ion Origin Plot: Scatter plot color-coded by whether the ion originated upstream (blue) or downstream (red) of the peak anode potential when biased.

2. Density Plot: 2-D histogram color-coded by ion density.

3. Energy Density Plot: 2-D histogram color-coded by ion density and weighted by kinetic energy.

Figures 44-46 show simulation results for each run period. The plots for run periods 1 and 2 were also combined to see the total back-bombarding ion distributions throughout each run period. To quantify the reduction of back-bombarding ions with the anode biased, ratios of

the number of ions incident on the photocathode, active area, or the laser spot(s) between grounded and biased anode simulations were calculated. Because the bunch charges used in each simulation were not equal, the number of back-bombarding ions in each simulation was normalized to the bunch charge, as shown in Eqs. 20 and 21. The ratios were also weighted by the ion kinetic energies. Table 18 shows the calculated ratios for run periods 1 and 2.

$$\text{Unweighted Ion Ratio} = \frac{\sum N_{\text{ion,0V}}/Q_{\text{0V}}}{\sum N_{\text{ion,961V}}/Q_{\text{961V}}} \tag{20}$$

$$\text{Energy-Weighted Ion Ratio} = \frac{\sum N_{\text{ion,0V}} E_{\text{ion,0V}}/Q_{\text{0V}}}{\sum N_{\text{ion,961V}} E_{\text{ion,961V}}/Q_{\text{961V}}} \tag{21}$$

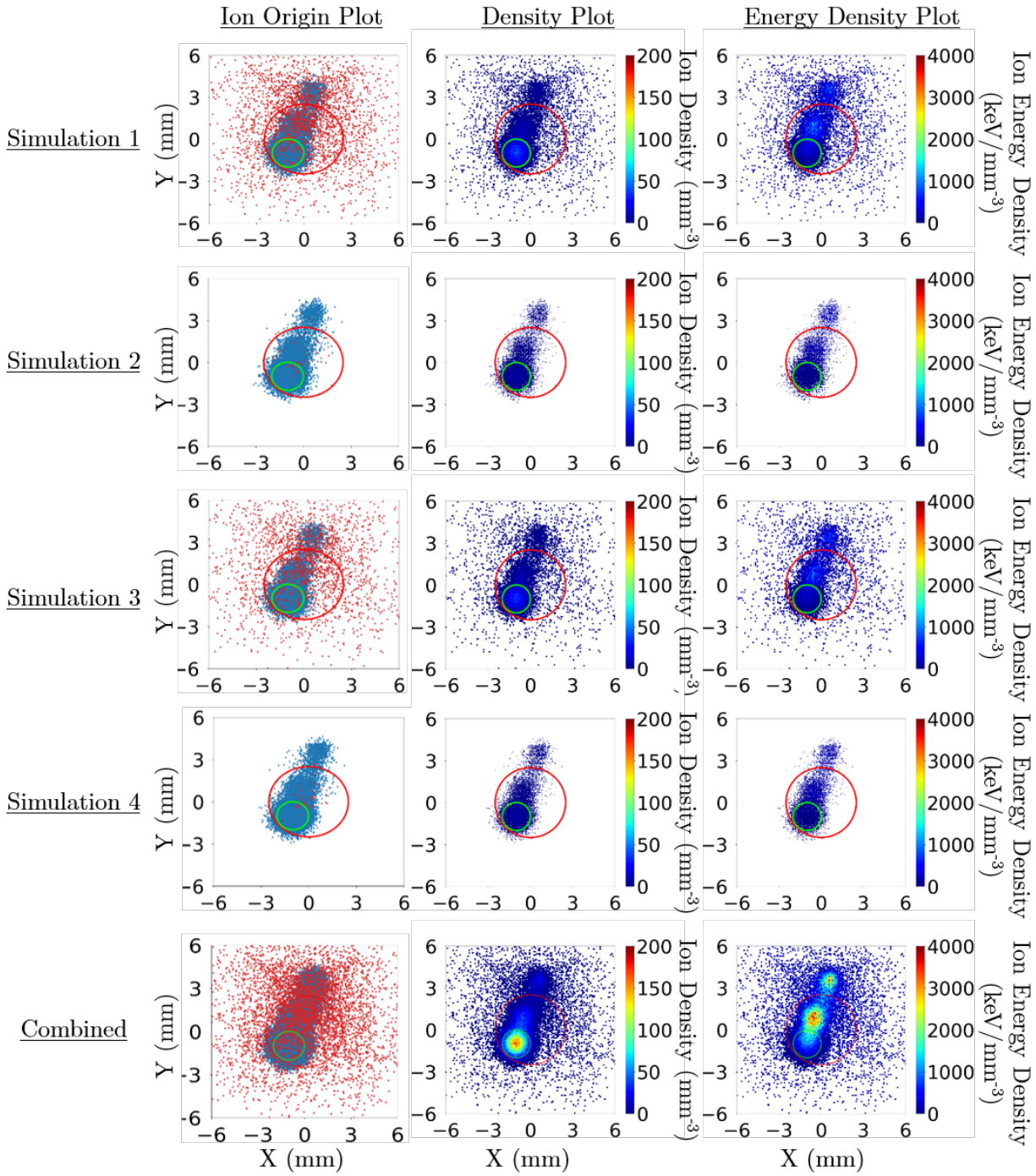

FIG. 44: Plots of the ion distribution at the photocathode for each simulation for run period 1. The red circle denotes the 5 mm diameter active area and the green circle denotes the 2 mm laser spot.

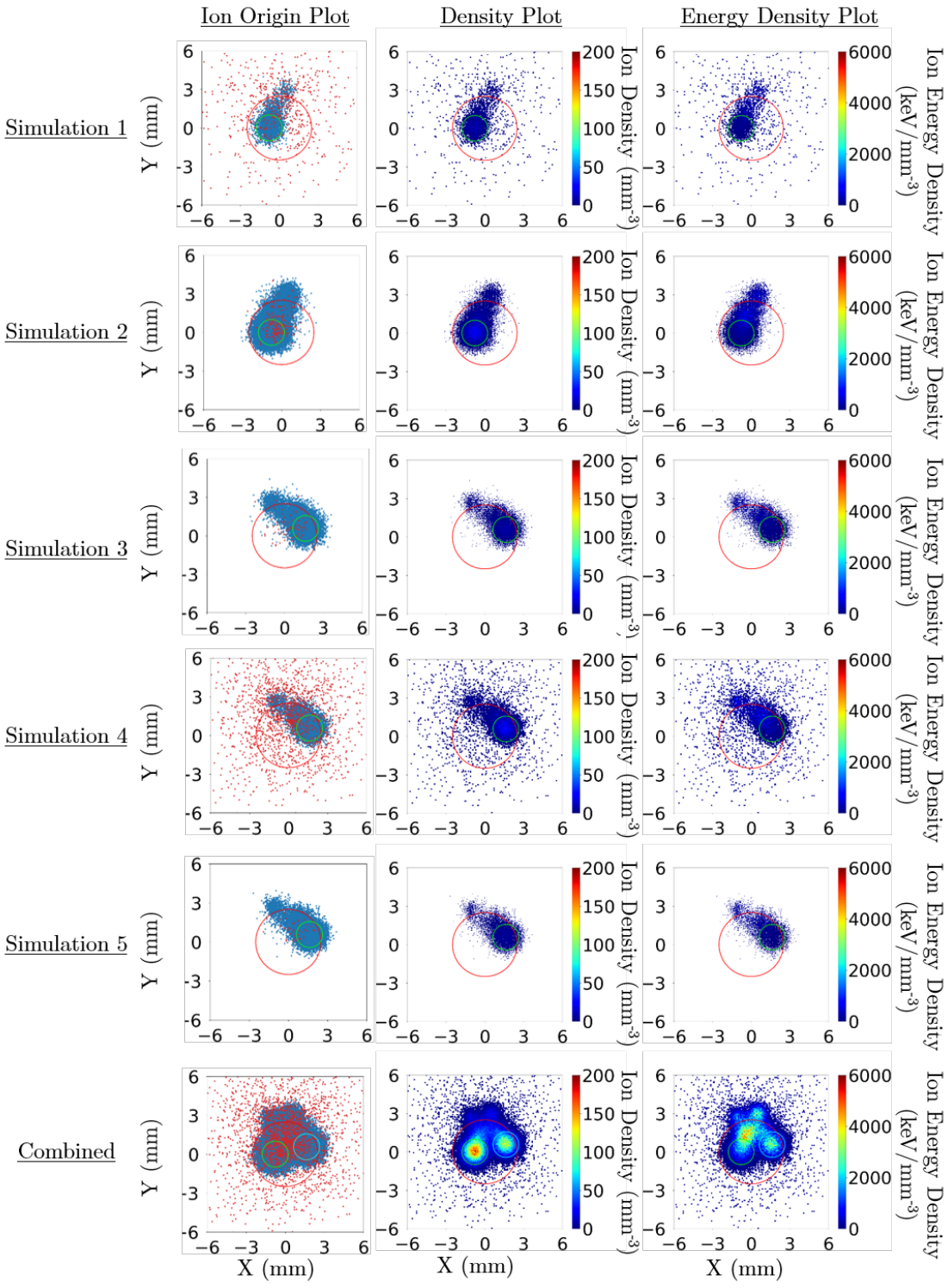

FIG. 45: Plots of the ion distribution at the photocathode for each simulation for run period 2. The red circle denotes the 5 mm diameter active area and the green circles denote the two 2 mm laser spots.

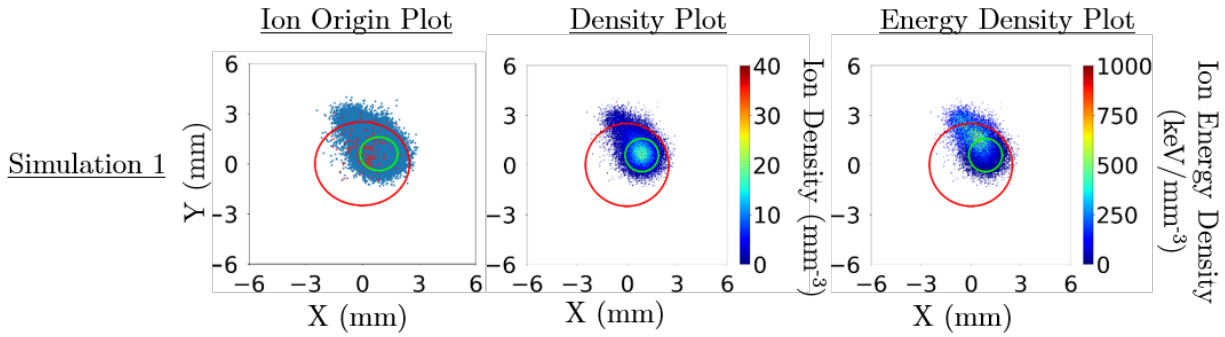

FIG. 46: Plots of the ion distribution at the photocathode for each simulation for run period 3. The red circle denotes the 5 mm diameter active area and the green circle denotes the 2 mm laser spot.

Run Period	Simulation #'s Used	Ion Region	Unweighted Ion Ratio	Energy-Weighted Ion Ratio
1	$\frac{1}{2}$	Photocathode	1.20	1.86
		Active Area	1.06	1.33
		Laser Spot	1.01	1.15
	$\frac{3}{4}$	Photocathode	1.20	1.86
		Active Area	1.06	1.31
		Laser Spot	1.00	1.14
2	$\frac{1}{2}$	Photocathode	1.22	1.88
		Active Area	1.08	1.37
		Laser Spot	1.02	1.15
	$\frac{4}{3}$	Photocathode	1.20	1.84
		Active Area	1.07	1.34
		Laser Spot	1.00	1.08
	$\frac{4}{5}$	Photocathode	1.21	1.87
		Active Area	1.07	1.34
		Laser Spot	1.00	1.10

TABLE 18: Ion ratios between grounded and biased anode simulations for run periods 1 and 2.

5.6 SYSTEMATIC STUDIES

GPT simulations were performed to study the effect of three possible uncertainties in the simulation results: the laser spot position, the relative active area position, and the gas species. For the laser spot position study, the active area remains centered at the origin while the laser spot is scanned across the active area. The purpose of this study is to determine the extent to which uncertainty in the laser spot position can affect the simulation results.

The relative active area position study repeats the laser spot position study, except that the *displacement* of the active area center from the laser spot center is kept fixed. That is, while the origin is assumed to be at the center of the photocathode, the active area is not, allowing the possibility that the active area was not aligned with the center photocathode in the experiments. Both laser spots used in run period 2 were used (denoted as the 1^{st} and 2^{nd} laser spot).

In both studies, the laser spot position was scanned in a 11×11 square grid of points centered with the photocathode with 0.5 mm separation. Simulations of a 1 mC electron bunch traveling through the CEBAF accelerator to viewer #1 were created for each laser spot using grounded and biased anode configurations. In each simulation, a uniform gas density of $10^{11}\,\text{m}^{-3}$ of either H_2, CO, or CH_4 gas was used.

Since each study entails running hundreds of simulations, each requiring its own analysis, batches of GPT simulations, as well as the individual analyses using the Python script, were run simultaneously. Density and energy density plots corresponding to each laser position were created for both grounded and biased anode configurations. Ion ratios between the respective grounded and biased anode simulations were also calculated for each laser spot position.

Plot grids containing density or energy density plots and tables containing ion ratios for each laser spot position were compiled efficiently using a Visual Basic Application (VBA) macro for Microsoft PowerPoint. The ion ratio tables contain ratios for all laser spot positions. The plot grids and ion ratio tables show how the ion distribution and ion ratios are affected by a change in laser position. The results of the two studies are described in the following subsections.

5.6.1 SYSTEMATIC STUDY SIMULATION SETUP

The initial conditions for the systematic study simulations, such as the simulation region, bunch parameters (with the exception of bunch charge), and field maps, were the same as

those for the biased anode experiment simulations. However, because the laser positions are different from the biased anode simulations, the strengths of the first two steering coil field maps were adjusted such that the beam passes straight through the solenoid for each laser spot position. To do this, the steering coil strengths were adjusted manually until the transverse coordinates of the bunch were within $x, y < 1\,\text{mm}$ and the transverse velocities were within $\beta_x, \beta_y < 0.001$ at the position of the solenoid ($z = 0.76\,\text{m}$).

Each simulation for each laser position was repeated with the biased anode configuration *without* adjusting the steering coil strengths to determine the effect of the biased anode configuration on the transverse beam position for comparison with the beam deflection test results in Ch. 4.1. The average beam displacement was $0.103\,\text{mm} \pm 0.078\,\text{mm}$, which is consistent with the beam deflection test results.

5.6.2 SYSTEMATIC STUDY RESULTS

Since the results of the systematic studies contain a substantial number plot grids and ion ratio tables, and the plot grids are too large to show in this thesis, a representative ion ratio table for H_2 ions incident on the photocathode in the laser spot position study is shown in Table 19. As another way to visualize the systematic study data, plots of the number of back-bombarding ions incident on the photocathode, active area, or laser spot, as a function of the laser spot position for both grounded and biased anode configurations are shown in Figs. 47-52. Each line is color coded by y-position: either 0 mm, ± 1 mm, or ± 2 mm. The active area is assumed to be at the center of the photocathode for purposes of these plots.

Laser X (mm) → Laser Y (mm) ↓	-2.5	-2.0	-1.5	-1.0	-0.5	0.0	0.5	1.0	1.5	2.0	2.5
2.5	1.22	1.20	1.21	1.23	1.20	1.19	1.21	1.20	1.23	1.22	1.21
2.0	1.19	1.24	1.24	1.24	1.19	1.21	1.21	1.22	1.20	1.23	1.21
1.5	1.21	1.21	1.20	1.23	1.23	1.22	1.22	1.22	1.23	1.20	1.22
1.0	1.22	1.24	1.23	1.22	1.22	1.23	1.22	1.25	1.20	1.22	1.23
0.5	1.23	1.19	1.24	1.22	1.23	1.21	1.20	1.23	1.23	1.20	1.23
0.0	1.23	1.20	1.23	1.21	1.21	1.22	1.22	1.21	1.22	1.18	1.20
-0.5	1.21	1.19	1.22	1.20	1.21	1.20	1.22	1.21	1.21	1.21	1.21
-1.0	1.21	1.20	1.25	1.19	1.22	1.22	1.21	1.23	1.21	1.22	1.22
-1.5	1.21	1.22	1.21	1.25	1.19	1.22	1.22	1.21	1.20	1.21	1.21
-2.0	1.21	1.25	1.19	1.20	1.22	1.21	1.20	1.20	1.19	1.22	1.20
-2.5	1.21	1.21	1.22	1.22	1.24	1.22	1.23	1.23	1.19	1.22	1.21

TABLE 19: Photocathode ion ratio table for H_2 gas in the laser spot position systematic study.

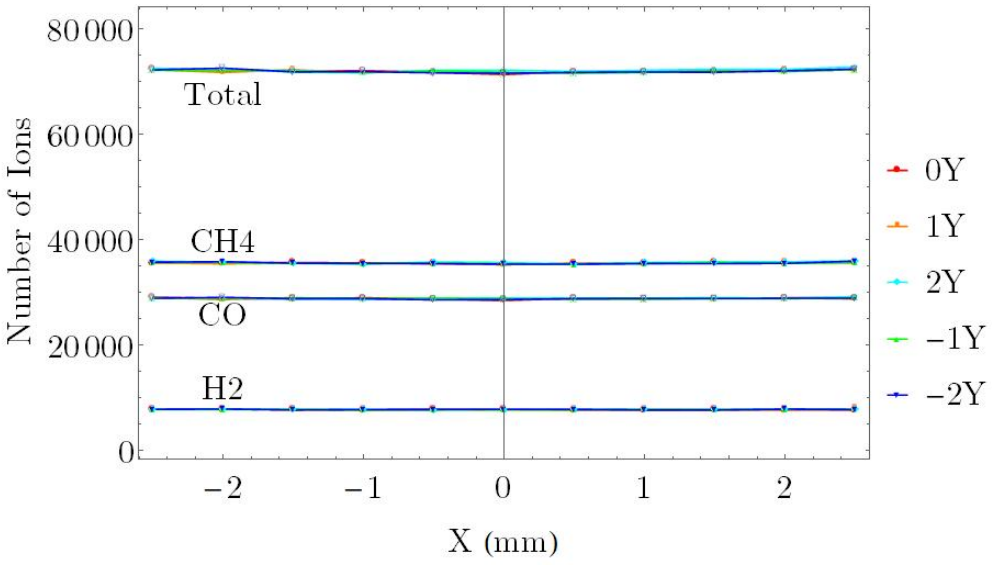

FIG. 47: Plot of number of back-bombarding ions on the photocathode vs laser spot position for a 0 V anode.

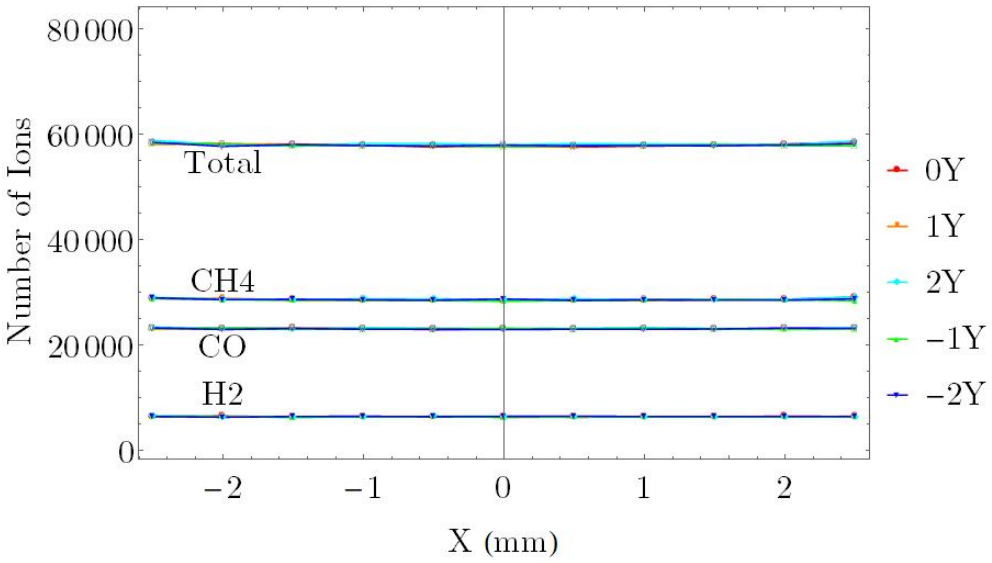

FIG. 48: Plot of number of back-bombarding ions on the photocathode vs laser spot position for a 961 V anode.

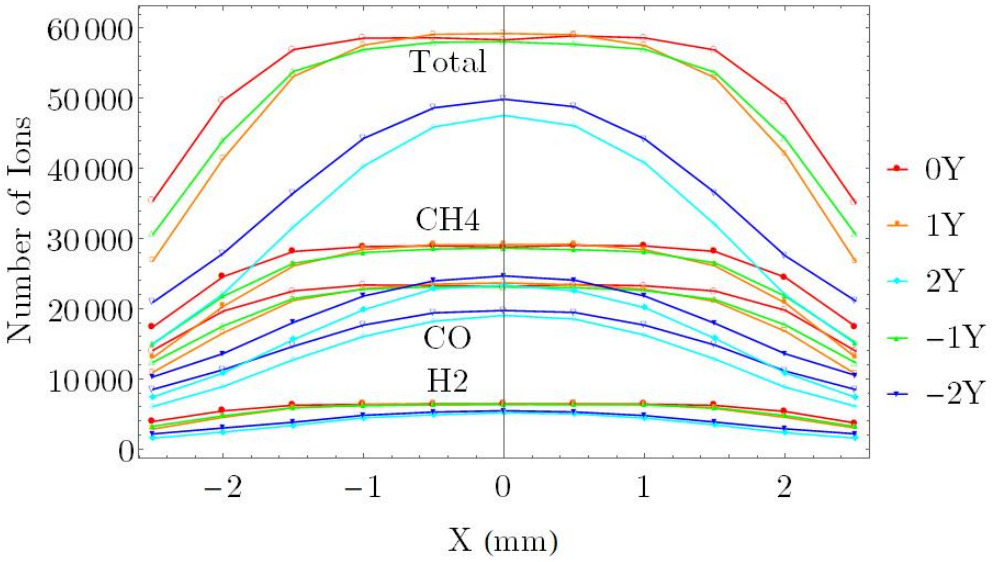

FIG. 49: Plot of number of back-bombarding ions on the active area vs laser spot position for a 0 V anode.

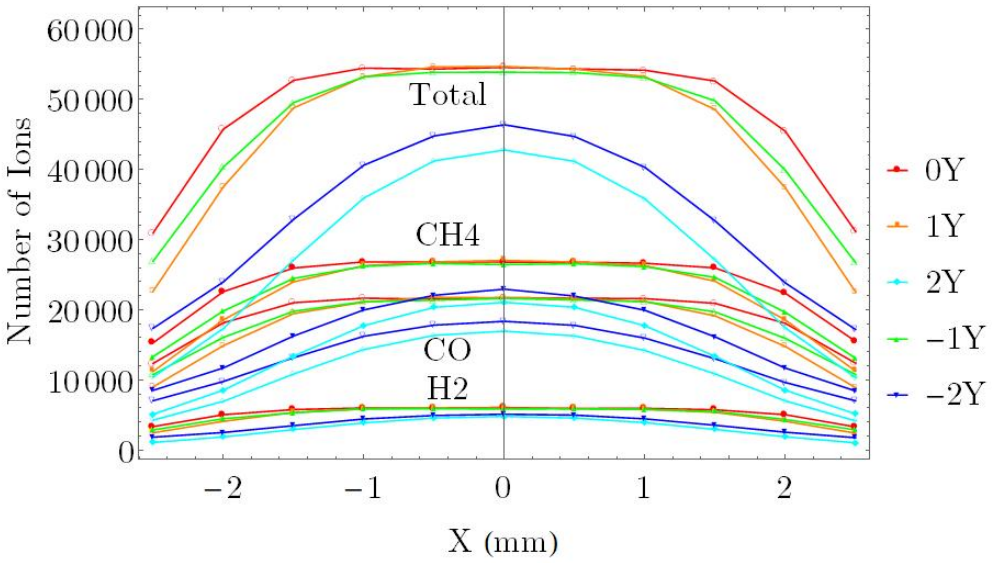

FIG. 50: Plot of number of back-bombarding ions on the active area vs laser spot position for a 961 V anode.

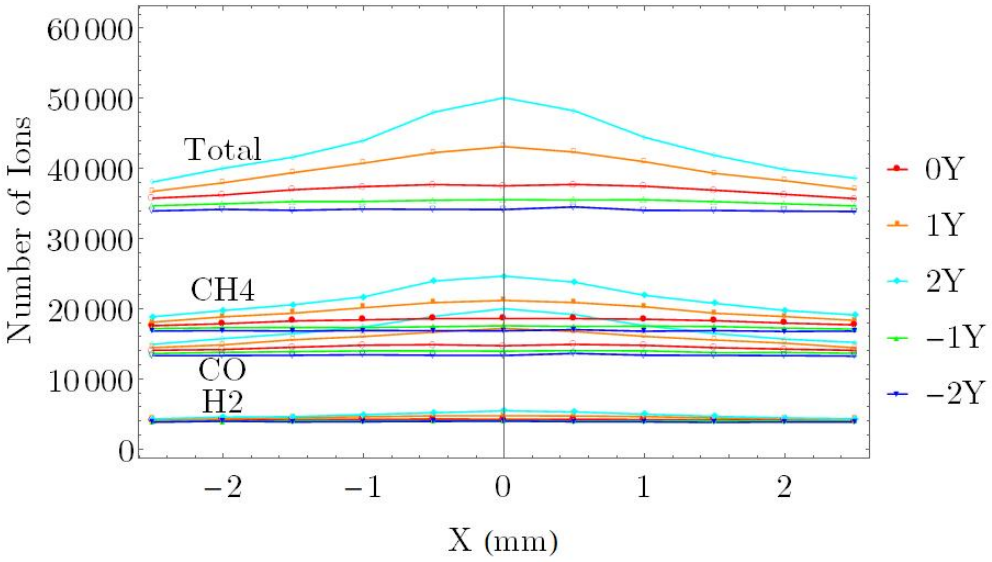

FIG. 51: Plot of number of back-bombarding ions on the laser spot vs laser spot position for a 0 V anode.

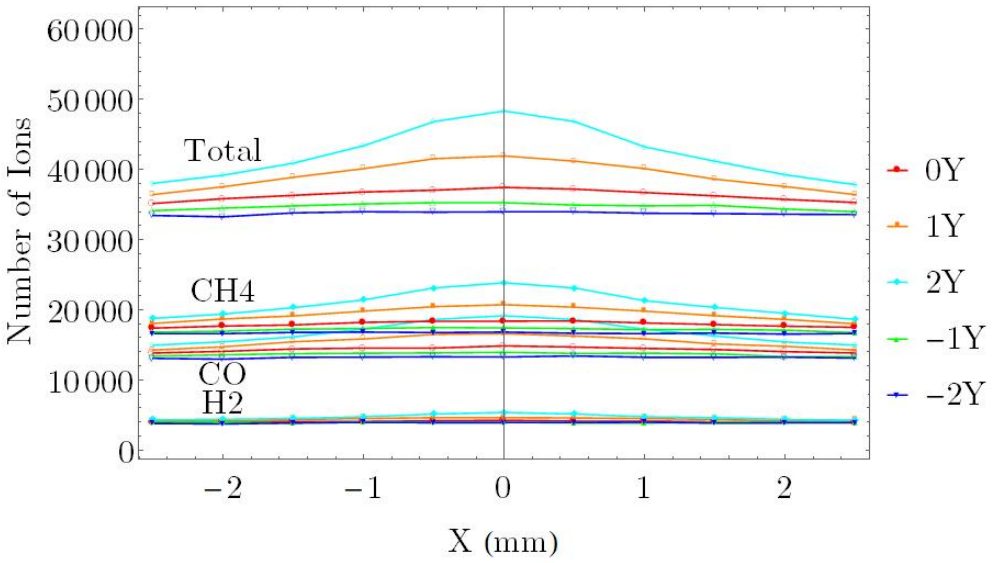

FIG. 52: Plot of number of back-bombarding ions on the laser spot vs laser spot position for a 961 V anode.

CHAPTER 6

DISCUSSION

Biasing the anode has proven to be an effective method to increase the charge lifetime of the CEBAF photo-gun. The configuration for the biased anode is straightforward to implement: the downstream flange of the photo-gun chamber was modified to allow a HV feedthrough and the anode support apparatus was modified to isolate the anode from the grounded beamline. The anode was then biased remotely because the power supply for the anode is separate from the power supply for the cathode.

Because the anode bias changes the electric potential of the photo-gun, it was expected that the biased anode configuration would affect the beam quality. The three tests performed to measure the effects on transverse beam position, transverse beam size, and time of flight showed that these effects are small and can be easily compensated for. The 50.5 ± 5.4 µm average transverse displacement of the beam was the most significant effect, but it can be compensated for with steering coils. The effects on the transverse beam size and time of flight were consistent with the resolution of the measuring equipment and thus were negligible. The QE measurement results in Figs. 32-34 show an exponential decay of QE during all three run periods regardless of anode bias. However, the QE decays at a slower rate when the anode is biased compared to when the anode is grounded. In Table 8, the charge lifetimes for a biased anode are 1.80 ± 0.51 times greater on average than the grounded anode charge lifetimes, which is a remarkable result considering the simplicity of implementing the biased anode configuration.

There are, however, several noteworthy observations from the QE measurement results. First, the charge lifetimes for grounded anode configurations generally improved during run periods 1 and 2 while charge lifetimes for biased anode configurations generally decreased. Indeed, the charge lifetime improvement decreased throughout run period 2 and there was no significant charge lifetime improvement in the latter half of run period 1. While it is possible that the anode bias power supply simply stopped working in the latter half of run period 1, the fact that there is also decrease in improvement in run period 2 suggests that the charge lifetime improvement depends on the QE at the laser spot at the time of anode biasing. One explanation is that back-bombarding ions from downstream of the anode remove weakly bound activation material via sputtering at the beginning of a run period. As

a result, if the anode is biased later in the run period, there is less weakly bound activation material, so there is less charge lifetime improvement.

Second, the QE measurements and charge lifetimes vary between lasers. The QE measurements for lasers A and C were generally higher compared to lasers B and D. However, the charge lifetimes were generally lower for lasers A and C compared to lasers B and D. These observations can be explained by:

1. Although the wavelength of all four lasers should be 780 nm, any variation in wavelength would affect the measured QE (see Eq. 9). However, this effect is likely small, since the four laser wavelengths are within 1 nm.

2. Because the QE is measured at the laser spot, any variation in the laser profiles would also affect the measured QE because the spatial distribution of QE throughout the active area is not uniform. In Fig. 17, there is clearly variation between the four laser profiles, albeit small.

3. Because the QE pattern across the active area is non-uniform, any variation in the laser position would also affect the measured QE. Nominally, all four lasers are incident on the same spot on the active area. Because the spatial distribution of QE throughout the active area is not uniform, the measured QE is affected by a change in laser position.

Looking at the QE scans and difference scans shown in Figs. 34-36, there are two main observations that can be made: there is QE degradation throughout entire active area regardless of the laser spot position, and the majority of the degradation occurs at the laser spot in run periods 1 and 3. Interestingly, the fact that these same two observations can be made for run period 3, when the anode configuration remained biased, and for run period 1, when the anode configuration was alternated between grounded and biased, suggests that the spatial distribution of QE degradation is independent of anode bias. In run period 2, the majority of the degradation occurred in between the two laser positions instead of at the laser positions. Looking closer at the individual QE difference scans, the majority of the degradation occurs near the laser spot, not at the laser spot, suggesting that the laser position may have drifted over the course of the run period.

When comparing these QE scan results to the GPT simulation results in Figs. 44-46, we see that the majority of the back-bombarding ions are incident on the laser spot, suggesting that the patterns of QE degradation in run periods 1 and 3 are correlated with the back-bombarding ion distribution. Comparing the ion density distributions to the ion

origin distributions, we see that the majority of ions that hit the laser spot originate from upstream of the anode. Because ions originate within the electron beam, ions originating upstream of the anode accelerate straight back to the laser position. On the other hand, the density distribution of back-bombarding ions originating downstream of the anode is spread out over the active area, suggesting that ions originating upstream of the anode contribute to the majority of the QE degradation at the laser spot, while ions originating downstream of the anode contribute to the degradation throughout the active area.

Looking at the energy-density distributions, we see that the highest energy density of ions originating upstream of the anode occurs within the tail of the distribution, implying that the further the ions are created from the photocathode, the higher their energy, and the more they are deflected towards the electrostatic center. Also, the energy density distribution of ions originating downstream of the anode is similar to the corresponding number density distribution. Ions originating downstream of the anode experience the full photo-gun potential before reaching the photocathode. Because the energy of these ions following ionization is four to five orders of magnitude less than their kinetic energy when reaching the photocathode (4 eV vs. 130 keV), there is little variation in energy when these ions reach the photocathode.

Looking at the combined ion distributions for run period 2, we see that the QE degradation is correlated with the ion energy-density distribution, not the ion (number) density distribution, suggesting that ion energy may also play a role in QE degradation. It is interesting that, as noted earlier, the pattern of QE degradation in run period 2 may have been affected by laser spot drift because the majority of the degradation occurred outside the laser spot, but the simulations producing the ion energy distributions had no laser drift! Thus, this correlation between QE degradation and ion energy density may be coincidental, but it warrants further study.

If our hypothesis that QE degradation is correlated with the number of ions incident on the laser spot is true, then the average charge lifetime improvement of 1.80 seen in Table 9 should be consistent with a factor of 1.80 decrease in the number of ions incident on the laser spot when the anode is biased. However, as Table 18 shows, there is at most a 2% decrease in the number of ions incident on the laser spot, or an 8-15% decrease when weighting by kinetic energy. The discrepancy is most likely due to the fact the distribution of ions from downstream of the anode is spread out over the active area instead of being concentrated at the laser spot. Interestingly, the energy-weighted ratios for ions incident on the photocathode do match the charge lifetime improvement of 1.80. This may simply be

coincidental, since the current measured in a QE measurement comes solely from the laser spot, not the entire photocathode. However, it does suggest a correlation between the charge lifetime improvement and ions that are incident outside of the laser spot.

If the reason why the ion ratios in Table 18 are not consistent with the lifetime improvement ratios in Table 9 is uncertainty in the laser spot position, then it is expected that at least one laser spot position would have a corresponding laser spot ion ratio of 1.80. However, the average ion ratio for ions at the laser spot for H_2 is 1.01 ± 0.02 and the corresponding average energy-weighted ratio is 1.13 ± 0.05. Therefore, even if the laser spot were misaligned by a $1 - 2$ mm, the difference between the number of ions incident on the laser spot between grounded and biased anode configurations is 1-3%, or 8-18% when weighting by kinetic energy.

When we compare the ion origin plots between different laser spot positions, we can see why there is little difference in the number of ions incident on the laser spot. When the laser position is moved, the distribution of ions from upstream of the anode changes, but the distribution of ions from downstream of the anode remains the same. The tail of the distribution of ions from downstream of the anode always points towards the electrostatic center, regardless of the residual gas species.

The ion ratios for ions incident on the active area are dependent on the relative laser spot position, but only when weighting by energy. That is, there is little variation in ion ratio when using the relative active area position from the 1st laser spot. But when using the 2nd laser spot, the ion ratios are 0.2 to 0.3 higher for laser spot positions with negative x coordinates. This correlation may be due to the 2nd laser spot being close to the upper right edge of the active area. As a result, if the x-coordinate of the laser spot position is negative, then a significant part of the ion distribution tail is outside the active area and thus would not contribute to the ion ratio. Thus, even if the active area were misaligned by 1-2 mm, the 15-25% variation in the energy-weighted ion ratios is still insufficient to explain the discrepancy between the charge lifetime improvement and the ion ratios at the laser spot.

One other possibility is that because heavier ions from residual gases such as CO, and CH_4 do not receive as much deflection as H_2 due to their larger masses, more of these ions would hit the laser spot. The partial pressures of CO and CH_4 are much lower than H_2 (see Fig. 14), but the ionization cross sections are 2-4 times higher (see Fig. 3), so more CO and CH_4 ions are produced than H_2 if the gas densities are the same. Figures 48 and 49 show that while there is no significant difference in the number of ions incident on the photocathode regardless of the laser spot position or gas species, the total number of ions decreases from

about 72,000 ions to 58,000, or about 20%, when the anode is biased. On the other hand, Figs. 50 and 51 show no significant difference in the number of ions incident on the active area when the laser spot is near the center of the photocathode. As the laser spot moves towards the edge of the photocathode, a greater portion of the laser spot is no longer in the active area, so these ions are not counted towards the active area ion total. The opposite is true for ions incident on the laser spot, as Figs. 52 and 53 show: when the laser spot is far from the electrostatic center, a lesser portion of the higher energy ions hit the laser spot because of their deflection towards the electrostatic center.

CHAPTER 7

CONCLUSIONS

The results of the biased anode experiments clearly demonstrate that biasing the anode improves the charge lifetime of the photocathode by 1.80 ± 0.51 on average. Because of its simplicity in implementation, this technique can be used in experiments that use GaAs photocathodes for polarized electron production in order to extend the operating lifetime of the photo-gun and meet the polarization and current requirements.

There is also clear utility of sophisticated simulation models to help explain and predict the QE degradation of the photocathode. The ionization custom element has been successfully developed and implemented in GPT to allow for the creation of these simulations. The Monte-Carlo algorithms implemented in the custom element routine ensure that the resulting simulations are probabilistic, as opposed to deterministic. The custom element has been benchmarked against theory and against IBSimu to ensure accuracy.

The simulation results show that while the back-bombarding ion energy density distribution correlates with the measured QE degradation, the charge lifetime improvement does not correlate with the energy-weighted ion ratio for ions incident on the laser spot, as our original hypothesis predicted. The systematic studies showed that these two results are not significantly affected by uncertainty in the laser spot position, the relative active area position, or the gas species used in the simulations.

There are several limitations to these simulations that can be improved upon to increase accuracy. First, the only residual gases currently supported by the custom element are H_2, CO, and CH_4. These were chosen because they are the most common residual gases in the CEBAF accelerator. However, other residual gases can be supported by the custom element provided their ionization parameters are known. Second, the ion energy routine uses a Maxwellian distribution of ion energies with the most probable energy of 4 eV to determine the kinetic energy of the ion after ionization. The assumption is that the residual gas is at room temperature and obeys the ideal gas law, which may not be true in every context. Thus, universal application of the custom element would require a more sophisticated theory. Lastly, because the custom element loops through every particle present in the simulation with every timestep, simulations that use a large number of electron macro particles or involve high ion production rates would require very long CPU times.

While ion damage mechanisms, such as sputtering and implantation, have been well studied, it is not known how these damage mechanisms degrade photocathode QE. Experiments can be performed to study each individual mechanism and determine which mechanisms lead to QE degradation. One experiment could be to bombard an activated GaAs photocathode with an ion beam made of a residual gas ion, such as H_2^+. Based on an analysis of the ion damage and simulations made using codes such as SRIM [83], one can determine the extent to which sputtering and implantation contribute to QE degradation. Once these mechanisms are known, the extensibility of GPT allows one to easily create other custom elements, or to modify existing ones, to improve upon the existing simulation model and increase its accuracy.

Another possible avenue for future experimental and simulation work is to study the time dependence of back-bombarding ions on QE degradation. The decrease in the charge lifetime improvement throughout each run period suggests a time dependence on QE decay: the rate of QE decay might be proportional to the already existing QE of the photocathode. In other words, the effect of back-bombarding ions on QE degradation might not depend on how the ions hit the photocathode, but rather when they hit the photocathode. This time dependence can be studied by analyzing the QE decay of photocathodes with different amounts of activation material, as the concentration of activation material determines the local QE of the photocathode for a given laser power [43, 84, 85, 86], and measuring the charge lifetime of each photocathode.

REFERENCES

[1] M. J. Fero *et al.*, "Precision standard model tests with polarized e^+e^- beams", SLAC, Menlo Park, CA, USA, Rep. SLAC-PUB-6678, Oct. 1994.

[2] G. Moortgat-Pick *et al.*, "Polarized positrons and electrons at the linear collider", *Phys. Rep.*, vol. 460, no. 4, pp. 131–243, 2008, doi:10.1016/j.physrep.2007.12.003

[3] J. Arrington, K. de Jager, and C. F. Perdrisat, "Nucleon form factors - a Jefferson Lab perspective", *J. Phys. Conf. Ser.*, vol. 299, p. 012002, 2011, doi:10.1088/1742-6596/299/1/012002

[4] D. Adhikari *et al.* (PREX Collaboration), "Accurate Determination of the Neutron Skin Thickness of ^{208}Pb through Parity-Violation in Electron Scattering", *Phys. Rev. Lett.*, vol. 126 p. 172502, Apr. 2021, doi:10.1103/PhysRevLett.126.172502

[5] E. L. Garwin, D. T. Pierce, and H. C. Siegmann, "Polarized photoelectrons from optically magnetized semiconductors", *Helv. Phys. Acta*, vol. 47, p. 393, 1974.

[6] G. Lampel and C. Weisbuch, "Proposal for an efficient source of polarized photoelectrons from semiconductors", *Solid State Commun.*, vol. 16, pp. 877–880, Apr. 1975, doi:10.1016/0038-1098(75)90884-4

[7] W. E. Spicer, "Photoemissive, photoconductive, and optical absorption studies of alkali-antimony compounds", *Phys. Rev.*, vol. 112, pp. 114–122, Oct. 1958, doi:10.1103/PhysRev.112.114

[8] T. Nishitani, M. Tabuchi, H. Amano, T. Maekawa, M. Kuwahara, and T. Meguro, "Photoemission lifetime of a negative electron affinity gallium nitride photocathode" *J. Vac. Sci. Technol. B*, vol. 32, no. 6, p. 06F901, Oct. 2014, doi:10.1116/1.4901566

[9] U. Fano, "Effects of configuration interaction on intensities and phase shifts", *Phys. Rev.*, vol. 124, pp. 1866–1878, Dec. 1961, doi:10.1103/PhysRev.124.1866

[10] B. I. Zakharchenya, V. G. Fleisher, R. R. Dzhioev, Yu. P. Veshchunov, and I. B. Rusanov, "Effect of optical orientation of electron spins in a GaAs crystal", *J. Exp. Theor. Phys. Lett.*, vol. 13, no. 4, pp. 195–197, Feb. 1971.

[11] D. T. Pierce, F. Meier, and P. Zürcher, "Negative electron affinity GaAs: A new source of spin-polarized electrons", *Appl. Phys. Lett.*, vol. 26, no. 12, pp. 670–672, 1975, doi:10.1063/1.88030

[12] D. T. Pierce and F. Meier, "Photoemission of spin-polarized electrons from GaAs", *Phys. Rev. B*, vol. 13, pp. 5484–5500, Jun. 1976, doi:10.1103/PhysRevB.13.5484

[13] G. D. Cates *et al.*, "The bates polarized electron source", *Nucl. Instrum. Methods Phys. Res. A: Accel. Spectrom. Detect. Assoc. Equip.*, vol. 278, no. 2, pp. 293–317, Jun. 1989, doi:10.1016/0168-9002(89)90846-2

[14] T. Maruyama, E. L. Garwin, R. Prepost, G. Zapalac, J. S. Smith, and J. D. Walker, "Observation of strain enhanced electron spin polarization in photoemission from InGaAs", *Phys. Rev. Lett.*, vol. 66, pp. 2376–2379, May 1991, doi:10.1103/PhysRevLett.66.2376

[15] T. Maruyama, E. L. Garwin, R. Prepost, G. H. Zapalac, "Electron-spin polarization in photoemission from strained GaAs grown on $GaAs_{1-x}P_x$", *Phys. Rev. B*, vol. 46, pp. 4261–4264, Aug. 1992, doi:10.1103/PhysRevB.46.4261

[16] R. Alley *et al.* "The stanford linear accelerator polarized electron source", *Nucl. Instrum. Methods Phys. Res. A: Accel. Spectrom. Detect. Assoc. Equip.*, vol. 365, no. 1, pp. 1–27, Nov. 1995, doi:10.1016/0168-9002(95)00450-5

[17] H. G. Andresen *et al.* (B2 Collaboration), "Operational experience with the MAMI-source of polarized electrons", in *Proc. Fifth European Particle Accelerator Conference (EPAC96)*, Sitges, Spain, 1997, pp. 1502–4.

[18] M. J. J. van den Putte *et al.*, "The polarized electron source at NIKHEF", *AIP Conf. Proc.*, vol. 421, no. 1, pp. 260–269, doi:10.1063/1.55027

[19] E. Tsentalovich *et al.*, "Development of a polarized electron source for the MIT-Bates Linear Accelerator Center", *Nucl. Instrum. Methods Phys. Res. A: Accel. Spectrom. Detect. Assoc. Equip.*, vol. 582, no. 2, pp. 413–428, Nov. 2007, doi:10.1016/j.nima.2007.09.009

[20] W. Hillert, M. Gowin, and B. Neff, "The 50 kV inverted source of polarized electrons at ELSA", *AIP Conf. Proc.*, vol. 570, no. 1, pp. 961–964, 2001, doi:10.1063/1.1384233

[21] C. K. Sinclair *et al.*, "Development of a high average current polarized electron source with long cathode operational lifetime", *Phys. Rev. ST Accel. Beams*, vol. 10, p. 023501, Feb. 2007, doi:10.1103/PhysRevSTAB.10.023501

[22] H. G. Andresen *et al.* (B2 Collaboration), "Operating experience with the MAMI polarized electron source", in *Proc. Workshop on Photocathodes for Polarized Electron Sources for Accelerators*, Stanford, CA, USA, Sep. 1993, SLAC-432, pp. 2–12.

[23] K. Aulenbacher *et al.*, "The MAMI source of polarized electrons", *Nucl. Instrum. Methods Phys. Res. A: Accel. Spectrom. Detect. Assoc. Equip.*, vol. 391, no. 3, pp. 498–506, Jun. 1997, doi:10.1016/S0168-9002(97)00528-7

[24] R. R. Mammei *et al.*, "Charge lifetime measurements at high average current using a K_2CsSb photocathode inside a DC high voltage photogun", *Phys. Rev. ST Accel. Beams*, vol. 16, no. 3, p. 033401, Mar. 2013, doi:10.1103/PhysRevSTAB.16.033401

[25] W. Liu, S. Zhang, M. Stutzman, and M. Poelker, "Effects of ion bombardment on bulk GaAs photocathodes with different surface-cleavage planes", *Phys. Rev. Accel. Beams*, vol. 19, no. 10, p. 103402, Oct. 2016, doi:10.1103/PhysRevAccelBeams.19.103402

[26] F. F. Umarov and A. A. Dzhurakhalov, "Radiation effects in materials", in *Ion Bombardment-Induced Surface Effects in Materials*, W. Monteiro, Ed. London, UK: IntechOpen, 2016, doi:10.5772/62731

[27] M. L. Stutzman and J. Grames, "Superlattice photocathode damage analysis", *AIP Conf. Proc.*, vol. 1149, no. 1, pp. 1032–1037, Aug. 2009, doi:10.1063/1.32155588

[28] S. A. E. Johansson and T. B. Johansson, "Analytical application of particle induced X-ray emission", *Nucl. Instr. Meth.*, vol. 137, no. 3, pp. 473–516, Sep. 1976, doi:10.1016/0029-554X(76)90470-5

[29] H. Salow, "Uber den Sekundaremissionfaktor elektronenbestrahlter Isolatoren (On the secondary electron yield of electron bombarded insulators)", *Z. Fur. Tech. Phys.*, vol. 21, no. 8, pp. 8–15, 1940.

[30] G. F. Dionne, "Effects of secondary electron scattering on secondary emission yield curves", *J. Appl. Phys.*, vol. 44, no. 12, pp. 5361–5364, Oct. 2003, doi:10.1063/1.1662156

[31] A. Chatterjee, K. Blaser, M. Ehrlichman, D. Rubin, and J. Shanks, "Fast Ion Instability at CESR-TA", in *Proc. 5th Int. Particle Accelerator Conf. (IPAC'14)*, Dresden, Germany, Jun. 2014, pp. 1638–1640, doi:10.18429/JACoW-IPAC2014-TUPRI036

[32] A. Chao, "Lecture notes on topics in accelerator physics", SLAC, Menlo Park, CA, USA, Rep. SLAC-PUB-9574, Nov. 2002.

[33] C. L. Olson, "Charge-neutralization processes for intense relativistic electron beams in low-pressure neutral gases", *Phys. Rev. A*, vol. 11, no. 1, pp. 288–296, Jan. 1975, doi:10.1103/PhysRevA.11.288

[34] R. Yang *et al.*, "Evaluation of beam halo from beam-gas scattering at the KEK Accelerator Test Facility", *Phys. Rev. Accel. Beams*, vol. 21, no. 5, p. 051001, May 2018, doi:10.1103/PhysRevAccelBeams.21.051001

[35] M. L. Stutzman *et al.*, "Characterization of the CEBAF 100kV DC GaAs photoelectron gun vacuum system", *Nucl. Instrum. Methods Phys. Res. A: Accel. Spectrom. Detect. Assoc. Equip.*, vol. 574, no. 2, pp. 213–220, May 2007, doi:10.1016/j.nima.2007.01.170

[36] C. D. Park, S. M. Chung, X. Liu, and Y. Li, "Reduction in hydrogen outgassing from stainless steels by a medium-temperature heat treatment", *J. Vac. Sci. Technol. A*, vol. 26, no. 5, pp. 116–1171, Aug. 2008, doi:10.1116/1.2956625

[37] J. Grames *et al.*, "Lifetime measurements using the Jefferson Lab load-lock electron gun", in *Proc. Spin'04*, Trieste, Italy, Oct. 2004, pp. 948–952, doi:10.1142/9789812701909_0191

[38] T. Maruyama *et al.*, "A Very high charge, high polarization gradient doped strained GaAs photocathode", *Nucl. Instrum. Meth. A*, vol. 492, no. 1-2 pp. 199–211, Oct. 2002, doi:10.1016/S0168-9002(02)01290-1

[39] Y. Sun, R. E. Kirby, T. Maruyama, G. A. Mulhollan, J. C. Bierman, and P. Pianetta, "The surface activation layer of GaAs negative electron affinity photocathode activated by Cs, Li, and NF_3", *Appl. Phys. Lett.*, vol. 95, no. 17, p. 174109, Oct. 1009, doi:10.1063/1.3257730

[40] L. Cultrera, A. Galdi, J. K. Bae, F. Ikponmwen, J. Maxson, and I. Bazarov, "Long lifetime polarized electron beam production from negative electron affinity GaAs activated with Sb-Cs-O: Trade-offs between efficiency, spin polarization, and lifetime", *Phys. Rev. Accel. Beams*, vol. 23, no. 2, p. 023401, Feb. 2020, doi:10.1103/PhysRevAccelBeams.23.023401

[41] J. K. Bae, A. Galdi, L. Cultrera, F. Ikponmwen, J. Maxson, and I. Bazarov, "Improved lifetime of a high spin polarization superlattice photocathode", *J. Appl. Phys.*, vol. 127, no. 12, p. 124901, Mar. 2020, doi:10.1063/1.5139674

[42] J. K. Bae, L. Cultrera, P. DiGiacomo, and I. Bazarov, "Rugged spin-polarized electron sources based on negative electron affinity GaAs photocathode with robust Cs_2Te coating", *Appl. Phys. Lett.*, vol. 112, no. 15, p. 154101, Apr. 2018, doi:10.1063/1.5026701

[43] Y. Wang *et al.*, "Thermal emittance and lifetime of alkali-antimonide photocathodes grown on GaAs and molybdenum substrates evaluated in a $-300\,\mathrm{kV}$ DC photogun", *Phys. Rev. Accel. Beams*, vol. 23, no. 10, p. 103401, Oct. 2020, doi:10.1103/PhysRevAccelBeams.23.103401

[44] A. Poncet, "Ion trapping, clearing, beam-ion interactions", CERN, Geneva, Switzerland, Rep. CERN-OPEN-2000-278, 1999.

[45] S. Full, A. Bartnik, I. V. Bazarov, J. Dobbins, B. Dunham, and G. H. Hoffstaetter, "Detection and clearing of trapped ions in the high current Cornell photoinjector", *Phys. Rev. Accel. Beams* vol. 19, no. 3, p. 034201, Mar. 2016, doi:10.1103/PhysRevAccelBeams.19.034201

[46] G. H. Hoffstaetter and C. Spethmann, "Equilibrium ion distribution in the presence of clearing electrodes and its influence on electron dynamics", *Phys. Rev. ST Accel. Beams*, vol. 11, no. 1, p. 014001, Jan. 2008, doi:10.1103/PhysRevSTAB.11.014001

[47] Y. Miyahara, "Investigation of trapped ions with a clearing electrode", *J. Vac. Sci. Technol. A*, vol. 5, no. 5, pp. 2927–2929, Mar. 1987, doi:10.1116/1.574267

[48] E. Bozoki and S. L. Kramer, "Ion shaking in the 200 MeV XLS-Ring", BNL, Upton, NY, USA, Rep. BNL-48679, 1992.

[49] E. Bozoki and D. Sagan, "On the shaking of ions in electron storage rings", *Nucl. Instrum. Methods Phys. Res. A: Accel. Spectrom. Detect. Assoc. Equip.*, vol. 340, no. 2, pp. 259–271, Feb. 1994, doi:10.1016/0168-9002(94)90101-5

[50] R. Alves Pires, D. Mohl, Y. Orlov, F. Pedersen, A. Poncet, and S. van der Meer, "On the theory of coherent instabilities due to coupling between a dense cooled beam and charged particles from the residual gas", in *Proc. PAC'89*, Chicago, IL, USA, Mar. 1989, pp. 800–802, doi:10.1109/PAC.1989.73254

[51] J. Grames *et al.*, "Ion back bombardment of GaAs photocathodes inside DC high voltage electron guns", in *Proc. PAC'05*, Knoxville, TN, USA, May 2005, pp. 2875–2877, doi:10.1109/PAC.2005.1591299

[52] J. Grames *et al.* "Milliampere beam studies using high polarization photocathodes at the CEBAF Photoinjector", in *Proc. PSTP'17*, Kaist, South Korea Oct. 2017, p. 014, doi:10.22323/1.324.0014

[53] J. Grames *et al.*, "A biased anode to suppress ion back-bombardment in a DC high voltage photoelectron gun", *AIP Conf. Proc.*, vol. 980, no. 1, pp. 110–117, Feb. 2008, doi:10.1063/1.2888075

[54] E. Pozdeyev, "Ion trapping and cathode bombardment by trapped ions in DC photoguns", *Phys. Rev. ST Accel. Beams*, vol. 10, no. 8, p. 083501, Aug. 2007, doi:10.1103/PhysRevSTAB.10.083501

[55] General Particle Tracer (GPT) Version 3.39, http://www.pulsar.nl/gpt

[56] CST Studio Suite, http://www.cst.com

[57] M. Reiser, "Linear beam optics with space charge", in *Theory and Design of Charged Particle Beams* Weinheim, Germany: Wiley VCH Verlag GmbH, 2008, pp. 163–272

[58] H. Bethe, "Zur Theorie des Durchgangs schneller Korpuskularstrahlen durch Materie (On the theory of the passage of fast corpuscular rays through matter)", *Ann. Phys. (Leipzig)*, vol. 397, no. 3, pp. 325–400, 1930, doi:10.1002/andp.19303970303

[59] F. F. Rieke and W. Prepejchal, "Ionization cross sections of gaseous atoms and molecules for high-energy electrons and positrons", *Phys. Rev. A*, vol. 6, no. 4, pp. 1507-1519, Oct. 1972, doi:10.1103/PhysRevA.6.1507

[60] Y. Kim and M. E. Rudd, "Binary-encounter dipole model for electron-impact ionization", *Phys. Rev. A*, vol. 50, no. 5, pp. 3954–3967, Nov. 1994, doi:10.1103/PhysRevA.50.3954

[61] R. Kazimi, "Simultaneous Four-hall Operation for 12 GeV CEBAF", in *Proc. IPAC'13*, Shanghai, China, May 2013, paper THPFI091, p. 3502.

[62] G. A. Krafft *et al.*, "CEBAF at 12 GeV", unpublished.

[63] G. Palacios-Serrano, "Electrostatic design and characterization of a 200 keV photogun and wien spin rotator", Ph.D. thesis, Phys. Dept. Old Dominion University, Norfolk, VA, 2021.

[64] Y. Wang, "Development of a 300 kV DC high voltage photogun and beam based studies of alkali antimonide photocathodes", Ph.D. thesis, Phys. Dept. Old Dominion University, Norfolk, VA, 2018.

[65] G. Palacios-Serrano, F. Hannon, C. Hernandez-Garcia, M. Poelker, and H. Baumgart, "Electrostatic design and conditioning of a triple point junction shield for a -200 kV DC high voltage photogun", *Rev. Sci. Instrum.*, vol. 89, no. 10, p. 104703, Oct. 2018, doi:10.1063/1.5048700

[66] J. R. Pierce, "Rectilinear electron flow in beams", *J. Appl. Phys.*, vol. 11, no. 8, pp. 548–554, 1940, doi:10.1063/1.1712815

[67] W. Liu *et al.*, "Record-level quantum efficiency from a high polarization strained GaAs/-GaAsP superlattice photocathode with distributed Bragg reflector", *Appl. Phys. Lett.*, vol. 109, no. 25, p. 252104, Nov. 2016, doi:10.1063/1.4972180

[68] L. W. James, G. A. Antypas, J. J. Uebbing, T. O. Yep, and R. L. Bell, "Optimization of the $InAs_xP_{1-x-}Cs_2O$ photocathode", *J. Appl. Phys.*, vol. 42, no. 2, pp. 580–586, 1971, doi:10.1063/1.1660067

[69] X. Jin *et al.*, "Low energy electron microscopy and Auger electron spectroscopy studies of Cs-O activation layer on p-type GaAs photocathode", *J. Appl. Phys.*, vol. 116, no. 17, p. 174509, Oct. 2014, doi:10.1063/1.4901201

[70] SAES, https://www.saesgetters.com/products-functions/products/dispensers/alkali-metals-dispensers

[71] M. Baylac *et al.*, "Effects of atomic hydrogen and deuterium exposure on high polarization GaAs photocathodes", *Phys. Rev. ST Accel. Beams*, vol. 8, no. 12, p. 123501, Dec. 2005, doi:10.1103/PhysRevSTAB.8.123501

[72] R. Suleiman, P. Adderley, J. Grames, J. Hansknecht, M. Poelker, and M. Stutzman, "High current polarized electron source", *AIP Conf. Proc.*, vol. 1970, no. 1, p. 050007, May 2018, doi:10.1063/1.5040226

[73] P. A. Adderley *et al.*, "Load-locked DC high voltage GaAs photogun with an inverted-geometry ceramic insulator" *Phys. Rev. ST Accel. Beams*, vol. 13, no. 1, p. 010101, Jan 2010, doi:10.1103/PhysRevSTAB.13.010101

[74] *SORB-AC Cartridge Pumps - MK5 Series*, Lainate, Italy, 2001, "https://www.saesgetters.com/sites/default/files/SORB%20AC%20General_0.pdf"

[75] MKS, https://www.mksinst.com/n/gas-correction-factors-for-ionization-vacuum-gauges

[76] Ophir Spiricon, https://www.ophiropt.com/user_files/laser/press-release/L11058.pdf

[77] BeamGage Professional Version 6.15.3, https://www.ophiropt.com/laser--measurement/beam-profilers/products/Beam-Profiling/Camera-Profiling-with-BeamGage

[78] Y. Wang, "CEBAF overview", presented at *25th Annual Hampton University Graduate Studies Program (HUGS 2010)*, Jefferson Lab, Newport News, Virginia, USA, June 2010, https://www.jlab.org/hugs/archive/Schedule2010/slides/Wang.pdf

[79] Experimental Physics and Industrial Control System (EPICS), https://epics.anl.gov/index.php

[80] M. Ahmad, "CEBAF Upgrade Bunch Length Measurements", Ph.D. Thesis, Phys. Dept., Old Dominion University, Norfolk, Virginia, 2016.

[81] J. R. Cash and A. H. Karp, "A variable order Runge-Kutta method for initial value problems with rapidly varying right-hand sides", *ACM Trans. Math. Softw.*, vol. 16, no. 3, pp. 201–222, Sep. 1990, doi:10.1145/79505.79507

[82] Ion Beam Simulator (IBSIMU), http://ibsimu.sourceforge.net

[83] The Stopping and Range of Ions in Matter (SRIM) Version SRIM-2013.00, http://srim.org

[84] N. Chanlek, R. Jones, J. Herbert, L. Jones, K. Middleman, and B. Militsyn, "Gallium arsenide preparation and QE lifetime studies using the ALICE photocathode preparation facility", in *Proc. IPAC'10*, Kyoto, Japan, May 2010, paper TUPEC018, p. 1752

[85] J. Biswas, E. Wang, M. Gaowei, W. Liu, O. Rahman, and J. T. Sadowski, "High quantum efficiency GaAs photocathodes activated with Cs, O_2, and Te", *AIP Adv.*, vol. 11, no. 2, p. 025321, Jan. 2021, doi:10.1063/5.0026839

[86] Y. Zhang *et al.*, "Improved activation technique for preparing high-efficiency GaAs photocathodes", *Opt. Mater. Express*, vol. 7, no. 9, pp. 3456–3465, Sep. 2017, doi:10.134/OME.7.003456

APPENDIX

The GPT ionization custom element code is split up into two files: a header file containing an info structure with function definitions and parameters for each supported gas species, and a file containing the main element code. Both files are listed and are fully described below.

HEADER FILE

7.0.1 GLOBAL AND ION-SPECIFIC PARAMETERS

The header file starts with declarations of the global parameters, which will be defined later during initialization. These global parameters are chosen by the user when declaring the function in the GPT input file. These parameters are:

- `axis, tf, L`: Coordinate system and the boundaries of the element.
- `set_electrons, set_gas,`...: Names of the primary electron, target gas particle (and ion), secondary electron, and scattered electron particle sets.
- `gasdensity, *z, *fz, *fz2, densitypoints`: Gas density (either constant or via 1-D density file).
- `IPRInfoFileName`: Name of GDF (GPT Data File) info file for ionization info output.

The ion-specific parameters `mass, charge` refer to the mass and charge of the ion. `radius` refers to the Plummer radius of the ion, which is used with certain GPT space charge elements. `Brmag` refers to the magnitude of the normalized velocity $\beta = v/c$ of the gas molecule prior to ionization. These parameters are "virtual", meaning that they will be defined later by one of the ion substructures listed later in the header file.

```
axis_handle axis;
gpttransform tf;
double L;

parset_handle set_electrons;
parset_handle set_gas;
parset_handle set_secondaryelectrons;
```

```
parset_handle set_scatteredelectrons;

double gasdensity;
double *z, *fz, *fz2 ;
int densityPoints = 0;

string IPRInfoFileName;

virtual double get_mass() = 0;
virtual double get_charge() = 0;
virtual double get_radius() = 0;
virtual double get_Brmag() = 0;
```

7.0.2 FUNCTION DEFINITIONS

Ionization Cross Section

The ionization cross section (ICS) is calculated using Eq. 2 in Ch. 2 [57], where `A1, A2` and `Ii` are ion-specific variables, and `G` is the Lorentz factor of the primary electron. Note that variables starting with "`gpt_`" are built-in physical constants: `gpt_me, gpt_c` and `gpt_qe` refer to the electron mass (in eV/c^2), speed of light (in m/s) and electric charge (in C) respectively.

```
virtual double get_A1() = 0;
virtual double get_A2() = 0;

double get_ICS(double A1, double A2, double Ii, double G)
{
  double betasquared = (G*G - 1.0) / (G*G);
  double Te = ((G - 1.0)*gpt_me*gpt_c*gpt_c) / (-gpt_qe);
  double ICStermone = (1.872e-24)*A1/betasquared;
  double ICStermtwo = (Ii / Te)*((Te / Ii) - 1.0);
  double ICStermthree = log((7.515e4)*A2*betasquared*G*G) - betasquared;
  double ICS = ICStermone * ICStermtwo * ICStermthree;
  return ICS;
}
```

Secondary Electron Differential Cross Section

The secondary electron differential cross section (SEDCS) is calculated using Eqs. 5 and 6 in Ch. 2 [60]. Gases H_2 and He use the `SEDCS_BED` calculation while other gases use the

`SEDCS_BEB` calculation. To minimize loss of precision, constants such as `hbar, bohrradius,` and `eRyd` are derived as much as possible from built-in constants or Planck's constant, which has a fixed value in SI units.

The SEDCS equations are generalized such that the user can choose both the gas species and the orbital from which the secondary electron originates (with the exception of H_2 and He, as there is only one orbital). Thus, the variables `U, B,` and `N` depend on the ion species as well as the orbital chosen. The definition of `S` depends on whether the BED or BEB calculation is used.

The `get_SEDCS` function is used by each ion substructure to calculate `S` and then calculate the SEDCS using either the BED or BEB equation. Having each ion substructure calculate the SEDCS individually allows the main element code to call `get_SEDCS` and always get the correct value regardless of the chosen target gas particle or orbital.

```
double plankconstant = 6.62607015e-34;
double hbar = plankconstant / (2 * gpt_pi);
double bohrradius =
(4.0 * gpt_pi*gpt_eps0*hbar*hbar) / (gpt_me*gpt_qe*gpt_qe);
double eRyd = (gpt_me*gpt_qe*gpt_qe*gpt_qe*gpt_qe) /
↪  (8*gpt_eps0*gpt_eps0*plankconstant*plankconstant) / (-gpt_qe);

double get_uratio(double U, double B) {return U/B;}
double get_tratio(double T, double B) {return T/B;}
double get_wratio(double W, double B) {return W/B;}

double get_yw(double wratio){return 1.0/(wratio+1.0);}
double get_dfdw(
  double c,
  double d,
  double e,
  double f,
  double y)
{
  double dfdw = c*y*y*y + d*y*y*y*y + e*y*y*y*y*y + f*y*y*y*y*y*y;
  return dfdw;
}

double SEDCS_BED_calc(
  double S,
  double B,
  double Ni,
  double N,
```

```
  double t,
  double u,
  double w,
  double dfdw)
{
  double termone = S/(B*(t+u+1.0));
  double termtwo = ( ((Ni/N)-2.0)/(t+1) ) * ( 1/(w+1) + 1/(t-w) );
  double termthree = (2-(Ni/N)) * ( (1/(w+1))*(1/(w+1)) + (1/(t-w))*(1/(t-w)) );
  double termfour = (log(t)/(N*(w+1))) * dfdw;
  double SEDCS = termone*(termtwo + termthree + termfour);
  return SEDCS;
}

double SEDCS_BEB_calc(
  double S,
  double B,
  double N,
  double t,
  double u,
  double w)
{
  double termone = S/(B*(t+u+1.0));
  double termtwo = (1/(w+1))*(1/(w+1)) + (1/(t-w))*(1/(t-w));
  double termthree = (1/(1+t)) * ( 1/(t-w) + 1/(w+1) );
  double termfour = log(t) * ( (1/(t-w))*(1/(t-w))*(1/(t-w)) +
  ↪  (1/(w+1))*(1/(w+1))*(1/(w+1)) );
  double SEDCS = termone*(termtwo - termthree + termfour);
  return SEDCS;
}

virtual double get_U() = 0;
virtual double get_B() = 0;
virtual double get_N() = 0;
virtual double get_SEDCS(double T, double W) = 0;
```

Integration Functions

The SEDCS and Maxwellian distributions (Eq. 43) are integrated numerically via the trapezoidal approximation $F(E)$ (Eq. 17). The integration interval is subdivided into partitions (i.e., `steps`). The number of partitions is chosen in the main element code to be sufficiently large to ensure precision without wasting CPU time. In the main element code,

the number of partitions is 500, though this may be later added to the list of global parameters to allow the user to set the precision.

```
double get_FE_SEDCS(int steps, double maxE, double sedcsT)
{
  double FEsum = 0;
  double stepsize = maxE / steps;
  for (int i = 0; i < steps; i++)
  {
    FEsum += 0.5*stepsize*(get_SEDCS(sedcsT, stepsize+i*stepsize) + get_SEDCS(sedcsT,
    ↪  stepsize + (i + 1)*stepsize));
  }
  return FEsum;
}

double get_Maxwell(double v, double a)
{
  double result = sqrt(2/gpt_pi) * ( v*v*exp(-1.0*(v*v)/(2*a*a)) )/( a*a*a );
  return result;
}

double get_FE_Maxwell(int steps, double maxV, double a)
{
  double FEsum = 0.0;
  double stepsize = maxV/steps;
  for(int i=0; i<steps; i++)
  {
    FEsum += 0.5 * stepsize
    * (get_Maxwell((stepsize + i*stepsize),a)
    + get_Maxwell((stepsize+(i+1)*stepsize),a));
  }
  return FEsum;
}
```

7.0.3 SWITCHES

The code uses three switches, coded as integers that are either 0 (OFF) or 1 (ON). If turned on, `SESwitch` and `ScatESwitch` put secondary and scattered electrons into their respective particle sets and add them to the simulation at the end of the timestep. If turned off, the code calculates the secondary and scattered electron parameters *as if* they were present. If the `IPRInfoSwitch` is turned on, ionization info is written to a separate GDF info file.

```
int SESwitch;
int ScatESwitch;
int IPRInfoSwitch;
```

7.0.4 IONIZATION INFO PARAMETER LIST

The ionization info file contains parameters for all ionization events in a simulation. The parameters are kept in vectors for easy reading/writing. The parameters for all ionizations in a given timestep are first written to temporary vectors (i.e., vectors appended by "`_temp`"). If the timestep fails, these vectors are cleared with the `clearAllTempVectors()` function. If the timestep succeeds, the parameters are added to the *end* of the respective vectors. After the simulation, these vectors are written to a GDF file (described later).

The complete list of ionization parameters are described in Table 20. For conciseness, only the first three parameters are shown in the code listing below.

```
vector<double> ID;
vector<double> n;
vector<double> tend;
...

vector<double> ID_TEMP;
vector<double> n_TEMP;
vector<double> tend_TEMP;
...

void clearAllTempVectors()
{
  vector<double>().swap(ID_TEMP);
  vector<double>().swap(n_TEMP);
```

```
  vector<double>().swap(tend_TEMP);
  ...
}

void addAllTempVectors()
{
  ID.insert(ID.end(), ID_TEMP.begin(), ID_TEMP.end());
  n.insert(n.end(), n_TEMP.begin(), n_TEMP.end());
  tend.insert(tend.end(), tend_TEMP.begin(), tend_TEMP.end());
  ...
}
```

7.0.5 ION INFO SUBSTRUCTURES

Each ion info substructure inherits the global info structure so that the main element code can choose an ion substructure and still have access to all functions and parameters in the global info structure. Ions that have more than one electron orbital have multiple orbital substructures for the ion substructure.

Each substructure contains info on the `mass, charge, radius, Brmag, A1, A2` parameters as well as the corresponding SEDCS calculation.

```
struct H2Info : public ionization_routine_9_info
{
        double get_mass()
        {
                double H2ionMass = 2.0 * gpt_mp + gpt_me;
                return H2ionMass;
        }
        double get_charge()
        {
                double H2ionCharge = -gpt_qe;
                return H2ionCharge;
        }
        double get_radius() { return 2e-9; }

        double get_Brmag() { return 6.353e-6; }

        double get_U()        { return 25.68; }
        double get_B()        { return 15.43; }
        double get_N()        { return 2.0 ; }
        double Ni = 1.173;
```

```
        double c = 1.1262;
        double d = 6.3982;
        double e = -7.8055;
        double f = 2.1440;

        double get_SEDCS(double T, double W)
        {
                double S =
                ↪  4*gpt_pi*bohrradius*bohrradius*get_N()*eRyd*eRyd/(get_B()*get_B());
                double y = get_yw( get_wratio(W,get_B()) );
                double dfdw = get_dfdw( c, d, e, f, y);
                double SEDCS = SEDCS_BED_calc(S, get_B(), Ni, get_N(),
                ↪  get_tratio(T,get_B()), get_uratio(get_U(),get_B()),
                ↪  get_wratio(W,get_B()), dfdw);
                return SEDCS;
        }

        double get_A1() { return 0.695; }
        double get_A2() { return 1.5668; }
};

struct HeInfo : public ionization_routine_9_info
{

        double get_mass()
        {
                double HeionMass = 4.0 * gpt_mp  + gpt_me;
                return HeionMass;
        }
        double get_charge()
        {
                double HeionCharge = -gpt_qe;
                return HeionCharge;
        }
        double get_radius() { return 2e-9; }

        double get_Brmag() { return 4.509e-6; }

        double get_U()        { return 39.51; }
        double get_B()        { return 24.59; }
        double get_N()        { return 2.0 ;}
        double Ni = 1.605;
        double c = 12.178;
        double d = -29.585;
        double e = 31.251;
```

```
        double f = -12.175;

        double get_SEDCS(double T, double W)
        {
                double S =
                ↪  4*gpt_pi*bohrradius*bohrradius*get_N()*eRyd*eRyd/(get_B()*get_B());
                double y = get_yw( get_wratio(W,get_B()) );
                double dfdw = get_dfdw( c, d, e, f, y);
                double SEDCS = SEDCS_BED_calc(S, get_B(), Ni, get_N(),
                ↪  get_tratio(T,get_B()), get_uratio(get_U(),get_B()),
                ↪  get_wratio(W,get_B()), dfdw);
                return SEDCS;
        }

        double get_A1() { return 0.745; }
        double get_A2() { return 0.6174; }
};

struct COInfo : public ionization_routine_9_info
{
        double get_mass()
        {
                double COionMass = 28.0 * gpt_mp  + 27*gpt_me;
                return COionMass;
        }
        double get_charge()
        {
                double COionCharge = -gpt_qe;
                return COionCharge;
        }
        double get_radius() { return 2e-9; }

        double get_Brmag() { return 1.700e-6; }

        double get_A1() { return 3.70; }
        double get_A2() { return 0.1773; }
};

struct COInfo_5sigma : public COInfo
{
        double get_B()        { return 14.01; }
        double get_U()        { return 42.26; }
        double get_N()        { return 2.0;}

        double get_SEDCS(double T, double W)
```

```
        {
                double B = get_B();
                double N = get_N();
                double S = 4*gpt_pi*bohrradius*bohrradius*N*eRyd*eRyd/(B*B);
                double SEDCS = SEDCS_BEB_calc(S, B, N, get_tratio(T,B),
                ↪  get_uratio(get_U(),B), get_wratio(W,B));
                return SEDCS;
        }
};

struct COInfo_1pi : public COInfo
{
        double get_B()        { return 17.66; }
        double get_U()        { return 54.30; }
        double get_N()        { return 4.0;}

        double get_SEDCS(double T, double W)
        {
                double B = get_B();
                double N = get_N();
                double S = 4*gpt_pi*bohrradius*bohrradius*N*eRyd*eRyd/(B*B);
                double SEDCS = SEDCS_BEB_calc(S, B, N, get_tratio(T,B),
                ↪  get_uratio(get_U(),B), get_wratio(W,B));
                return SEDCS;
        }
};

struct COInfo_4sigma : public COInfo
{
        double get_B()        { return 21.92; }
        double get_U()        { return 73.18; }
        double get_N()        { return 2.0;}

        double get_SEDCS(double T, double W)
        {
                double B = get_B();
                double N = get_N();
                double S = 4*gpt_pi*bohrradius*bohrradius*N*eRyd*eRyd/(B*B);
                double SEDCS = SEDCS_BEB_calc(S, B, N, get_tratio(T,B),
                ↪  get_uratio(get_U(),B), get_wratio(W,B));
                return SEDCS;
        }
};

struct COInfo_3sigma : public COInfo
```

```
{
        double get_B()        { return 41.92; }
        double get_U()        { return 79.63; }
        double get_N()        { return 2.0;}

        double get_SEDCS(double T, double W)
        {
                double B = get_B();
                double N = get_N();
                double S = 4*gpt_pi*bohrradius*bohrradius*N*eRyd*eRyd/(B*B);
                double SEDCS = SEDCS_BEB_calc(S, B, N, get_tratio(T,B),
                ↪  get_uratio(get_U(),B), get_wratio(W,B));
                return SEDCS;
        }
};

struct CH4Info : public ionization_routine_9_info
{
        double get_mass()
        {
                double CH4ionMass = 16.0 * gpt_mp  + 9*gpt_me;
                return CH4ionMass;
        }

        double get_charge()
        {
                double CH4ionCharge = -gpt_qe;
                return CH4ionCharge;
        }

        double get_radius() { return 2e-9; }

        double get_Brmag() { return 2.249e-6; }

        double get_A1() { return 4.23; }
        double get_A2() { return 0.264; }
};

struct CH4Info_1t2 : public CH4Info
{
        double get_B()        { return 14.25; }
        double get_U()        { return 25.96; }
        double get_N()        { return 6.0; }

        double get_SEDCS(double T, double W)
```

```
        {
                double B = get_B();
                double N = get_N();
                double S = 4*gpt_pi*bohrradius*bohrradius*N*eRyd*eRyd/(B*B);
                double SEDCS = SEDCS_BEB_calc(S, B, N, get_tratio(T,B),
                ↪  get_uratio(get_U(),B), get_wratio(W,B));
                return SEDCS;
        }
};

struct CH4Info_2a1 : public CH4Info
{
        double get_B()        { return 25.73; }
        double get_U()        { return 33.05; }
        double get_N()        { return 2.0;}

        double get_SEDCS(double T, double W)
        {
                double B = get_B();
                double N = get_N();
                double S = 4*gpt_pi*bohrradius*bohrradius*N*eRyd*eRyd/(B*B);
                double SEDCS = SEDCS_BEB_calc(S, B, N, get_tratio(T,B),
                ↪  get_uratio(get_U(),B), get_wratio(W,B));
                return SEDCS;
        }
};

struct CH3Info : public ionization_routine_9_info
{
        double get_mass()
        {
                double CH3ionMass = 15.0 * gpt_mp  + 8*gpt_me;
                return CH3ionMass;
        }

        double get_charge()
        {
                double CH3ionCharge = -gpt_qe;
                return CH3ionCharge;
        }

        double get_radius() { return 2e-9; }

        double get_Brmag() { return 2.33e-6; }
```

```
        double get_A1() { return 4.23; }
        double get_A2() { return 0.264; }
};

struct CH3Info_1a2 : public CH3Info
{
        double get_B()        { return 9.84; }
        double get_U()        { return 30.40; }
        double get_N()        { return 1.0;}

        double get_SEDCS(double T, double W)
        {
                double B = get_B();
                double N = get_N();
                double S = 4*gpt_pi*bohrradius*bohrradius*N*eRyd*eRyd/(B*B);

                double SEDCS = SEDCS_BEB_calc(S, B, N, get_tratio(T,B),
                ↪  get_uratio(get_U(),B), get_wratio(W,B));
                return SEDCS;
        }
};

struct CH3Info_1e : public CH3Info
{
        double get_B()        { return 15.64; }
        double get_U()        { return 26.46; }
        double get_N()        { return 4.0;}

        double get_SEDCS(double T, double W)
        {
                double B = get_B();
                double N = get_N();
                double S = 4*gpt_pi*bohrradius*bohrradius*N*eRyd*eRyd/(B*B);

                double SEDCS = SEDCS_BEB_calc(S, B, N, get_tratio(T,B),
                ↪  get_uratio(get_U(),B), get_wratio(W,B));
                return SEDCS;
        }
};

struct CH3Info_2a1 : public CH3Info
{
        double get_B()        { return 24.57; }
        double get_U()        { return 34.18; }
        double get_N()        { return 2.0;}
```

```
        double get_SEDCS(double T, double W)
        {
                double B = get_B();
                double N = get_N();
                double S = 4*gpt_pi*bohrradius*bohrradius*N*eRyd*eRyd/(B*B);
                double SEDCS = SEDCS_BEB_calc(S, B, N, get_tratio(T,B),
                ↪  get_uratio(get_U(),B), get_wratio(W,B));
                return SEDCS;
        }
};
```

MAIN ELEMENT CODE

7.0.6 LIBRARIES AND FORWARD DECLARATIONS

The following are a list of C++ libraries used in the ionization custom element, a header for the IPR info file, and forward declarations of the main ionization routine `ionization_routine_9_err` and the IPR info file routine `ionization_routine_9_ter`. The header file is referenced here so that it can be used by the main element code.

```
#include <stdio.h>
#include <math.h>
#include "elem.h"
#include <iostream>
#include <vector>
#include <limits>
#include <string>
#include <algorithm>
#define STEPNAME "IPRInfo"
#define STEPMAJ  1
#define STEPMIN  00
#include "ionization_routine_infostructures.h"

using namespace std;

static double ionization_routine_9_err(double t, double dt, const double *xstart, const
↪  double *xend, const double *xerr, void *vinfo);
static void ionization_routine_9_ter(void *vinfo);
```

Parameter	Description
ID	Particle ID
n	Number of elementary particles this macro-particle represents
tend	Simulation time at the END of the current timestep
G	Primary electron Lorentz factor
eGBr_x, eGBr_y, eGBr_z	Primary electron normalized momentum components
eInitialEnergy	Primary electron kinetic energy (eV)
ICS	Ionization cross section (m^2)
P	Ionization probability
eRand	Random number for determining whether an ionization event occurs
seRand	Random number for determining the secondary electron energy W
seG	Secondary electron Lorentz factor
seBr_x, seBr_y, seBr_z	Secondary electron normalized velocity (β) components
seGBr_x, seGBr_y, seGBr_z	Secondary electron normalized momentum ($\gamma\beta$) components
gasG	Target gas particle Lorentz factor
gasBr_x, gasBr_y, gasBr_z	Target gas particle normalized velocity components
gasEnergy	Target gas particle kinetic energy (eV)
ionRand	Random number for determining ion kinetic energy
ionEnergy	Ion kinetic energy (eV)
ionG	Ion Lorentz factor
ionBr_x, ionBr_y, ionBr_z	Ion normalized velocity components
ionGBr_x, ionGBr_y, ionGBr_z	Ion normalized momentum components
ionWr_x, ionWr_y, ionWr_z	Ion position components
eFinalEnergy	Scattered electron kinetic energy (eV)
eBrFinal_x, eBrFinal_y, eBrFinal_z	Scattered electron normalized velocity components
eGBrFinal_x, eGBrFinal_y, eGBrFinal_z	Scattered electron normalized momentum components

TABLE 20: List of ionization parameters to be written to IPR info file.

7.0.7 INITIALIZATION ROUTINE

User-Defined Parameters

The custom element uses an initialization routine to define all parameters, both global and ion-specific, before calling the main ionization routine. To do this, the element first ensures that the correct number of function arguments (9) are set when the ionization custom element is called in the input file. If not, the simulation is terminated and the resulting error message shows the correct syntax. Note that the element coordinate system (ECS) specification is not counted towards the number of elements, as there are multiple ways to define the coordinate system in a GPT element and each require a different number of arguments. Thus, the custom element uses `gptbuildECS` to determine the type of ECS specification.

Each function argument is read as a `string` or `double` using the GPT functions `gptgetargstring` and `gptgetargdouble`. The ion substructure used by the custom element is decided by the second and third arguments, `gasname` and `orbitalname`. The possible gas and orbital name combinations are listed in Table 21. Note that in the case of "H_2" or "He", there is no specific orbital name, so anything can be put for the orbital name specification. However, an orbital name specification must still be included. An orbital name of "default" will choose the orbital with the lowest ionization energy.

The `electronSet`, `ionSet`, `secondaryelectronSet,` and `scatteredelectronSet` arguments are particle set names and are read as strings. The electron and ion particle set names are required, while the secondary and scattered electron particle set names are optional. Providing a name for the secondary and/or scattered electron particle sets their respective switches to ON. A non-string name turns the switches to OFF. To turn on the IPR info file routine, the 8th argument must be a string of the form "<filename>.gdf" (such as "IPRInfoFilename.gdf").

```
void ionization_routine_9_init(gptinit *init)
{
        gptbuildECS(init);

        int numarg = gptgetargnum(init);
        if (numarg !=9)
                gpterror("Syntax: %s(ECS, L, gasname, orbitalname, electronSet, ionSet, 0
                ↪  [or secondaryelectronSet], 0 [or scatteredelectronset], 0 [or
                ↪  IPRInfoFilename.gdf], gasdensity [or gasdensity.gdf])\n",
                ↪  gptgetname(init));
```

```
const char *gasname = gptgetargstring(init, 2);
const char *orbitalname = gptgetargstring(init,3);
ionization_routine_9_info *info = nullptr;
if (!strcmp(gasname, "H2")) info = new H2Info;
if (!strcmp(gasname, "He")) info = new HeInfo;
if (!strcmp(gasname, "CO"))
{
        if( !strcmp(orbitalname, "5sigma") || !strcmp(orbitalname, "default") )
        ↪  info = new COInfo_5sigma;
        else if(!strcmp(orbitalname, "1pi")) info = new COInfo_1pi;
        else if(!strcmp(orbitalname, "4sigma")) info = new COInfo_4sigma;
        else if(!strcmp(orbitalname, "3sigma")) info = new COInfo_3sigma;
        else gpterror("Invalid orbital name for CO: %s.~Choices are: 5sigma
        ↪  (default), 1pi, 4sigma, 3sigma", orbitalname);
}
if (!strcmp(gasname, "CH4"))
{
        if( !strcmp(orbitalname, "1t2") || !strcmp(orbitalname, "default") ) info
        ↪  = new CH4Info_1t2;
        else if(!strcmp(orbitalname, "2a1")) info = new CH4Info_2a1;
        else gpterror("Invalid orbital name for CH4: %s.~Choices are: 1t2
        ↪  (default), 2a1", orbitalname);
}
if (!strcmp(gasname, "CH3"))
{
        if( !strcmp(orbitalname, "1a2") || !strcmp(orbitalname, "default") ) info
        ↪  = new CH3Info_1a2;
        else if( !strcmp(orbitalname, "1e")) info = new CH3Info_1e;
        else if( !strcmp(orbitalname, "2a1")) info = new CH3Info_2a1;
        else gpterror("Invalid orbital name for CH3: %s.~Choices are: 1a2
        ↪  (default), 1e, 2a1", orbitalname);
}
if (info == nullptr)
        gpterror("Unknown Gas Species; %s", gasname);

if(gptgetargtype(init,6)==GPTTYPE_STRING) info->SESwitch = 1; else info->SESwitch
↪  = 0;
if(gptgetargtype(init,7)==GPTTYPE_STRING) info->ScatESwitch = 1; else
↪  info->ScatESwitch = 0;
if(gptgetargtype(init,8)==GPTTYPE_STRING) info->IPRInfoSwitch = 1; else
↪  info->IPRInfoSwitch = 0;

info->axis = init->axis;
gptconcattransform(&info->tf, &init->axis->a, &init->e);
```

```
info->L = gptgetargdouble(init, 1);
info->set_electrons = parset_handle(gptgetargstring(init, 4),
↪  create_if_not_present);
info->set_gas = parset_handle(gptgetargstring(init, 5), create_if_not_present);

if(info->SESwitch == 1)
{
        info->set_secondaryelectrons = parset_handle(gptgetargstring(init, 6),
        ↪  create_if_not_present);
        gptwarning("Secondary Electrons ON\n");
}
else
{
        gptwarning("Secondary Electrons OFF\n");
}

if(info->ScatESwitch ==1)
{
        info->set_scatteredelectrons = parset_handle(gptgetargstring(init, 7),
        ↪  create_if_not_present);
        gptwarning("Scattered Electrons ON\n");
}
else
{
        gptwarning("Scattered Electrons OFF\n");
}

if(info->IPRInfoSwitch==1)
{
        info->IPRInfoFileName = gptgetargstring(init,8);

        const char *filename = gptgetargstring(init,8);
        gptwarning("Writing IPR Info to %s", filename);
}
```

Gas Density

The 9$^{\text{th}}$ argument decides the gas density calculation: If the 9$^{\text{th}}$ argument is a `double`, then the gas density is constant throughout the specified ionization region (e.g. a `gasdensity` of "10^11" will set a uniform gas density of 10^{11} m^2). The density can also be calculated from a data file containing 1-D density data with points (z, ρ) with $z \in [z_{\min}, z_{\max}]$. In this case, the

Gas name (`gasname`)	Orbital name (`orbitalname`)	Ionization Energy (eV)
H_2 ("H2")	N/A	15.43
He ("He")	N/A	24.59
CO ("CO")	5σ ("5sigma" or "default")	14.01
	1π ("1pi")	17.66
	4σ ("4sigma")	21.92
	3σ ("3sigma")	41.92
CH_4 ("CH4")	$1t_2$ ("1t2" or "default")	14.25
	$2a_1$ ("2a1")	25.73
CH_3 ("CH3")	$1a_2$ ("1a2" or "default")	9.84
	1e ("1e")	15.64
	$2a_1$ ("2a1")	24.57

TABLE 21: List of supported gas and orbital names with their respective ionization energies.

9^{th} argument is a filename of the form "<filename>.gdf". The density is then extrapolated to a 2-D cylindrically symmetric density map using a natural cubic spline:

$$\begin{aligned} \rho_r(r,z) &= -\frac{1}{2} r \rho_z'(z) \\ \rho_z(r,z) &= \rho_z(z) - \frac{1}{4} r^2 \rho_z''(z) \\ \rho''(z_{\min}) &= \rho''(z_{\max}) = 0 \end{aligned} \tag{22}$$

```
if(gptgetargtype(init,9)==GPTTYPE_STRING)
{
          const char *filename = gptgetargstring(init,9) ;
        gptwarning("Reading vacuum data from %s", filename);
        struct gdff ingdff ;
          gdfsrinit( filename, &ingdff ) ;
        struct gdfmem gm ;
          gdfrmem( &ingdff, &gm, GDFR_ABORTONERROR | GDFR_READARRAYS ) ;

          struct gdfdata *ds=gptinputgetgroup(&gm,filename,"z") ;
        int points ;
          info->z=gptinputdoublearray(ds,"z",&points) ;
          info->densityPoints = points ;
```

```
            info->fz = gptinputdoublearraypoints(ds,"density",points) ;

            for(int i=0 ; i<(points-1) ; i++ )
              if( info->z[i]>=info->z[i+1] )
                        gpterror( "%s: z array must be in ascending order\n",
                        ↪  filename ) ;

            info->fz2 = (double *)gptmalloc(points*sizeof(double)) ;
            if( spline( info->z, info->fz, info->fz2, points, NATSPLINE, NATSPLINE
            ↪  ) )
                      gpterror( "%s: Error calculating spline\n", filename )
                      ↪  ;
}
else
{
        info->gasdensity = gptgetargdouble(init, 9);
}

double density = info->gasdensity ;
if( info->densityPoints!=0 )
{
        double z = 0.5*(ecs_rstart[2]+ecs_rend[2]) ;
        if( splint( info->z, info->fz, info->fz2, info->densityPoints, z,
        ↪  &density ) )
                       continue ;
}
```

Callback Functions

Callback functions are used to define where and when routines in the main element code are called in the GPT kernel. The first callback function, `odeadderrfunction`, tells the GPT kernel that the main ionization routine, `ionization_routine_9_err`, will be called during every simulation timestep. The "`_err`" suffix means that if the return value of `ionization_routine_9_err` is greater than 1, the timestep fails and is retried with a shorter timestep, a feature which is important in the two Monte-Carlo routines described later. The second callback function, `gptaddmainfunction`, with the specification `GPTMAINFNC_TER`, tells the GPT kernel that the IPR info routine, `ionization_routine_9_ter`, will be called upon completion of the simulation.

```
odeadderrfunction(ODEFNC_USR, ionization_routine_9_err, info);
gptaddmainfunction(GPTMAINFNC_TER, ionization_routine_9_ter, info);
```

7.0.8 MAIN IONIZATION ROUTINE

Ionization Conditions

During every timestep, the main ionization routine loops over all particles present in the simulation to determine its eligibility for ionization. To do this, the routine accesses the `pars` array, which is indexed from 0 to `numpar`, and checks the attributes (i.e., members) of all particles currently present in the simulation. The particle is eligible for ionization if it meets the following conditions:

1. The particle is present in the simulation and will not be removed during this timestep. Any GPT element that removes a particle (such as `zminmax` or `rmax`) will set the attribute `alive` to false and/or the `tokill` attribute to true.

2. The particle is an electron. The electron must be in the primary, secondary, or scattered electron particle sets. If the particle is in any other particle set, it is not eligible for ionization.

3. The coordinate system of the particle matches the coordinate system of the ionization region. The coordinate system of the particle is handled by the `axis` attribute and is set during initialization.

4. The particle will not leave the ionization region during the timestep. To test this, a displacement vector is calculated from the position of the particle at the start and end of the timestep. The particle is eligible to ionize if the *center* of the displacement vector is within the ionization region. Of course, it is possible for a particle to enter and exit the ionization region during a single timestep, especially if the timestep is large, or if the general motion of the particle is complex. Thus, using the center of the displacement vector introduces uncertainty in the simulation. However, because the probability of a single electron to ionize is extremely small, it is unlikely that this uncertainty will be significant for most applications.

5. The kinetic energy of the particle T_e is above the ionization energy I_i of the target gas particle and the calculated ionization cross section is positive (which is always true provided $T_e > I_i$). The kinetic energy is calculated using the Lorentz factor at

the end of the timestep instead of the interpolated value. Because the kinetic energy could change substantially during a timestep, especially in complex simulations, this approximation does introduce some uncertainty, though as with the previous ionization condition, the uncertainty is negligible for most applications.

```
static double ionization_routine_9_err(double t, double dt, const double *xstart, const
↪  double *xend, const double *xerr, void *vinfo)
{
        double tstart = t;
        double tend = t + dt;

        ionization_routine_9_info *info = (struct ionization_routine_9_info *)vinfo;

        for (int i = 0; i < numpar; i++)
        {
                if (pars[i].alive == false || pars[i].tokill == true)
                {
                        continue;
                }

                if (pars[i].set != info->set_electrons && pars[i].set !=
                ↪  info->set_secondaryelectrons && pars[i].set !=
                ↪  info->set_scatteredelectrons)
                {
                        continue;
                }
                if (pars[i].axis != info->axis)
                {
                        continue;
                }

                double ecs_rstart[3], ecs_rend[3];
                const double *rstart = &xstart[pars[i].odeoff];
                const double *rend = &xend[pars[i].odeoff];
                gpttoUCS(&info->tf, rstart, ecs_rstart);
                gpttoUCS(&info->tf, rend, ecs_rend);

                if (fabs(ecs_rstart[2] + ecs_rend[2]) > info->L)
                {
                        continue;
                }

                double G = pars[i].G;
```

```
double eInitialEnergy = (G - 1.0)*gpt_me*pow(gpt_c, 2) / (-gpt_qe);
double eGBr[3];
for (int j = 0; j < 3; j++)
{
        eGBr[j] = pars[i].GBr[j];
}
double eBr[3];
for (int j = 0; j < 3; j++)
{
        eBr[j] = eGBr[j] / G;
}
double dr[3];
for (int j = 0; j < 3; j++)
{
        dr[j] = rend[j] - rstart[j];
}
double d = gptVECLEN(dr);
if (eInitialEnergy < info->get_B())
{
        continue;
}
if (info->get_ICS(info->get_A1(),info->get_A2(),info->get_B(),G) < 0.0)
{
        continue;
}
```

Monte Carlo Algorithm For Calculating the Ionization Probability

A Monte Carlo algorithm is used to determine if the electron macro-particle ionizes. The routine starts by calculating the ionization probability `P` using Eq. 1. If `P` is greater than one for *any* electron macro-particle, then the timestep fails and is retried with smaller timesteps until `P` is less than one for all particles. While it is unlikely that `P` can be greater than one, since the ionization cross section for a single electron is tiny, `P` can become greater than one if the electron macro particle represents an extremely large number of electrons (i.e., `pars[i].n` is large) or if the gas density is high.

Once `P` is less than one for all particles, a random number between 0 and 1, `eRand`, is chosen using the built-in function `dblpprand()`, which is optimized for parallel computing. If `eRand` is greater than `P`, then the particle does not ionize and the simulation continues to the next particle. If the particle does ionize, then parameters for the ion are set and the routine continues to the secondary electron routine to calculate the secondary electron and

ion energies.

```
double P = density * d * info->get_ICS(info->get_A1(),info->get_A2(),info->get_B(),G) *
↪  pars[i].n;
if(P>1.0)
{
        info->clearAllTempVectors();
        return 20;
}

double eRand = dblpprand();
if (eRand > P)
{
        continue;
}

double ionMass = info->get_mass();
double ionCharge = info->get_charge();
double ionRadius = info->get_radius();
double ionTime = tend;
```

Monte Carlo Algorithm For Secondary Electron and Ion Energies

To calculate the secondary electron and ion energies, the custom element uses a Monte Carlo algorithm (described in section 5.4.3), with a precision of 500 integration steps. The maximum secondary electron energy is defined as $E_{max} = 0.75(T - B)$, as the SEDCS becomes unphysical past this energy. Because the Maxwellian distribution is a distribution of speeds, not energies, the most probably ion energy of 4.0 eV is converted to the most probable ion speed before the MC routines. Each MC routine consists of selecting a random number between 0 and 1 and a do-while loop that determines the closest value of $g(E)$ to the random number. As a safety precaution, the do-while loop terminates the simulation after 500 steps, though it is extremely unlikely that this error would occur because the secondary electron energy is likely to be low.

```
double seMass = gpt_me;
double seCharge = gpt_qe;
double seRadius = 2e-9;
double seTime = tend;

const int nsteps = 500;
```

```
double Emax = 0.75*(eInitialEnergy - info->get_B());
double FEsumtoEmax = info->get_FE_SEDCS(nsteps, Emax, eInitialEnergy);

double mostProbIonEnergy = 4.0;
double ionRestMass = ionMass*gpt_c*gpt_c/(-gpt_qe);
double a = sqrt((gpt_c*gpt_c*0.5) * (1-(ionRestMass/(mostProbIonEnergy+ionRestMass)) *
↪  (ionRestMass/(mostProbIonEnergy+ionRestMass))));

double mostProbIonSpeed=sqrt(2)*a;
double Vmax = 5.0*mostProbIonSpeed;
double FEsumtoVmax = info->get_FE_Maxwell(nsteps,Vmax,a);

double seRand = dblpprand();
double W;
double gEstepsize = Emax/nsteps;
int sej = 0;
do
{
        double gE = info->get_FE_SEDCS(nsteps, gEstepsize + sej*gEstepsize,
        ↪  eInitialEnergy) / FEsumtoEmax;

        if(seRand<gE)
        {
                W = gEstepsize + sej*gEstepsize;
                break;
        }
        sej++;

        if(sej>nsteps) gpterror("Possible infinite loop in secondary electron energy
        ↪  routine, sej exceeded %i",nsteps);
}while(true);

double ionRand = dblpprand();
double ionSpeed = 0.0;
double gEstepsizeMaxwell = Vmax / nsteps;
int ionj = 0;
do
{
    double gEMaxwell = info->get_FE_Maxwell(nsteps, gEstepsizeMaxwell +
    ↪  ionj*gEstepsizeMaxwell, a) / FEsumtoVmax;

        if (ionRand<gEMaxwell)
        {
                ionSpeed = gEstepsizeMaxwell + ionj*gEstepsizeMaxwell;
```

```
                break;
        }
        ionj++;

        if(ionj>nsteps) gpterror("Possible infinite loop in ion energy routine, ionj
        ↪  exceeded %i", nsteps);
}while(true);
```

Kinematic Parameters

Once the energies of the secondary electron and ion energies are determined, the remaining kinematic parameters of all particles involved in ionization can be calculated. The directions of the secondary electron, ion, and gas molecule are randomized. For each of these, a vector whose components are random numbers between 0 and 1 is created. The sign of each component is also chosen at random: if a random number is below 0.5, then the component is negative, otherwise it is positive. The vectors are normalized by dividing each component by the magnitude of the vector using the built-in `gptVECLEN` function.

Once the unit vectors are known, the normalized velocity of each particle are constructed. The normalized velocity vector is constructed based on the Lorentz factor γ:

$$\vec{\beta} = \left|\vec{\beta}\right| \hat{\beta} \tag{23}$$
$$\left|\vec{\beta}\right| = \frac{\gamma^2 - 1}{\gamma^2}$$

The normalized momentum vectors are simply the product of the Lorentz factor and the normalized velocity. The scattered electron parameters are chosen based momentum and energy conservation.

Once kinematic parameters of all particles are known, the ion is placed at a random location along the trajectory of the primary electron in the timestep, defined as the displacement between the electron's initial and final positions. The secondary and scattered electrons are then placed at the ion's position.

```
double ionBrmag = ionSpeed/gpt_c;
double ionG = 1/sqrt(1-(ionBrmag*ionBrmag));
double ionEnergy = (ionG-1)*ionRestMass;

double seG = W*(-gpt_qe) / (gpt_me*pow(gpt_c, 2)) + 1.0;
```

```
double serndBr[3];
double serndmag1 = dblpprand();
double serndmag2 = dblpprand();
double serndmag3 = dblpprand();
double serndmag[3] = { serndmag1, serndmag2, serndmag3 };
for (int j = 0; j < 3; j++)
{
        double rndsign = dblpprand();
        double brsign;
        if (rndsign < 0.5)
        {
                brsign = -1.0;
        }
        else
        {
                brsign = 1;
        }
        serndBr[j] = serndmag[j] * brsign;
}

double gasrndBr[3];
double gasrndmag1 = dblpprand();
double gasrndmag2 = dblpprand();
double gasrndmag3 = dblpprand();
double gasrndmag[3] = { gasrndmag1, gasrndmag2, gasrndmag3 };
for (int j = 0; j < 3; j++)
{
        double rndsign = dblpprand();
        double brsign;
        if (rndsign < 0.5)
        {
                brsign = -1;
        }
        else
        {
                brsign = 1;
        }
        gasrndBr[j] = gasrndmag[j] * brsign;
}

double seunitBr[3], gasunitBr[3];
for (int j = 0; j < 3; j++)
{
        seunitBr[j] = serndBr[j] / gptVECLEN(serndBr);
}
```

```
for (int j = 0; j < 3; j++)
{
        gasunitBr[j] = gasrndBr[j] / gptVECLEN(gasrndBr);
}
double seBrmag = sqrt(1 - (1 / (seG*seG)));

double seBr[3];
for (int j = 0; j < 3; j++)
{
        seBr[j] = seunitBr[j] * seBrmag;
}

double seGBr[3];
for (int j = 0; j < 3; j++)
{
        seGBr[j] = seG*seBr[j];
}

double gasBrmag = info->get_Brmag();
double gasG = 1 / sqrt(1 - (gasBrmag*gasBrmag));

double gasBr[3];
for (int j = 0; j < 3; j++)
{
        gasBr[j] = gasunitBr[j] * gasBrmag;
}

double gasEnergy = (gasG - 1.0)*(info->get_mass()+gpt_me)*gpt_c*gpt_c / (-gpt_qe);

double eP[3], eunitBr[3], seP[3];
double Pconvfactor = 5.344286e-28;
double eBrmag = gptVECLEN(eBr);
for (int j = 0; j < 3; j++)
{
        eunitBr[j] = eBr[j] / eBrmag;
        eP[j] = gpt_me*gpt_c*sqrt(G*G - 1.0)*Pconvfactor*eunitBr[j];
        seP[j] = gpt_me*gpt_c*sqrt(seG*seG - 1.0)*Pconvfactor*seunitBr[j];
}

double gasP[3];
for (int j = 0; j < 3; j++)
{
        gasP[j] = info->get_mass()*gpt_c*sqrt(gasG*gasG - 1.0)*Pconvfactor*gasunitBr[j];
}
```

```
double totEnergy = eInitialEnergy + gasEnergy - W - info->get_B();

double ionrndBr[3];
double ionrndmag1 = dblpprand();
double ionrndmag2 = dblpprand();
double ionrndmag3 = dblpprand();
double ionrndmag[3] = { ionrndmag1, ionrndmag2, ionrndmag3 };
for (int j = 0; j < 3; j++)
{
        double rndsign = dblpprand();
        double brsign;
    if (rndsign < 0.5)
        {
                brsign = -1.0;
        }
        else
        {
                brsign = 1;
        }
        ionrndBr[j] = ionrndmag[j] * brsign;
}

double ionunitBr[3];
for (int j = 0; j < 3; j++)
{
        ionunitBr[j] = ionrndBr[j] / gptVECLEN(ionrndmag);
}

double ionBr[3];
for (int j = 0; j < 3; j++)
{
        ionBr[j] = ionBrmag*ionunitBr[j];
}

double ionP[3];
for (int j = 0; j < 3; j++)
{
        ionP[j] = ionMass*gpt_c*sqrt(ionG*ionG - 1.0)*Pconvfactor*ionunitBr[j];
}

double ionGBr[3];
for (int j = 0; j < 3; j++)
{
        ionGBr[j] = ionBr[j] * ionG;
}
```

```
double ionWr[3];
double lambda = dblpprand();
for (int j = 0; j < 3; j++)
{
        ionWr[j] = rstart[j] + lambda*dr[j] + (1.0 - lambda)*(tend - tstart)*ionBr[j] *
        ↪  gpt_c;
}

double seWr[3];
for (int j = 0; j < 3; j++)
{
        seWr[j] = ionWr[j];
}

double eFinalEnergy = totEnergy - ionEnergy;
double eGFinal = (eFinalEnergy*(-gpt_qe)) / (gpt_me*gpt_c*gpt_c) + 1.0;
double eBrFinalmag = sqrt(1 - (1 / (eGFinal*eGFinal)));

double ePFinal[3];
for (int j = 0; j < 3; j++)
{
        ePFinal[j] = eP[j] + gasP[j] - ionP[j] - seP[j];
}
double ePFinalmag = gptVECLEN(ePFinal);
double ePFinalunit[3];
for (int j = 0; j < 3; j++)
{
        ePFinalunit[j] = ePFinal[j] / ePFinalmag;
}

double eBrFinal[3];
for (int j = 0; j < 3; j++)
{
        eBrFinal[j] = eBrFinalmag*ePFinalunit[j];
}

double eGBrFinal[3];
for (int j = 0; j < 3; j++)
{
        eGBrFinal[j] = eGFinal*eBrFinal[j];
}
```

Particle Sets and IPR Info Parameters

The ion is added to the ion particle set using the `gptaddparmnqnartid` built-in function. It is assigned a unique ID so that it can be differentiated from other particles. If the secondary and scattered electron switches are turned on, then these particles are added to their respective particle sets. The `parset_handle` class ensures that each particle is added to the correct particle set. Afterwards, the number of macro particles the electron macroparticle represents, `pars[i].n` is reduced by one.

If the IPR info file switch is turned on, then ionization parameters are added to the end of the temporary vectors. If the timestep does not fail for any particle, then the temporary vectors are added to the main vectors and are then cleared for the next timestep.

```
parset_handle ionSet = info->set_gas;
gptaddparmqnartid(&*ionSet, ionWr, ionGBr, ionMass, ionCharge, 1.0, axis_handle(),
↪  ionRadius, ionTime, 0);

if(info->SESwitch == 1)
{
        parset_handle seSet = info->set_secondaryelectrons;
        gptaddparmqnartid(&*seSet, seWr, seGBr, seMass, seCharge, 1.0, axis_handle(),
        ↪  seRadius, seTime, 0);
}

if(info->ScatESwitch == 1)
{
        parset_handle scatESet = info->set_scatteredelectrons;
        gptaddparmqnartid(&*scatESet, seWr, eGBrFinal, seMass, seCharge, 1.0,
        ↪  axis_handle(), seRadius, seTime, 0);
}

double *parsnmacro = (double *)&(pars[i].n);
*parsnmacro = pars[i].n - 1.0;

if(info->IPRInfoSwitch==1)
{
        info->ID_TEMP.push_back(pars[i].ID);
        info->n_TEMP.push_back(pars[i].n);
        info->tend_TEMP.push_back(tend);
        ...
}
} //end of for-loop over all particles
```

```
if(info->IPRInfoSwitch==1)
{
        info->addAllTempVectors();
        info->clearAllTempVectors();
}

return 0;
}
```

7.0.9 IPR INFO FILE ROUTINE

After a simulation successfully completes with the IPR info file switch turned on, the IPR vectors are written to a GDF file specified by `IPRInfoFileName`. To do this, the custom element uses the routine `ionization_9_ter` called at the end of a simulation. The routine first determines the size (i.e., length) of the IPR vectors by calling `info->tend.size()`. In the rather unlikely event that the size of the vector is greater than the maximum integer `INT_MAX` (about 2.1×10^9), the routine terminates the simulation to prevent an overflow error.

To write the vectors to a GDF file, the custom element uses a similar procedure to the built-in writesteps() function. The GDF routines are initialized by invoking the built-in `gdff` structure and calling `gdfswinit()` and `gdfwmainhead()` to define the file name and header. The file is then filed with data from from each IPR vector via `gdfwarr` or `gptoutputdoublearray` depending on if the file was originally empty.

Once the simulation completes, the IPR info file can be opened similarly to the simulation result file in GPT or converted to a text file using the built-in `gdf2a` program in order to view the IPR info data.

```
static void ionization_routine_9_ter(void *vinfo)
{
        struct ionization_routine_9_info *info = (struct ionization_routine_9_info
        ↪  *)vinfo;

        if(info->IPRInfoSwitch!=1) return;

        if(info->tend.size() > INT_MAX) gpterror("Overflow in IPRInfo: %i >
        ↪  INT_MAX",info->tend.size());

        int size = static_cast<int>(info->tend.size());
```

```
        if( !info->IPRInfoFileName.empty() )
            {
                struct gdff gdff ;
                gdfswinit(info->IPRInfoFileName.c_str(), nullptr, &gdff ) ;
                gdfwmainhead( &gdff, STEPNAME, STEPMAJ, STEPMIN, "", 0, 0 ) ;

                gdfwarr(&gdff,"ID",        size,t_dbl,&info->ID.front());
                gdfwarr(&gdff,"nmacro",        size,t_dbl,&info->n.front());
                gdfwarr(&gdff,"tend",        size,t_dbl,&info->tend.front());
                ...

                gdfwclose(&gdff) ;
        }
        else
        {
                gptoutputdoublegroup("IPRInfo",0) ;

                gptoutputdoublearray("ID",        &info->ID.front(),        size) ;
                gptoutputdoublearray("nmacro",        &info->n.front(),        size) ;
                gptoutputdoublearray("tend",        &info->tend.front(),        size) ;
                ...

                gptoutputendgroup() ;
        }
}
```

VITA

J. T. Yoskowitz
Department of Physics
Old Dominion University
Norfolk, VA 23529

Education

- Ph.D. Accelerator Physics, Old Dominion University, Norfolk, Virginia, May 2022.
- M.S. Accelerator Physics, Old Dominion University, Norfolk, Virginia, January 2018.
- B.S. Physics, Union College, Schenectady, New York, June 2016.

Recent Publications

- **J. T. Yoskowitz** et al., "Improving the operational lifetime of the CEBAF photo-gun by anode biasing", Proc. 12th In. Particle Accelerator Conf. (IPAC '21), Campinas, SP, Brazil (Virtual Conference), May 2021.
- **J. T. Yoskowitz** et al., "Simulating electron impact ionization using a general particle tracer (GPT) custom element", Proc. 12th In. Particle Accelerator Conf. (IPAC '21), Campinas, SP, Brazil (Virtual Conference), May 2021.
- **J. T. Yoskowitz** et al., "New simulations for ion-production and back-bombardment in GaAs Photo-guns", in Proc. PSTP 2019, vol. 379, p. 040, 2020. doi: 1022323/1.379.0040
- Y. Wang, **J. T. Yoskowitz** et al., "Thermal emittance and lifetime of alkali-antimonide photocathodes grown on GaAs and molybdenum substrates evaluated in a -300 kV DC photo-gun", Phys. Rev. Accel. Beams, vol. 23, no. 10, p. 103401, 2020. doi: 10.1103/PhysRevAccelBeams.23.103401
- C. Hernandez-Garcia, **J. T. Yoskowitz**, et al., "Compact -300 kV DC inverted insulator photogun with biased anode and alkali-antimonide photocathode", Phys. Rev. Accel. Beams, vol. 22, no. 11, p. 113401, 2019. doi: 10.1103/PhysRevAccelBeams.22.113401

Typeset using LaTeX.