TUM School of Natural Sciences

TUM

Technische Universität München

Towards the Creation of an Electron–Positron Plasma: Advanced Single-Particle Simulations for APEX

Stefan Nißl

Abstract

The vast majority of the universe is in the plasma state. Virtually all of it, in turn, consists of electrons and ions. In the vicinity of intense gamma-ray sources, however, a more exotic type of plasma can form, known as electron–positron pair plasma. In a pair plasma, the two species of particles involved are oppositely charged but have the same mass. This mass equality is the reason why pair plasmas are much less susceptible to instabilities than their electron–ion counterparts. For this reason, it is of great interest to study such a plasma.

The APEX collaboration has the ambitious but achievable goal of creating the first magnetized low-energy electron–positron pair plasma in the laboratory. Magnetic dipole fields and stellarators are particularly well suited to confine such plasmas, both of which have different properties, which is why APEX follows both paths. Since the availability of positrons is very limited, these devices are table-top sized to reduce the number of positrons required to achieve a sufficiently small Debye length, a prerequisite for a plasma. Nonetheless, positrons from a reactor-based source must first be accumulated in several stages before an intense pulse can be injected into a confinement device. Electrons, on the other hand, are extremely cheap and readily available via thermionic emission.

As mentioned above, positrons are very precious, so special care must be taken to ensure that they can be injected into the confinement devices with as little loss as possible. For this reason, Proto-APEX was built as a test device, which successfully demonstrated that a nearly lossless transfer of positrons from the incoming beamline into the magnetic dipole field of a supported permanent magnet via the E×B drift is possible. Due to the time-limited positron beam times and the limited diagnostic capabilities of Proto-APEX, numerical simulations are a key part of understanding the behavior of positrons in the confinement devices in detail.

For this thesis, single-particle simulations were performed to investigate, among other things, how the radial deposition of positrons can be manipulated during injection, what limits the confinement time and how it can be extended, and how the trap can be emptied efficiently. The simulations were performed with a previously developed code that can propagate particles through electric potentials and analytically calculated magnetic fields. To remove the dependency on other programs, it has been extended here with its own electric potential solver, which supports arbitrary user-defined geometries on a regular grid as well as an octree data structure. The latter allows the geometry of the upcoming devices to be resolved with higher precision without the aggressive memory consumption scaling of a regular grid.

Zusammenfassung

Der weitaus größte Teil des Universums befindet sich im Plasmazustand. Praktisch alles davon besteht wiederum aus Elektronen und Ionen. In der Nähe von intensiven Gammastrahlenquellen kann sich jedoch auch eine exotischere Art von Plasma bilden, das so genannte Elektron–Positron-Paarplasma. In einem Paarplasma sind die beiden beteiligten Teilchenarten gegensätzlich geladen, haben aber die gleiche Masse. Diese Massengleichheit ist der entscheidende Faktor dafür, dass Paarplasmen im Vergleich zu Plasmen, die aus Elektronen und Ionen bestehen, viel weniger anfällig für Instabilitäten sind. Aus diesem Grund ist es von großem Interesse, ein solches Plasma genauer zu untersuchen.

Die APEX-Kollaboration hat das ehrgeizige, aber erreichbare Ziel, das erste magnetisierte niederenergetische Elektron–Positron-Paarplasma im Labor zu erzeugen. Für den Einschluss solcher Plasmen eignen sich vor allem magnetische Dipolfelder und Stellaratoren. Diese beiden Methoden haben unterschiedliche Vorteile, weshalb APEX beide Ansätze verfolgt. Da Positronen nur in sehr begrenzten Mengen zur Verfügung stehen, werden beide Geräte tischgroß sein, um die Anzahl der Positronen zu verringern, die benötigt werden, um eine ausreichend kleine Debye-Länge zu erreichen, was eine Voraussetzung für die Erzeugung eines Plasmas ist. Dennoch müssen Positronen aus einer reaktorbasierten Quelle zunächst in mehreren Stufen akkumuliert werden, bevor ein intensiver Puls in eine Einschlussvorrichtung injiziert werden kann. Im Gegensatz dazu stehen Elektronen durch thermionische Emission in praktisch unbegrenzten Mengen sofort zur Verfügung.

Wie bereits erwähnt, sind Positronen sehr kostbar, so dass besonders darauf geachtet werden muss, dass diese möglichst verlustfrei in die Einschlussvorrichtungen injiziert werden können. Aus diesem Grund wurde Proto-APEX als Testvorrichtung gebaut, mit der erfolgreich gezeigt werden konnte, dass ein nahezu verlustfreier Transfer von Positronen von der eingehenden Strahlführung in das magnetische Dipolfeld eines auf einem Stab befestigten Permanentmagneten über die E×B-Drift möglich ist. Aufgrund der zeitlich begrenzten Verfügbarkeit von Positronen und der beschränkten diagnostischen Möglichkeiten von Proto-APEX sind numerische Simulationen ein wichtiger Bestandteil, um das Verhalten von Positronen in den Einschlussvorrichtungen im Detail zu verstehen.

Für diese Arbeit wurden Einzelteilchensimulationen durchgeführt, um unter anderem zu untersuchen, wie die radiale Verteilung der Positronen während der Injektion manipuliert werden kann, welche Faktoren die Einschlusszeit begrenzen und wie diese verlängert werden kann, und wie die Falle effizient geleert werden kann. Die Simulationen wurden mit einem zuvor entwickelten Code durchgeführt, der Teilchen durch elektrische Potentiale und analytisch berechnete Magnetfelder bewegen kann. Um weniger abhängig von anderen Programmen zu sein wurde die Software hier um einen eigenen Löser für elektrische Potentiale erweitert, der beliebige benutzerdefinierte Geometrien sowohl auf einem regulären Gitter als auch auf einer Octree-Datenstruktur unterstützt. Letztere erlaubt es, die Geometrien der zukünftigen Einschlussvorrichtungen mit höherer Genauigkeit aufzulösen, ohne die aggressive Skalierung des Speicherverbrauchs eines regulären Gitters.

Contents

Chapter 1

Introduction

In the universe, it is estimated that more than $99\,\%$ of the visible matter is in the plasma state [1], from stars to the interstellar medium. This state of matter, which consists of a mixture of charged particles, can also be observed on Earth e.g. as lightnings or as auroras, which are caused by the interaction of the solar winds with the magnetosphere. Plasmas also have a wide range of applications, ranging from plasma cutters, to extreme ultraviolet (EUV) light sources needed for state-of-the-art lithography processes, to fusion reactors that seek to generate energy by fusing atomic nuclei.

Plasmas usually consist of ionized atomic nuclei and free electrons, so the masses differ by at least a factor of 1836 as in the case of a hydrogen plasma, but there are also plasmas in which both species have the same mass, the so-called pair plasmas. These pair plasmas exist in relativistic jets, such as those emitted by pulsars or active galactic nuclei [2, 3, 4], where the enormous luminosity of gamma radiation interacting with the surrounding matter can produce large amounts of electron–positron pairs via pair production.

One of the properties that makes the study of pair plasmas so interesting is their extraordinary stability in parameter regimes of experimental interest [5, 6], which arise from the mass equality of the involved species. An example of this is the ion acoustic wave, broadly sketched in Figure 1.1. When a density fluctuations occurs in an electron–ion plasma, the electrons can quickly leave the region of elevated density, whereas the ions remain behind due to their greater inertia. The resulting charge separation creates an electric field, which pushes the ions away from the region of elevated density and the electrons back to the ions, but, due to their inertia, the ions overshoot, causing an oscillation that is damped by Coulomb collisions and Landau damping. Considering the same density fluctuation in an electron–positron plasma, no oscillation occurs, because electrons and positrons leave the region of elevated density simultaneously and therefore no electric fields are generated.

The goal of the APEX (A Positron–Electron eXperiment) collaboration is to create a magnetized, low-energy, electron–positron plasma in the laboratory for the first time. There are two magnetic confinement devices that are particularly
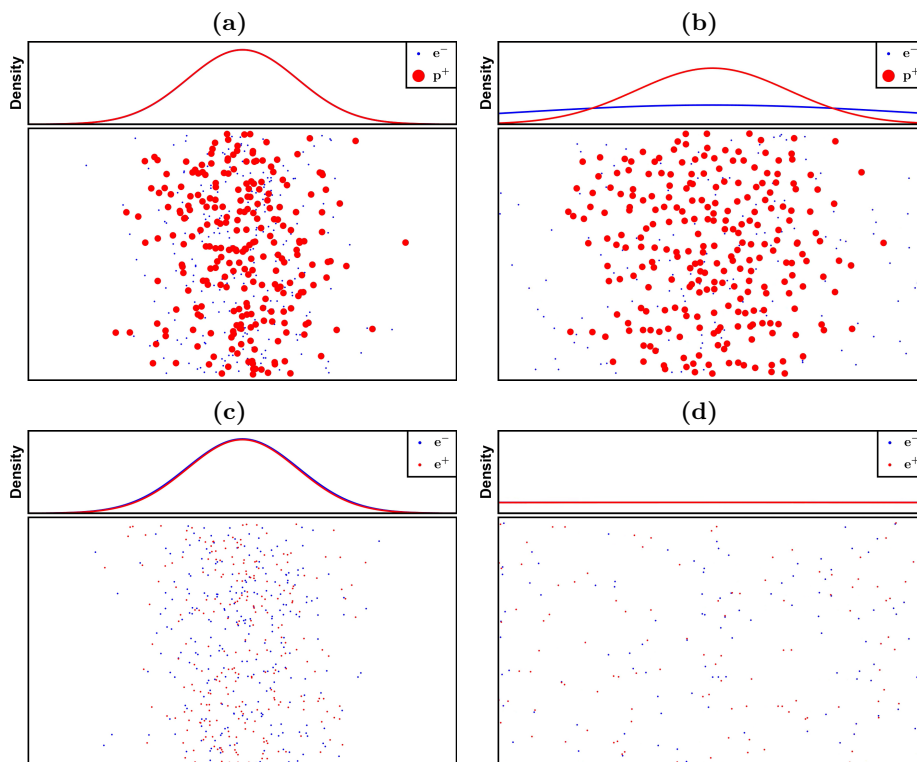
Figure 1.1: Qualitative illustration of the evolution of a density fluctuation in case of electrons and ions (a+b), and electrons and positrons (c+d). In this simple simulation, the magnetic field pointed from left to right and all species had the same initial temperature.

suitable for confining a neutral pair plasma: one is a magnetic dipole and the other is a stellarator. Both paths are pursued by the APEX collaboration with APEX-D, in which the magnetic dipole field is generated by a superconducting, levitating coil, and with EPOS (Electrons and Positrons in an Optimized Stellarator) [7], a tabletop-sized stellarator which also uses superconducting coils.

To create the conditions for a plasma, the Debye length must be much smaller than the size of the plasma. In case of a pair plasma, the Debye length λ_D is given by [8]:

$$\lambda_D = \sqrt{\frac{\epsilon_0 k_B T}{2ne^2}} \propto \sqrt{\frac{T}{n}} \tag{1.1}$$

with the vacuum permittivity ϵ_0, the Boltzmann's constant k_B, the temperature T, the density $n = n_{e^+} = n_{e^-}$, and the elementary charge e. For the target parameters of APEX-D and EPOS (volumes of about $10\,\text{L}$, a plasma size of about $8\,\text{cm}$, and temperatures lower than $1\,\text{eV}$), the required number of positrons is therefore in the order of $10^{10} - 10^{11}$ [9]. Currently, there is no source that can deliver this many positrons in the short time ($< 100\,\mu\text{s}$) required to fill the confinement devices. For this reason, it is planned to capture and accumulate positrons from a reactor-based source (see Chapter 2) in several stages

(discussed in detail in Chapter 3) before they are injected into one of the devices, as illustrated in the grand scheme of the APEX collaboration in Figure 1.2.
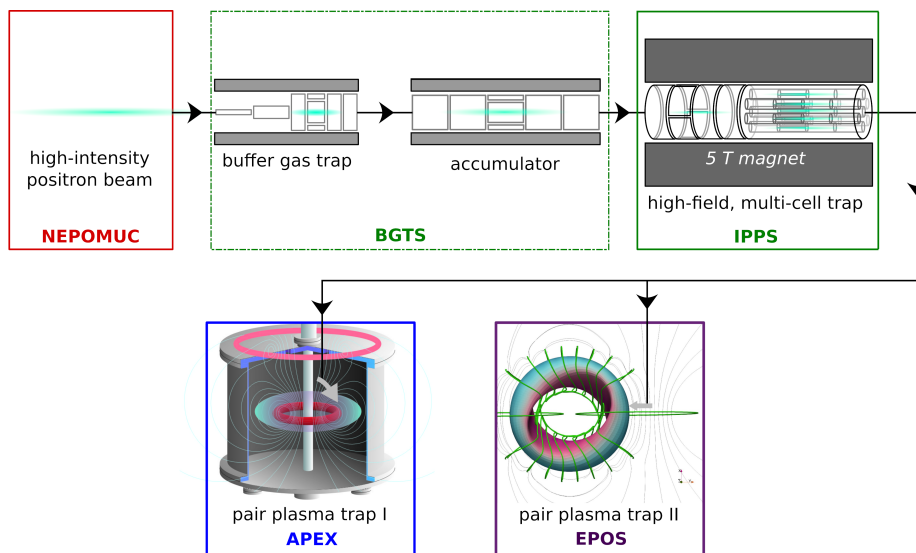
Figure 1.2: The grand scheme of the APEX collaboration. The positrons from a reactor-based source are accumulated in several stages to inject an intense pulse into either APEX (which will be called APEX-D in the following to better distinguish it from the collaboration) or EPOS (modified version of [10]).

It is one thing to achieve the necessary densities for a pair plasma, but care must also be taken to allow sufficient time to study the plasma before the positrons annihilate. Figure 1.3 gives an overview of the main processes that limit the lifetime of the plasma in the targeted range. One can see that three-body recombination, direct annihilation and positronium formation via radiative recombination do not have a significant detrimental effect on the positron lifetime. However, direct annihilation on residual gas molecules and atoms, like nitrogen and helium (which is used to cool the levitated, superconducting coil, see Section 4.2), and charge exchange with atomic hydrogen (outgassed from the stainless steel vacuum chamber [11]) will result in a limitation of the expected lifetime in the order of tens of minutes, if the positron energy is less than $10\,\mathrm{eV}$, which is still more than sufficient to investigate occurring plasma phenomena.

To date, experiments have been performed mainly with a predecessor device of APEX-D, called Proto-APEX (see Section 4.1). In this device, the magnetic dipole field is generated by a supported permanent magnet, and the positron beam is injected directly without accumulation, so no plasma densities could be achieved. Nevertheless, important milestones were achieved with Proto-APEX. Among other things, our team was able to inject the positron beam into the dipole field with an efficiency of $100\,\%$ [12], to hold these trapped positrons in the dipole field for more than one second [13], and to show that positrons can be injected into an already existing electron cloud [14, 15].
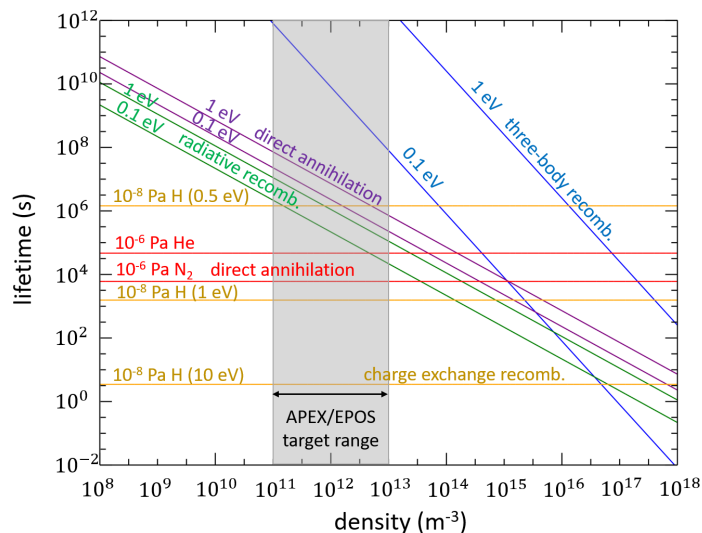
Figure 1.3: Overview of the main annihilation processes in a electron–positron plasma for different energies [10].

The focus of this dissertation is on single-particle simulations of positrons in Proto-APEX. These simulations are used to study injection, confinement, and ejection of positrons independently of the reactor cycle, as well as to gain insights that are not accessible experimentally or are very difficult to obtain. The simulations were performed with AlGeoJ, a program written in Java, which was previously developed in-house [16] and was extended here with an electrostatic field solver.

In the next chapter, the positron source used for APEX is described in more detail. The plan of how these positrons will be accumulated to produce an intense pulse is elaborated in Chapter 3. Chapter 4 focuses on the setup of Proto-APEX and the status of APEX-D. The simulation framework is presented in depth in Chapter 5, including the calculation of the electric and magnetic fields and the propagation of the particles in these fields. Chapter 6 will introduce a more sophisticated way of describing and solving the electric potentials, using octrees as the underlying data structure. The continuous injection of positrons into Proto-APEX is investigated in Chapter 7 by simulations and comparisons with experimental data, in particular how the radial deposition can be manipulated with the available free parameters. Which factors limit the confinement time of positrons and how the confinement time can be improved are the subject of Chapter 8. In Chapter 9, a previous beam time is recreated in the simulation framework as an example, which also yielded an unexpected result. Various ways of efficiently removing confined positrons from the dipole trap are explored by simulations in Chapter 10. Chapter 11 examines whether the trap can also be filled with multiple positron pulses and whether this is already testable in Proto-APEX. Finally, there is a summary and an outlook on future plans and developments in Chapters 12 and 13 respectively.

Chapter 2

The positron source NEPOMUC

In order to achieve a Debye length that is small enough to reach plasma conditions, a correspondingly strong positron source is needed.

A widespread source for positrons are radionuclides, which emit positrons via β^+ decay. The most commonly used isotope for creating a positron beam is ^{22}Na, since it has a half-life of 2.6 years and strong sources of up to $50\,\text{mCi}$ $(1.85\,\text{GBq})$ are commercially available [17]. Radioisotopes are often used in conjunction with a moderator, e.g. solid neon [18], to reduce the kinetic energy of the emitted positrons. This is done by implanting the positrons in the chosen moderation material, where most of them are lost by annihilation and, depending on the geometry, by transmission without thermalization, but a small fraction is re-emitted with energies in the low eV range.

Another way to create positrons is by using a linear accelerator to shoot electrons onto a high-Z target in order to produce bremsstrahlung [19]. If the photon energy is higher than $1022\,\text{keV}$, it can be converted into an electron–positron pair via pair production, which in turn can be moderated and formed into a beam. One such accelerator based positron source is, for example, operated by National Institute of Advanced Industrial Science and Technology (AIST) in Japan [20].

The high-energetic gammas necessary for pair production can also be produced by nuclear reactors, either as a byproduct of the fission process [21] or by capturing the emitted neutrons in a metal target in the vicinity of the reactor core. A reactor-based source that utilizes thermal neutron capturing is the NEutron-induced Positron source MUniCh (NEPOMUC) [22] at the Research Neutron Source Heinz Maier-Leibnitz (FRM II) in Garching bei München. Since the FRM II is located on the same campus as the IPP, NEPOMUC was naturally the first choice for conducting pair-plasma research, which is why all experiments of APEX to date have been carried out there and the future equipment will also be tailored to this positron source. For this reason, NEPOMUC will now be described in more detail.

2.1   The primary beam

The so-called primary beam of NEPOMUC is created by thermal neutrons from the FRM II reactor core hitting a ^{113}Cd-enriched cap mounted on the SR11 beam tube, as illustrated in Figure 2.1. Since the cross section of ^{113}Cd for capturing thermal neutrons is about 20600 barn, the ^{113}Cd$(n, \gamma)^{114}$Cd process is the most dominant. As the neutron binding energy of ^{113}Cd is ≈ 9 MeV, this energy is released as a cascade of γ-rays during the de-excitation, with about 2.25 photons with energies above 1.5 MeV being emitted with each neutron captured [23]. The platinum structure inside the beam tube, directly behind the cadmium cap, acts as a converter material for the photons, since the cross section for pair production is proportional to Z^2; it also serves as a moderator for the positrons created (positron work function $\Phi^+ = (-1.95 \pm 0.05)$ eV [24]), since platinum has better long-term stability and is easier to handle than tungsten [25], although the latter has better moderation efficiency. With the help of a high voltage applied to the structure, the positrons can be extracted and then magnetically (in fields of 4 mT $- 7$ mT) guided to the different experiments, providing them with over $1 \cdot 10^9$ e$^+$/s with energies between 400 eV and 1000 eV.
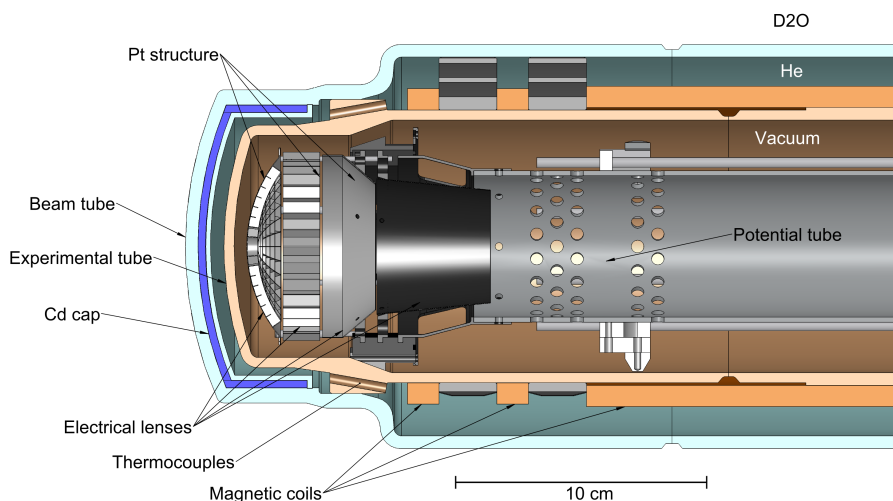
Figure 2.1: Cross-section of the in-pile positron source inside the SR11 beam tube [26].

2.2   The remoderated beam

In order to enhance the brightness of the positron beam, the 1000-eV primary beam can be guided onto another moderator, as shown in Figure 2.2, which consists of a tungsten single crystal (in situ annealed W(100) with a positron work function of $\Phi^+ = -3.0$ eV [27]) in back reflection geometry [28]. The energy of the beam can be varied between 5 eV and 20 eV, but APEX used mainly the 5-eV beam for most of the experiments.
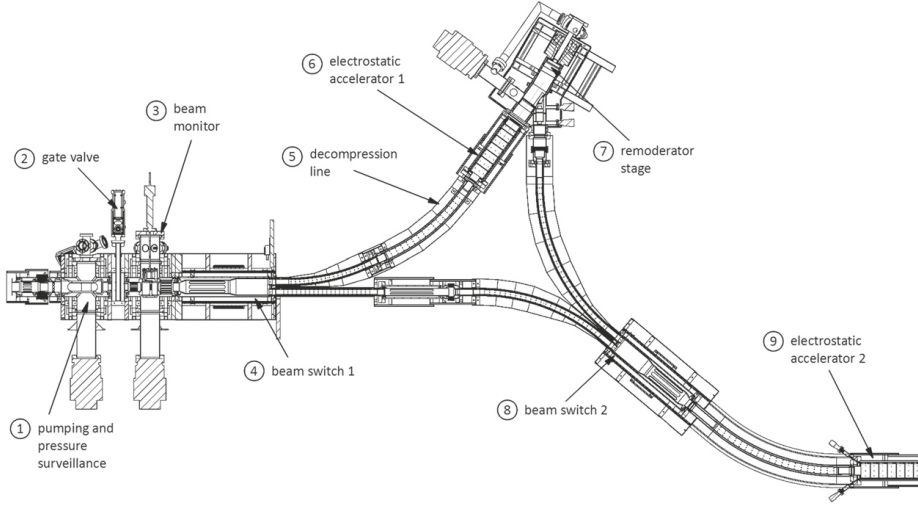
Figure 2.2: Overview of the beam line of NEPMOUC at the remoderation stage [22]. The primary beam can either be directly guided to the experiments or routed to the remoderation stage (7) beforehand via beam switches (4, 8).

This remoderated beam was characterized at the location of the experiment using a retarding field analyzer (RFA) [29]. At the energy of $5\,\mathrm{eV}$ and when measured in $5\,\mathrm{mT}$, the width of the beam was determined to be about $3\,\mathrm{mm}$ (FWHM) with an intensity over $2 \cdot 10^7\,\mathrm{e^+/s}$.

The distribution of the perpendicular energy $E_\perp$ was found to be well described by a Maxwell–Boltzmann distribution:

$$f\left(E_\perp\right) = \frac{1}{k_B T_\perp} \exp\left(-\frac{E_\perp}{k_B T_\perp}\right) = \frac{1}{\langle E_\perp \rangle} \exp\left(-\frac{E_\perp}{\langle E_\perp \rangle}\right) \tag{2.1}$$

where the product of the Boltzmann constant k_B and the perpendicular temperature $T_\perp$ was set to the mean perpendicular energy, which was measured to be $\langle E_\perp \rangle = 0.78\,\mathrm{eV}$.

The parallel energy is Gaussian-distributed:

$$f\left(E_\parallel\right) = \frac{1}{\sqrt{2\pi}\sigma_\parallel} \exp\left(-\frac{\left(E_\parallel - \langle E_\parallel \rangle\right)^2}{2\sigma_\parallel^2}\right) \tag{2.2}$$

$$= \frac{1}{\sqrt{2\pi}\langle E_\perp \rangle} \exp\left(-\frac{\left(E_\parallel - \langle E_\parallel \rangle\right)^2}{2\langle E_\perp \rangle^2}\right) \tag{2.3}$$

where the spread of the parallel energy $\sigma_\parallel$ can be assumed to be the mean perpendicular energy. The mean parallel energy was determined to be $\langle E_\parallel \rangle = 5.16\,\mathrm{eV}$. These values were also used to initiate the energy distributions of the positrons in most of the simulations, as described in Chapter 5.

7

Chapter 3

Positron accumulation

To create a pair plasma, up to 10^{11} positrons are needed, which would take NEPOMUC between one hundred and a few thousand seconds to deliver. Because confinement times cannot be expected to be this long, and because injection into magnetic traps will most likely be done in short pulses, the continuous beam from NEPOMUC must be accumulated for a considerable period of time so that an intense positron pulse can be rapidly injected into the experiment to reach the desired density. This accumulation is done in several stages to incrementally increase the number of stored positrons. For logistical reasons, design and construction of the entire system has been divided into two projects, namely the buffer-gas trap (BGT) with subsequent accumulator, and the multi-cell trap. Since these projects are an essential part of the grand scheme of APEX (see Figure 1.2), they will be described in more detail in this chapter.

3.1   Buffer-gas trap and accumulator

The principle of a BGT is based on a Penning-Malmberg (PM) trap, which confines charged particles with the same sign of charge radially with a magnetic field and prevents axial losses with electric potentials at its ends. In addition, a BGT uses a gas to reduce the energy of incoming particles through inelastic collisions in order to trap them inside the potential well [30]. To prevent the incoming particles from being reflected from the potential at the end of the trap, a relatively high buffer gas pressure in the order of 10^{-2} mbar is present in the first stage to ensure a high probability of an inelastic collision in the first pass. Since a high gas pressure also increases the likelihood of annihilation, multiple stages with progressively larger electrode diameters are used, operated at progressively lower pressures (via differential pumping) and electrostatic potentials, as illustrated in Figure 3.1. The most efficient buffer gas for positrons is N_2, since the cross section for its electronic excitation is larger than that for positronium formation in the range of 9 eV to 11 eV [31]. Once the positrons have lost enough energy to reach the third stage, they must be cooled. Instead of nitrogen, CF_4 is used as it is the most efficient cooling gas due to its large vibrational excitation cross section [32].
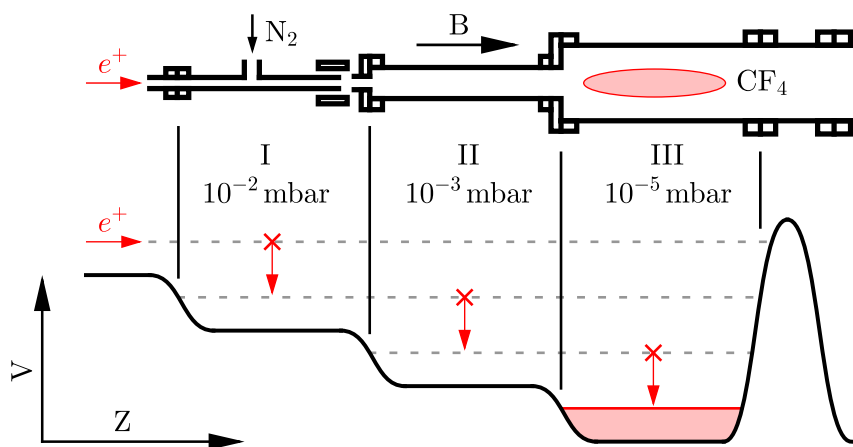
Figure 3.1: General working scheme[1] of a three-stage BGT. The upper part shows the structure of the electrodes and the lower part shows the profile of the electric potential along the central axis, parallel to the magnetic field. The positrons lose energy by inelastic collisions with the buffer gas, which causes them to be trapped due to the stepped potential well in regions of progressively lower gas pressure.

The BGT (stage I+II) and accumulator system (stage III) that will be used for APEX is shown in Figure 3.2. The BGT has an expected trapping efficiency of 5-10 % and will eject its trapped positrons into the accumulator about every second. These accumulated positrons ($> 10^8$) are then transferred to the multi-cell trap approximately every 60 s [33]. Magnetic field asymmetries, which cause heating of the trapped particles, are a general problem of Penning-Malmberg traps. In addition to carefully aligning the electrodes within the field, a common technique also used in the BGT, accumulator, and multi-cell trap is the rotating wall technique, in which sinusoidal voltages (with correspondingly offset phases) are applied to the wall sectors of a segmented electrode [34]. The heating caused by the rotating wall is compensated for by the gases that are already in the system.

The BGT works best for incoming positrons with energies around 10 eV due to the dominant cross section for positronium formation above 11 eV [17], which is why the BGT will be installed directly after the remoderator of NEPOMUC. This also has the advantage that all other experiments of NEPOMUC have the possibility to use the low energy positrons (< 1 eV) provided by the BGT system.

---

[1]Vectorized and modified version of "Buffer-gas-trap.jpg". Created by Physicscomputingfacility. Distributed under Creative Commons Attribution-Share Alike 4.0 International. Wikimedia Commons. 28 March 2019.
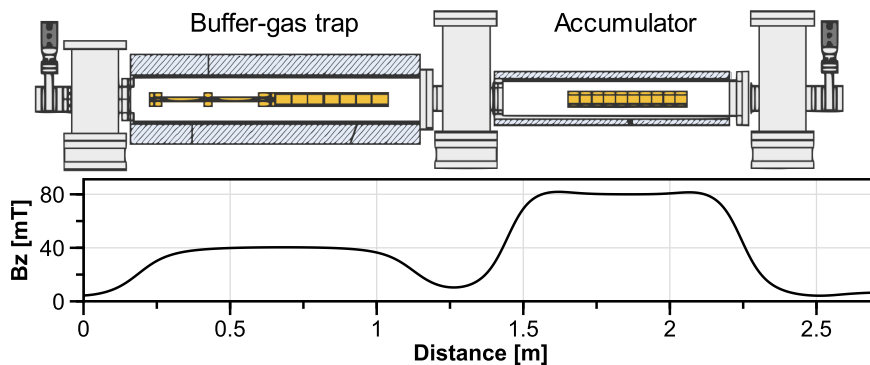
Figure 3.2: Illustration of the buffer gas trap and the accumulator with the magnetic field strength along the central axis [33]. The positrons are injected from the left.

3.2   Multi-cell trap

The number of positrons stored in a PM trap is limited, mainly for technical reasons, since potentials in the kilovolt range are needed to confine large space charges. Instead of using high voltages, several smaller PM traps could be used, like shown in Figure 3.3, to realize the storage of 10^{10} to 10^{11} positrons. This so-called multi-cell trap [35] consists of a larger master-cell that takes the positrons from the accumulator and shifts the plasma radially via autoresonant excitation of the diocotron mode using the rotating wall to fill the off-axis cells [36]. Since it can take several hours to fill all the storage cells, special care must be taken to avoid positron losses in the meantime. This includes a clean UHV environment, the use of the rotating wall to keep the plasmas compressed, and careful alignment of the cells within the magnetic field [37]. To counteract the inevitable heating, e.g. by the rotating wall itself, cooling is nevertheless required. Since a cooling gas would also increase the risk of annihilations, the entire trap is placed in a magnetic field of several tesla to use cyclotron radiation as the main cooling mechanism. It is not yet clear if all storage cells can be emptied at the same time to create one intense positron pulse or if they need to be emptied in multiple batches. In Chapter 11, some preliminary simulation results are presented to see if such an injection of positrons via multiple pulses would be possible for Proto-APEX.
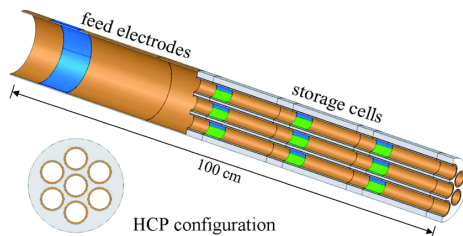
Figure 3.3: One possible design of a multi-cell.trap [17]. Positrons are transferred from the master-cell ("feed electrodes") to the 21 hexagonally closed packed (HCP) storage cells by using the "diocotron mode" [36]. Each cell has its own set of rotating wall electrodes.

Chapter 4

Magnetic confinement devices

This chapter provides a more detailed description of the current and future magnetic confinement devices used by the APEX collaboration. All these devices have in common that the incoming positrons are transferred via the E×B drift [38, p. 18] (with the electric field vector $\vec{E}$ and the magnetic field vector $\vec{B}$)

$$\vec{v}_{E\times B} = \frac{\vec{E} \times \vec{B}}{B^2} \tag{4.1}$$

from the beam line into the confinement region [39].

4.1   The prototype trap: Proto-APEX

The setup of Proto-APEX, whose primary results have been to verify the E×B drift injection technique and subsequent positron confinement [40, 41], is shown in Figure 4.1. The positron beam from NEPOMUC, most of the time the re-moderated 5-eV beam, is magnetically guided to the experiment, where a set of guide field coils couples the magnetic field of the beam line to the dipole field. The guide field coils are centered around the beam tube, which is offset by 4 cm relative to the main chamber. Inside the beam tube is a deflector, a cylindrical electrode that can be biased to prevent the positron beam from reaching the experiment. The beam first passes through a diagnostic chamber; there, a target plate can be inserted to measure the incoming beam intensity, or a microchannel plate (MCP) can be inserted to visualize the beam. The positrons then pass through two sets of steering coils, which are used to fine-tune the position of the injection into the main chamber.

Once the beam enters the main chamber two oppositely charged plates provide the necessary electric field for the positrons to drift onto the closed field lines of the dipole, hence the name E×B plates. In order to reduce the disturbance of the successfully injected positrons by this strong electric field, a grounded

shield plate is located directly beside them. The central dipole field is generated by a supported neodymium magnet (height: 40 mm, diameter: 28 mm, 600 mT at the poles), which is housed in a copper casing (height: 46 mm, diameter: 34 mm) so that it can be biased to prevent positrons from annihilating on it. The magnet, the E×B plates, and the shield plate are surrounded by a cylindrical wall (radius: 9 cm), which is divided into several segments, all of which can be biased separately. The lower ring is divided into eight equal segments, while the top ring is divided into a 1/8 segment and a 7/8 segment. Normally, all segments are grounded except for RW1 and Top1, highlighted in Figure 4.1, which are both positively biased during the injection for most experiments conducted, as this dramatically increases injection efficiency by preventing positrons from hitting the wall if they are not directly confined. This allows a positron to be electrostatically reflected by the wall electrodes and magnetically mirrored by the dipole field (or electrostatically reflected due to the magnet case bias) within the electric field of the E×B plates, sometimes even multiple times, until the positron is ultimately confined in the dipole field (more on this in Section 7.1).
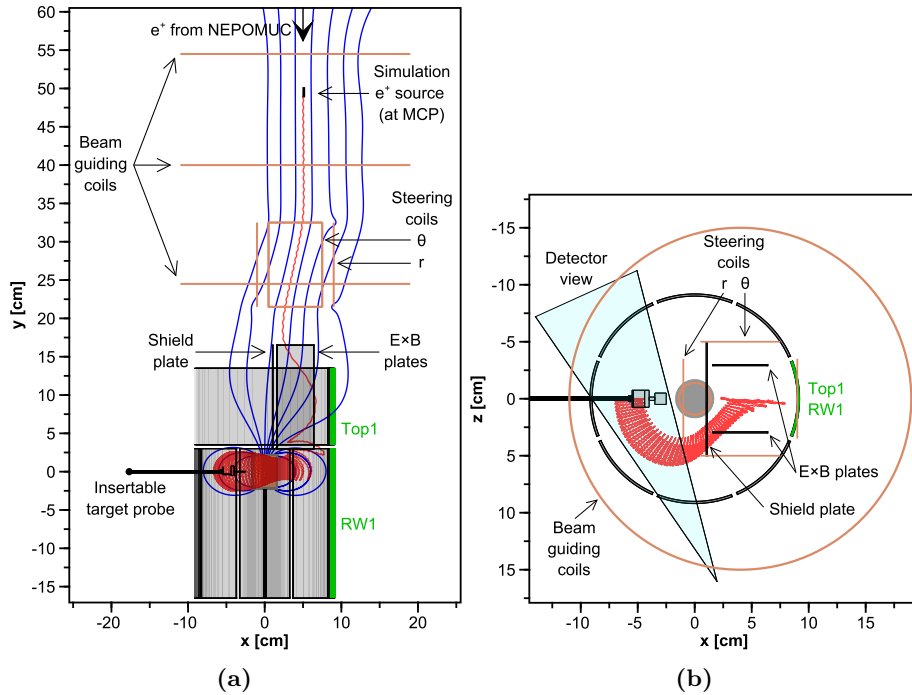

Figure 4.1: Side view (a) and top view (b) of Proto-APEX, as modeled in the simulation program. Magnetic field lines are depicted in blue and a sample positron trajectory is shown in red.

Once confined, the positrons travel toroidally around the magnet due to the grad-B drift $\vec{v}_{\nabla B}$ and curvature drift $\vec{v}_R$ [38, p. 20–21]:

$$\vec{v}_{\nabla B} = -\frac{E_\perp}{q} \frac{\vec{\nabla} B \times \vec{B}}{B^3} \tag{4.2}$$

$$\vec{v}_R = \frac{2E_\parallel}{qR_c^2} \frac{\vec{R}_c \times \vec{B}}{B^2} \tag{4.3}$$

with the kinetic energy E perpendicular and parallel to the magnetic field B, the charge q, and the radius of curvature R_c pointing away from the center of the approximated circle. For low-energies (at which the gyromotion is fast compared to the drifts) the positrons perform three distinct motions in the dipole field (values given correspond to the remoderated 5-eV beam): they gyrate in the magnetic field at the cyclotron frequency ($f \approx (280 \pm 70)\,\text{MHz}$), bounce between the poles ($T \approx (280 \pm 40)\,\text{ns}$) and drift toroidally around the magnet ($T \approx (15 \pm 4)\,\text{µs}$). On the opposite side of the chamber with respect to the injection region, a target probe can be inserted, acting as an annihilation target for the positrons. With this target probe, the injection efficiency and the radial distribution of the positrons can be measured either by counting the annihilation gammas with a collimated bismuth germanium oxide (BGO) scintillation detector or by directly measuring the incoming current. The effect of the target probe on the measurement is discussed in Section 7.2.

4.2   The levitated dipole trap: APEX-D

Proto-APEX has produced excellent results, but is not suitable for creating plasma conditions because the loss cone onto the magnet can only be plugged for either positrons or electrons at the same time. For this reason, APEX-D [42] will use a levitated superconducting coil (GdBaCuO-tape from THEVA, $T_C = 92\,\text{K}$), which of course does not have this loss cone because the field lines do not intersect material surfaces.

In Figure 4.2, the 3D model is shown. The setup consists of two chambers. In the lower chamber, the cooling assembly is situated. At the beginning of an experimental cycle, the normally conducting floating coil (radius: $\approx 7.5\,\text{cm}$, mass: $< 2\,\text{kg}$, $30\,\text{kA} - \text{turns}$, $1\,\text{T}$) is placed inside this cooling assembly and enclosed by a removable lid. Next, the high-temperature superconducting coil within the assembly is ramped up to produce a strong magnetic field. Helium is injected into the assembly to provide good thermal contact between the floating coil and the assembly itself, which is cooled to $20\,\text{K}$ with a cryocooler. Once the temperature of the floating coil is well below its critical temperature, the charging coil is ramped down to induce a current in the floating coil. After the helium has been pumped out of the cooling assembly, the lid is removed, and the floating coil is mechanically lifted into the upper experimental chamber. Once inside experimental chamber, a copper, water-cooled lifting coil is engaged to levitate the floating coil in the chamber, while the lifter is retracted. This configuration, with the lifting coil on top, is stable for slide and tilt motions, but the vertical motion of the floating coil is unstable [43]. Three laser sensors are therefore used to provide position data to a feedback system that controls the current of the lifting coil to
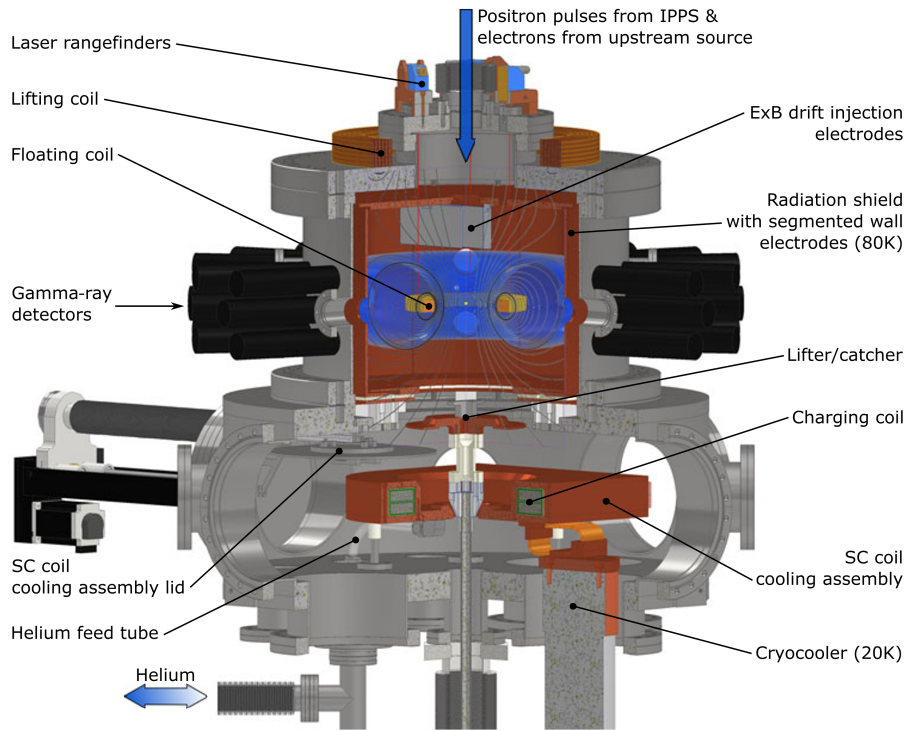
Figure 4.2: Computer model of APEX-D with the cooling assembly and cahrging coil in the lower chamber, the leviating coil in the upper experimental chamber, and the lifting coil at the top of the chamber [10].

keep the floating coil in place. The main factors that limit the levitation time are the current decay due to the residual resistance in the floating coil and the heating due to thermal radiation from the environment. The effect of the radiation heating is reduced by gold-plating the floating coil and connecting the electrode structure in the experimental chamber to the cryocooler, thus acting as a radiation shield. When the levitation can no longer be maintained, the lifter is raised to capture the floating coil and safely return it to the cooling assembly, where the cycle can be repeated once the coil has warmed above its critical temperature.

As the main diagnostic tool, the chamber will be surrounded by 48 cylindrical BGO-detectors (crystal size: $1\,\text{in} \times 1\,\text{in}$), which are directly connected to three V1730 digitizer cards from CAEN [44]. These cards also support on-board pulse processing to directly return timing and energy information for each incoming detector signal, enabling a quick data analysis and perhaps even a tomographic reconstruction. In the future, high-purity germanium detectors may be added to provide better energy resolution, which, in conjunction with a gas, could potentially provide an upper limit on the positron energy [45]. Another option would be to inject pellets or cold gas jets into the plasma to locally increase the annihilation rate to obtain radial density profiles [9].

4.3 The optimized stellarator trap: EPOS

Besides the dipole configuration, a stellarator is also suitable for confining a pair plasma. Therefore, EPOS (Electrons and Positrons in an Optimized Stellarator) is being developed in parallel to APEX-D to cover a wider physical range for studying pair plasmas [7]. One of the main physical differences will be that instead of using a floating coil, which requires cooling and warming cycles, to generate the confining magnetic field, EPOS will use many 3-D coils, which will be mechanically mounted, allowing the device to operate in a steady state. As a result, the magnetic topology will also be different. Instead of having field lines that close after one pass around the coil, which is the case for a levitated dipole, a stellarator will have flux surfaces with correspondingly longer magnetic connection lengths. An overview of the major differences between the two configurations is given in Table 4.1.

Property	Stellarator	Levitated dipole
Time scale	Steady state	Requires cooling/warming cycles
Wider relevance	Fusion	Astrophysics
Flux expansion	Negligible	Large
Topology	Magnetic flux surfaces	Field lines close after one pass
B-field connection length	Long	Short
Drift orbit confinement	Requires optimization	Confined (due to axissymmetry)
Coils	Many 3-D coils	2–3: floating, lifting, charging
Current leads	Permanently attached	Detachable/inductively charged
Positron fuelling	Pulsed fuelling	Possibility of steady state

Table 4.1: A comparison of the different characteristics of a stellarator and a levitating dipole [10].

At the time of writing, EPOS is still in the development phase, but it is likely to be a quasi-axisymmetric stellarator, optimized to reduce collisionless transport. The optimization process will also pay special attention to the robustness to manufacturing tolerances, and since EPOS is table-top sized, it is an ideal candidate for testing this. EPOS is also breaking new ground by attempting to 3D print the metal frame around which the high-temperature superconducting tape is wrapped.

Chapter 5

The simulation framework

A very helpful counterpart to experiments are numerical simulations. They are able to explain experimental results by visualizing complete trajectories of particles, gather data that would be inaccessible otherwise and can assist designing future experiments. Furthermore, numerical simulations are of course not dependent on the limited availability of positrons, which makes them an essential tool of the quest to create an electron–positron plasma, as it allows for a rapid iteration and feasibility studies of e.g. electrode configurations.

At the current stage of the experiments, the achievable positron densities are low enough to assume that positrons act as single particles without any interaction between other particles. This greatly simplifies the simulations, since only a global description of the electric and magnetic fields at each point is needed, and the iterative propagation of the particles through these fields is independent of other particles and thus can be trivially parallelized. None of the known free or commercial simulation software combined a fast, analytical calculation of the magnetic fields of coils, a solver for electric potentials in arbitrary geometries, and a particle pusher fast enough to numerically explore the large parameter space of APEX in a reasonable time. Therefore, AlGeoJ was developed from scratch to fill this gap.

This chapter describes in detail all the components needed to run particle simulations. First, a particle pusher algorithm with excellent energy conserving properties is presented to efficiently propagate particles through electric and magnetic fields. In Section 5.2, the implemented analytical formulas for the calculation of the magnetic fields are given for each type of coil used in APEX. Section 5.3 describes how the electric potentials are calculated, including multiple acceleration techniques. Several ways to obtain the electric field vector from a discrete potential are presented in Section 5.4. Section 5.5 covers the available options in AlGeoJ for describing the electrode geometries. Finally, Section 5.6 illustrates a method for generating a realistic spatial distribution of particles from an image.

5.1 Particle pusher

The trajectory of a charged particle is, in our case, determined solely by the Lorentz force:

$$\vec{a} = \frac{d\vec{v}}{dt} = \frac{q}{m}(\vec{E} + \vec{v} \times \vec{B}) = \vec{\Sigma} + \vec{v} \times \vec{\Omega} \tag{5.1}$$

where q and m is the charge and the mass, respectively, of a particle, $\vec{E}$ the electric field vector, $\vec{B}$ the magnetic field vector, $\vec{\Sigma} = \frac{q}{m}\vec{E}$ and $\vec{\Omega} = \frac{q}{m}\vec{B}$. The velocity can be replaced by the average of the old and new velocity:

$$\frac{\vec{v}_{new} - \vec{v}_{old}}{\Delta t} = \vec{\Sigma} + \frac{\vec{v}_{new} - \vec{v}_{old}}{2} \times \vec{\Omega} \tag{5.2}$$

With the time step Δt and

$$\vec{A} = \vec{\Omega}\Delta t/2 \tag{5.3}$$

$$\vec{C} = \vec{v}_{old} + \Delta t \left(\vec{\Sigma} + \vec{v}_{old} \times \vec{\Omega}/2 \right) \tag{5.4}$$

Equation (5.2) can be solved for the new velocity vector by [46, p. 30]:

$$\vec{v}_{new} = \frac{\vec{C} + \vec{A}\vec{A} \cdot \vec{C} - \vec{A} \times \vec{C}}{1 + A^2} \tag{5.5}$$

The new position x_{new} is subsequently given by:

$$\vec{x}_{new} = \vec{x}_{old} + \Delta t \cdot \vec{v}_{new} \tag{5.6}$$

This is a variant of the second-order leapfrog integrator, which conserves phase space volume [47, p. 231] and is therefore a much better choice for propagating particles through electromagnetic fields than, for example, the fourth-order Runge-Kutta method [16].

5.2 Magnetic fields

In APEX, charged particles are confined in the main dipole field, so special care was taken to avoid any ferromagnetic materials inside the experimental chamber that may deform the field. This has the positive side effect of simplifying the calculation of the magnetic field since the field at any given point can be assumed to be the superposition of the generated fields of each individual coil.

In general, the magnetic field produced by a current carrying wire can be calculated with the Biot–Savart law:

$$\vec{B}(\vec{r}) = \frac{\mu_0}{4\pi} \int_C I d\vec{l} \times \frac{\vec{r} - \vec{r}'}{|\vec{r} - \vec{r}'|^3} \tag{5.7}$$

From a simulation standpoint, this is far from ideal, as a line integral must be calculated for each individual step along the trajectory for each individual coil. Fortunately, there is an analytical solution for each type of coil used in the experiment, which allows one to avoid an inaccurate approximation of the magnetic field on a discrete grid.

5.2.1 Circular current loop

Probably the simplest coil is a single, circular, current-carrying loop. Such loops are very well suited to model the beam line coils, since the large distance of these coils to the positron beam makes their physical dimension negligible. With the radius of the coil R, the distance to the symmetry axis ρ, the distance to plane z, the ampere-turns I and

$$\alpha^2 = R^2 + \rho^2 + z^2 - 2R\rho \tag{5.8}$$

$$\beta^2 = R^2 + \rho^2 + z^2 + 2R\rho \tag{5.9}$$

$$k^2 = 1 - \alpha^2/\beta^2 \tag{5.10}$$

$$C = \mu_0 I/\pi \tag{5.11}$$

the magnetic field of a current loop in cylindrical coordinates is given by [48]:

$$B_\rho = \frac{Cz}{2\alpha^2\beta\rho} \left[\left(R^2 + \rho^2 + z^2\right) E\left(k^2\right) - \alpha^2 K\left(k^2\right) \right] \tag{5.12}$$

$$B_z = \frac{C}{2\alpha^2\beta} \left[\left(R^2 - \rho^2 - z^2\right) E\left(k^2\right) + \alpha^2 K\left(k^2\right) \right] \tag{5.13}$$

$K(k)$ and $E(k)$ are the complete elliptic integrals of the first and second kind, respectively. These functions can be calculated quite efficiently with the arithmetic-geometric mean described in Appendix B.

5.2.2 Continuous finite solenoid

In Proto-APEX, the dipole magnetic field is created by a neodymium permanent magnet. One way to reproduce the magnetic field of the magnet would be to stack multiple of the aforementioned circular current loops, but this approach would require about 75 loops to achieve an accuracy better than 1% on the equatorial plane at a radius of 9 cm compared to the limit of an infinite number of windings (see Figure 5.1). At smaller radii, were the windings of the solenoid play a more significant role (see Figure 5.2), even more loops would be required.
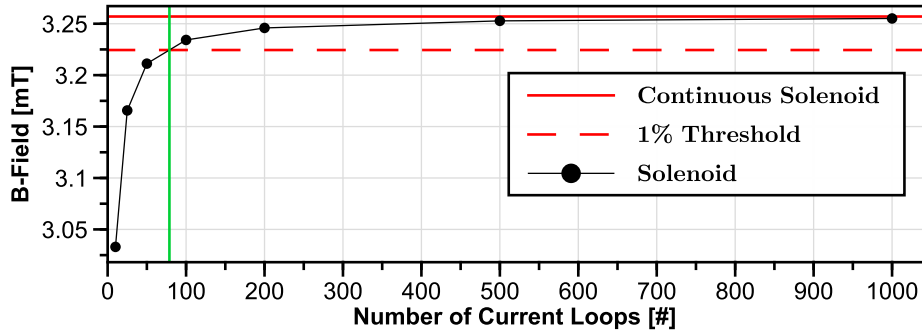
Figure 5.1: Comparison of the B-field strengths of the Proto-APEX magnet modeled by a continuous finite solenoid and a solenoid with a certain number of windings [16]. Measured on the midplane at a radius of 9 cm.

Instead of using a stack of circular current loops to model the magnet, it is computationally much more efficient to calculate the field with a closed formula for a finite continuous solenoid where the field is generated by a surface current. With the length of the solenoid L, the radius R, the current I, the distance to the center of the solenoid along the symmetry axis z, the distance to the symmetry axis ρ and

$$\zeta_\pm = z \pm L/2 \tag{5.14}$$

$$h^2 = \frac{4R\rho}{(R+\rho)^2} \tag{5.15}$$

$$k^2 = \frac{4R\rho}{(R+\rho)^2 + \zeta^2} \tag{5.16}$$

the magnetic field of a continuous finite solenoid can be calculated in cylindrical coordinates by [49]

$$B_\rho = \frac{\mu_0 I}{2\pi L} \sqrt{\frac{R}{\rho}} \left[\frac{k^2 - 2}{k} K\left(k^2\right) + \frac{2}{k} E\left(k^2\right) \right]_{\zeta_-}^{\zeta_+} \tag{5.17}$$

$$B_z = \frac{\mu_0 I}{4\pi L} \frac{1}{\sqrt{R\rho}} \left[\zeta k \left(K\left(k^2\right) + \frac{R-\rho}{R+\rho} \Pi\left(h^2, k^2\right) \right) \right]_{\zeta_-}^{\zeta_+} \tag{5.18}$$

with the three complete elliptic integral $K(k)$, $E(k)$, and $\Pi(n, k)$. How these can be computed efficiently is described in Appendix B.

(a) Solenoid   **(b)** Continuous solenoid.
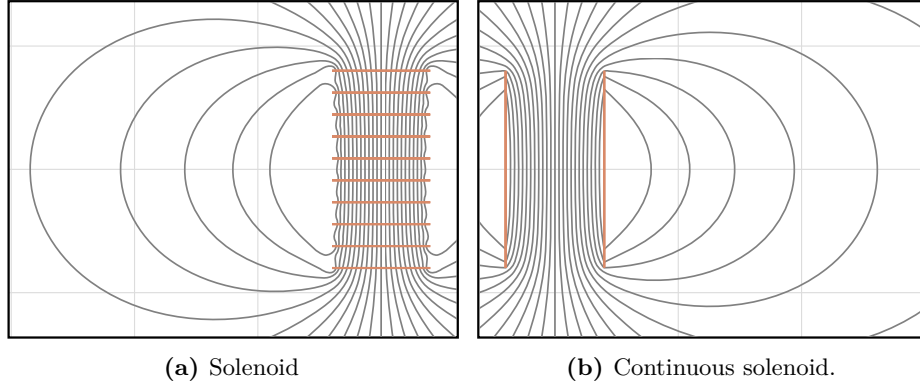
Figure 5.2: Comparison of the field lines generated by a solenoid (a) consisting of 10 current loops and a continuous finite solenoid (b) [16].

5.2.3   Rectangular coils

The steering coils, that are used to fine adjust the positron beam position right before it enters the main chamber, have a rectangular shape, as seen in Figure 4.1. Of course, these hand-wound coils do not have a perfect rectangular shape in real life, so they could well be modeled with circular current loops, but since analytical expressions exist for calculating the magnetic field of a rectangular coil, one of them was used instead as described in the following.

Assuming a coil with side lengths of $2a$ and $2b$ that is centered in the x-y-plane, like shown in Figure 5.3, then with

$$C_1 = -C_4 = a + x \tag{5.19}$$

$$C_2 = -C_3 = a - x \tag{5.20}$$

$$d_1 = d_2 = y + b \tag{5.21}$$

$$d_3 = d_4 = y - b \tag{5.22}$$

and the distances r_n from the point of evaluation to the corners of the coil

$$r_1 = \sqrt{(a+x)^2 + (y+b)^2 + z^2} \tag{5.23}$$

$$r_2 = \sqrt{(a-x)^2 + (y+b)^2 + z^2} \tag{5.24}$$

$$r_3 = \sqrt{(a-x)^2 + (y-b)^2 + z^2} \tag{5.25}$$

$$r_4 = \sqrt{(a+x)^2 + (y-b)^2 + z^2} \tag{5.26}$$

the magnetic field of a rectangular coil is given in Cartesian coordinates by [50]

$$B_x = \frac{\mu_0 I}{4\pi} \sum_{n=1}^{4} \left[\frac{(-1)^{n+1} z}{r_n [r_n + d_n]} \right] \tag{5.27}$$

$$B_y = \frac{\mu_0 I}{4\pi} \sum_{n=1}^{4} \left[\frac{(-1)^{n+1} z}{r_n \left[r_n + (-1)^{n+1} C_n \right]} \right] \tag{5.28}$$

$$B_z = \frac{\mu_0 I}{4\pi} \sum_{n=1}^{4} \left[\frac{(-1)^n d_n}{r_n \left[r_n + (-1)^{n+1} C_n \right]} - \frac{C_n}{r_n [r_n + d_n]} \right] \tag{5.29}$$
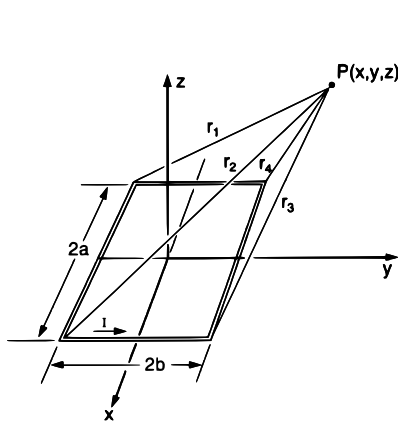
Figure 5.3: Rectangular current loop in the x-y plane. P is the point of evaluation [50].
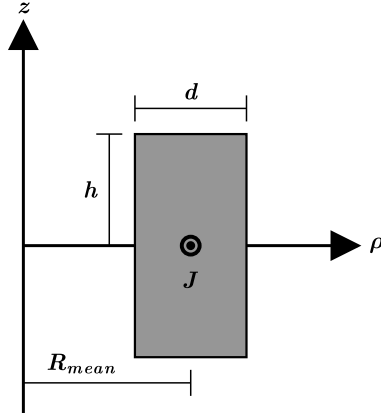
Figure 5.4: Illustration of the dimensions of a circular coil with rectangular cross section (based upon [51])

5.2.4   Circular coil with rectangular cross section

In the future, the main dipole field of APEX will be produced by a levitated superconducting coil (see Chapter 4). Since it is a coil, particles can of course fly through it unhindered, unlike Proto-APEX, where they are inevitably stopped by the magnetic case. In addition, it cannot be biased, which ultimately leads to particles being able to stay in close proximity to the coil, making an approximation of the magnetic field by a singular current loop less suitable. For that reason, a semi-analytic method was implemented to calculate the magnetic field of a circular coil with rectangular cross section.

For a coil with dimension as illustrated in Figure 5.4 and the uniform current density J, the magnetic field in cylindrical coordinates at a position (ρ, z) is given by [51]:

$$B_\rho = \frac{\mu_0 J r}{2\pi} \sum_{i=1}^{2} \sum_{j=1}^{2} (-1)^{i+j} P_\rho(R_i, Z_j) \tag{5.30}$$

$$B_z = \frac{\mu_0 J r}{2\pi} \sum_{i=1}^{2} \sum_{j=1}^{2} (-1)^{i+j} P_z(R_i, Z_j) \tag{5.31}$$

With the scaling transformation

$$R_{1,2} = \frac{R_{mean} \mp d/2}{\rho} \tag{5.32}$$

$$Z_{1,2} = \frac{\mp h - z}{\rho} \tag{5.33}$$

$$R_0 = \sqrt{R^2 + 1 - 2R\cos\varphi + Z^2} \tag{5.34}$$

and

$$P_\rho(R, Z) = \int_0^\pi \left(R_0 + \cos\varphi \ln(R_0 + R - \cos\varphi) \right) \cdot \cos\varphi \, d\varphi \tag{5.35}$$

$$P_z(R, Z) = \int_0^\pi Z \ln(R_0 + R - \cos\varphi) + \frac{1}{2}\cos\varphi \ln\left(\frac{R_0 - Z}{R_0 + Z} \right) \tag{5.36}$$

$$- \sin\varphi \arctan\left(\frac{Z(R - \cos\varphi)}{R_0 \sin\varphi} \right) d\varphi \tag{5.37}$$

For the simulation of trajectories, however, this is very time-consuming, since eight integrals have to be numerically solved for each step. Nevertheless, with the help of this implementation and an optimization procedure, a combination of previously described coils can probably be found that produces a good approximation of the magnetic field of the APEX coil, which can make trajectory simulations feasible again.

5.2.5 Ideal magnetic dipole

The point of magnetic reflection is mostly expressed under the condition that the magnetic moment $\mu = mv_\perp^2/(2B)$ is adiabatically invariant. In a dipole field, however, this is only valid to the lowest order. If higher orders are also considered, the point at which a charged particle is mirrored depends also on the phase of its gyration motion and on its parallel velocity [52]. In order to investigate the resulting effects in more detail, an ideal magnetic dipole is a good choice:

$$\vec{B}(\vec{m}, \vec{r}) = \frac{\mu_0}{4\pi} \frac{3(\vec{m} \cdot \hat{r})\hat{r} - \vec{m}}{r^3} \tag{5.38}$$

with the magnetic dipole moment $\vec{m}$.

5.3 Electric potentials

The first Maxwell equation states for the electric field E that

$$\nabla \vec{E} = \frac{\rho}{\epsilon} \tag{5.39}$$

with the charge density ρ, the permittivity ϵ. Therefore, charges generate a divergent electric field. In the electrostatic case, the electric field can also be expressed as the gradient of the electric potential Φ:

$$\vec{E} = -\nabla \Phi \tag{5.40}$$

Applying the ∇ operator to Equation (5.40) and setting it equal to Equation (5.39) results in the Poisson's equation

$$\Delta \Phi(x, y, z) = -\frac{\rho(x, y, z)}{\epsilon} \tag{5.41}$$

with the Laplace operator Δ. If the charge density $\rho(x, y, z) = 0$ then the Poisson equation simplifies to the Laplace equation:

$$\Delta \Phi(x, y, z) = 0 \tag{5.42}$$

5.3.1 Discretization of the potential

In general, the Poisson equation or the Laplace equation, respectively, can only be solved analytically for very simple, idealized cases. The calculation of the electric potential therefore requires a numerical approach. To make it numerically solvable in the first place, the problem must be discretized. At first sight, an approach with cylindrical coordinates seems to be a good fit for the Proto-APEX potentials, but the disadvantages in handling the shield plate, the E×B plates and the target probe outweigh its advantages. This leaves only a regular Cartesian grid for which the solution of an electric potential is derived in the following, including multiple methods to reduce the required computational resources.

The second order derivative of a function can be written as the limit of the second order central difference quotient

$$
\begin{aligned}
f''(x) &= \lim_{h \to 0} \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} \\
&= \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + \mathcal{O}(h^2)
\end{aligned}
\tag{5.43}
$$

where h is a small step. Combining Equations (5.42) and (5.43) results in the local approximation of the Laplace equation:

$$
\begin{aligned}
\Delta\Phi(x,y,z) &= \frac{\Phi(x+h,y,z) - 2 \cdot \Phi(x,y,z) + \Phi(x-h,y,z)}{h^2} + \\
&\quad \frac{\Phi(x,y+h,z) - 2 \cdot \Phi(x,y,z) + \Phi(x,y-h,z)}{h^2} + \\
&\quad \frac{\Phi(x,y,z+h) - 2 \cdot \Phi(x,y,z) + \Phi(x,y,z-h)}{h^2} \\
&= 0
\end{aligned}
\tag{5.44}
$$

For a discretized potential on a grid $\Phi_{x,y,z}$, with x, y, z being the array indices, Equation (5.44) can be written as:

$$
\begin{aligned}
\Delta\Phi_{x,y,z} &= \Phi_{x+1,y,z} + \Phi_{x-1,y,z} + \Phi_{x,y+1,z} + \\
&\quad \Phi_{x,y-1,z} + \Phi_{x,y,z+1} + \Phi_{x,y,z-1} - 6 \cdot \Phi_{x,y,z} \\
&= 0
\end{aligned}
\tag{5.45}
$$

This can then be easily solved for $\Phi_{x,y,z}^k$, with k indicating the kth iteration:

$$
\Phi_{x,y,z}^{k+1} = \frac{1}{6} \cdot (\Phi_{x+1,y,z}^k + \Phi_{x-1,y,z}^k + \Phi_{x,y+1,z}^k + \Phi_{x,y-1,z}^k + \Phi_{x,y,z+1}^k + \Phi_{x,y,z-1}^k)
\tag{5.46}
$$

This means that if the potential on the grid is correct, the potential at every grid point is the average of its six direct neighbors as illustrated in Figure 5.5. It does, however, not provide an instant global solution for the potential, therefore it is necessary to iterate over all grid points until the maximum change between iterations is smaller than a given threshold or until a maximum number of iterations is reached. The threshold itself must be greater than the machine precision used to ensure that it can be reached at all.

(a) 2D.   **(b)** 3D.
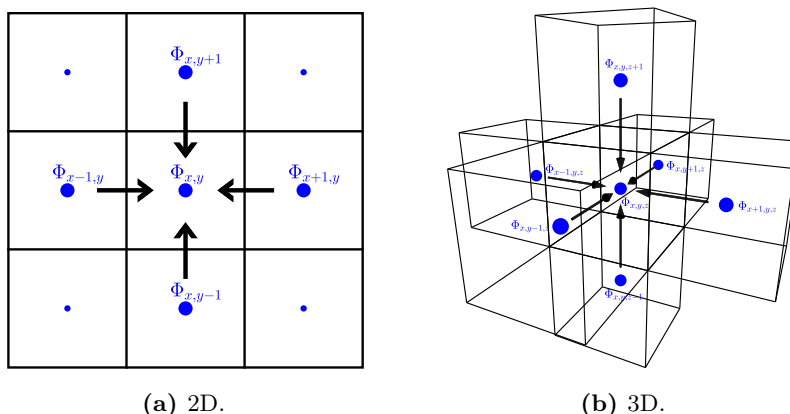
Figure 5.5: Illustration of a local Laplace iteration step in 2D and 3D.

5.3.2   Boundary conditions

Before solutions for the different electrical potentials can be calculated, some boundary conditions have to be defined. In this case, Dirichlet boundary conditions (which require a function to take given values on the boundary of the domain) are the logical choice, since no adjustments to the derived formulas need to be made for them.

Figure 5.6 shows the two different boundary types, namely the domain boundary and object boundaries. If the boundary of the domain is set to $0\,\mathrm{V}$, which is usually the case since only electrodes within the domain are biased, then no adjustments need to be made because when a potential array is created, all values are set to 0 in the first place and while iterating over the cells with e.g. Equation (5.48), the boundary needs to be omitted anyway (see Algorithm 3). With the domain boundary alone, one can of course not yet calculate a usable potential. A simple way to discretize the geometry is to check whether the center of a cell is inside the object or not. In order to achieve better coverage, especially for rounded objects, it can be helpful to also consider cells whose centers are close to the object (compare Figures 5.6a and 5.6b). In the case of Proto-APEX, a threshold of $20\,\%$ of the grid spacing yielded good results. Once the geometry cells are found, these cells can be initialized in the potential array. The set value can correspond either directly to the desired potential, or $1\,\mathrm{V}$, which allows for a quick change of the potential strength by simply scaling it,without the need to recalculate the full potential.

In the following pseudocode examples, the method ISGEOMETRY(x, y, z) returns either true or false, depending on whether the cell at the passed position was marked as geometry. The actual implementation can be done using a boolean array in which the geometry marker for each cell is stored. This way each cell has to be checked only once and ISGEOMETRY is reduced to a single array access. The additional memory demand is relatively small, because a boolean array, depending on the used programming language, requires much less memory than a potential array (e.g. in Java: boolean/float $= 1/4$) and only one array is necessary per domain.

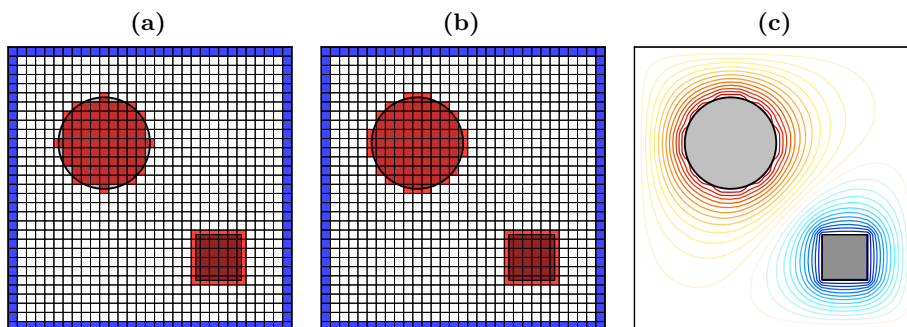**Figure 5.6:** The discretization on a regular grid. Cells with geometry are colored in red. The domain boundary cells are set to $0\,\mathrm{V}$ and highlighted in blue. In (a), only cells were marked as geometry if their centers were completely within the object. In (b), cells located at a distance up to $20\,\%$ of the grid spacing from the object were also considered. The solved potential is shown in (c), where the cylinder and the cube are oppositely charged.

5.3.3   Successive over-relaxation

An improvement of Equation (5.46), also known as the Jacobi method, is the so called Gauss-Seidel method:

$$\Phi_{x,y,z}^{k+1} = \frac{1}{6} \cdot (\Phi_{x+1,y,z}^{k} + \Phi_{x-1,y,z}^{k+1} + \Phi_{x,y+1,z}^{k} + \Phi_{x,y-1,z}^{k+1} + \Phi_{x,y,z+1}^{k} + \Phi_{x,y,z-1}^{k+1}) \quad (5.47)$$

The Gauss-Seidel method is very similar to the Jacobi method with the only difference that values are updated in-place and previously updated values are directly used. This has the benefit that a temporary array storing the updated values can be omitted. A negative side effect of the Gauss-Seidel method is, however, that the local update causes an asymmetry in the direction of the iteration, but this asymmetry should be small once the potential has converged. Another disadvantage to the Jacobi method is that the iteration cannot be parallelized in a straightforward way since this would require a synchronization mechanism between threads and their work block boundaries.

To speed up the convergence, the Gauss-Seidel can be modified to the successive over-relaxation method (SOR) [53][47, p. 308]:

$$\begin{aligned}\Phi_{x,y,z}^{k+1} = &(1-\omega) \cdot \Phi_{x,y,z}^{k} + \frac{\omega}{6} \cdot (\Phi_{x+1,y,z}^{k} + \Phi_{x-1,y,z}^{k+1} + \Phi_{x,y+1,z}^{k} + \\ &\Phi_{x,y-1,z}^{k+1} + \Phi_{x,y,z+1}^{k} + \Phi_{x,y,z-1}^{k+1})\end{aligned} \quad (5.48)$$

with the SOR parameter ω and $1 < \omega < 2$. If $\omega = 1$ then Equation (5.48) is identical to the Gauss-Seidel method (Equation (5.47)). The optimal choice of ω is unfortunately highly dependent on the problem and can usually only be found heuristically. Once found, however, the number of iteration until convergence is reached can be reduced quite drastically (a factor of 10 was achieved in the example shown in Figure 5.7). For the calculation of the Proto-APEX potentials $\omega = 1.9$ was chosen.

5.3.4 Multigrid preconditioning

The term "multigrid" can refer to a variety of schemes to solve a discretized partial differential equation system, with the basic idea being to first solve the problem on a coarser grid and then to apply this solution to the fine grid. The different schemes differ mainly in the number of levels of coarser grids, and in what order the fine grid is restricted to the coarser grids and the coarse grids prolongated to the finer ones, respectively [47, p. 308].

Solving a potential using the multigrid method alone in a domain with a lot of geometry proves to be difficult, as this requires special considerations during the restriction and prolongation between the different grid levels. However, the potential array can be preconditioned using a basic multigrid method to further decrease the number of expensive SOR iterations on the finest grid. In this instance, the potential is first solved on the coarsest grid with $1/4$ of the resolution ($d = 4$, see Algorithm 3 in Appendix E.1), then the values are prolongated (described in Algorithm 4 in Appendix E.1) to a grid with half the resolution of the finest grid and solved again ($d = 2$). Finally, the values are prolongated once again to the finest grid and relaxed using the SOR method (now $d = 1$) until convergence. Even with this simple approach, the number of expensive, full SOR iterations can be decreased by over $50\,\%$ while the dependence on the parameter ω is weakened at the same time (see Figure 5.7).
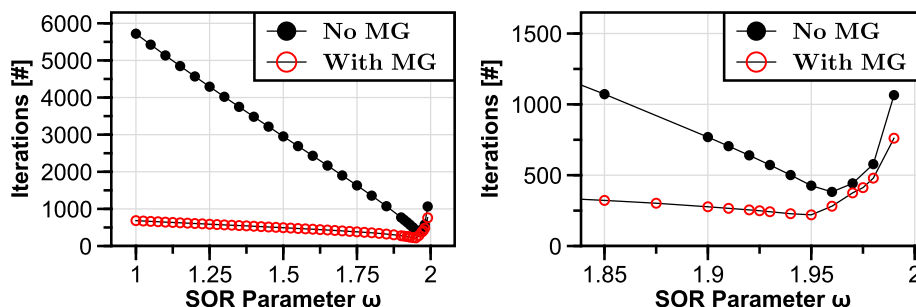
Figure 5.7: The effect of different SOR parameters on the number of full SOR iterations needed until convergence. The data shown here corresponds to the magnet case potential of Proto-APEX (see Figure 4.1) with $201 \times 201 \times 401$ grid points.

5.3.5 Symmetry

An obvious way to reduce the amount of system memory occupied by the potential arrays and to decrease the time to solve these potentials is to exploit mirror symmetries inside the domain. Looking at the geometry of Proto-APEX, a symmetry in z-direction can be seen (see Figure 4.1). It can also be seen that the geometries of the mainly used electrodes (Top1, RW1, E×B plates and the magnet case) are themselves symmetric in the z-direction and consequently so are their potentials, except for the ExB potential which is antisymmetric. When solving such a (anti-)symmetric the potential via SOR (see 5.3.3), special care must be taken at the symmetry plane which depends on the type of symmetry

(antisymmetric or not) and the number of grid points in the symmetry direction (even or not). This is accomplished by an additional boundary iteration after each SOR iteration. A pseudocode implementation of this is given in Algorithm 5 in Appendix E.2.

If the multigrid preconditioner is used (see 5.3.4), the methods for propagating potential values from a coarse grid to the next finer grid have to be adapted as well, like shown in Algorithm 6 in Appendix E.2.

Finally, to also ensure that a (anti)-symmetric potential is still defined in the whole domain, the methods to retrieve a calculated potential value at a specific grid point have to be modified. Implementations of both cases are given in Algorithm 7 in Appendix E.2.

When the symmetry of an electrodes is exploited, the system memory consumption of its potential is naturally halved and so is the time necessary to solve the potential since the SOR has to be applied to only halve the grid points. Another benefit is that the potential is actually symmetric compared to one where the symmetry is not explicitly used because the SOR iterations are done diagonally which causes a small asymmetry.

Algorithm 1: Complete algorithm to compute the potential on a grid using SOR with multigrid preconditioning and symmetry.

Input: The potential array: Φ
The size of the domain: nx, ny, nz
The multigrid preconditioning level: d (1,2,4,...)
The successive over-relaxation parameter: ω

Function: COMPUTEPOTENTIAL()

   $\Phi \leftarrow$ INITPOTENTIALARRAY(nx, ny, nz)

   level $\leftarrow 3$

   while level > 0 **do**

      $d \leftarrow 2^{\text{level}-1}$

      repeat

         ITERATIONSOR(Φ, d, ω)

         if Φ has a symmetry **then**

            ITERATEBOUNDARY$(\Phi, nx, ny, nz, d, \omega)$

      until Φ converged **or** max. # of iterations reached

      if level > 1 **then**

         PROLONGATE(Φ, d)

         if Φ has a symmetry **then**

            PROLONGATEBOUNDARY()

      level $\leftarrow$ level-1

5.4 Calculation of the electric field vector

To iteratively propagate a charged particle through a potential landscape, it is necessary to know the electric field vector at every location, as discussed in Section 5.1. In the following, multiple ways to differentiate a discrete scalar field are presented.

Before that, a few necessary functions have to be defined.

The electric field vector exactly at a grid point can be directly calculated with

$$\vec{E}_{x,y,z} = \begin{pmatrix} -(\Phi_{x+1,y,z} - \Phi_{x-1,y,z})/(2 \cdot d) \\ -(\Phi_{x,y+1,z} - \Phi_{x,y-1,z})/(2 \cdot d) \\ -(\Phi_{x,y,z+1} - \Phi_{x,y,z-1})/(2 \cdot d) \end{pmatrix} \tag{5.49}$$

This is the discretized version of Equation (5.40), where the gradient was replaced by the central difference quotient in the corresponding direction, with d being the grid spacing. At the domain boundaries, the central difference quotient has to be replaces with the forward and backward difference, respectively.

Assuming that x lies between the values p_0 and p_1, the value at position x can be linearly interpolated with

$$\mathrm{l}(p_0, p_1, x) = \left(1.0 - \frac{x - \lfloor x \rfloor}{d}\right) \cdot p_0 + \frac{x - \lfloor x \rfloor}{d} \cdot p_1 \tag{5.50}$$

where $0 \leq x \leq nx - 1$ with nx being the number of grid points in the domain in x-direction. This can be extended to two dimensions, also known as bilinear interpolation:

$$\mathrm{b}(p_0, p_1, p_2, p_3, x, y) = \mathrm{l}(\mathrm{l}(p_0, p_1, x), \mathrm{l}(p_3, p_2, x), y) \tag{5.51}$$

Linear interpolation in three dimensions is called trilinear interpolation:

$$\mathrm{t}(p_0, \ldots, p_7, x, y, z) = \mathrm{l}(\mathrm{b}(p_0, p_1, p_2, p_3, x, y), \mathrm{b}(p_4, p_5, p_6, p_7, x, y), z) \tag{5.52}$$

In Figure 5.8, the different linear interpolation steps are illustrated. In case that $p_0, \ldots, p_7$ are vectors, the trilinear interpolation is applied to each component of the vectors separately. It should be noted that the order in which the axes are interpolated does not have any influence on the result.

5.4.1 Nearest-neighbor

The easiest way to retrieve an electric field vector at an arbitrary position inside the domain is to find the closest grid point and use Equation (5.49) to calculate the field vector at that point:

$$\vec{E}(x, y, z) = \vec{E}_{\mathrm{nint}(x),\mathrm{nint}(y),\mathrm{nint}(z)} \tag{5.53}$$

Of course, this is only a very rough approximation and therefore hardly suitable for the simulation of trajectories. A possible application would be the fast visualization of field vectors, under the condition that the density of grid points in the viewing area is sufficiently high.
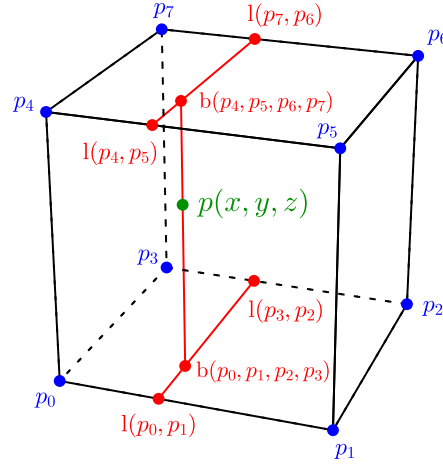
Figure 5.8: Illustration of trilinear interpolation at the point p (green) from the given values $p_0, \ldots, p_7$ (blue). Red points are intermediate results.

5.4.2 Difference quotient with bilinear interpolation

Another method to obtain an electric field vector within a cell is to first perform a bilinear interpolation on each side of the cell and then use these values to form the difference quotient for each direction:

$$
\vec{E}(x,y,z) = \begin{pmatrix} -\Big(\mathrm{b}\big(\Phi_{\lceil x \rceil, \lfloor y \rfloor, \lfloor z \rfloor}, \Phi_{\lceil x \rceil, \lceil y \rceil, \lfloor z \rfloor}, \Phi_{\lceil x \rceil, \lceil y \rceil, \lceil z \rceil}, \Phi_{\lceil x \rceil, \lfloor y \rfloor, \lceil z \rceil}, y, z\big) - \\ \quad \mathrm{b}\big(\Phi_{\lfloor x \rfloor, \lfloor y \rfloor, \lfloor z \rfloor}, \Phi_{\lfloor x \rfloor, \lceil y \rceil, \lfloor z \rfloor}, \Phi_{\lfloor x \rfloor, \lceil y \rceil, \lceil z \rceil}, \Phi_{\lfloor x \rfloor, \lfloor y \rfloor, \lceil z \rceil}, y, z\big)\Big)/d \\[2ex] -\Big(\mathrm{b}\big(\Phi_{\lfloor x \rfloor, \lceil y \rceil, \lfloor z \rfloor}, \Phi_{\lceil x \rceil, \lceil y \rceil, \lfloor z \rfloor}, \Phi_{\lceil x \rceil, \lceil y \rceil, \lceil z \rceil}, \Phi_{\lfloor x \rfloor, \lceil y \rceil, \lceil z \rceil}, x, z\big) - \\ \quad \mathrm{b}\big(\Phi_{\lfloor x \rfloor, \lfloor y \rfloor, \lfloor z \rfloor}, \Phi_{\lceil x \rceil, \lfloor y \rfloor, \lfloor z \rfloor}, \Phi_{\lceil x \rceil, \lfloor y \rfloor, \lceil z \rceil}, \Phi_{\lfloor x \rfloor, \lfloor y \rfloor, \lceil z \rceil}, x, z\big)\Big)/d \\[2ex] -\Big(\mathrm{b}\big(\Phi_{\lfloor x \rfloor, \lfloor y \rfloor, \lceil z \rceil}, \Phi_{\lceil x \rceil, \lfloor y \rfloor, \lceil z \rceil}, \Phi_{\lceil x \rceil, \lceil y \rceil, \lceil z \rceil}, \Phi_{\lfloor x \rfloor, \lceil y \rceil, \lceil z \rceil}, x, y\big) - \\ \quad \mathrm{b}\big(\Phi_{\lfloor x \rfloor, \lfloor y \rfloor, \lfloor z \rfloor}, \Phi_{\lceil x \rceil, \lfloor y \rfloor, \lfloor z \rfloor}, \Phi_{\lceil x \rceil, \lceil y \rceil, \lfloor z \rfloor}, \Phi_{\lfloor x \rfloor, \lceil y \rceil, \lfloor z \rfloor}, x, y\big)\Big)/d \end{pmatrix}
$$
(5.54)

This produces good results within a cell, but the values change discontinuously when the next cell is entered. This method is nevertheless useful when a short computation time is more important than accurate trajectories, such as for a quick assessment of the effect of an electrode voltage change on the general trajectory of a particle.

5.4.3 Trilinear vector interpolation

To obtain a globally continuous gradient one can first calculate the field vectors at the vertices of a cell using Equation (5.49) and then the field vector at each desired position within the cell by trilinear interpolation:

$$
\begin{aligned}
\vec{E}(x,y,z) = \mathrm{t}\Big(&\vec{E}_{\lfloor x \rfloor, \lfloor y \rfloor, \lfloor z \rfloor}, \vec{E}_{\lceil x \rceil, \lfloor y \rfloor, \lfloor z \rfloor}, \vec{E}_{\lceil x \rceil, \lceil y \rceil, \lfloor z \rfloor}, \vec{E}_{\lfloor x \rfloor, \lceil y \rceil, \lfloor z \rfloor}, \\
&\vec{E}_{\lfloor x \rfloor, \lfloor y \rfloor, \lceil z \rceil}, \vec{E}_{\lceil x \rceil, \lfloor y \rfloor, \lceil z \rceil}, \vec{E}_{\lceil x \rceil, \lceil y \rceil, \lceil z \rceil}, \vec{E}_{\lfloor x \rfloor, \lceil y \rceil, \lceil z \rceil}, x, y, z \Big)
\end{aligned}
\tag{5.55}
$$

This method is well suited for accurate propagation of particles in the electric field; it is, however, numerically quite expensive. In the case of non-time-dependent potentials, the field vectors at the vertices can be cached and reused for the next trilinear interpolation to reduce the computational cost as long as a particle is inside the same cell.

5.4.4 7-point stencil

The last possibility mentioned here to obtain the field vector is to form the difference quotient by taking a step of the length of half the grid spacing in each direction and determining the value of the electric potential at this point by trilinear interpolation.

$$
\vec{E}_{x,y,z} = \begin{pmatrix} -(\Phi(x+d/2,y,z) - \Phi(x-d/2,y,z))/d \\ -(\Phi(x,y+d/2,z) - \Phi(x,y-d/2,z))/d \\ -(\Phi(x,y,z+d/2) - \Phi(x,y,z-d/2))/d \end{pmatrix}
\tag{5.56}
$$

This is the method of choice here, since it delivers a globally continues gradient, like the previous one, but with better performance for time-varying potentials. In case d is chosen to be smaller than the grid spacing, the obtained results are very similar to Equation (5.54).

5.5 Defining the geometry

In most cases, the geometry can be defined by geometric primitives (like cuboids, cylinders, spheres...) that are simply placed in the 3D workspace to form the desired simulation boundaries. For these geometric primitives, it is mathematically straightforward to check whether a point is inside or outside the object, which is essential to initialize and solve the electric potential, as discussed previously, but also necessary to determine if a particle hit an object and consequently should be stopped.

More complex geometries can be created by using constructive solid geometry (CSG). With CSG, the geometry is described by a set of geometric primitives together with boolean operators (union, difference and intersection). This has the benefit that the object created maintains the accuracy of the geometric primitives used. For objects created by CSG it is also straightforward to check if a point is inside by simply checking the point against all primitive geometries while considering the same boolean operators.

If even more complicated structures are needed that cannot be easily generated by CSG, e.g. the stellarator shape of EPOS, triangle meshes can be used. However, finding out whether a point is inside the mesh is not trivial. A common way is to start a ray from the point in question and count the number of intersections with the object. If it is an odd number then the point is inside the mesh, otherwise it is outside. Another option is to check if the normal of the triangle at the closest intersection has the same (inside) or opposite (outside) direction of the ray. In the STL file format, a widely used format to exchange triangulated surfaces, the vertices v_1, v_2, v_3 of a triangle are ordered by the right-hand rule in such a way that the normal n calculated by $\vec{n} = (v_2 - v_1) \times (v_3 - v_1)$ is pointing outwards. One can see that both approaches are computationally very expensive, since for each point to be tested, all triangles must be checked to see if they intersect the test ray, for example via the Möller–Trumbore ray-triangle intersection algorithm [54]. Instead of testing every triangle, it is therefore much more efficient to partition the mesh into voxels and only test the triangles in the voxels that the test ray traverses. Here, a simple regular grid is used for partitioning, but nested grids, octrees (see Chapter 6) and especially k-d trees are also common choices. The first obstacle in using a regular grid is determining the grid resolution. A widely used scheme is given by [55]

$$N_x = d_x \sqrt[3]{\frac{\lambda N}{V}}, \quad N_y = d_y \sqrt[3]{\frac{\lambda N}{V}}, \quad N_z = d_z \sqrt[3]{\frac{\lambda N}{V}} \tag{5.57}$$

with the grid resolution $N_{x,y,z}$, the size of the bounding box of the specific dimension $d_{x,y,z}$, the volume of the bounding box V, and the number of triangles N. For the free parameter λ, which is used to optimize the performance, values between 3 and 5 seem to be a good choice [56]. In the next step, the triangles have to be sorted into the created voxels. This can be done by taking the bounding box of the triangle and adding it to every voxel the bounding box intersects. A more accurate method would be to actually check if the triangle overlaps the voxel [57] but the cost for that usually outweighs the benefits [56]. Once all triangles have been sorted, one can already test if a point is inside the mesh by finding the voxel that contains the point, picking any standard-basis vector (preferably the one pointing towards the closest side of the bounding box) and checking all voxels in that direction for triangle intersections. After all intersections have been found it can directly be determined if the point is inside by comparing the direction of the chosen basis vector to the normal of the closest face, as mentioned earlier. Checking the parity of the number of intersections, however, is usually not straightforward, since triangles may be assigned to multiple voxels and therefore be tested for intersection multiple times. This can be avoided by, for example, mail-boxing [58], where each triangle keeps a note of whether it has already been tested, or by removing all duplicate values from the list of intersections. If one is more interested in finding the intersection points of the ray with the triangle mesh (as depicted in Figure 5.9), the initial step is to find the voxels that the ray traverses. This can be accomplished by using an 3D extension of a Digital Differential Analyser (DDA), which are typically used to rasterize lines in computer graphics. A common 3D-DDA is presented by Amanatides et. al [58]. Like mentioned before, special care must be taken to avoid duplicated intersection points.
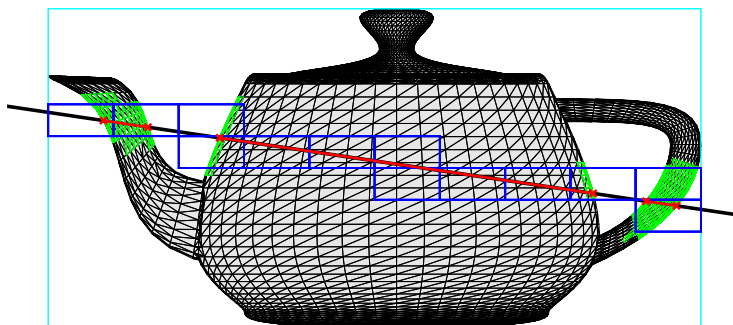
Figure 5.9: Illustration of a ray (black line) intersecting a triangle mesh[1]. The bounding box of the mesh is shown in cyan. The voxels that are traversed by the ray are shown in blue and the tested triangles are highlighted in green. For illustration clarity, a low resolution ($N_{x,y,z} = 10$) of the regular partition grid was chosen instead of calculating it with Equation (5.57). The intersection points of the ray with the mesh and the sections of the ray inside the object are shown in red.

5.6 Image based sample generation

To obtain good predictions, e.g. for the injection efficiency, it is essential to know the spatial extent of the positron beam. As a good first approximation, one can assume a Gaussian distribution of the beam [29]. However, since there are strong variations between beam times caused by upstream changes, such as modifications of the remoderator or the steering of the beam to the experiment, it would be useful to use a microchannel plate (MCP) image (such as the example shown in Figure 5.10), which is typically obtained during the beam centering process at the beginning of each beam time, to generate a more realistic spatial distribution.

To generate sample positions that follow an image-based distribution, inverse transform sampling is a good choice. This works by first generating a uniformly distributed random number x between 0 and 1, and then by calculating $C^{-1}(x)$, the inverse of the cumulative distribution function C. For a discrete distribution, C can be calculated in a straightforward manner by just accumulating the individual probability values.

In the case of an image, the probability values are determined by the brightness of the individual pixels (see Figure 5.10d and Algorithm 8 in Appendix E.3). After calculating the cumulative distribution function and storing the result in an array, $C^{-1}(x)$ can be simply evaluated by searching for the best insertion index of x into the array. The index found corresponds directly to a pixel of the image, which in turn provides a location. Since an image provides only a discretized spatial distribution limited by the resolution of the image, the location within a pixel is undefined. It can be assumed, however, that values within a pixel are uniformly distributed (see Algorithm 9 in Appendix E.3).

---

[1]"Utah teapot (solid).stl". Created by Nik Clark. Distributed under Creative Commons CC0 1.0. Wikimedia Commons. 27 May 2015.
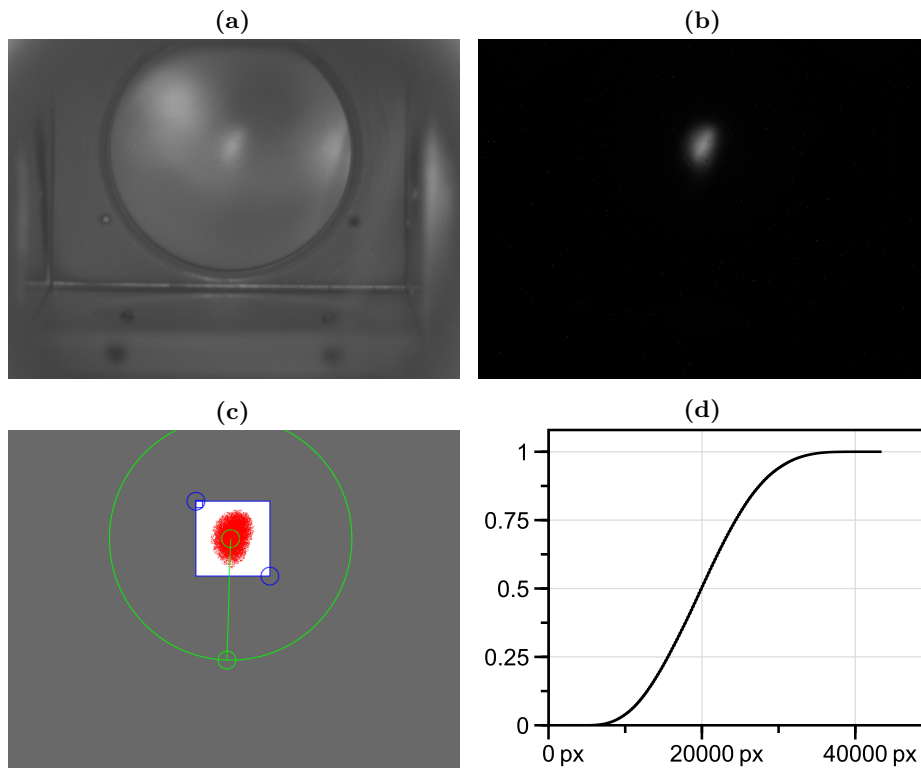
(a)

(b)

(c)

(d)

Figure 5.10: Figure (a) shows an image of the illuminated MCP. In (b), the illumination was turned off, clearly showing the positron beam hitting the MCP. The green line in (c) serves as reference scale (12.5 mm). The green circle is just a help to find the center of the MCP more easily. The blue rectangle limits the sampling area considered. Test sampled are shown in red. The normalized, cumulated brightness values for the selected area are shown in (d).

5.7 Summary

This chapter introduced all the components necessary to simulate single particles in static electromagnetic fields, namely expressions to analytically compute the magnetic fields of all coils used in Proto-APEX, a solver for electric potentials (including various acceleration methods), several methods to differentiate the electric potentials to obtain a field vector, and a particle pusher to propagate particles through these fields. It was also explained how the geometry can be defined in AlGeoJ either by primitive objects, by using these primitives to create more complex objects via CSG, or by loading STL files. Finally, it was shown how to extract a 2D distribution from an image, which can be used to generate realistic positron starting positions based on a MCP image of the NEPOMUC beam.

In the next chapter, a potential solver with a more sophisticated data structure will be presented, which will be able to resolve finer details of the geometry more efficiently than the regular grid solver presented here, by only locally increasing the grid density.

The Chapters 7 to 11 will all focus on the single-particle simulations of positrons in Proto-APEX, all of which have been carried out using this simulation framework.

In the introduction to this chapter it was mentioned that this simulation framework can be used to visualize trajectories and discover positron loss channels that would be very difficult to determine experimentally. Examples of these use cases are given in Appendix D and Appendix F, respectively. Last but not least, the user interface of AlGeoJ is shown and explained in Appendix G.

Chapter 6

Octree-based field solver

The regular cartesian grid has been found to be a very efficient method for modeling the electrical potential in most cases. However, geometries that are smaller than the grid size, curved surfaces or geometries that are angled in relation to the grid cannot be accurately reproduced. Reducing the grid size will alleviate these problems but only to a limiting degree due to the aggressive $\mathcal{O}(n^3)$ scaling between the grid size and the number of grid points and memory consumption, respectively. A more elegant approach would be to only increase the grid resolution locally where it is beneficial, especially near surfaces, but keeping a lower resolution everywhere else.

The method of choice here is the octree with cell-centered values where some cells are subdivided into eight equal sub-cells therefore the name oct(Greek) + tree = eight + tree. The octree is the three-dimensional extension of the binary tree, whereas the quadtree is its two-dimensional variant. Compared to an unstructured mesh, the locations of the cells follow a logical order, which should result in smaller memory footprint and higher performance. The relation with the Cartesian mesh also allows the procedure to be reused, with adjustments, for solving the potentials, as described in Section 5.3.3.

6.1   Terminology

Before addressing the various aspects of implementing an octree-based electric potential, it is necessary to define some terminology first, which is illustrated in Figure 6.1.

At the base of every octree there is a root cell. If a cell is subdivided into child cells, it is called a parent cell. In the case of the octree, a parent cell has always exactly eight child cells and therefore a degree of 8. If a cell is not subdivided, it is called a leaf cell with degree 0. Here, the child cells are labeled from A to H to encode their spatial position within their parent cell (see Figure 6.6). Every cell has neighboring cells. If the neighboring cell has the same parent, it is a sibling cell. The level of a child cell is one higher than the level of its parent cell, with the root cell being at level 0.
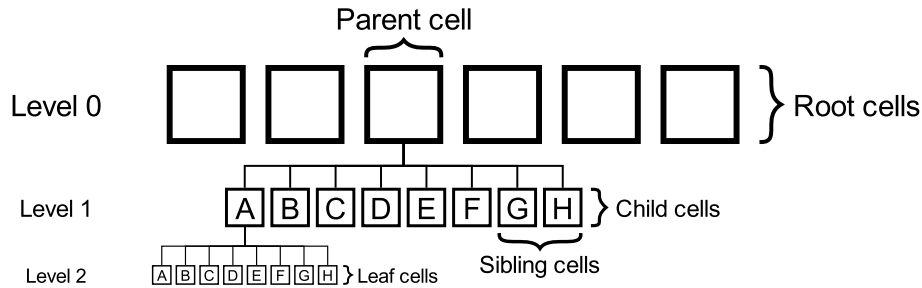
Figure 6.1: Illustration of the terminology used to describe tree data structures.

Since the electrical potential requires a minimum resolution, it would be very inefficient to start with just one root cell and keep subdividing it until the desired base resolution is reached, as this would cost valuable memory to store this unused portion of the tree, and would also make it more expensive to find a particular cell within the tree. For these reasons, a full 3D array of root cells is initialized at the desired minimum resolution, which is equivalent to the case of the regular grid. Since many root cells will be subdivided at the end of the refinement process, it would be more accurate to speak of a forest of octrees, but for simplicity the data structure will be referred to simply as octree in the following.

Algorithm 2 shows an example of an implementation of a cell class in Java with all variables; however, depending on the cell type, not all are needed.

Algorithm 2: The cell class implemented in Java.

```
public class Cell {
    int        xi, yi, zi;    \\Location in the root cell array
    Cell       parent;
    Cell [ ]   childs;
    byte       level;
    boolean    geometry;
    float [ ]  potential;
}
```

6.2 Tree balancing

To simplify the handling of octrees, a balanced tree is often desired. This means that the subdivision level of a cell differs from all neighboring cells by a maximum of ±1 (see Figure 6.2). This condition has the consequence that all cells of the boundary of the domain must not be subdivided. To achieve and maintain a balanced tree, the easiest way is to check the levels of all the neighbors of a cell each time it is subdivided, and then recursively subdivide them if necessary.
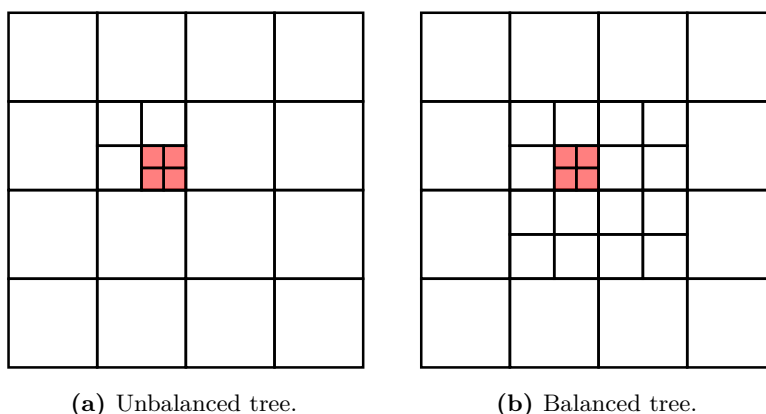

| (a) Unbalanced tree. | (b) Balanced tree. |

Figure 6.2: In an unbalanced tree (a), the subdivision level of a cell is not subject to any rules. In a balanced tree (b), the subdivision level of a cell may only differ by ±1 from all its neighbors.

6.3 Grid refinement

The first step in solving the potential on an octree is to determine which cells should be subdivided and which not. Currently, only static geometries are taken into account, especially their surfaces, since the interior of an object is not important for the simulation.

One option to find cells that are close to the surface is to check if the center of a cell is within an object or not, as described in Section 5.3.2, but with the addition of also checking all eight corners of a cell. If at least one corner, but not all, are within the object then the surface intersects the cell, and the cell should therefore be subdivided. This has the problem, however, that cells that are outside an object but very close to its surface are not considered for subdivision despite being able to deliver an overall better coverage of the geometry. The result can slightly be improved by also checking if the cell lies within a threshold to the object. To reach higher refinement levels, the process can be repeated for the previously subdivided cells. This method falls apart, however, when one dimension of the object is smaller than the grid size, for example a thin plate.

Another way to refine the octree, which is not afflicted by the previously mentioned issues, is a ray-tracing-based scheme, illustrated in Figure 6.3. First, the bounding box of the object is found to limit the volume of interest. Then,
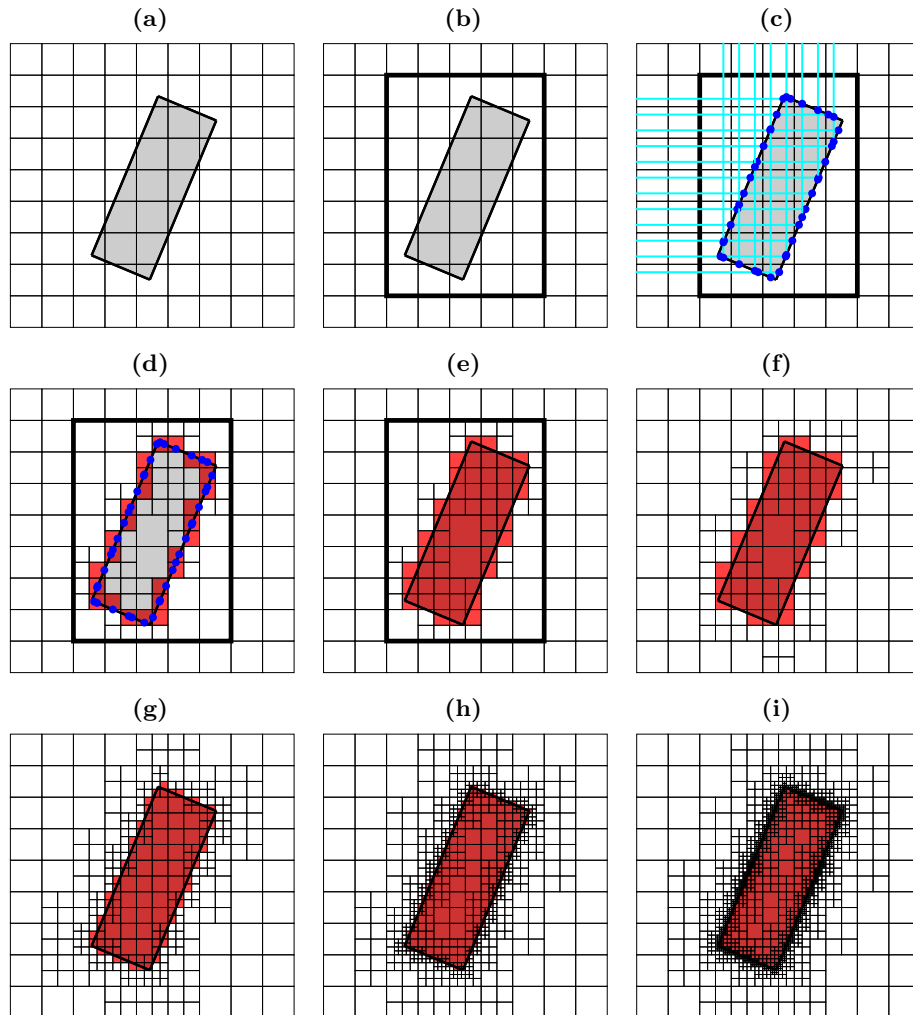
Figure 6.3: Overview of the different octree grid refinement steps: (a) before refinement, (b) bounding box, (c) finding surface points via ray tracing, (d) subdividing cells and marking them as geometry, (e) mark remaining cells inside the object as geometry. (f) Repeated refinement with slightly larger object. (g) two level subdivision, (h) three level, (i) four level.

the surface of an object is sampled at the desired resolution with axis-parallel rays from each direction. The surface samples found can be used to directly determine their corresponding cells, which can subsequently be subdivided and marked as geometry. If the desired, the procedure can be repeated with a slightly larger object to guarantee that it is completely surrounded by cells at the highest refinement level that are not tagged as geometry. Last but not least, since only the surface of an object is processed, all cells inside the bounding box must be checked to see if they are inside the object so that they can be marked as geometry cells as well. This ray-tracing-based refinement method also works for two-dimensional objects, in contrast to the previously mentioned method, but

it fails to refine one-dimensional objects. This is not a critical issue, however, since one-dimensional geometries do not occur in the experimental setup. If they do, a simple solution would be to directly generate samples along the object and refine the octree with them. Another advantage of ray tracing is that finding the surface points is very easy even for complex objects that are created using constructive solid geometry (CSG) [59].

6.4   Calculation of the potential

After the octree has been refined and all eligible cells have been marked as geometry and initialized with the desired voltages, solving the potential can be addressed. Since the grid spacing is no longer constant, but instead changes as one moves from coarse to refined cell regions, and vice versa, the discretization of the potential, as described in Section 5.3.1, must be re-derived.

The second order central difference quotient of a function $f(x)$ for unequal sub-intervals is given by [60]:

$$f''(x) = \frac{2\left[h^+ f(x - h^-) - (h^- + h^+)f(x) + h^- f(x + h^+)\right]}{h^- h^+ (h^- + h^+)} \tag{6.1}$$

In case of equal sub-intervals $h^+ = h^- = h$, this simplifies to the usual second order difference quotient as given in Equation (5.43).

With Equation (6.1), the Laplace equation can then be solved for the potential again, resulting in the octree equivalent of Equation (5.46):

$$\Phi(x, y, z) = \left(\frac{1}{h_x^+ h_x^-} + \frac{1}{h_y^+ h_y^-} + \frac{1}{h_z^+ h_z^-}\right)^{-1} \cdot$$
$$\left[\frac{\Phi(x + h_x^+, y, z)}{h_x^+ (h_x^+ + h_x^-)} + \frac{\Phi(x - h_x^-, y, z)}{h_x^- (h_x^+ + h_x^-)} + \frac{\Phi(x, y + h_y^+, z)}{h_y^+ (h_y^+ + h_y^-)} + \right. \tag{6.2}$$
$$\left. \frac{\Phi(x, y - h_y^-, z)}{h_y^- (h_y^+ + h_y^-)} + \frac{\Phi(x, y, z + h_z^+)}{h_z^+ (h_z^+ + h_z^-)} + \frac{\Phi(x, y, z - h_z^-)}{h_z^- (h_z^+ + h_z^-)}\right]$$

With this, the successive over-relaxation method (see Equation (5.48)) is then given by:

$$\Phi_{x,y,z}^{k+1} = (1 - \omega) \cdot \Phi_{x,y,z}^k + \omega \cdot \left(\frac{1}{h_x^+ h_x^-} + \frac{1}{h_y^+ h_y^-} + \frac{1}{h_z^+ h_z^-}\right)^{-1} \cdot$$
$$\left[\frac{\Phi(x + h_x^+, y, z)}{h_x^+ (h_x^+ + h_x^-)} + \frac{\Phi(x - h_x^-, y, z)}{h_x^- (h_x^+ + h_x^-)} + \frac{\Phi(x, y + h_y^+, z)}{h_y^+ (h_y^+ + h_y^-)} + \right. \tag{6.3}$$
$$\left. \frac{\Phi(x, y - h_y^-, z)}{h_y^- (h_y^+ + h_y^-)} + \frac{\Phi(x, y, z + h_z^+)}{h_z^+ (h_z^+ + h_z^-)} + \frac{\Phi(x, y, z - h_z^-)}{h_z^- (h_z^+ + h_z^-)}\right]$$

The choice of h and the used potential value Φ is determined by the subdivision level of the cell in that specific direction. For a balanced octree, three cases are possible:

Case: cell level < neighbor cell level

If the neighboring cell is one level further subdivided than the current cell (see Figure 6.4a), then $\Phi(x, y, z)$ is the average of the four neighboring cell values and $h = 0.75$. Since the octree is balanced, these neighboring cells are not allowed to be further subdivided, so no other cell level combinations have to be considered.

Case: cell level = neighbor cell level

This case is straightforward, as it is equivalent to the regular grid. $\Phi(x, y, z)$ is simply the value of the neighboring cell and $h = 1$ (Figure 6.4b).

Case: cell level > neighbor cell level

The most difficult case is when the neighboring cell has a lower level. Here, $\Phi(x, y, z)$ has to be interpolated as seen in Figure 6.4c with $h = 1.5$. The difficulty arises from the fact that the three cells surrounding the neighboring cell could also be subdivided and thus have the same level as the cell currently being considered. Therefore, all eight combinatorial possibilities have to be treated individually. Fortunately, by looking at Figure 6.8 and using symmetry transformations, these can be reduced to just four different interpolations that need to be implemented.
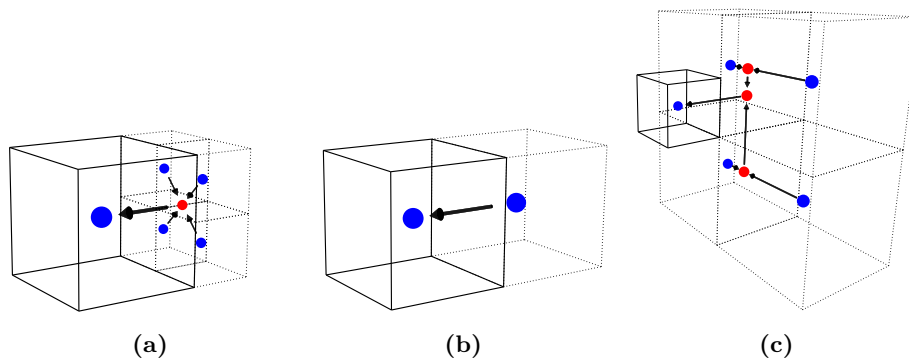

(a)	**(b)**	**(c)**

Figure 6.4: Overview of the three different types of neighboring cells (shown with dotted lines) in a balanced octree: (a) higher level, (b) same level, (c) smaller level. Blue dots represent existing values, red dots represent interpolated values. The three cells in case (c) that do not touch the black cell may also be subdivided.

The SOR method can now be applied iteratively to each cell, or recursively if the cell is subdivided. During the iteration, all h and Φ can be cached in a temporary object instead of local variables to reduce the workload of Java's garbage collector. To check when the potential has converged, several test cells are selected at the beginning. Before each iteration, the current values of the potential are cached in order to compare them with the new values after the iteration. If for all test cells the difference between the two values is smaller than a certain threshold, the potential can be considered as converged. In the current implementation, 26 cells are selected as test cells along each of the three principal axes, the six diagonals in the principal axis planes, and the four space diagonals of the domain.

6.4.1 Multigrid preconditioning

Just like in the case of the regular grid, the potential in an octree can be preconditioned using the multigrid method. The method implemented here is relatively basic, since it ignores the refinement and only uses the root cells at the lowest level. The problem with this, however, is that refined parent cells should not store potential values, because this would unnecessarily take up memory space as only values of cells at the highest refinement level are used by the SOR and the simulation in general. The solution to this is to use the value of a child cell at the highest refinement level instead and, after performing the preconditioning, to copy the value of this child cell to its sibling cells. The potential can be solved with the above described SOR method afterwards.

6.4.2 Load balancing for multithreading

To further reduce the time needed to numerically solve the electric potential, multithreading can be utilized also for the octree. For the octree, however, it was implemented in a different way compared to the regular grid. For regular grids, each potential is stored in its own 3D array that can be processed by one thread, but no more. This means that if more threads are available than there are potentials, the full performance of the CPU is not utilized, which is quite often the case since modern (2022) mid- to high-end CPUs usually have eight threads or more, but only about four potentials are present in the simulation.

The version implemented here is based on assigning approximately the same number of cells to each thread for processing. This is done by slicing the 3D root cell array in one direction and then determining the number of cells in each slice. These slices are then grouped into blocks in such a way that they contain roughly the same number of cells. Two constraints can limit the number of blocks and thus the number of working threads: Each block should consist of at least four slices to avoid interference between two threads if they would process adjacent slices, and it should also contain at least 100 000 cells, because starting a thread always involves an overhead. However, these constraints have a very small impact on the actual solution time, since they only come into effect for small octrees, which can be solved quickly anyway.

6.5   Potential interpolation

Now that the electric potential can be solved, there is still a way missing to interpolate the potential between the cell centric values. The method chosen here is based on trilinear interpolation, just like in the case of the regular grid. The difficulty, however, is that each corner cell can either be subdivided or not, resulting in $2^8 - 1 = 255$ possible variations. Fortunately, these can be reduced to 21 unique variations, shown in Figure 6.5, by exploiting rotational and reflective transformations. This is one of the reasons why the octree is required to be balanced, otherwise there would be exponentially more variations, which would render this approach impractical.
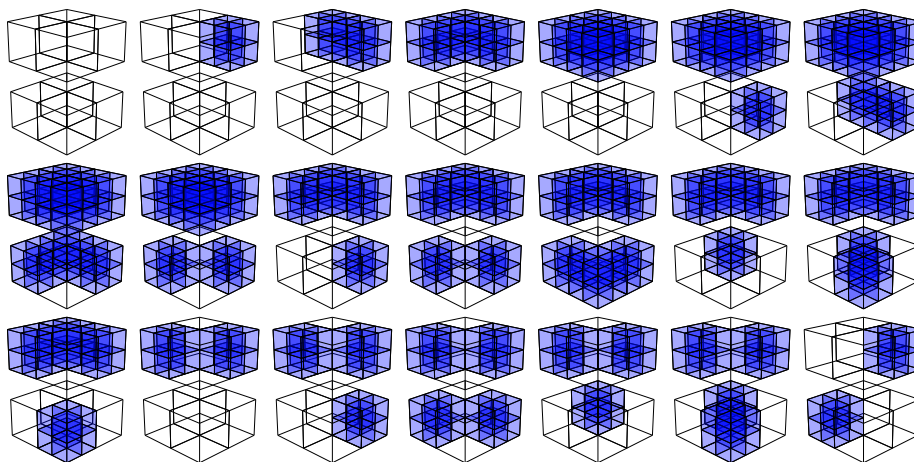

Figure 6.5: The 21 unique permutations of cells under rotation and reflection transformation in a balanced octree.

The interpolation procedure starts by finding the nearest vertex that is surrounded by exactly eight cells. These cells can then be checked in the order shown in Figure 6.6 (or any other as long as the same order is consistently used) to see if they are subdivided or not. This allows the cell variation to be determined and any necessary transformation to be applied, which is described in more detail in Algorithm 10 in Appendix E.4. Before the trilinear interpolation can be performed, the irregular volume spanned by the cell centers must be partitioned into rectangular cuboids, as exemplified in Figure 6.7 for the case of one subdivided cell. The partitioning into cuboids is not arbitrary, since it must be ensured that the interpolated value is continuous at the transition between adjacent volumes. To ensure this, the partitioning is predefined for each possible interface, as shown in Figure 6.8. Last but not least, the vertices of the corresponding cuboid that do not coincide with cell centers have to be interpolated (by linear or bilinear interpolation of neighboring cells), then the final trilinear interpolation can be performed. Once all 21 permutations have been partitioned and the corresponding interpolation rules have been implemented, continuous values of the potential can be retrieved in the whole domain, as demonstrated in Figure 6.9.
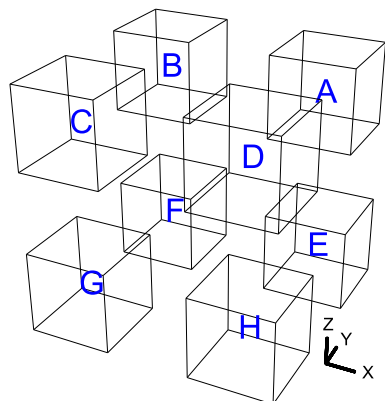
Figure 6.6: The cell naming scheme used for children cells as well as for cells that surround a common vertex like seen in Figure 6.7. In the code, A–H were defined as constants with values 0–7. This allows for a more understandable handling of cell configurations.
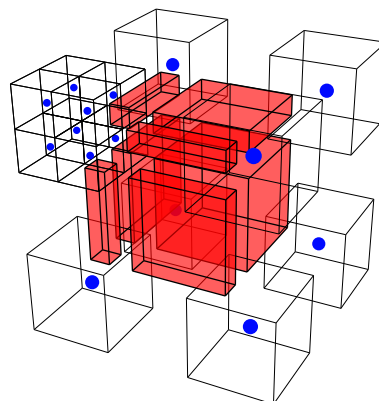
Figure 6.7: For interpolation, the volume between cells is split up into smaller volumes (red cuboids) in which trilinear interpolation can be easily performed. The blue dots represent the existing values. How the volume is split up on each side is constrained by its specific type, as shown in Figure 6.8.
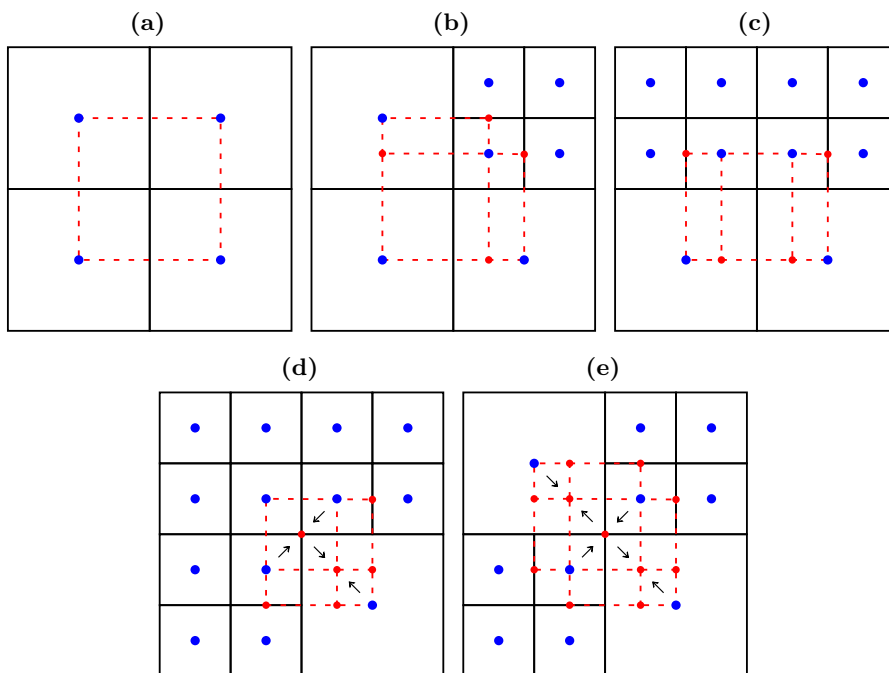

Figure 6.8: The five possible interfaces between cells. Blue dots represent existing values, red dots represent interpolated values. The red dashed lines visualize how the volume must be divided at this boundary.
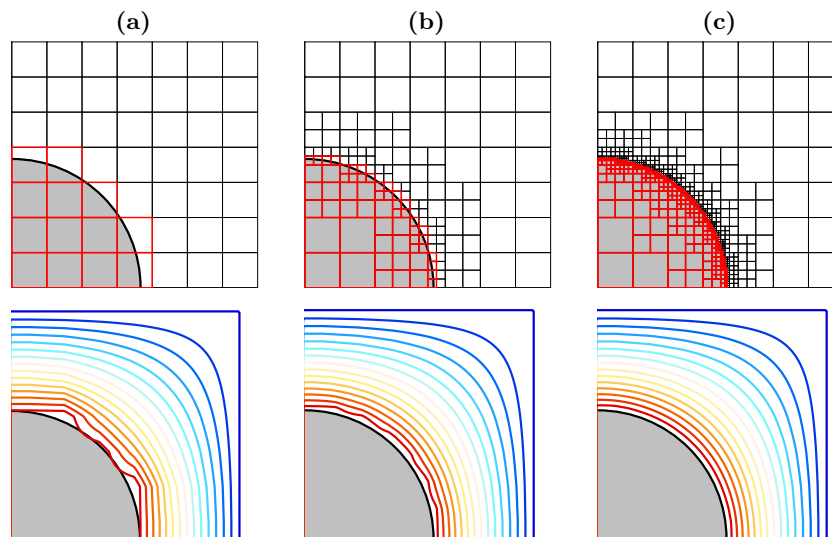
Figure 6.9: Illustration of the grid (top) and the interpolated potential (bottom) of a charged cylinder (grey, only a quarter section is shown) on (a) a regular grid, (b) an octree at level 2 and (c) at level 4.

6.6   Calculation of the electric field vector

As a final step, before simulations can be performed, a method must be implemented to calculate the electric field vector.

6.6.1   7-point stencil

The simplest method is to use the 7-point stencil explained in Section 5.4, since it requires only potential evaluations. To avoid discontinuities in the transition between regions with different refinement levels, d (see Equation (5.56)) must be chosen smaller than half the length of the smallest cell.

6.6.2   Adaptive 7-point stencil

The 7-point stencil method can be improved by adaptively choosing d based on the level of the cell at the evaluation point as well as on the levels of the three closest neighbor cells in each direction. An implementation of this adaptive 7-point stencil is outlined in Algorithm 11 in Appendix E.4.

6.6.3   Difference quotient with bilinear interpolation

Another possible way to calculate the electric field vector is based on the difference quotient method already presented in Section 5.4. The evaluation is performed in the same partitioned cuboids that are used for the potential interpolation like shown in Figure 6.7. However, the implementation effort is very high, since all 21 possible cell permutations (see Figure 6.5) have to be dealt with separately, as already for the interpolation. For this reason, this method has not yet been implemented, but it promises to be significantly faster than the other two methods, since finding the eight cells to determine the permutation is the most costly part and this only needs to be done once instead of six times.

6.7   Outlook

The main focus of the current implementation was functionality, which is why there is some room for optimization, especially in memory usage. The next step will be to create a separate class for each individual cell type with only the required variables, instead of a single class for all types containing a correspondingly large number of unnecessary variables, as shown in Algorithm 2. Another option is to encode the hierarchy, position, and level in a single, unique ID [61], which would allow the cells to be stored in a hashmap. This should reduce the memory requirements by quite a bit at the cost of slower retrieval time of these variables due to the additional decoding step. Furthermore, it should be easy to exploit a symmetry within the domain, if such a symmetry exists.

Since Java is constantly being developed further, an increase in performance is also possible in the future, e.g. with JEP (Java Enhancement Proposal) 401 [62], which has the goal to allow the creation of primitive classes.

Currently, only the surfaces of objects are refined. One idea would be to extend this to areas with steep gradients by using information from the preconditioning step, which would allow a more accurate particle propagation in these areas.

Chapter 7

Continuous injection of positrons

In previous experiments, lossless injection of a 5-eV positron beam into Proto-APEX was demonstrated [12]. This was achieved by optimizing the effective potential energy, which is explained in more detail below. The primary diagnostic to investigate these injected positrons is the insertable target probe, making it important to examine to what extent it can possibly falsify the measurement of the radial distribution. After that, a study is presented on how the radial distribution can be manipulated by changing the free parameters of the experiment. The results obtained during this work and presented here have also been previously published in [63].

7.1   Effective potential energy

The range of motion of a single charged particle in our dipole traps is limited by a combination of electrostatic reflection and magnetic mirroring. An elegant way to describe this is in terms of the effective potential energy [64]

$$U_{eff} = e\phi_E + \mu B \tag{7.1}$$

experienced by a particle with charge e, magnetic moment $\mu = mv_\perp^2/(2B)$, mass m, and perpendicular velocity $v_\perp$ in an electric potential ϕ_E and magnetic field B. Since the magnetic moment can generally be assumed to be an an adiabatic invariant, the effective potential boundary associated with a particle's total energy therefore limits the guiding center drift motion of said particle. In Figure 7.1, it can be seen how this can be used to visualize where particles are reflected as they stream along the magnetic field lines. It shows vertical slices of three simulated trajectories of positrons with the same initial total energy but with different initial pitch angles. The dashed contours indicate surfaces at which the effective potential energy of a particle is equal to its total energy, i.e., the boundaries at which the particle is reflected. This underlines how important the perpendicular energy spread of the incoming positron beam is, as positrons with a high perpendicular velocity may not be successfully injected, especially if

the magnetic field is stronger like in the upcoming APEX-D experiment. It also shows that particles with larger pitch angles are excluded from areas close to the magnet where stronger magnetic fields are present, causing them to be deposited further out, potentially reducing the confinement time of these particles, as described in more detail in Chapter 8.
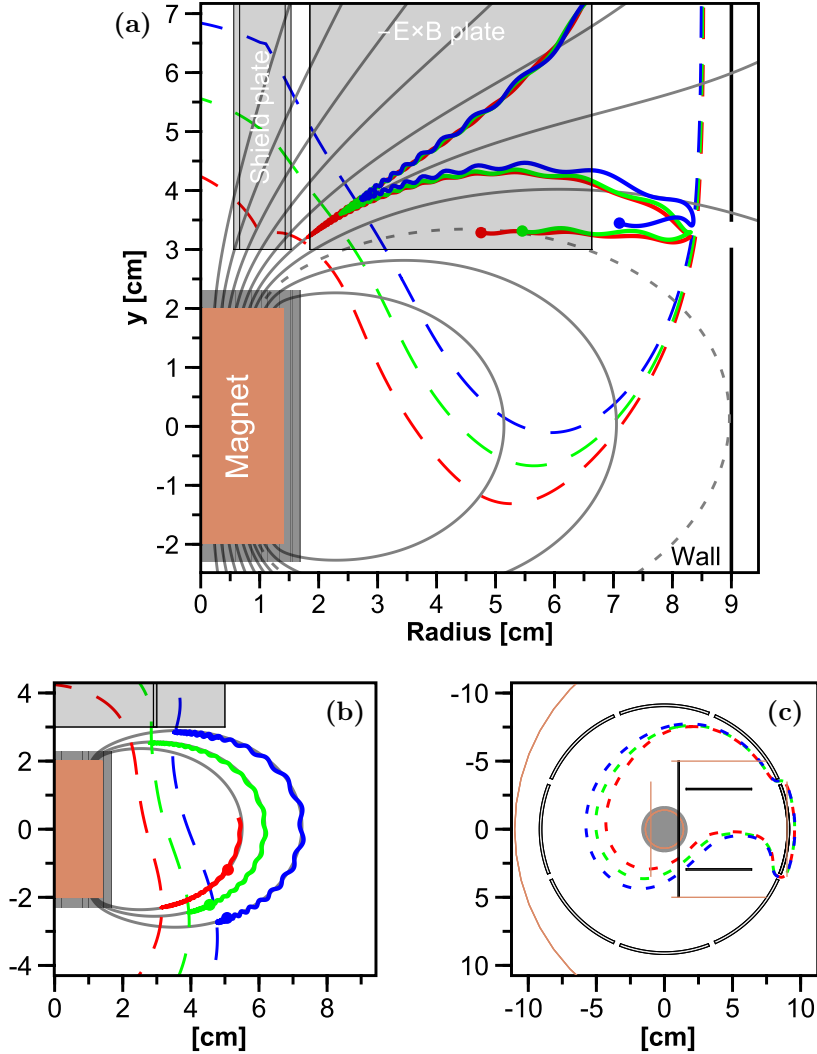

Figure 7.1: Surfaces of maximum effective potential energy (dashed lines) for positrons with $E_0 = 5\,\mathrm{eV}$ and different initial pitch angles $\alpha = \arctan(v_{\perp 0}/v_{\parallel 0})$. Red: $\alpha = 10°$, Green: $\alpha = 20°$, Blue: $\alpha = 30°$. (a) shows the trajectories of these positrons during drift injection projected onto the vertical plane at a toroidal angle of $5°$. The dot at the end of the trajectory indicates the particle position at this plane. The dotted field line (grey) defines the boundary of the confinement region. (b) shows the most recent parts of the trajectories of the same particles at a toroidal angle of $90°$ with the corresponding field lines and effective potential energy surfaces. (c) shows the extent of the effective potential energy surfaces at the mid-plane [63].

7.2 Perturbation caused by the target probe

The insertable target plate is the main diagnostic for determining the positron injection efficiency and additionally allows the measurement of the radial distribution of injected positrons. The probe is used two ways: by measuring the positron current it collects or by counting the positron annihilation photons, using the collimated BGO detector viewing the probe (see Figure 4.1b). An important question raised by this diagnostic is how strongly positron trajectories are affected by the insertion of a grounded electrode into the already complex 3D electrostatic landscape of the confinement region.
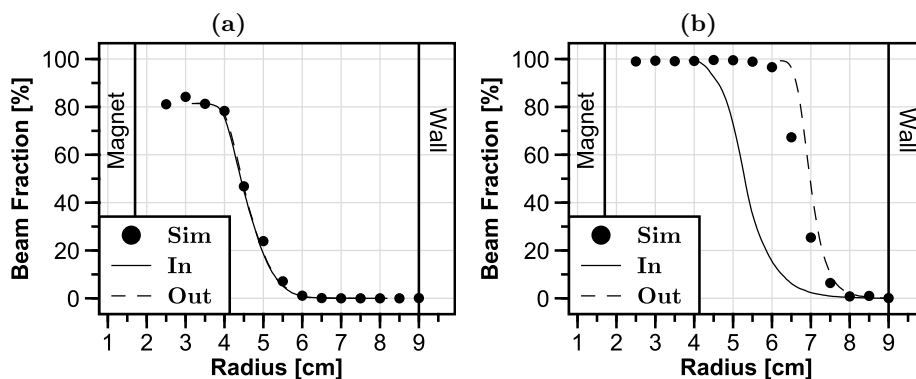

Figure 7.2: Simulation of the radial positron distribution at a magnet case bias of 0 V (a) and 30 V (b). The dashed line ("Out") shows the cumulative distribution in the unperturbed case with the target probe retracted. The solid line ("In") shows the cumulative distribution of the maximal perturbed case with the target probe fully inserted. The black dots ("Sim") show the simulated counts on a target probe that is inserted step wise from the wall towards the magnet, as it is in the experiment. Inserting the probe into the trap creates integrated profiles, so the true radial distributions are their derivatives (Figure adapted from Figure 2 in [63]).

An answer to this question is provided by the simulation results of two extreme cases shown in Figure 7.2. Three different simulation methods are compared for the scenario where the magnet case is biased to either 0 V (a) or 30 V (b). The graphs labeled as "In" and "Out" correspond to simulations where the target probe was either fully inserted (like in Figure 7.3) or fully retracted. For the data points labeled "Simulation" the target probe was stepwise inserted into the trap (just like in the experiments) and the potentials recalculated for every position. Looking at Figure 7.2a, it is immediately apparent that all three simulation methods produce the same result, confirming the assumption that other potentials within the trap, except the magnet case bias, have no influence on the measurement of radial distribution of positrons. Figure 7.2b demonstrates that when the magnet case is biased, the positron trajectories are significantly affected by the fully inserted target probe.. However, this effect is significantly weakened if, as in the experiment, the measurement is performed by a stepwise insertion of the target probe ("Simulation"). Moreover, since the radial distribution of the

injected positrons is farther away from the magnet, the higher the bias voltage is (see Figure 7.8), this reduces the perturbation effect even further. Negative voltages were not investigated as they are generally unfavorable for efficient injection and unsuitable for long-term confinement.
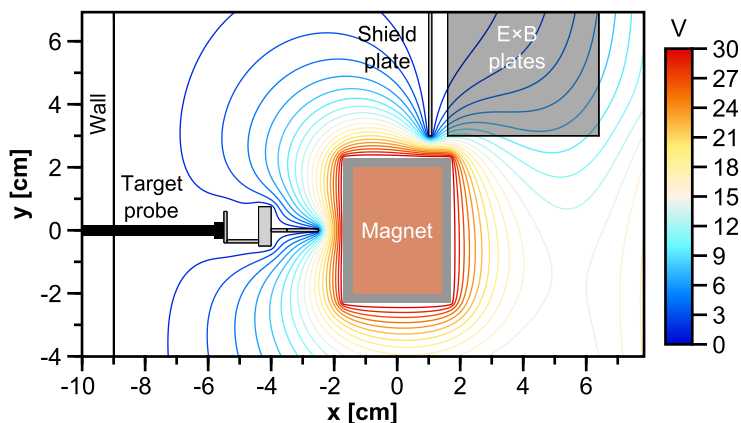

Figure 7.3: Perturbation of the electric potential caused by the grounded target probe if the magnet case is biased to 30 V [63].

In a comparison of experiment and simulation, which is shown in Figure 7.4, the positron injection efficiency and the mean position of the radial distribution could be reproduced by the simulation, although the spread of the distribution is not as large as in the experiment. Uncertainties of the experiment which are not taken into account in the simulation might explain this discrepancy, such as, among others, a tilting of the magnet, an angled positron beam at the injection port caused by misaligned beam line coils, Earth's magnetic field, or other stray fields present in the experimental hall.

Thus, although the grounded target probe may indeed affect the measurement of the radial distribution of positrons due to the deformation of the electrostatic landscape when the magnet case is biased, it is nevertheless an important and useful tool to determine the positron distribution and injection efficiency.
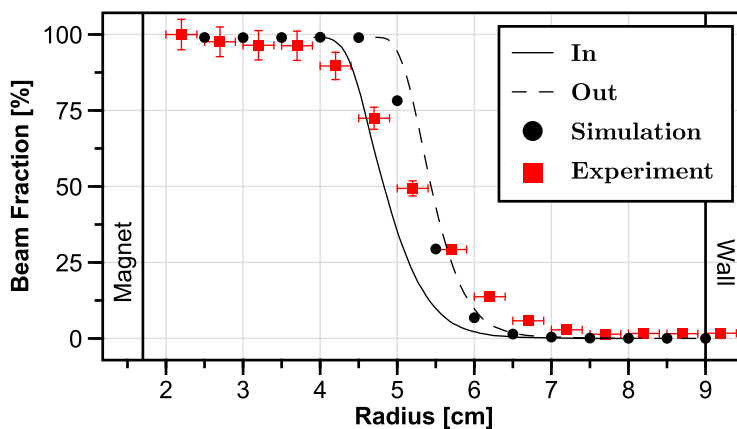
Figure 7.4: Comparison of an experimentally measured radial positron distribution (red squares) with the corresponding simulation (black dots) using the same parameters. In addition, the simulated cumulative distributions are shown with a fully inserted (In) and retracted target probe (Out), respectively. The magnet case was biased to $8\,\mathrm{V}$ [63].

7.3   Manipulation of the radial deposition profile

In the previous section, it was verified that the insertable target probe is a viable diagnostic to measure the radial distribution of positrons. This section investigates to what extent the free parameters can influence that radial distribution. This would help to optimize previous experiments and to design future experiments. In confinement experiments, for example, it was observed that positrons closer to the wall have a shorter confinement time [13]. When studying the compression of trapped positrons by oscillating wall biases, distributions with larger radii are advantageous to verify a successful compression more clearly. In future experiments, higher positron densities might be achieved by accumulating multiple positron pulses in the trap. For this, it might be useful to deposit the different pulses at different radii.

In the following, the comparison of simulations and experimental data is used to investigate how the position of the radial positron distribution changes when the free parameters are varied. The free parameters considered are the voltages applied to the E×B plates, Top1, RW1, and the magnet case and the currents of the two steering coils I_r and I_θ. In the simulations, 200 particles were launched for each bin and the perturbation of the target probe was taken into account. Just like in the previous section, all profiles shown here are the integrated radial positron densities. The values of the parameters that are not scanned are listed in Appendix C. The color scale of the histograms shows the normalized fraction of positrons hitting the target probe relative to the total count of injected positrons. The maximum value of each row is therefore the injection efficiency for that particular parameter value.

7.3.1 E×B plate voltage

The effect on the radial distribution and injection efficiency by varying the voltage applied to the E×B plates can be seen in Figure 7.5. For the chosen steering coil settings (Appendix C), the highest injection efficiency is achieved in the range between 200 V and 300 V. The simulations can be used not only to reproduce the injection efficiency, but also to explain why they are not injected or where they annihilate outside the field of view of the detector: below 50 V, positrons hit the shield plate, and between 50 V and 150 V they either are reflected back up the beam line or hit the wall before reaching the target probe. At voltages above 350 V, the mean position of positrons is further outside, so many positrons already hit the wall before they could reach the target probe, but these positrons still annihilate in the field of view of the gamma detector. This is the reason why a signal is detected at high voltages even with fully retracted target probe (upper right corner). In terms of manipulating the radial deposition of positrons with the E×B bias, the possibilities are quite limited without affecting the injection efficiency.

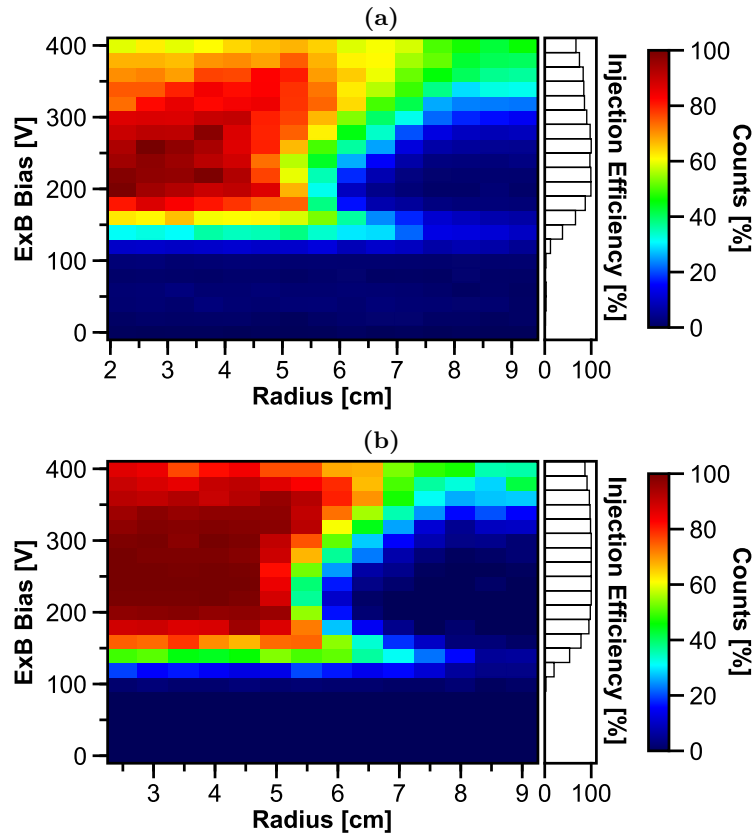**Figure 7.5:** Normalized 2D histogram of counts on the target probe as the radial position of the probe and the E×B bias are varied. (a) Experimental data of an injected 5-eV positron beam. (b) Simulation result using the same settings [63].

7.3.2 Top1 voltage

Figure 7.6 shows that the Top1 bias has no influence on the radial distribution. The injection efficiency is also unaffected from the applied voltage as long as it is higher than the beam energy of 5 eV to prevent positrons from hitting the Top1 electrode. However, other simulations have shown that if the E×B voltage is significantly higher than in the case shown, the bias of the Top1 electrode is no longer necessary for good injection and a voltage between 0 V and 8 V can very well shift the mean position about 1 cm outwards.
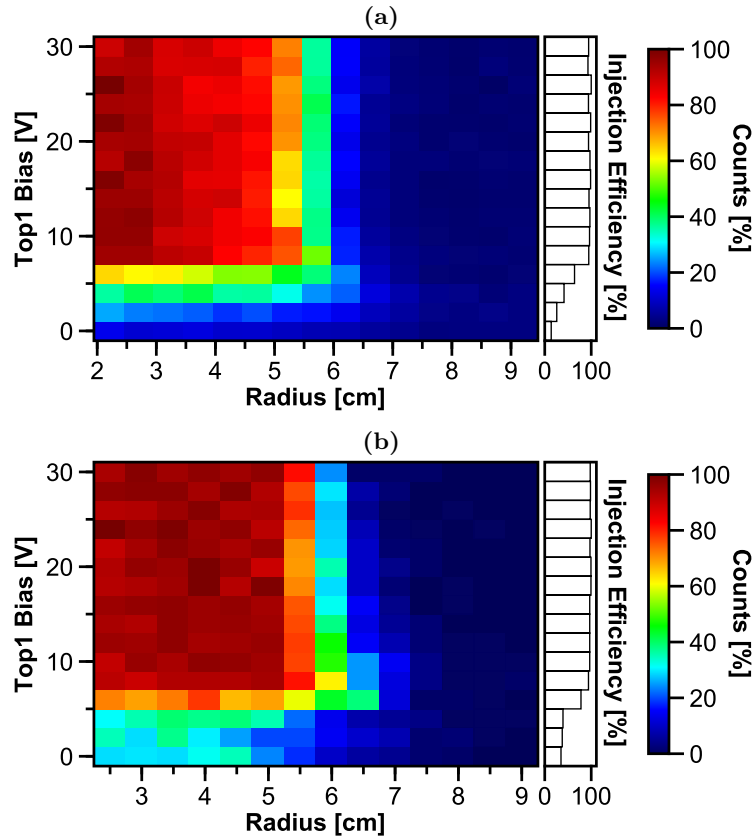

Figure 7.6: Normalized 2D histogram of counts on the target probe as the radial position of the probe and the Top1 bias are varied. (a) Experimental data of an injected 5-eV positron beam. (b) Simulation result using the same settings [63].

7.3.3   RW1 voltage

The voltage applied to the RW1 electrode has, like the Top1 electrode, no real influence on the radial distribution once the voltage is higher than the beam energy, as one can see in Figure 7.7. At lower voltages, positrons tend to be injected closer to the wall, but this also increases the probability of hitting the wall and thereby decreases the injection efficiency measured at the target probe. Since the Top1 and RW1 electrode are both, generally speaking, necessary for good injection and have very similar effects on the radial deposition, they can be paired together, which shortens the optimization time at the beginning of every beam time and also frees up a power supply.
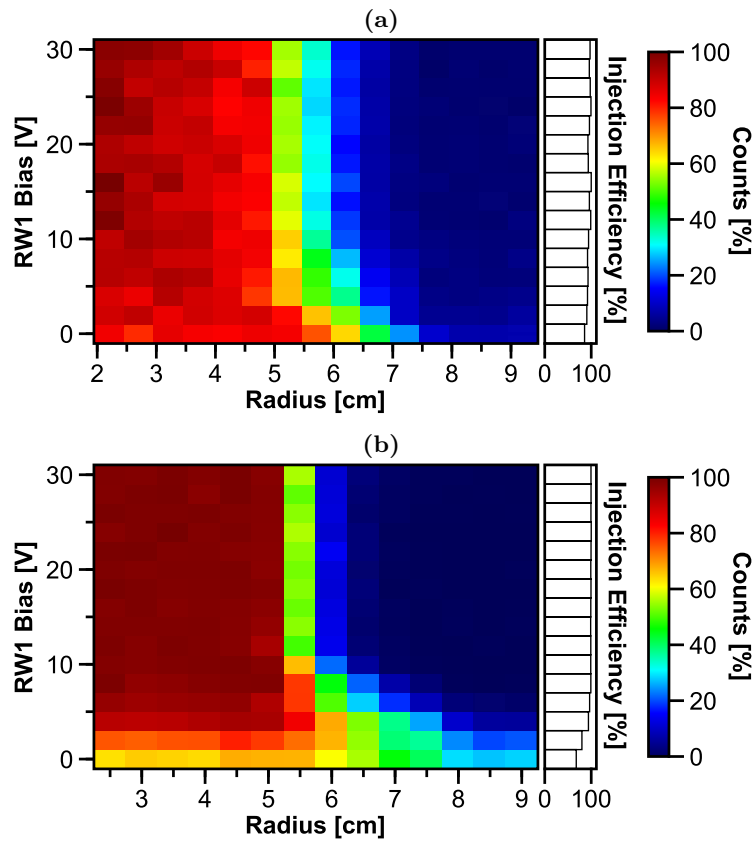

Figure 7.7: Normalized 2D histogram of counts on the target probe as the radial position of the probe and the RW1 bias are varied. (a) Experimental data of an injected 5-eV positron beam. (b) Simulation result using the same settings [63].

7.3.4   Magnet case voltage

As both the simulation and the experiment show in Figure 7.8, the applied voltage on the magnet case is very suitable for manipulating the radial position of the injected positrons in Proto-APEX, as long as it is high enough to prevent positrons with low perpendicular energy from hitting it. Further simulations show that this trend continues beyond the shown 10 V up until 40 V without degrading the injection efficiency. This correlation is easily explained in terms of the effective potential energy introduced in Section 7.1. Increasing the magnet case bias causes the positrons to be reflected earlier, similar to positrons with increasingly larger pitch angles as shown in Figure 7.1a, which in turn causes the positron to be deposited further out (see Figure 7.1b).
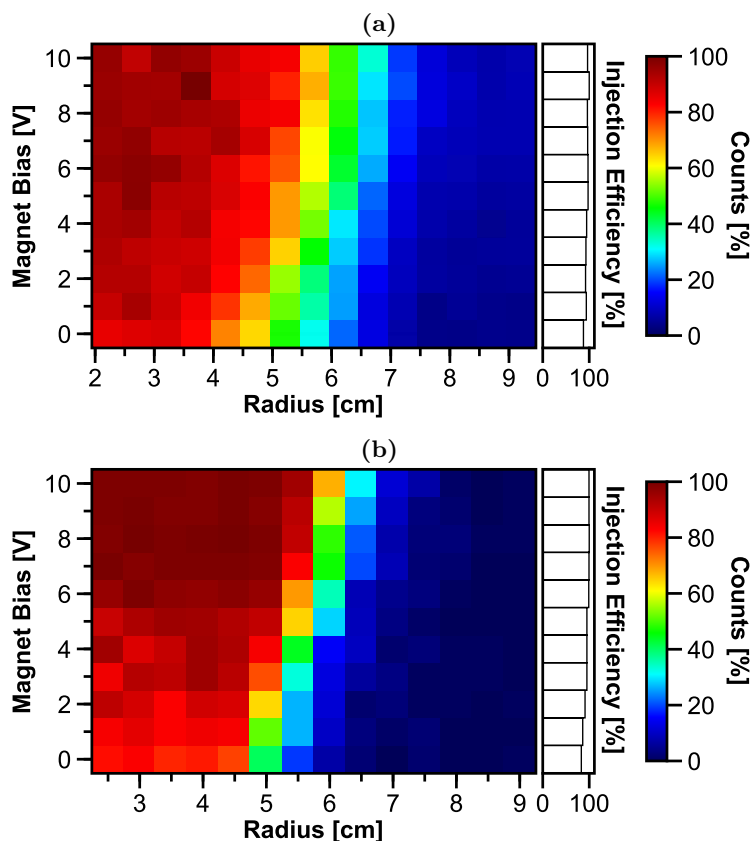

Figure 7.8: Normalized 2D histogram of counts on the target probe as the radial position of the probe and the magnet case bias are varied. (a) Experimental data of an injected 5-eV positron beam. (b) Simulation result using the same settings [63].

7.3.5 Steering coil currents

No experimental data are available for the comparison of coil current and target probe position, so simulations must suffice for this, but since the simulations have been able to reproduce the experiments very well so far, they should be sufficient for a rough estimate of the occurring effects.

In Figure 7.9, one can see that there is not much room to vary the steering coil current I_θ without reducing the injection efficiency considerably. If the current is higher than the optimal value, the positron beam is guided towards the positively charged E×B plate which has the consequence that the positrons are reflected back up the beam line before even entering the trap. If the current is too low, they enter the trap but are reflected as well and subsequently hit the wall in the injection region or, if the current is even lower, the positrons directly hit the negatively charged E×B plate. The main purpose of the steering coils for the θ-direction is therefore to center the incoming beam with respect to the E×B plates.
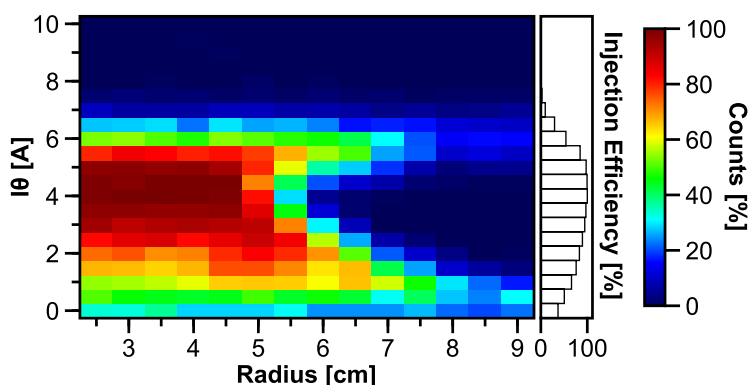

Figure 7.9: Normalized 2D histogram of counts on the target probe as it is inserted and the steering coil current I_θ is varied [63].

The steering coil current I_r is a particularly useful "knob" to control the radial deposition of positrons during injection, as the simulation in Figure 7.10 shows. In the range from $-2\,\mathrm{A}$ to $-8.5\,\mathrm{A}$, the distribution can be shifted by about $3\,\mathrm{cm}$ while maintaining lossless injection. At higher currents, the positrons start hitting the wall, and at very low currents, they hit the top of the shield plate. Some are even injected on the other side of the plate, causing them to eventually reflect back into the beam line. At high I_r currents and small radii, the drop of the count rate is caused by positrons at large radii missing the rod of the target probe (diameter: $3\,\mathrm{mm}$) since some may have a large enough gap between mid-plane crossings (this can also be seen at E×B biases in Figure 7.5).
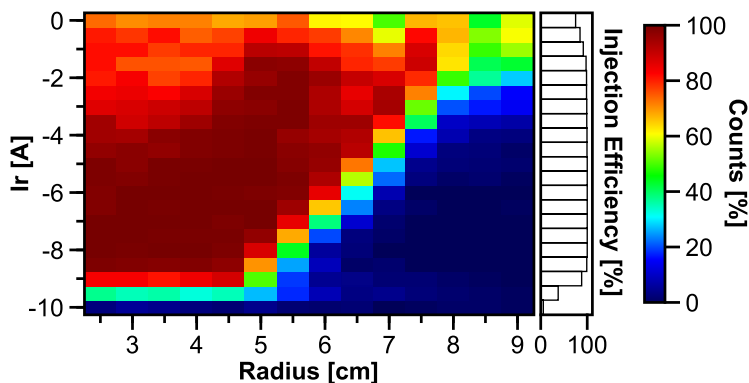
Figure 7.10: Normalized 2D histogram of counts on the target probe as it is inserted and the steering coil current I_r is varied [63].

7.4 Conclusion

It was discovered by simulations and experimental data that from all free parameters present in the Proto-APEX experiment, the applied voltage to the magnet case and the steering coil current I_r have the desirable property of manipulating the radius of positron deposition during injection without compromising the injection efficiency. Here, only one parameter was changed at a time. The multidimensional parameter space, however, is highly non-linear (as seen in Figure 7.11 as well as in Appendix F), so it is likely that the radius can be changed with even finer control by simultaneously varying multiple parameter at once.

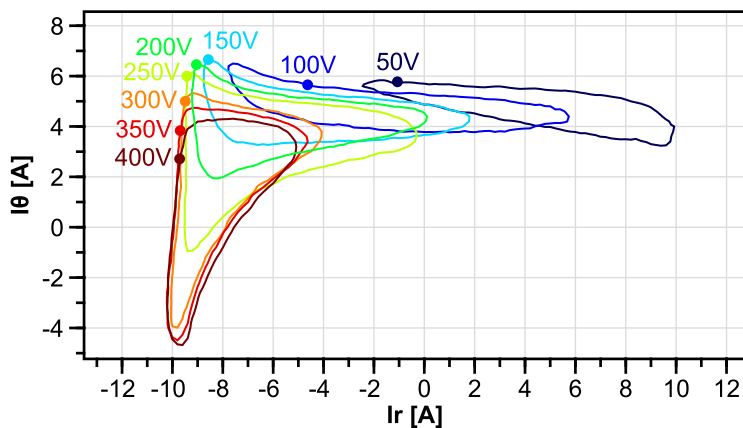**Figure 7.11:** $I_r - I_\theta$ steering coil current regions with an injection efficiency of $> 50\%$ for a range of E×B voltages [63].

In this chapter, it was also shown that the effective potential energy can be used to describe the complex behavior of positrons in electric and magnetic fields, which is particularly useful for understanding the injection of positrons into magnetic confinement devices.

Finally, the target probe was tested to see if its insertion into Proto-APEX could perturb the trajectories of the positrons, which would cause a discrepancy between the measured radial distribution and the actual distribution. It has been found that the target probe does affect the radial distribution when the magnet case is biased, but this effect is small and does not outweigh the benefits of having this tool. A target probe will most likely also be installed in APEX-D, so the perturbation effect should not exist, since the levitated coil has no leads attached and therefore cannot be biased at all, but it remains to be seen how far the coil is charged up over time by the surrounding positrons and/or electrons.

Chapter 8

Confinement of positrons

Besides the efficient injection discussed in the previous chapter, the confinement of the positrons is very important, since a longer confinement time allows more possibilities to study the pair plasma. Previous experiments in Proto-APEX had already yielded good confinement times, but the corresponding simulations revealed the factors that prevented even longer confinement times [13]. This chapter will expand on this topic by investigating, through simulations, how to extend the confinement time of positrons in Proto-APEX.

First, the procedure of the confinement experiments is described in detail in Section 8.1, followed by an explanation of the measures taken to simulate the confinement of positrons in Section 8.2. Section 8.3 takes a closer look at the results and limitations of the previous confinement experiments. The main part, Section 8.4, explores several ways to remove or reduce these limitations in order to improve the positron confinement time in Proto-APEX. Last but not least, a brief look at future experiments will be given in Section 8.5.

8.1 Setup of the experiments

In Proto-APEX, injected positrons are pushed towards the wall as they drift toroidally around the magnet, as seen in Figure 8.1, until they ultimately annihilate on the wall after just one rotation due to the potential of the positively charged E×B plate. Hence, the effective potential energy (see Section 7.1) necessary for the injection prohibits confinement times longer than one toroidal rotation, which is about 10 to 15 µs for a 5-eV positron.

To achieve longer confinement times, an electrode switching scheme must therefore be employed (Figure 8.2) [13]. The idea is to inject positrons into the trap until the loss rate equilibrates with the injection rate, so for at least 15 µs, and then bias the deflector to prevent positrons from reaching the main chamber, as well as switch off the voltages applied to the E×B plates and the wall electrodes. Since the effective potential energy is now symmetric, the confinement time is extended; however, only about 300 positrons can be trapped simultaneously for the positron rate provided by the remoderated beam of NEPOMUC. At the time of the experiment, only two BGO detectors were available. The covered

solid angle of these two detectors in combination with the low annihilation rate was therefore not sufficient to be able to record meaningful life times straight away. To achieve a better signal-to-noise ratio, the potentials of the electrodes were turned on again after a certain time to provoke an ejection of the trapped positrons, which resulted in a strong "dump" signal. Nevertheless, the whole cycle had to be repeated thousands of times to compensate for the low number of trapped positrons and the small covered solid angle. In order to determine the confinement time, the so-called hold time (between filling and dumping) was varied and the relative dump signals compared.
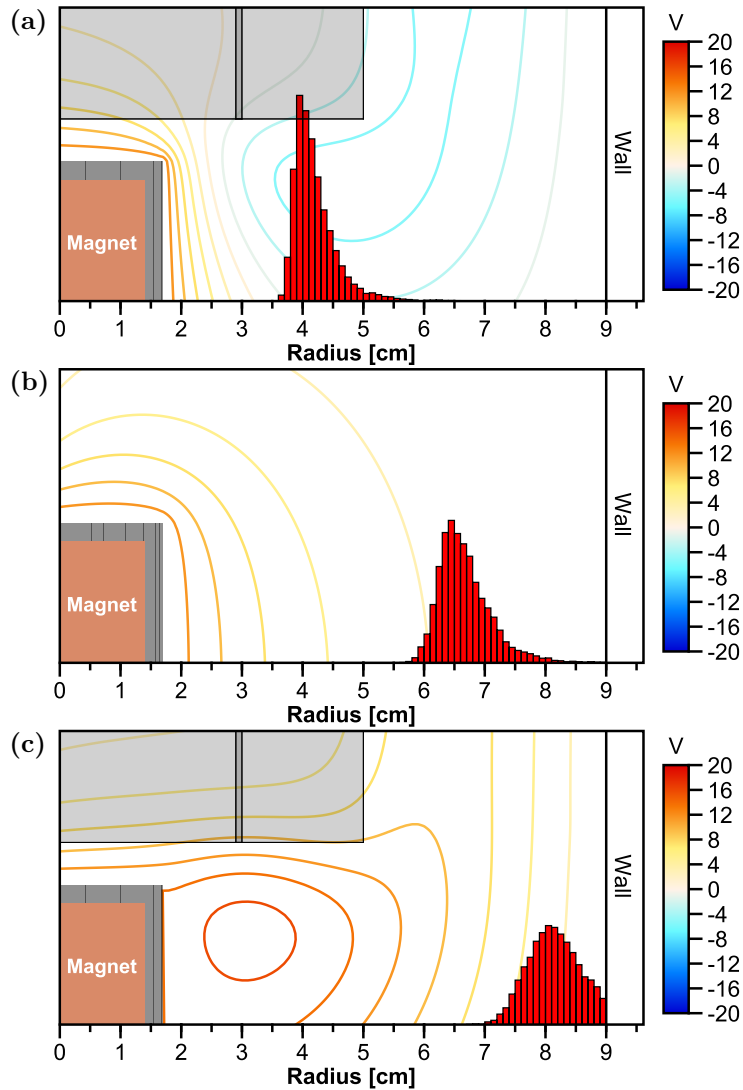

Figure 8.1: Simulated radial distribution of 5000 positrons on the mid-plane at different toroidal angles (red histogram): at 90° (a) at 180° (b) and at 270° (c). The electric potential is illustrated by the contour lines [63].
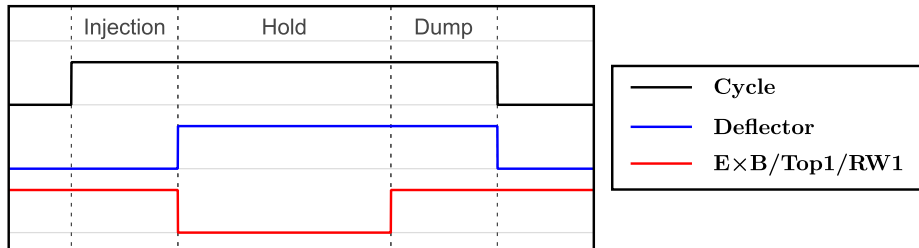
Figure 8.2: Illustration of the confinement cycle used in Proto-APEX.

8.2 Setup of the simulations

In the simulation of positron confinement, a signal-to-noise ratio does, of course, not exist because the position of a positron is known at all times. Under the assumption that the main loss channel is elastic scattering with the residual gas, some simplifications have to be made to make the simulation computationally feasible. First, elastic scattering is implemented by randomizing the direction of the positrons' velocity vector. This can be done by picking a random point on a unit sphere, for example with [65]

$$x = \sqrt{1 - u^2} \cos \theta \tag{8.1}$$

$$y = \sqrt{1 - u^2} \sin \theta \tag{8.2}$$

$$z = u \tag{8.3}$$

where $\theta \in [0, 2\pi)$ and $u \in [-1, 1]$, and then scaling this unit vector by the original velocity magnitude. Secondly, the small time step needed to resolve the gyromotion makes it costly to simulate particles for more than $10\,\mathrm{ms}$ which is, however, of the same order as the expected mean collision time in the experiment. It is therefore necessary to conduct the simulation with a collision frequency in the order of $1\,\mathrm{MHz}$ to reduce the simulation time to a reasonable duration. This collision frequency is still lower than the bounce frequency between the magnet poles but higher than the toroidal rotation. To compensate for missed collisions of the positrons with the E×B plates or the shield plate, a positron is stopped once the vertical distance to the equatorial plane is large enough so that it would have the possibility to hit the plates.

8.3 Previous results

So far, the record for the longest confinement of positrons measured in Proto-APEX was found to have a timescale of $(1.5 \pm 0.1)\,\mathrm{s}$ when fitted with a single exponential decay curve [13]. This confinement time, which corresponded to approximately 200 collisions of a positron with background neutrals, is consistent with positrons diffusing radially outwards and reaching field lines that intersect the E×B plates and shield plate (see Figure 8.3). In the same beam time, another experiment was performed where the magnet was grounded. This drastically reduced the confinement time to only $(0.28 \pm 0.04)\,\mathrm{s}$ — or about 25 collisions — and is consistent with the dominant loss mechanism being positrons scattering into the loss cone of the permanent magnet and hitting it, consequentially.
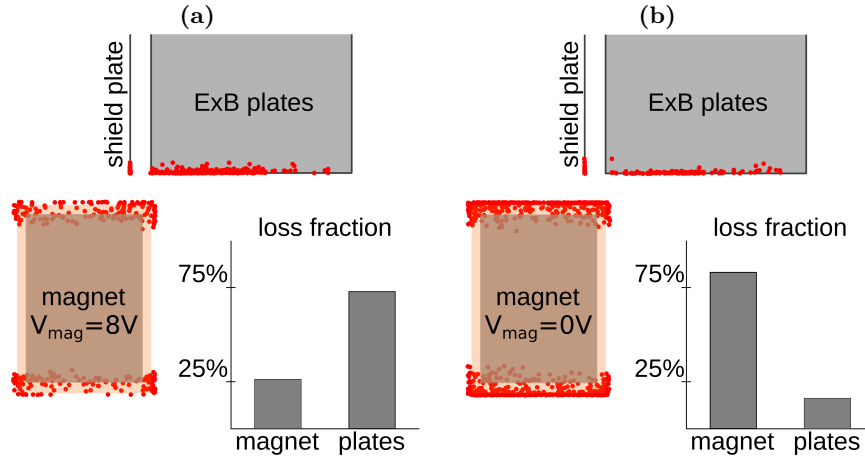
Figure 8.3: In the simulations [13], about three-quarters of the confined positrons end up hitting the plates and about one-quarter scatters into the loss cone of the magnet (a). If the magnet is grounded, most positrons annihilated on the magnet (b).

8.4 Improving the confinement time

In the following, simulations are used to investigate whether the confinement time can be extended, and if so, by how much. The collision frequency of all following simulations was set to $1\,\mathrm{MHz}$. This value has the positive side effect that the confinement time can be easily converted into the number of collisions, which in turn facilitates an estimation of the expected confinement time for different gas pressures. Figure 8.4 shows the result of a simulation, that is used as a baseline
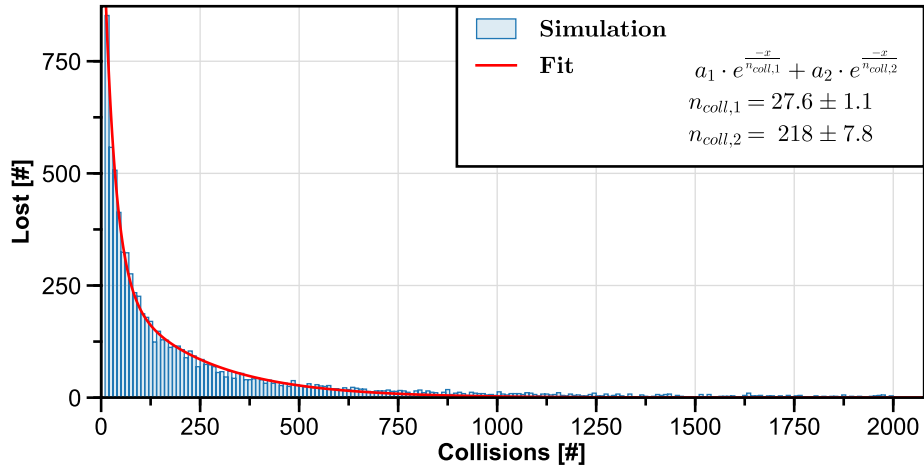

Figure 8.4: Reference confinement simulation. The data is fitted well by a two-phase exponential decay function, where the fast decay ($n_{coll,1}$) corresponds to the scattering into the magnet loss cone and the longer one ($n_{coll,2}$) to positrons hitting the plates.

reference. The data is very well described by an exponential decay with two phases, which can be justified by the fact that the scattering of positrons into the loss cone of the permanent magnet is a much faster process (≈ 20 collisions) than drifting outwards and subsequently hitting the plates (> 200 collisions).

Note that this simulation result should not be directly compared with the previously mentioned experimental data, because the settings used in the simulation are slightly different and the amount of available data was insufficient to resolve these two loss processes.

8.4.1 Shorter plates

Since longer confinement times are mainly limited by the plates, it would be logical to shorten them so that they no longer intersect the field lines of the confinement volume. Before doing so, however, it must be ensured that this does not reduce the injection efficiency, as this is essential to achieve plasma densities in the future. Fortunately, single-particle simulations can easily answer this question. In Figure 8.5, the injection efficiency — defined as fraction of launched positrons reaching the target probe — was simulated for a range of plate lengths. It shows that, without needing to change any other parameters, the plates can be shortened by over 40 mm without deteriorating the injection efficiency. By changing the injection parameters, the plates could be shortened even further, but would come at the cost of also reducing the parameter space for good injection.
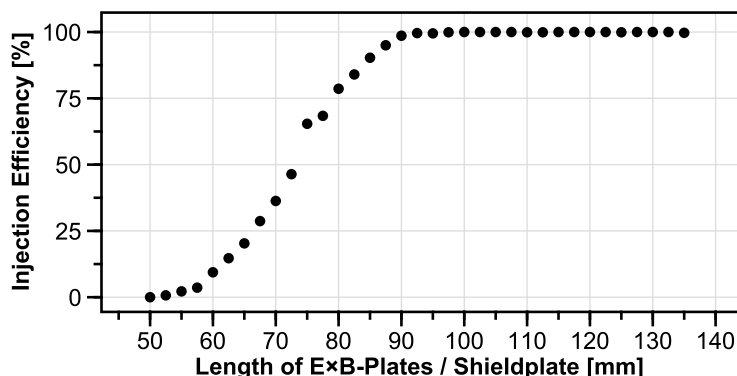
Figure 8.5: An injection efficiency of over 99 % can be maintained with shorter plates up to 90 mm without changing any parameters.

Not only is the high injection efficiency maintained with shorter plates, they also have the positive side effect of reducing the influence of the E×B potential on the trajectories of transiently trapped positrons, which can be seen in Figure D.1 in the Appendix. Since the positively charged E×B-plate is now further away from the confinement region, the positrons hit the wall later, which in turn means that more positrons can be captured in each injection-hold-dump cycle, resulting in a slightly better signal-to-noise ratio.

For the confinement simulation (Figure 8.6), a plate length of 115 mm was chosen. It shows that by merely shortening the plates, the confinement time could be extended by more than 65 %. There are still two loss processes present: the faster loss process is still the scattering of positrons into the loss cone of the magnet, the slower one is now the outwards drift and the subsequent annihilation on the wall.
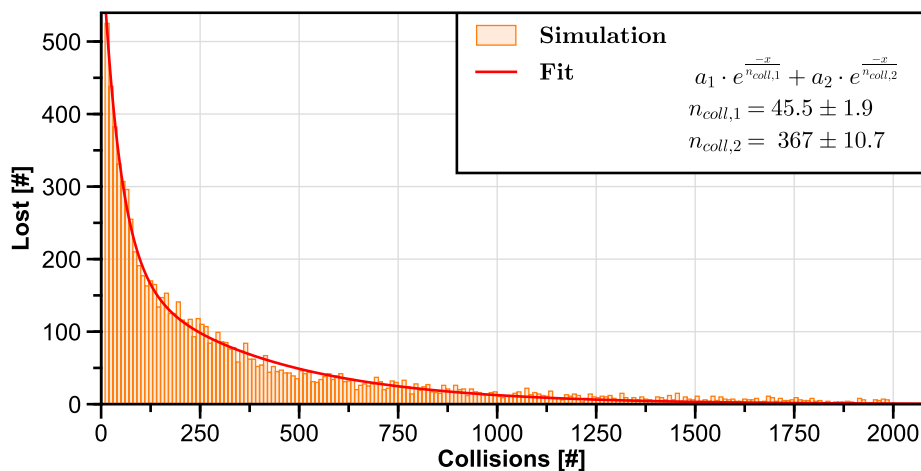

Figure 8.6: Simulation result of the confinement time when the height of the E×B plates and shield plate is reduced from 135 mm of the reference simulation (Figure 8.4) to 115 mm. The two phases of the exponential decay fit correspond to the scattering into the magnet loss cone ($n_{coll,1}$) and the annihilation on the wall after drifting outwards ($n_{coll,2}$).

8.4.2   Higher magnet bias

So far, the magnet case was biased to 8 V. As the previous simulations show, this is not enough to electrostatically prohibit 5-eV positrons from scattering into the loss cone. The next step is therefore to verify that this loss process can be stopped by applying a higher bias to the magnet case. The simulation result shown in Figure 8.7 indeed confirms that biasing the magnet case to 20 V will stop positrons from hitting the magnet, which in turn increases the overall confinement time by 50 % compared to just using shorter plates (Figure 8.6) and by a factor of 2.5 compared to the reference simulation (Figure 8.4). Since changing the potential applied to magnet case also effects the radial distribution of the injected positrons, as discussed in Section 7.3, special care was taken to maintain the same radial distribution as before (see Figure 8.8) by counteracting the shift by simultaneously adjusting the I_r steering coil current.
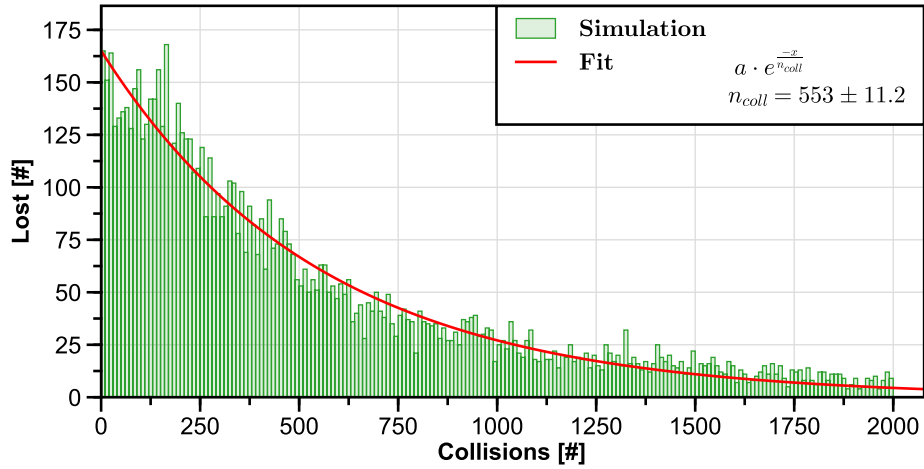
Figure 8.7: Simulation of the confinement time with shorter plates (Figure 8.6) and an increased magnet case bias (from 5 V to 20 V). Positrons are now lost only by drifting outward, to the wall electrodes.

8.4.3   Radial distribution

After shortening the plates and increasing the magnet case bias, the only loss channel remaining is the outward drift of positrons and the annihilation on the wall. To further increase the confinement time, one could use the knowledge presented in Section 7.3 on how different free parameters affect the radial distribution to try to inject positrons closer to the magnet, which would have the consequence that they would need more collisions until they end up hitting the wall. Figure 8.8 illustrates how the radial distribution is shifted inward, towards the magnet, when the magnet case bias is lowered to 15 V (which is still high enough to prevent positrons from hitting it) and the steering coil current I_r is set to -7 A instead of -6.2 A.
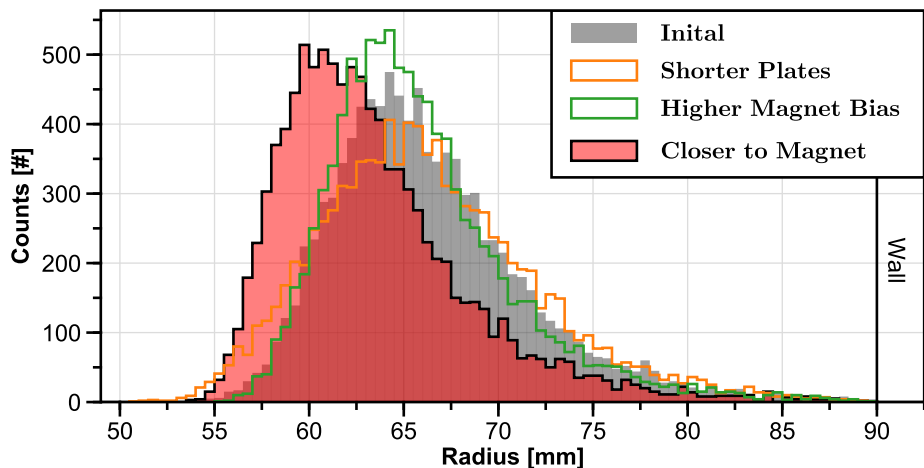

Figure 8.8: Comparison of the different radial distributions of injected positrons used in the simulations.

In the simulation (Figure 8.9), this radial shift of just 5 mm significantly improved the confinement time. Compared to the previous simulation, it was almost doubled, resulting in a confinement time five times longer than the initial reference simulation.
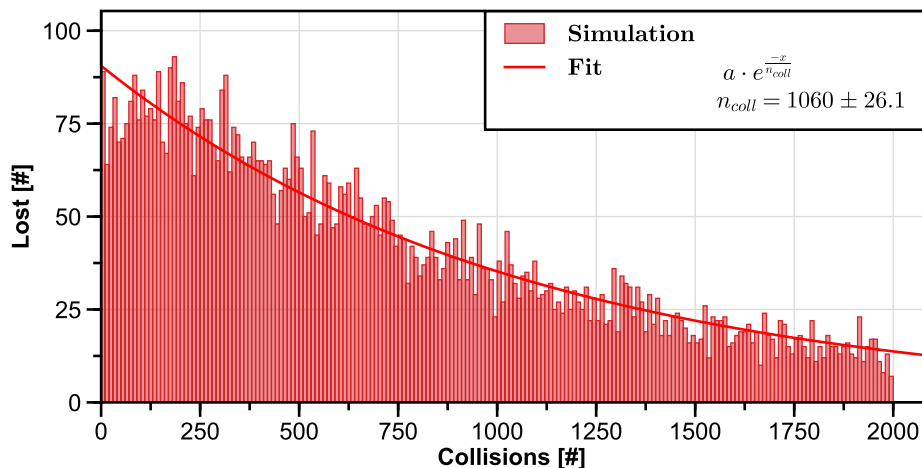

Figure 8.9: Simulation of the confinement time of positrons in Proto-APEX if the plates are shortened, the magnet case bias increased, and the radial distribution is shifted towards the magnet.

8.5   Plans for future experiments

In this chapter it was shown that the confinement time of positrons can be dramatically extended by shortening the E×B plates, by increasing the bias of the magnet case, and by depositing the positrons closer to the magnet during injection, using the methods described in Section 7.3. All of these methods can be easily applied to Proto-APEX for the next beam time.

In addition, the data acquisition system was completely revamped since the last experiments. This includes the switch from an analog setup and using an oscilloscope to bin the raw data to a digitizer based setup, which is able to tag events with a resolution of 2 ns or even lower, depending on the loaded digitizer firmware. The digitizers also make it possible to increase the number of BGO detectors from two to at least eight, which will significantly boost the amount of data collected. Each digitizer card can support up to 16 channels, but the layout at the experiment site will need to be significantly modified to create enough space to use all of them at the same time. Another advantage of the digitizers is that they can record annihilation events not only during the dump, but also during the entire hold phase, which were ignored by the old acquisition setup. This could also eliminate the need for the dump phase, which would simplify the confinement experiments.

One of the most important questions to be answered by future experiments is how different types of gases and pressures affect the confinement time of positrons. For APEX-D, for example, a gas might be injected to increase the volumetric annihilation rate, which, by a tomographic reconstruction, could give an indication of the positron densities. Proto-APEX can be used for preliminary studies to investigate which gas would be best suited for this purpose.

Chapter 9

Beam time reproduction workflow

The 13th beam time assigned to Proto-APEX started September 25, 2018 and lasted until October 3, 2018. This chapter is intended to serve as an example of how the simulation model is initialized from data collected during beam times and how such a model can reveal previously unknown phenomena.

After determining the parallel and perpendicular energy distributions and the shape and position of the incoming positron beam in Sections 9.1 and 9.2, respectively, the experimental results are compared with the simulations in Section 9.3. The dump signal is then examined in more detail in Section 9.4. Finally, positrons that are not ejected during the dump, as discovered by the simulations, are discussed in Section 9.5.

9.1 Energy distribution

Since the positrons in the simulation are launched inside a magnetic field, both the parallel and perpendicular energy distributions must be known to properly initialize their velocity vectors.

9.1.1 Parallel energy distribution

The parallel energy of the positron beam can be measured by ramping up the bias of a plate, acting as a potential barrier, while counting the annihilation gammas of positrons which are still able to hit said plate. In the beam time described in this chapter, this was done with the grid in Beam Monitor 2 (BM2) — an upstream measurement station in the NEPOMUC beam line — as well as the target at the MCP. Fitting the measurement of the latter one with an error function (see Figure 9.1) results in a Gaussian-distributed parallel energy of $(16.2 \pm 3.2)\,\text{eV}$.
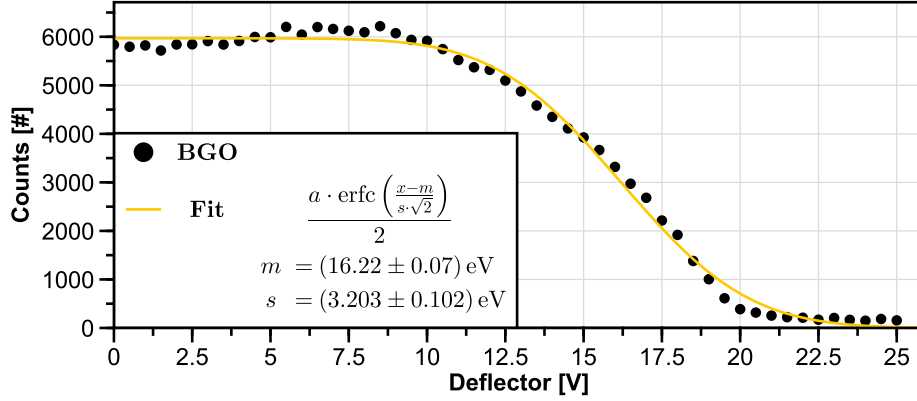
Figure 9.1: Parallel energy distribution of the positron beam measured at the MCP target by ramping up the bias of the deflector and counting annihilation gammas with a BGO detector.

9.1.2   Perpendicular energy distribution

The perpendicular positron energy cannot be measured directly with the current experimental setup, so it was linearly interpolated between $E_\perp = 1.22$ eV at $E_\parallel = 11.77$ eV and $E_\perp = 1.3$ eV at $E_\parallel = 20.7$ eV. These values were obtained from retarding field analyzer measurements performed to characterize the NEPOMUC positron beam [29]. This gives the Maxwell-Boltzmann distribution:

$$f(E_\perp) = 1/a \cdot e^{-E_\perp/a} \tag{9.1}$$

$$a = 1.26\,\text{eV} \tag{9.2}$$

9.2   Beam shape and position

The NEPOMUC positron beam can be visualized at the experiment with an insertable MCP. The raw image (Figure 9.2a) captured at the beginning of the beam time can be processed and cropped (Figure 9.2b) so that the shape of the beam can be used directly in the simulation to generate appropriately distributed samples, as described in more detail in Section 5.6. Since the MCP diameter is known to be 25 mm, the beam size can be inferred from it accordingly.

For accurate simulations, the position of the positron beam must also be known (in addition to its shape and size), but this is difficult to determine from an MCP image alone, since the MCP is manually inserted and therefore not guaranteed to be perfectly centered in the beam line. One way to find the beam position — and to compensate for any misalignment of the beam line coils and steering coils — is to do a scan of the initial beam position in the simulation using the optimal settings of the experiment in order to find the location where the maximum number of positrons are injected (Figure 9.3). Here, a positron was counted as injected if it successfully made half a toroidal rotation, which is equivalent to hitting an ideal, fully inserted target probe. The best injection efficiencies of 70 % to 80 % were achieved with the center of the beam shifted by −1 mm towards the magnet and 12 mm towards the negatively charged E×B plate relative to the center of the beam line.
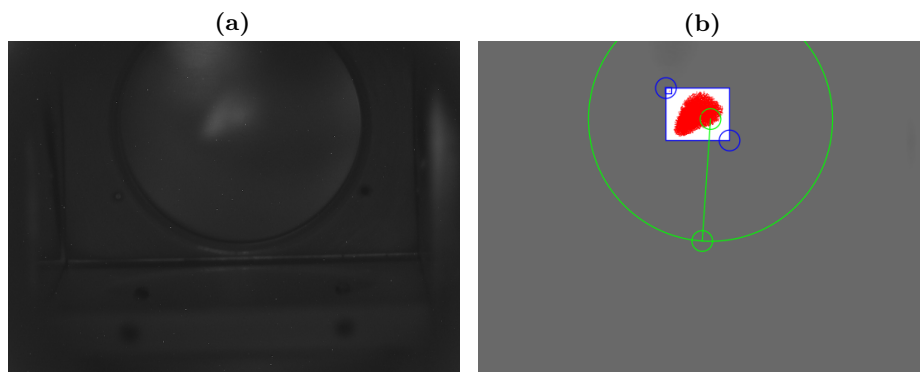
Figure 9.2: (a) The raw MCP image. (b) The inverted, denoised, and brightness-adjusted (23 %) image with the reference scale (green, $\varnothing= 25\,\mathrm{mm}$), sampling area (blue), and test samples (red).
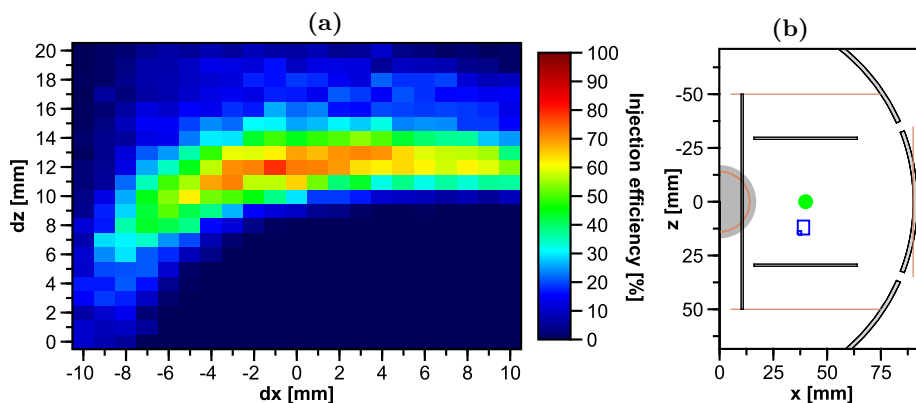
Figure 9.3: The maximum injection efficiency (70 % to 80 %, here 79 %) was achieved when the position of the center of the sampling area (blue rectangle in (b), see also Figure 9.2b) was shifted by $dx = -1\,\mathrm{mm}$ and $dz = 12\,\mathrm{mm}$ relative to the center of the beam tube (green point). A positron was considered injected when it toroidally drifted 180°.

9.3 Comparison of experiment and simulation

Now that all simulation parameters have been determined, one can attempt to reproduce the experimental data with simulations to confirm the overall consistency.

First, a 2D scan of the steering coil currents I_r and I_θ was performed that reproduced well the experimentally recorded data (see Figure 9.4). Since technical problems occurred during the beam time that prevented any current measurements that would allow the injection efficiency to be measured, it was also the task of the simulations to quantify it. In the simulation, about 62 % of the injected positrons were stopped within the detector view cone by the fully inserted target probe. However, the high beam energy of 16 eV allows a not

unimportant fraction of positrons to miss the rod of the target probe and are therefore not detected. If these are included, the injection efficiency reaches about 70 %. Another consideration is that the low resolution of the parameter scan skips over the optimal value of $I_\theta = 5.5\,\mathrm{V}$. The injection efficiency can therefore be concluded to be in the 70 % to 80 % range, as already seen in Figure 9.3. The limiting factors were the low voltages used ($V_{Magnet} = 12\,\mathrm{V}$, $V_{Top1} = 14\,\mathrm{V}$ and $V_{RW1} = 20\,\mathrm{V}$), which could not prevent positrons from the high-energy tail of the beam from annihilating on these electrodes. Biasing all of these electrodes to 25 V, injection efficiencies higher than 90 % were achieved in the simulations.
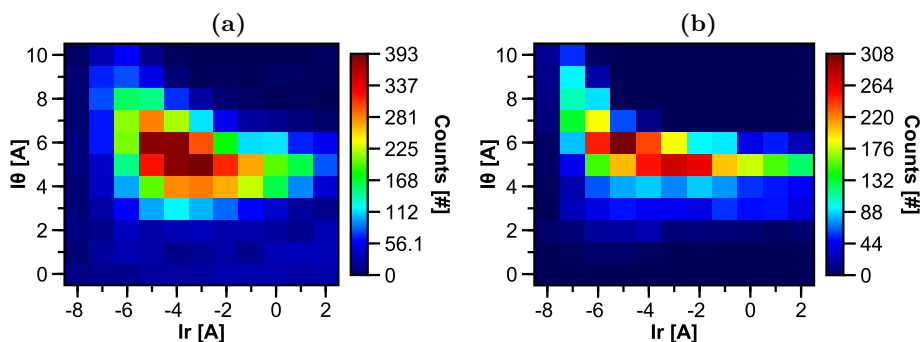

Figure 9.4: Comparison of an $I_r - I_\theta$ scan recorded during the beam time (a) with the simulation (b, 500 e^+/bin). In both, experiment and simulation, the counts were detected by a detector looking at the target probe. Counts are not normalized.

The radial distribution of the injected positrons was also compared with simulations. After fitting the data with an error function, the distribution means were found to agree quite well with each other, as seen in Figure 9.5. In contrast, the width of the simulated distribution is much larger than the measured one. An explanation for this would be magnetic perturbations, above all a tilting of the magnet, which, however, cannot be integrated into the simulation without a great deal of additional effort and guessing of free parameters. The radial steps of the experimental data differ slightly from those shown in Figure 7.4 because the target probe was changed to be fully automatic instead of manually operated.

9.4   Dump signal delay

During the analysis of the recorded confinement data, a clear delay of the dump signal of about 40 µs relative to the actual trigger signal became apparent, as seen in Figure 9.6.

When the actual voltage waveforms of the individual electrodes are recorded with an oscilloscope (see Figure 9.7), the reason for the delay is obvious. It turns out that the self-built switching electronics have a lag relative to the rising trigger signal (for the "dump") of about 25 µs, as well as a slow rise time, especially for the E×B plates.
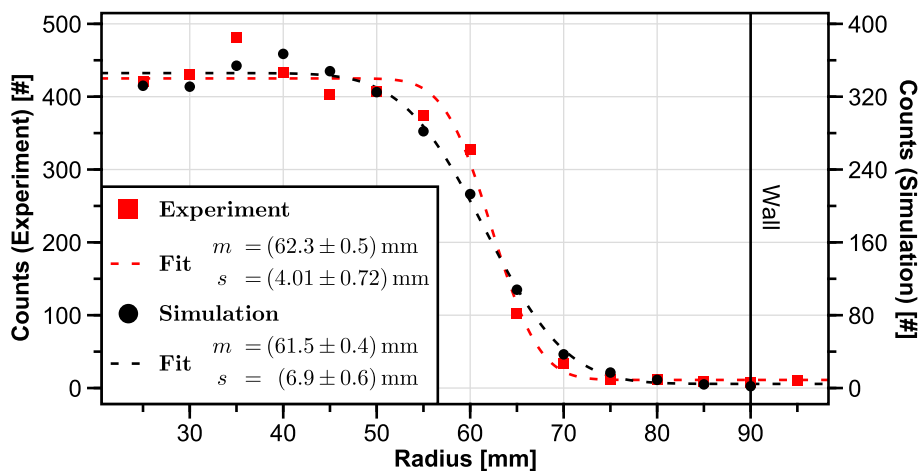
Figure 9.5: Comparison of radial positron distributions. Both, the experimentally measured profile (red squares) and the corresponding simulation (black dots) using the same parameter, were fitted with the error function $a \cdot \text{erfc}\left(\frac{x-m}{s \cdot \sqrt{2}}\right)/2 + y_0$. The mean values agree well. The spread of the simulated distribution is, however, higher than the experimentally recorded data.
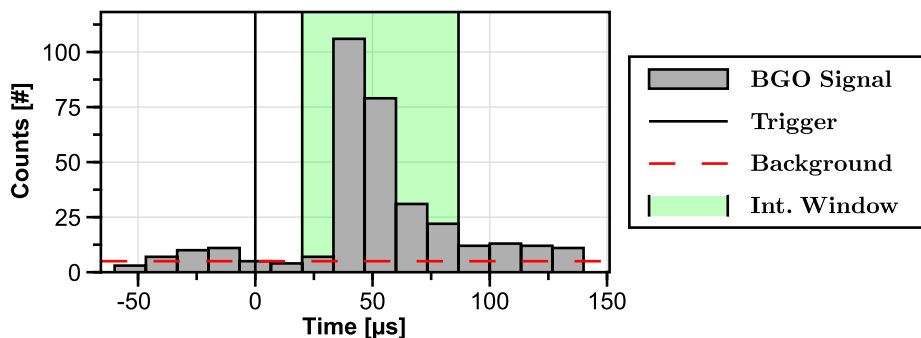
Figure 9.6: Accumulated experimental data recorded during the dump phase.

In order to investigate in more detail how the delay and the slow rise times affect the ejection of positrons, a simulation was performed using the recorded voltage curves of the respective electrons. In the simulation, the hold-phase (see Figure 8.2) is triggered at $10\,\mu s$ and the electrodes turn off a few microseconds later. $10\,000$ particles were started randomly between $0\,\mu s$ and $15\,\mu s$. The particle injection ends $5\,\mu s$ after the hold-phase trigger to make sure that the trap is completely filled. The dump phase is triggered at $50\,\mu s$ and the electrodes slowly turn on $\sim 25\,\mu s$ later. In the simulation result shown in Figure 9.8, one can see that the peak of the dump signal is at $93\,\mu s$, so around $40\,\mu s$ after the trigger, which is in very good agreement with the experiment. Interestingly, a smaller peak can be seen in the simulation result prior to the main peak, presumably caused by the relative fast switch on of the RW1 electrode. The analysis of this peak revealed that it contains only positrons closest to the wall and which have a small total kinetic energy. In the experimental data this smaller peak is not
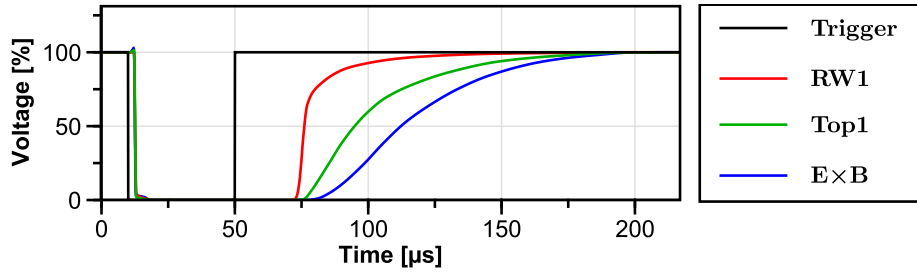
Figure 9.7: Timing scheme used in the simulation. The timing curves of the electrodes were recorded with an oscilloscope. The noisy, raw oscilloscope traces were smoothed with a moving average, interpolated with C²-splines and normalized for usability purposes. The switching times of the electrodes were recorded with following amplitudes: RW1: 0 V to 20 V, Top1: 0 V to 15 V, ExB: 0 V to 10 V.

visible, which is probably due to the limited number of recorded events and the low time resolution. After this beam time, the gamma detection and timing electronics received a major upgrade (Chapter 4), including fast high-voltage switches, so it is unlikely that this feature will be seen in the future despite the significant improvement of the data recording capabilities.
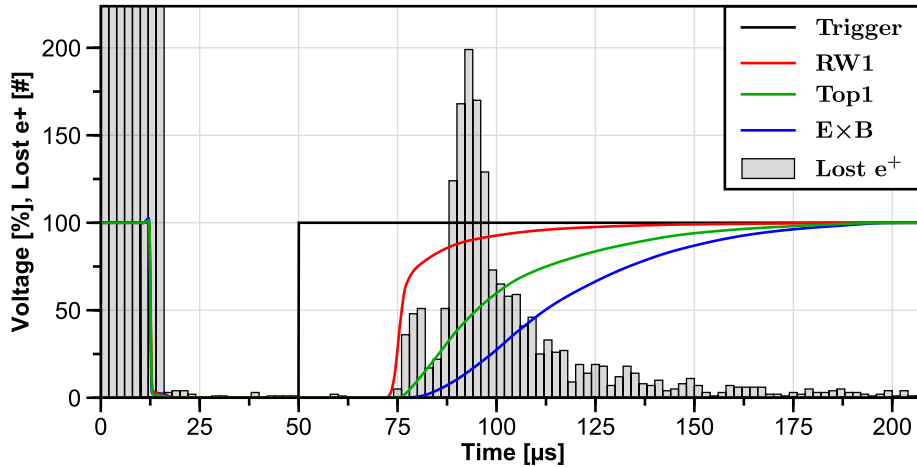
Figure 9.8: Simulation of the dump using the voltage curves that were recorded during the beam time. During the injection phase, many positrons are lost because they were either not successfully injected or ejected after one toroidal rotation due to the still biased electrodes.

9.5   Positrons confined after the dump

When evaluating the simulation result, surprisingly another peak appeared, namely at the end of the simulated period at 250 μs (see Figure 9.9). This can only happen if positrons are still flying when the simulation is terminated which in turn implies that these positrons are confined even with all electrodes fully biased. Since over 10 % of all confined positrons are not dumped, it is worth to investigate why this is the case.
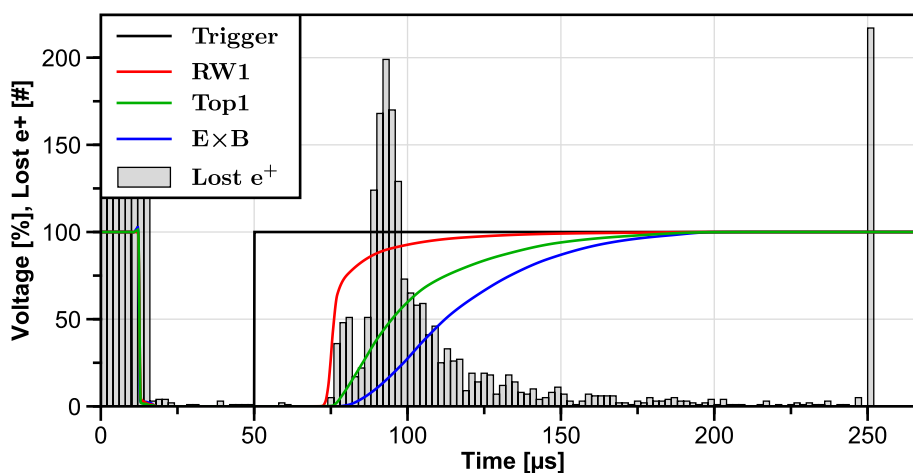

Figure 9.9: When plotting the data of Figure 9.8 over the full range, another peak appears at 250 μs, which was the predefined termination time of the simulation. This means that over 10 % of the confined positrons were still confined even after the dump and with all electrodes fully biased again.

Looking at Figure 9.10, where all positrons are shown with start and stop times, one can distinguish two populations of "long-confined" positrons (green points): one with a start time earlier than 5 μs and one with a start time later than ~ 10 μs. The first population consist of positrons that are injected way before the hold-phase starts and are not ejected by the injection potentials after the first toroidal rotation (an example of such a trajectory is shown in Figure D.2 in the Appendix). This size of this population is drastically reduced if the voltage applied to RW1 is increased from 20 V to 25 V. For 5-eV positrons, this population does not exist. In the second population are positrons that are started right before the hold phase-starts. All long-confined positrons have in common that they are located in the first half of the trap ($z > 0$, see Figure 4.1b) during the switch-off of the electrodes, which explains the gap between the two populations. In Figure 9.11a, the radial distribution of all confined positrons is plotted. It shows that the long-confined positrons are all located close to the permanent magnet. Furthermore, these positrons have a high kinetic energy, especially perpendicular energy (see Figure 9.11), which is the reason why they are able to stay confined even with all electrodes fully biased. A possible explanation for this above-average kinetic energy is that the long-confined positrons gain energy

during the ramp down of the potential of the negatively charged E×B-plate. This would also explain why these long-confined positrons are mostly located within the region of influence of the E×B potential during the switch-off.

High-energy positrons in the trap are unfavorable for the goal of achieving plasma conditions, as stated in Equation (1.1), so a fine-tuned injection and switch-off order is required in the future. On the other hand, they might provide some favorable properties for stacking multiple pulses to increase the number of positrons in the trap in combination with a cooling gas. So far, long-confined positrons have only been seen in the simulations, therefore an experimental proof is required to rule out any numerical effects. How such an experiment could look like is described in Chapter 10.
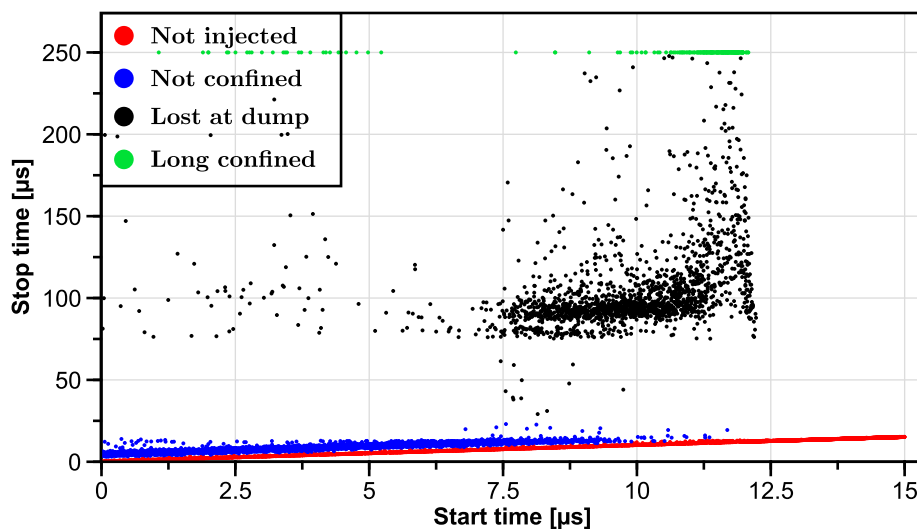
Figure 9.10: Each point represents the start and stop time of a single positron in the simulation. Positrons that are not successfully injected are shown in red. Injected positrons that are lost after one toroidal rotation due to the still-biased electrodes are shown in blue. If linear fits are applied to these two sets, the vertical difference of the two fits gives the average confinement time with active potentials, which is about 4.5 μs. This number together with the beam intensity allows for a rough estimate of how many positrons can be confined in each cycle. Considering only positrons with a start time <5 μs, one can deduce the injection efficiency to be ∼75 %. The black dots are positrons that were confined and lost during the dump. Long-confined positrons are highlighted in green.
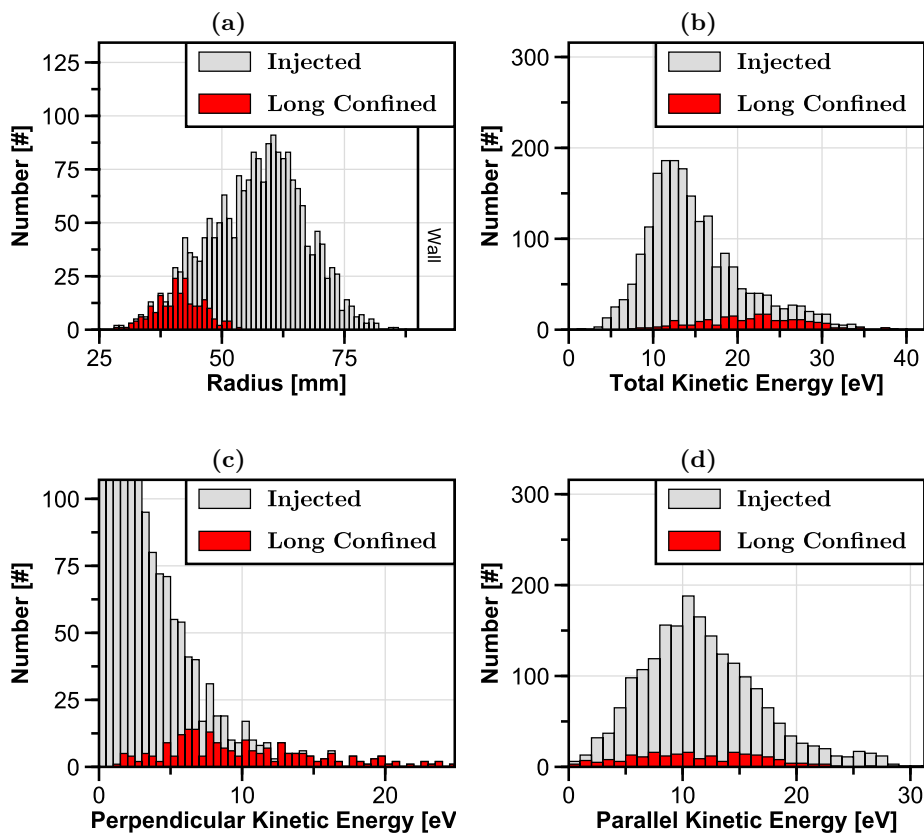
Figure 9.11: Illustrations of the radial and energy distributions of all confined positrons at their first mid plane crossing after the dump trigger. One can see that all long-confined positrons were located very close to the magnet (a) and that their total kinetic energy (b) is much higher than the average. This is due to the high perpendicular energy (c). There seems to be no correlation between the confinement time and the parallel energy (d).

9.6   Summary

This chapter has shown the general procedure of using experimentally collected data to extract energy distributions, as well as using the position and shape of the positron beam to initialize the simulation model. The results of the subsequent simulations were in good agreement with the experiments and reproduced the delay of the dump signal relative to the trigger signal. The simulations also revealed that positrons can gain energy if they are in the injection region while the electric potentials are turned off. These positrons are even able to resist the dump and remain trapped, so Chapter 10, which explores new dump schemes, will also address the removal of these high-energy positrons.

Chapter 10

Simulations of different dump scenarios

In Chapter 8, which discussed the confinement of positrons, it was mentioned that a dump scheme was used in the experiments to quickly empty the trap at different times after the injection, in order to estimate the confinement time. Dumping the trap content by switching the potentials required for the injection (E×B plates, RW1, Top1) back on again results in a strong signal, but it is apparently not sufficient to completely empty the trap. The simulations presented in Section 9.5 suggest that some positrons gain energy during the potential switch-off after the injection phase, allowing them to remain confined even after the dump.

The following describes how different dumping schemes were simulated in order to find a way to confirm the presence of these long-confined positrons, to determine how to remove them, and to investigate if there are schemes that provide additional information about the trapped positrons. First, the current dumping scheme and the influence of the switching speed on the positron ejection will be discussed in Section 10.1. Several ways of using the magnet case bias to eject confined positrons in Proto-APEX are explored in Section 10.2. Finally, in Section 10.3, different dumping schemes are investigated using the wall electrodes, which will be particularly useful for APEX-D.

All simulations were initialized with the identical 10 000 positrons used in the simulation shown in Figure 8.9, which achieved the longest confinement time. The dump was triggered at 50 µs and the simulation was terminated at 500 µs. Data labeled with "Fast" used the newly acquired fast high-voltage switches from Stahl-Electronics [66] (measured rise time: 116 ns), while "Ramp" used a linear ramp of 100 µs, if not stated otherwise. The magnet case was always biased to 15 V during the whole simulation, unless it is explicitly mentioned that it was switched as well.

10.1 Current dumping scheme

So far, the trap was emptied by simply switching the injection potentials back on after the hold phase (here: $V_{E\times B} = 220\,\text{V}$, $V_{RW1} = V_{Top1} = 18\,\text{V}$). The self-built switches can empty the trap in roughly $100\,\mu s$, but they have a delay of about $50\,\mu s$ before the potentials are strong enough to eject positrons and they are not able to eject the last $4\,\%$ of the content composed of the high-energetic positrons close to the magnet (see Figure 10.1).
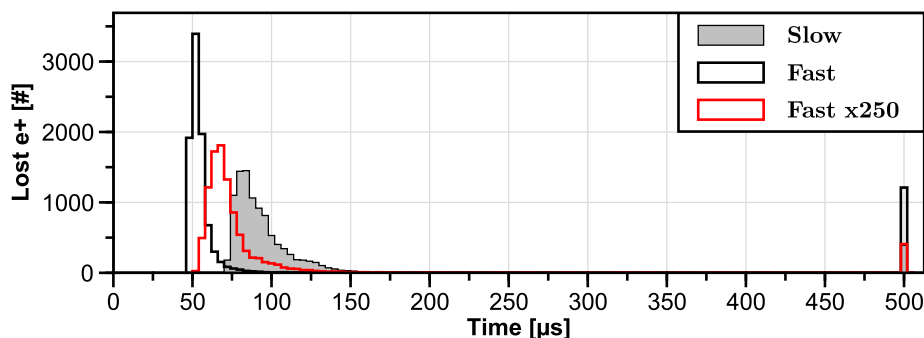
Figure 10.1: Result of the dump simulation for different switching speeds of the potentials. "Slow" uses the timings previously shown in Figure 9.7. "Fast" uses the new high-voltage switches or, respectively, slowed down by a factor of 250.

The high-voltages switches from Stahl-Electronics can dump most of the positrons in less than $50\,\mu s$ and without a significant delay, however, over $12\,\%$ of the trapped positrons remain confined. Due to the fast rise time in the order of $100\,\text{ns}$, the dump is very chaotic, with positrons hitting all wall electrodes as well as the shield plate and the E×B plates, as shown in Figure 10.2c. What is not shown is that about $1.5\,\%$ of the positrons were ejected from the main trap into the beam line.

These unfavorable properties can be mitigated by slowing down the rise time of the switch. At a factor of 250, almost no positrons annihilate on the plates anymore (Figure 10.2b) and the fraction of positrons that stay confined drops again to $4\,\%$. The annihilation locations on the wall remain fairly spread out compared to the slow switch, where most positrons are lost on RW8 (see Figure 10.2a). When slowing down the rise time of the fast switch by a factor of 1000, the number of positrons hitting RW3-RW6 is further reduced, though the fraction of long-confined positrons does not change significantly.

Figure 10.2: Locations of the positrons at the end of the simulations. The annihilation positions are shown in red, positrons that are still confined are illustrated in blue. (a) Slow switch, (b) fast switch slowed down by a factor of 250, (c) fast switch.

10.2   Magnet case bias

In Chapter 8, it was concluded that a high magnet case bias is necessary for good confinement. It can therefore be assumed that a negative voltage applied to the magnet case has a correspondingly bad effect on the confinement time. The simulation shows that a fast switch from $+15\,\text{V}$ to $-150\,\text{V}$ indeed dumps nearly the whole trap content almost instantaneously onto the magnet case, as shown and illustrated in Figure 10.3 and Figure 10.4a. Only 140 of the initially started 10 000 positrons remain confined, which can be further reduced to 24 by using $-200\,\text{V}$ and only 4 at $-250\,\text{V}$.
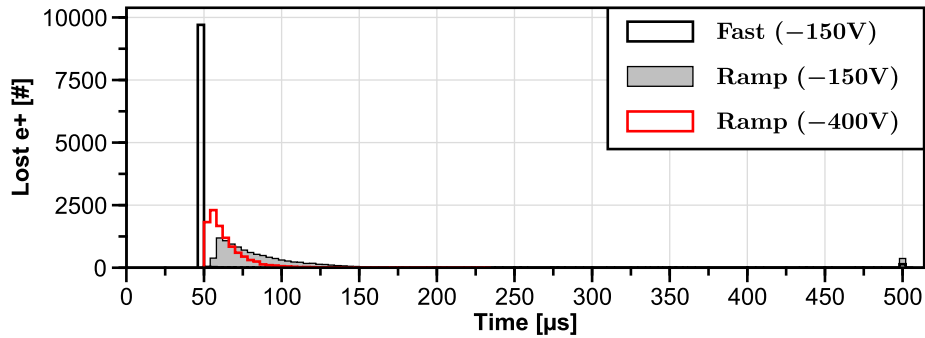
Figure 10.3: Simulation of dumping the trap content by changing the potential applied to the magnet case.
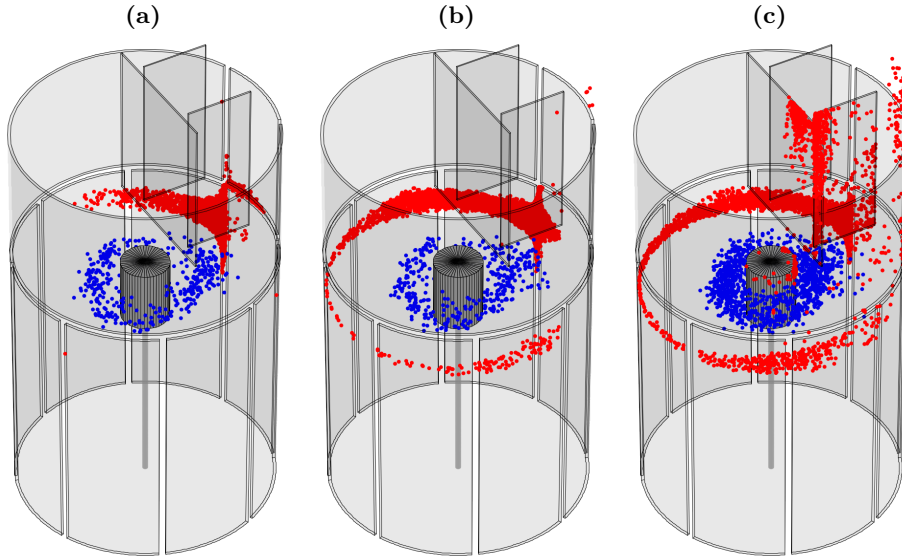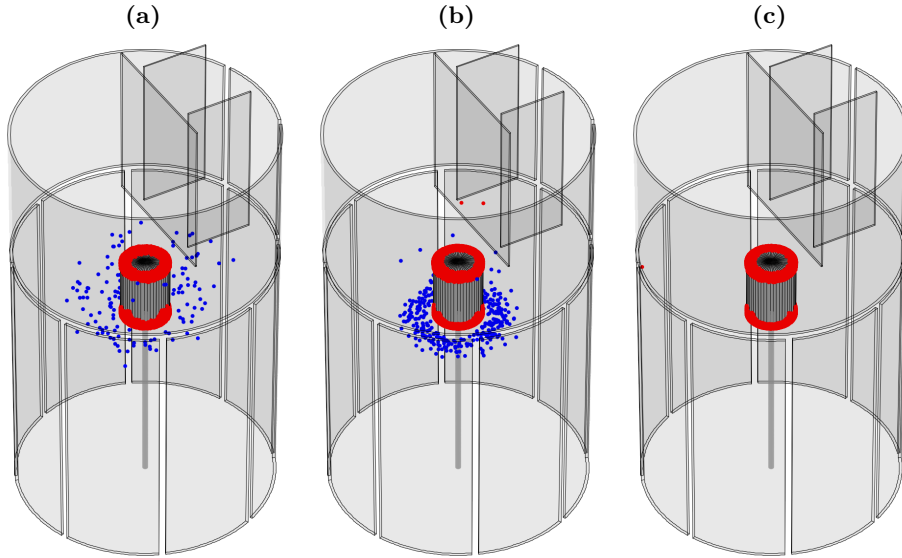
Figure 10.4: Locations of the positrons at the end of the simulations. The annihilation positions are shown in red, positrons that are still confined are illustrated in blue. (a) Fast switching of the magnet case bias to $-150\,\mathrm{V}$, (b) ramp to $-150\,\mathrm{V}$, (c) ramp to $-400\,\mathrm{V}$.

10.2.1   Ramp

If instead of switching the magnet case potential as quickly as possible, it is changed linearly, one can avoid possible pile-up effects in the gamma detectors, but also gain insight into the magnetic moment distribution of the trapped positrons, since the time of annihilation on the magnet case depends only on the applied potential and thus on time itself, as shown in Figure 10.5. About $3.5\,\%$ of all positrons are not dumped if a voltage of $-150\,\mathrm{V}$ is used. The small top–down asymmetry of the electric potential caused by the plates and the support rod of the magnet is the reason why most long-confined positrons accumulate at the bottom of the magnet, which can be seen in Figure 10.4b. Figure 10.4c shows that all positrons can be dumped if the applied voltage is increased to $-400\,\mathrm{V}$. One has to note that the E×B-drift induced by the high potential of the magnet case exceeds the grad-B and curvature-drift, resulting in a direction reversal of the toroidal rotation.
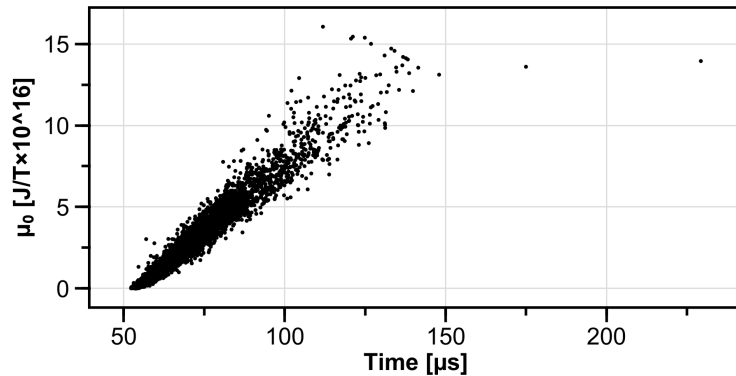
Figure 10.5: Dependency of the magnetic moment μ_0 of a positron on the time it hits the magnet case if the voltage applied to it is changed linearly from $15\,\mathrm{V}$ at $50\,\mu\mathrm{s}$ to $-400\,\mathrm{V}$ at $150\,\mu\mathrm{s}$.

10.2.2   Delayed magnet

Since the previous simulations confirmed that all confined positrons, even the highly energetic ones, can be dumped by switching the magnet case bias to a high negative voltage, this could potentially be used to verify the presence of these long-confined positrons in the experiments. In Figure 10.6, the simulation result shows that if the magnet is switched after the first main dump (here $200\,\mu\mathrm{s}$ later), virtually all remaining positrons are dumped. A closer look at the positrons hitting the magnetic case (Figure 10.7) reveals that they are indeed those long-confined particles already described in Section 9.5, that are located in proximity to the magnet and simultaneously have a high total energy.
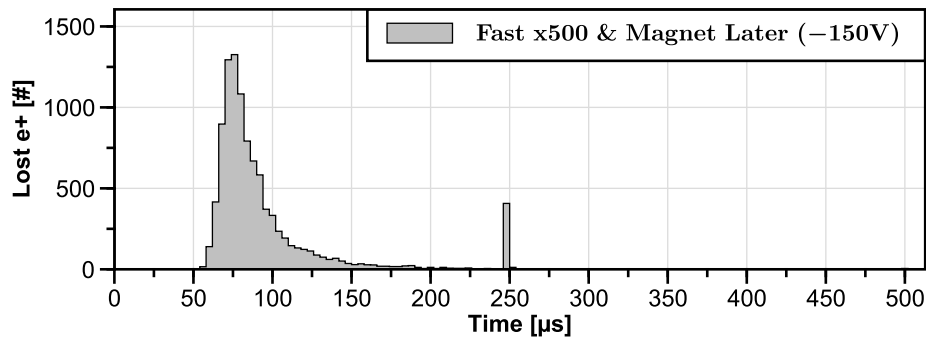
Figure 10.6: Simulation of the dump using the default scheme (switching on the injection potentials), but with the rise time slowed down by a factor of 500. At $250\,\mu\mathrm{s}$, the potential of the magnet case is switched from $15\,\mathrm{V}$ to $-150\,\mathrm{V}$.
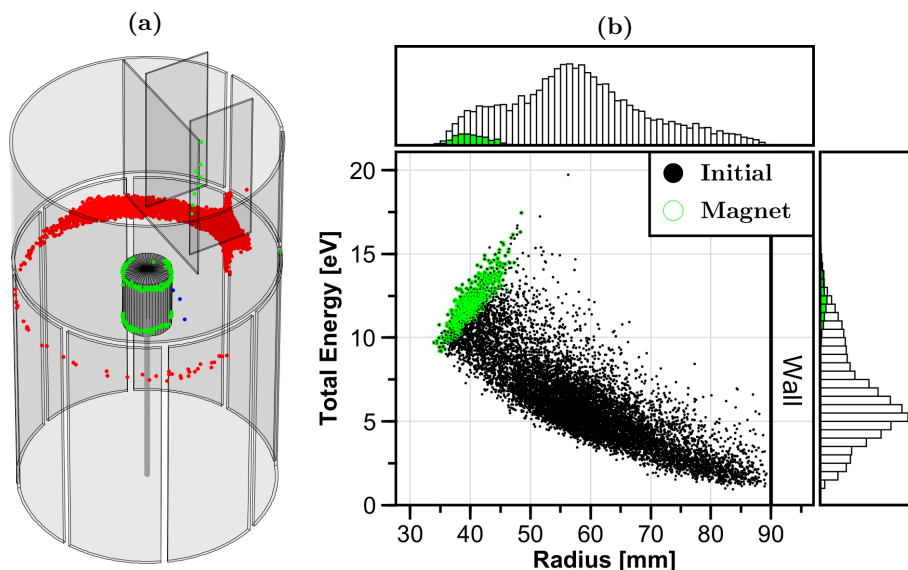
Figure 10.7: (a) Positrons that hit the magnet case are highlighted in green, positrons that are still confined are illustrated in blue, and all other annihilation positions are shown in red. (b) Scatter plot of the initial radial and total energy distributions of the simulated positrons. The positrons that hit the magnet are highlighted green.

10.3   Wall segment

Switching the magnet case bias to a high negative voltage seems to be a viable way to dump the complete trap content of Proto-APEX, according to the simulations. Of course, this is no longer possible for the next stage of the experiment, APEX-D, in which a levitating superconducting coil is used to generate the dipole magnetic field, since the levitation prohibits any control over the electric potential of the coil housing. This brings the wall electrodes back into focus, since these will also be present in APEX-D.

The simplest possible option would be to just use the RW1 and Top1 electrode to dump the trap content by switching the injection potentials back on again. However, the simulation reveals that this is anything but effective, with over 26 % of the positrons still trapped in the end as Figure 10.8 and Figure 10.9a show.

The next logical step to increase the efficiency of the dump is to increase the applied voltage. Since it is technically not straightforward to switch between three states (injection: 18 V, hold: 0 V, dump: 150 V) with a switch that has only two states, RW5 will take over the task of the dump. For that, the electrode structure was modified in the simulation so that RW5 spans the whole height of the trap (see Figure 10.9b). As Figure 10.8 shows, when RW5 is quickly switched to 150 V, only 3.5 % of the positrons remain confined. At 500 V, the number of long-confined positrons drops to only 6 of the initially started 10 000. Interestingly, these high voltages cause the trajectories of the still trapped positrons to

be mainly restricted to the region between the magnet and RW5, which can be clearly seen in Figure 10.9b. Some example trajectories are given in Figure D.3 in the Appendix.
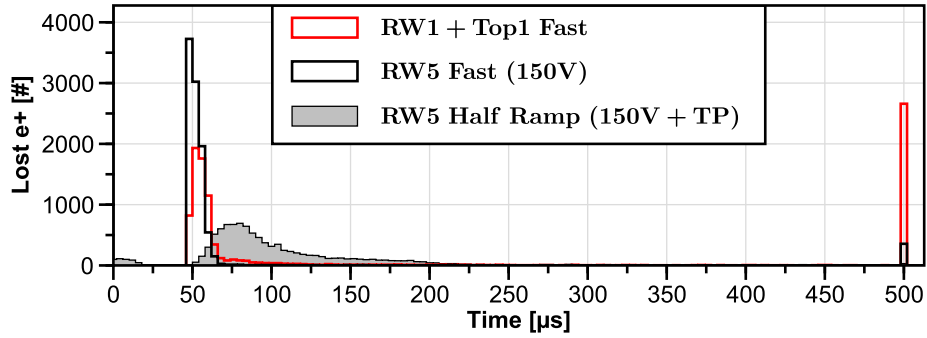
Figure 10.8: Comparison of dump simulations using different wall configurations: switching RW1 and Top1 to the injection biases of 18 V, a full-height RW5 switched to 150 V, and a vertically split full-height RW5 with the second half linearly ramped to 150 V within 400 μs with the target probe inserted 1 cm.
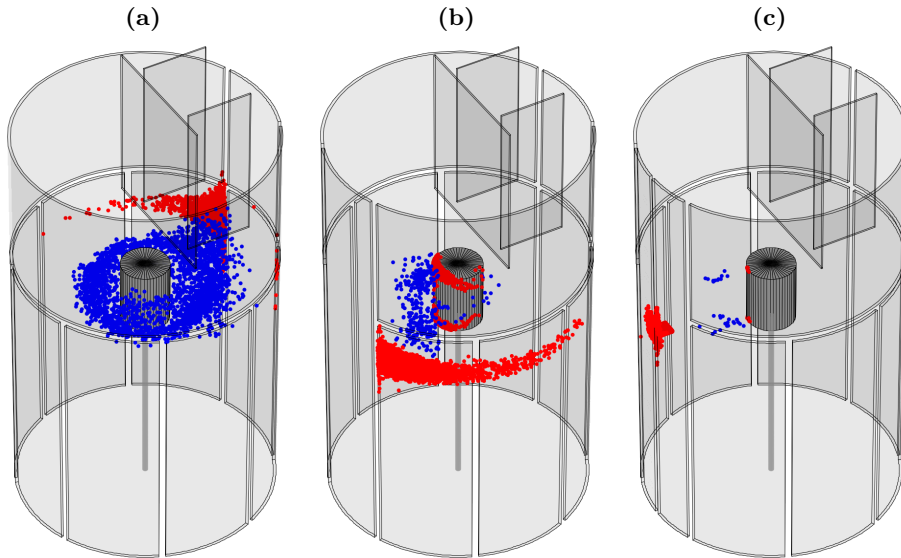
Figure 10.9: Locations of the positrons at the end of the simulations. The annihilation positions are shown in red, positrons that are still confined are illustrated in blue. (a) Switching RW1 and Top1 to the injection biases of 18 V, (b) full-height RW5 switched to 150 V, (c) vertically split full-height RW5, second half linearly ramped to 150 V within 400 μs with the target probe inserted 1 cm.

89

10.3.1 Limiter

Biasing RW5 to dump positrons opens up new possibilities. One question would be, for example, what happens during the dump if a limiter was inserted — or the target probe in case of Proto-APEX. In order for the positrons to be able to hit the target probe when the full-height RW5 electrode is biased, it had to be split vertically into two halves with the first one being grounded and only the second half being biased, as illustrated in Figure 10.9c.

For the simulation, the second half of RW5 was linearly ramped to 150 V within 400 µs and the target probe was inserted 1 cm into the confinement region. At the end of the simulation at 500 µs, only 30 of 10 000 initiated positrons were still confined; these were also restricted to orbits between the magnet and RW5, as mentioned before. Figure 10.8 shows that about 4.5 % of the positrons were lost within the first 25 µs. The reason for that is that the target probe was not present during the injection simulation of the started positron ensemble. Since the target probe was now inserted, however, positrons at the outer radii inevitably hit it immediately at the start of the simulation or as soon as they toroidally drifted far enough around the magnet to be able to hit the probe. An inserted target probe therefore slightly decreases the injection efficiency, but on the other hand virtually all confined positrons will annihilate at or near the target probe, as Figure 10.9c shows, which potentially allows for a better signal with a correspondingly adjusted gamma detector arrangement.

When the potential of RW5 is slowly ramped, as in this case, the time a positron hits the target probe is actually correlated to its initial radius, as shown in Figure 10.10. This dumping scheme is therefore an easy-to-implement diagnostic tool to retrieve the radial distribution of the confined positrons.
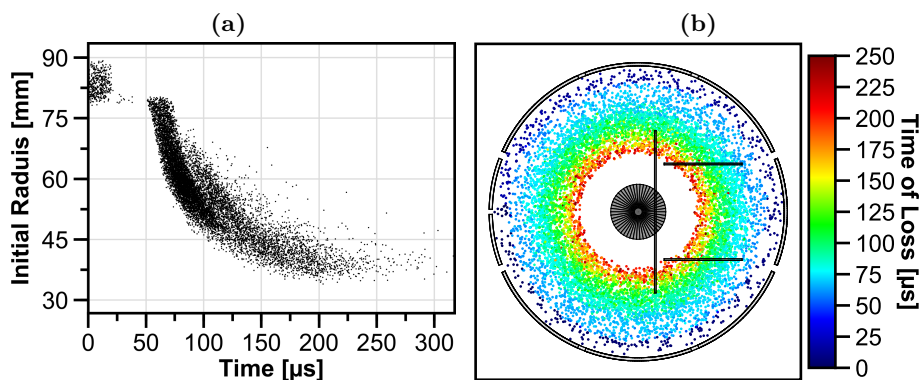
Figure 10.10: Illustrations of the dependence between the radius of a positron and the time of its annihilation. Positrons lost before 50 µs are the result of an simulation artifact caused by the fact that the target probe was not present during the simulation which created the started positron ensemble.

10.4   Conclusion

In Chapter 8, the confinement of positrons in Proto-APEX was discussed. There it was mentioned that in order to get a strong annihilation signal, the trap contents were dumped by turning the injection potentials back on. In Chapter 9, however, positrons were observed in the simulations that were able to resist being ejected because they gained energy when the electric potentials were turned off during the injection. Therefore, the goal of this chapter was to investigate different dump schemes that are capable of ejecting all positrons or that may have other advantages.

The simulations showed that using the new fast high-voltage switches (instead of the self-built electronics) actually worsened the efficiency of the dump. Slowing it down by a factor of 250 resulted in a very similar dump signal to the default dump scheme.

For Proto-APEX, a very efficient dumping scheme is to apply a high negative voltage to the magnet case. However, the rapid annihilation of all confined positrons could be problematic, as it could lead to pile-up signals in the BGO detectors. By increasing the voltage linearly to $-400\,\text{V}$ instead of switching instantly, the trap can be completely emptied without the risk of overloading the detectors. This also has the advantage of inferring the magnetic momentum distribution of confined positrons. The simulations also showed that by switching the magnet bias after the default dump, the existence of high-energy long-confined positrons can be confirmed.

Finally, different dumping schemes were simulated using only wall electrodes, as this is more relevant for the upcoming APEX-D experiment. The most promising scheme uses the slightly inserted target probe while linearly increasing the bias of one half of the split RW5 electrode to a high positive voltage. This scheme empties the trap almost completely, including the high-energy positrons. It also has the advantage that the time at which a positron is lost depends on its radial position before the dump, which could be used to reconstruct the radial distribution profile.

Chapter 11

Pulsed injection of positrons

Until now, positrons have been continuously injected into the trap. However, it is still uncertain whether this will remain the method of choice in the future or whether it will be necessary to fill the trap with multiple positron pulses. One reason for a pulsed injection could be, that it is not yet certain whether it is feasible to eject all positrons of the multi-cell trap (Section 3.2) at once or whether several ejection cycles are required to do so. Also, no definitive method has been chosen yet how electrons and positrons should enter the trap simultaneously. This could, for example, be accomplished by creating a non-neutral plasma of one species and then injecting the other, by injecting both species at the same time, or, if the positrons from the multi-cell trap need to be pulsed, an alternating injection of positrons and electrons.

In this chapter, it is investigated if positrons can effectively be injected via multiple pulses and to which extent the pulsed injected positrons differ from those that were injected continuously. Finally, Section 11.3 shows how a successful injection of multiple positron pulses could be experimentally confirmed in Proto-APEX.

11.1   Setup of the simulation

For the simulation, the $1\,\mu s$ long pulse shown in Figure 11.1a was used to switch the injection potentials (E×B plates, Top1, RW1, see Figure 4.1). The rise and fall times are $116\,ns$ and $157\,ns$ long, respectively, obtained from measurements of the recorded time traces of the high-voltage switches from Stahl-Electronics [66], which were also used in Chapter 10. During each pulse, 1000 positrons were started within a $100\,ns$ time window.

Figure 11.1b shows how the injection efficiency depends on the delay between the launch of the positrons and the start of the pulse. Since the positrons are initiated in the beam line (see Figure 4.1), they need about $200\,ns$ to reach the main chamber, so the $100\,\%$ injection efficiency is still maintained even with no delay. As the choice of the delay has no discernible effect, it has been set to $0\,ns$ in all of the following simulations.
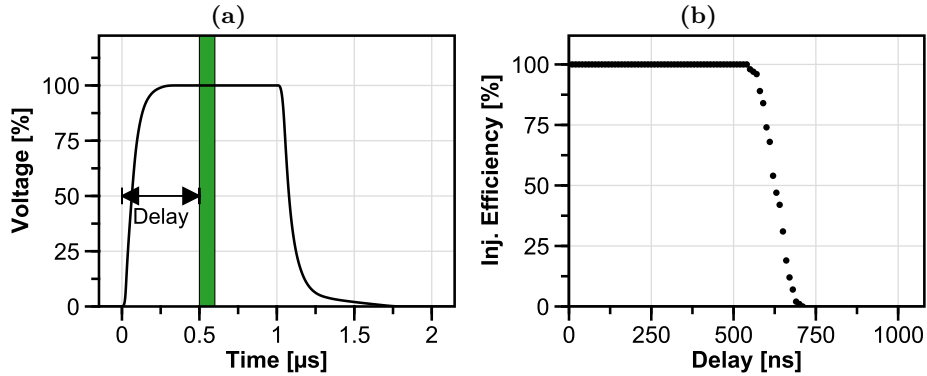
Figure 11.1: (a) Pulse used for switching the electrodes. The 100 ns positron launch window is shown in green. (b) Injection efficiency when varying the delay of the positron launch relative to pulse start.

11.2   Simulation result

In the first simulation, the pulse shown in Figure 11.1a was used to switch the electrodes five times every 10 µs while also starting 1000 positrons with every pulse. The result, as seen in Figure 11.2, indicates that more than 4600 of the total 5000 positrons launched are confined at the end of the simulation at 50 µs (10 pulses: $\approx 7300/10\,000$, 15 pulses: $\approx 8750/15\,000$). This is unexpected, since a simulation with a continuously filled trap showed that about 17 % of the confined positrons are lost when an empty pulse occurs. Repeated simulations with 100 and 10 000 positrons per pulse yielded the same ratio of trapped to launched positrons. The same ratio was also obtained with a five times smaller time step (0.1 ns instead of 0.5 ns) and a two times longer rise/fall time of the used pulse.
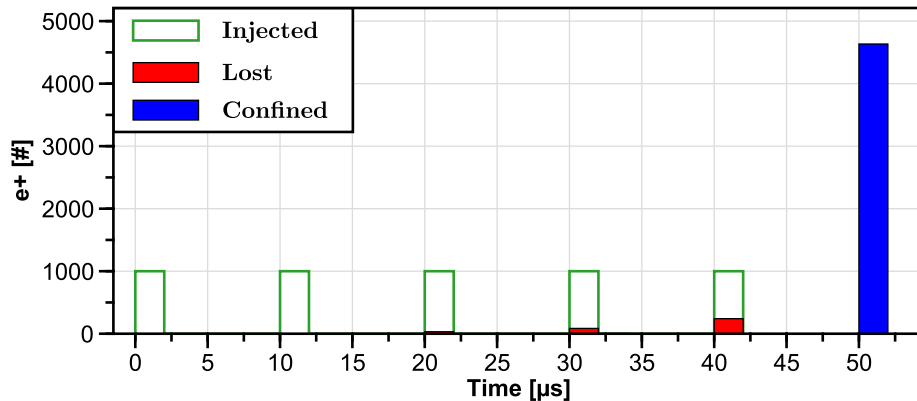
Figure 11.2: Simulation result after five injection pulses (green) with 1000 positrons each. Positrons lost during a pulse are illustrated in red. The blue bin at 50 µs shows the number of positrons that remained confined until the end of the simulation.

(a)

(b) **(c)**

(d) **(e)**
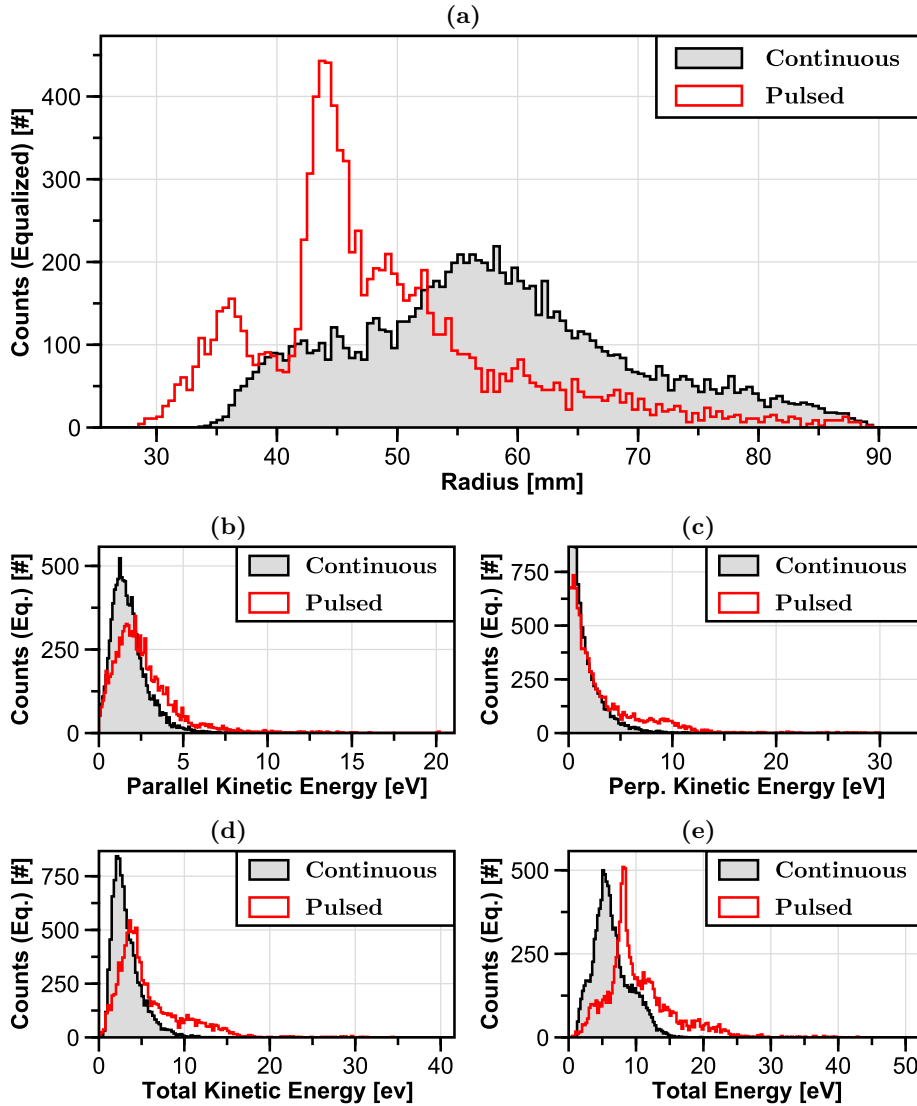
Figure 11.3: Radial and energy distribtions of the positrons that remained confined at the end of the simulation compared to the ones resulting from a continuous injection. The distributions of the pulsed simulation were scaled up to match the number of particles of the continous injection simulation.

Looking at Figure 11.3, which shows comparisons of the radial and energy distributions of confined positrons between continuous and pulsed injection, reveals that the confined positrons are closer to the magnet and have noticeably more energy in case of a pulsed injection than if they were injected continuously. These findings are qualitatively identical to the ones found in Section 9.5, where the switch-off of the injection potentials resulted in a highly energetic positron population close to the magnet which can withstand the ejection when the potentials are switched back on again. In addition, due to the short pulse length, the positrons do not have time to move outwards to the wall due to the asymmetric

electric potential while they drift toroidally around the magnet (see Figure 8.1), so the majority of freshly injected positrons stays close to the magnet after the potentials are switched off.

The energy gain is caused by the switching of the injection potentials, as mentioned earlier, and since they are switched on and off multiple times during the pulsed injection, this results in multiple branches of positrons with different energies close to the magnet ($< 45\,\mathrm{mm}$), as illustrated in Figure 11.4. In total, three branches can be distinguished: a low energy branch consisting of positrons with an average kinetic energy of about $5\,\mathrm{eV}$, a medium energy branch between $10\,\mathrm{eV}$ and $20\,\mathrm{eV}$, and a branch of high-energy positrons with kinetic energies up to $35\,\mathrm{eV}$. Moreover, it can be seen that the switching of the potentials mainly affects the positrons that have already been trapped, since the positrons injected with the last pulse (red) are relatively close together in terms of energy and radius compared to the positrons that were injected previously, especially during the first pulse. With two pulses, only the medium energy branch is visible, but with three pulses and more, all three branches are visible. Since even with 15 pulses only these three branches are produced, this indicates that the number of branches is independent of the number of pulses. The reason for the branching itself and the fact that there are exactly three branches is the task of a future investigation. However, the fact that there are highly energetic positrons present, which are rather disadvantageous for achieving plasma conditions, a cooling gas is most likely needed when using pulsed injection.
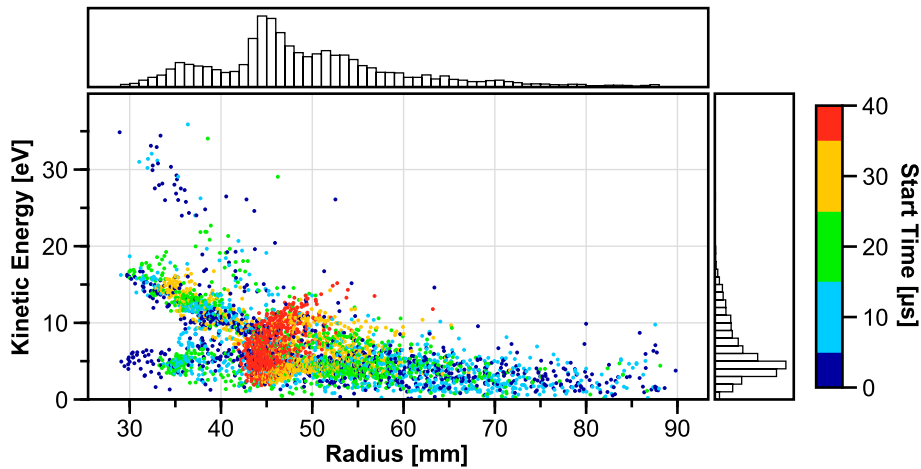

Figure 11.4: Energy and radial distribution of the confined positrons at the end of the simulation. The color indicates during which pulse the positrons were injected.

11.3   Realistic simulation

To see if an experimental proof of a successful pulsed injection is possible with the equipment currently available, the simulation will need some adjustment.

First, the pulse duration was increased to $10\,\mu s$ since this is the limit of the fast high-voltage switches from Stahl-Electronics. Second, due to the longer pulses, the time between pulses was increased to $50\,\mu s$ to give the positrons enough time to spread out evenly in the trap. Furthermore, neither the BGT nor the MCT is installed, so no pulses can be injected, only the continuous remoderated positron beam of NEPOMUC, which can provide $5\cdot10^7\,e^+/s$. Since the deflector would be biased when the injection potentials are off to prohibit positrons from reaching the chamber, positrons were launched only in a $20\,\mu s$ time frame around each pulse to ensure that a maximum number of positrons enter the trap during the pulse, but also to avoid the unnecessary simulation of positrons that cannot be injected in the first place. The used timing scheme is summarized in Figure 11.5.
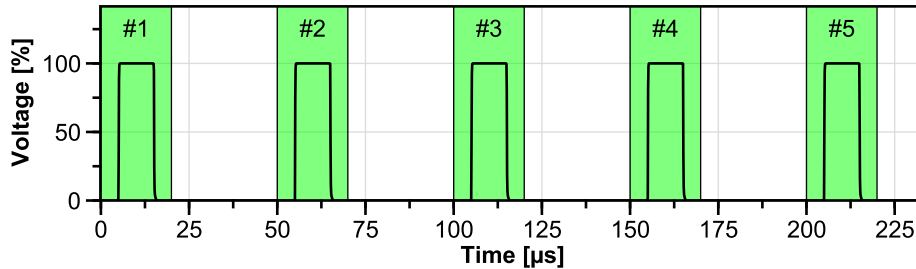
Figure 11.5: Timing scheme of the simulation. 1000 positrons are launched in the beam line within a time frame of $20\,\mu s$ every $50\,\mu s$ (highlighted in green). The electrodes are switched on for $10\,\mu s$ for every pulse.

Looking at the simulation result shown in Figure 11.6, it can be seen that the difference in confined positrons between only one injection pulse and two pulses is about $25\,\%$, and between one pulse and three pulses about $40\,\%$, so an experimental discrimination is within the realm of possibility. When using any dump scheme presented in Chapter 10 that ensures that all positrons are dumped, including the ones which gained energy during the switching, a differentiation of two and three injection pulses might also be possible despite the small difference of only $10\,\%$ in trapped positrons. Discrimination of more than three pulses is unlikely, since about as many positrons are lost as can be injected in one pulse.

An experiment with two injection pulses can directly be realized with the DG645 digital delay generator [67], as shown in Figure 11.7. Here, the magnet bias is used to dump the trap content at the end of the second hold phase, like described in Section 10.2. The T0 output of the delay generator (or a second pulse generator) can also be used to gate the data acquisition and/or to reset the time stamp of the digitizer [44]. This setup also allows for a quick reconfiguration of the experiment to cross-check the results, e.g. by using only the first or only the second pulse, or by switching the potentials in the second pulse but without opening the deflector.
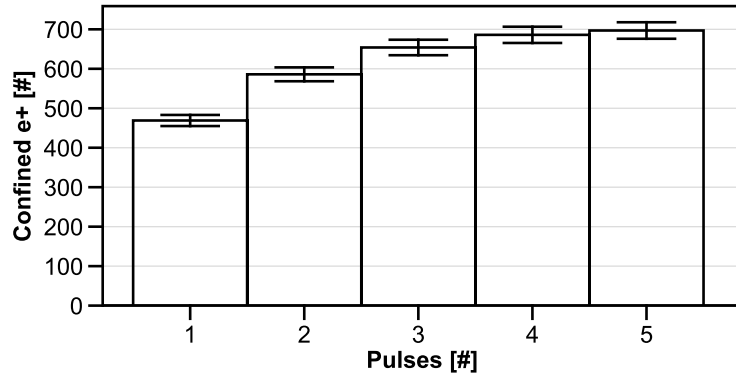
Figure 11.6: Number of confined positrons at the end of the simulation for different numbers of injection pulses.
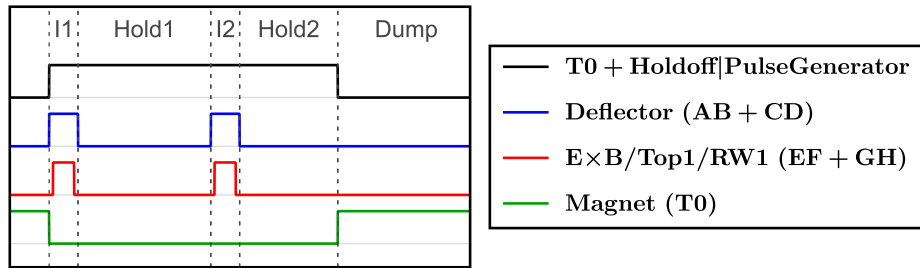
Figure 11.7: Possible timing scheme with two pulses using the DG645 digital delay generator [67]. The channel configurations are given in the parentheses. High levels mean "on", low levels mean "off" except for the magnet where the low level refers to the injection/confinement setting and a high level to the dump setting (see Section 10.2). The T0 output or another pulse generator is used for synchronization with the data aquisition.

11.4 Conclusion

This chapter investigated the possibility of injecting multiple pulses of positrons into a magnetic confinement device. This is important because it is not yet clear whether all the positrons accumulated in the multi-cell trap can be ejected at once or whether several pulses are needed.

The simulations performed showed that such a pulsed injection is indeed possible, but they also indicated that the repeated switching of the injection potentials causes the positrons to gain energy, similar to what was observed in Section 9.5. This means that a cooling gas is most likely needed to reduce the energy of the positron so that a small Debye length can be achieved. Furthermore, the simulations showed that pulsed injection can already be tested in Proto-APEX by repeatedly blocking the incoming continuous positron beam from NEPOMUC.

Chapter 12

Summary

The APEX collaboration plans to accumulate the positrons of the NEPOMUC beam using a buffer-gas trap system and a multi-cell trap, and then inject this intense positron pulse together with electrons into a magnetic dipole trap with a levitating, superconducting coil (APEX-D) or a stellarator (EPOS) to create the world's first magnetized low-energy pair plasma. Pair plasmas are particularly interesting to study because they have exceptional stability properties compared to an ion–electron plasma due to the equal masses of the species. So far, experiments have been conducted with a predecessor device, Proto-APEX, in which the central dipole field is generated by a permanent magnet.

This device has been extensively studied by numerical single-particle simulations, using analytical formulae to describe the magnetic fields, a Lorentz-equation solver to calculate the electric potentials inside the trap, and the Boris algorithm to propagate positrons in these fields. The simulation framework was described in detail at the beginning, including the introduction of an electric potential solver using an octree data structure, which is capable of resolving small geometric details in large volumes by refining the mesh in important areas while maintaining a coarse resolution everywhere else. This work has also shown, through simulations and comparisons with available experimental data, that the radial deposition of positrons during injection can be considerably altered by varying the steering coil current I_r as well as the magnet case bias.

This knowledge could immediately be used to show that positrons can be trapped for longer if they are deposited closer to the magnet. Simulations have also shown that the E×B plates, which can be hit by positrons and thus limit their confinement time, can be shortened without compromising the injection efficiency. If, in addition, the magnet case is biased high enough to prevent positrons from scattering into the loss cone, the total confinement time was increased by a factor of five.

After modelling a previous beam time in the simulation software, it was discovered that positrons can gain energy when the injection potentials are switched off. When the potentials are switched back on to empty the trap, these positrons are able to resist and remain trapped.

Several new methods of emptying the trap content were then investigated. Biasing the magnet case to a high negative voltage gave the best results, including the ejection of the high-energy positrons mentioned above. It was also shown that by dumping the content by turning on the injection potentials and then biasing the magnet case after a delay to eject the remaining high energy positrons, their existence or non-existence could be experimentally verified. Another promising option would be to bias a wall electrode to a high positive voltage in combination with a limiter. This method would also work in APEX-D and would allow to obtain some information about the radial distribution of the confined positrons if the voltage was increased slowly enough.

Finally, this work has shown that the trap can be filled by several positron pulses in case it is not possible to empty the multi-cell trap in one large pulse. However, this results in a complex energy and radial distribution which would most likely require the use of an additional cooling gas inside APEX-D to lower the energy of the positrons in order to achieve plasma conditions. Last but not least, simulations have shown that pulsed injection can already be verified experimentally in Proto-APEX using the continuous positron beam of NEPOMUC.

Chapter 13

Outlook

Since FRM II is not due to restart until the beginning of 2024 at the earliest, it was decided to carry out experiments with Proto-APEX at AIST in Japan. The focus of these experiments will be on the improvement of the confinement and the field testing of the gamma detection system. After Proto-APEX was removed, the RFA was installed at the open beam port, which will be used to characterize the NEPOMUC positron beam before and after the installation of the BGT and accumulator once the reactor is back in operation.

From the simulation point of view, APEX-D will become more and more important. For these simulations, the use of the octree potential solver will be crucial, which will be further optimized, especially in terms of memory usage by exploiting symmetries within the boundary and by assigning a unique ID to the cells to store them efficiently in a hashmap. To also reduce the time needed to solve the electric potentials on the regular grid, there are plans to backport some aspects of the octree solver, such as the convergence test and the multithreading implementation. Since in a charged cloud or in a plasma the assumption of single particles becomes less and less true with increasing number of particles, it is conceivable to use the software presented here and interface it with a PIC code.

Since this software is written in Java, improvements can be expected through future enhancements to Java itself. Project Lilliput, for example, has the goal to reduce the size of object headers from currently 96/128 bit (with and without compressed class pointers) to 64 bit and potentially even 32 bit [68], which would noticeably reduce the memory consumption of octrees. Project Valhalla aims to completely rethink Java's distinction between primitive values (int, double, . . .) and objects [62]. This includes the introduction of value classes and primitive classes, which would both reduce the memory consumption of the cells of an octree and also increase the performance in general. Project Panama wants to bring the Java virtual machine and native code closer together [69]. This includes adding an API for vector computation to make use of SIMD (Single Instruction, Multiple Data) capabilities of modern CPUs.

Appendices

A Symbols

Symbol/Name	Description
μ_0	Vacuum permeability
$\vec{a}$	Vector
$\hat{a}$	Unit vector
$a \leftarrow b$	Assignment operator
$a \mathbin{\%} b$	Modulo operator
$a \ll b$	Left shift operator
$a \mid b$	Bitwise OR operator
nint(x)	Round to closest integer
$\lfloor x \rfloor$	Floor function
$\lceil x \rceil$	Ceiling function
erfc(x)	Complementary error function

B Elliptic integrals

To calculate the magnetic field of a current loop (see Section 5.2.1) and a continuous finite solenoid (see Section 5.2.2), respectively, two or all three kinds of the complete elliptic integrals have to be evaluated at each point. There are many ways available to solve these integrals including brute force numeric integration and power series. Here, they were computed very efficiently by using the method of the arithmetic-geometric mean [70, Eqs. 19.8(i)].

The arithmetic–geometric mean, $\mathrm{agm}(x, y)$, is defined as

$$a_0 = x \tag{1}$$

$$g_0 = y \tag{2}$$

$$a_{n+1} = \frac{1}{2}\left(a_n + g_n\right) \tag{3}$$

$$g_{n+1} = \sqrt{a_n \cdot g_n} \tag{4}$$

where the two interdependent sequences a_n and g_n converge quadratically to the same value.

B.1 Complete elliptic integral of the first kind

The complete elliptic integral of the first kind can be calculated directly using the arithmetic–geometric mean [70, Eq. 19.8.5]:

$$K(k) = \int_0^{\frac{\pi}{2}} \frac{d\theta}{\sqrt{1 - k^2 \sin^2 \theta}} = \frac{\pi}{2 \cdot \mathrm{agm}\left(1, \sqrt{1 - k^2}\right)} \tag{5}$$

B.2 Complete elliptic integral of the second kind

With the additional sequence

$$c_n = \sqrt{a_n^2 - g_n^2} \tag{6}$$

which convergence quadratically to 0, the complete elliptic integral of the second kind is given by [70, Eq. 19.8.6]:

$$E(k) = \int_0^{\frac{\pi}{2}} \sqrt{1 - k^2 \sin^2 \theta}\, d\theta \tag{7}$$

$$= \frac{\pi}{2 \cdot \mathrm{agm}\left(1, \sqrt{1 - k^2}\right)} \cdot \left(a_0^2 - \sum_{n=0}^{\infty} 2^{n-1} c_n^2\right) \tag{8}$$

$$= K(k) \cdot \left(a_1^2 - \sum_{n=2}^{\infty} 2^{n-1} c_n^2\right) \tag{9}$$

Because the second kind depends on the elliptical integral of the first kind and both are required in any case for the calculation of the magnetic field, it is useful to calculate the first and the second kind at the same time to save some valuable computing power.

B.3   Complete elliptic integral of the third kind

For the evaluation of the last integral, the following sequences are necessary:

$$p_{n+1} = \frac{p_n^2 + a_n g_n}{2 p_n}, \quad p_0 = \sqrt{1 - \alpha} \tag{10}$$

$$\epsilon_n = \frac{p_n^2 - a_n g_n}{p_n^2 + a_n g_n} \tag{11}$$

$$Q_{n+1} = \frac{1}{2} Q_n \frac{p_n^2 - a_n g_n}{p_n^2 + a_n g_n}, \quad Q_0 = 1 \tag{12}$$

Here p_n converges quadratically to the same value as a_n and g_n, ϵ_n and Q_n converge to 0 quadratically and faster than quadratically, respectively. The complete elliptic integral of the third kind can now be computed with [70, Eq. 19.8.7]:

$$\Pi(\alpha, k) = \int_0^{\frac{\pi}{2}} \frac{d\theta}{\left(1 - \alpha \sin^2 \theta\right) \sqrt{1 - k^2 \sin^2 \theta}} \tag{13}$$

$$= \frac{\pi}{4 \cdot \operatorname{agm}\left(1, \sqrt{1 - k^2}\right)} \cdot \left(2 + \frac{\alpha}{1 - \alpha} \sum_{n=0}^{\infty} Q_n\right) \tag{14}$$

$$= \frac{K(k)}{2} \cdot \left(2 + \frac{\alpha}{1 - \alpha} \sum_{n=0}^{\infty} Q_n\right) \tag{15}$$

C Parameter settings

Table 1: Parameter settings of the experiments and simulations.

Figure	I_r [A]	I_θ [A]	E×B [V]	Magnet [V]	Top1 [V]	Top2 [V]	RW1 [V]	RW2-8 [V]
7.1	-9	4	210	12	12	0	12	0
7.2a	-8.3	4	220	0	14	0	20	0
7.2b	-8.3	4	220	30	14	0	20	0
7.4	-8.3	4	220	8	14	0	14	0
7.5	-8.3	4	-	8	14	0	20	0
7.6	-8	4	180	8	-	0	14	0
7.7	-8	4	200	8	22	0	-	0
7.8	-8.3	4	220	-	14	14	20	0
7.9	-8.3	-	220	8	14	0	20	0
7.10	-	4	220	8	14	0	20	0
7.11	-	-	-	0	12	0	12	0
8.1	-9	3	210	12	12	0	12	0
8.3a	-8.3	4	220	8	14	0	22	0
8.3b	-8.3	4	220	0	14	14	22	0
8.4	-4	-1	220	8	18	0	18	0
8.5	-5.5	-1	230	12	12	0	12	0
8.6	-3.7	-1	220	8	18	0	18	0
8.7	-6.2	-1	220	20	18	0	18	0
8.9	-7	-1	220	15	18	0	18	0
9.3	-4	5.5	250	12	14	0	20	0
9.4	-	-	250	12	14	0	20	0
9.5	-4	5.5	250	20	15	0	20	0
9.6	-4	5.5	250	20	15	0	20	0
9.8	-4	5.5	250	20	15	0	20	0
11.2	-7	-1	220	15	18	0	18	0
11.6	-7	-1	220	15	18	0	18	0
D.1	-9	3	210	12	12	0	12	0
F.x	-9	3	210	12	12	0	12	0

D Trajectories of positrons

D.1 Shorter E×B plates
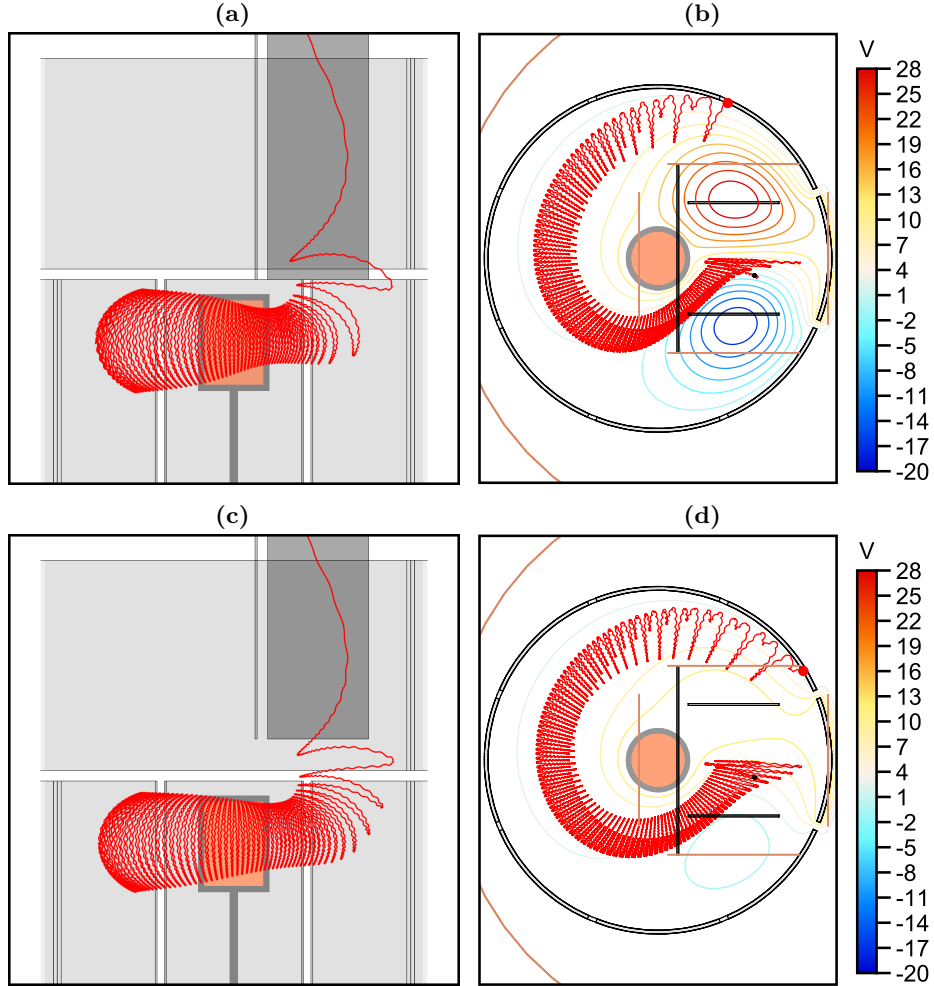
Figure D.1: Simulation of a trajectory in traps with different plate lengths: 135 mm((a) & (b)) and 115 mm (c) & (d)). The sample positron trajectory shown in red was started with $E_{\parallel 0} = 5\,\mathrm{eV}$ and $E_{\perp 0} = 0.5\,\mathrm{eV}$. [63].

D.2 Long-confined positrons
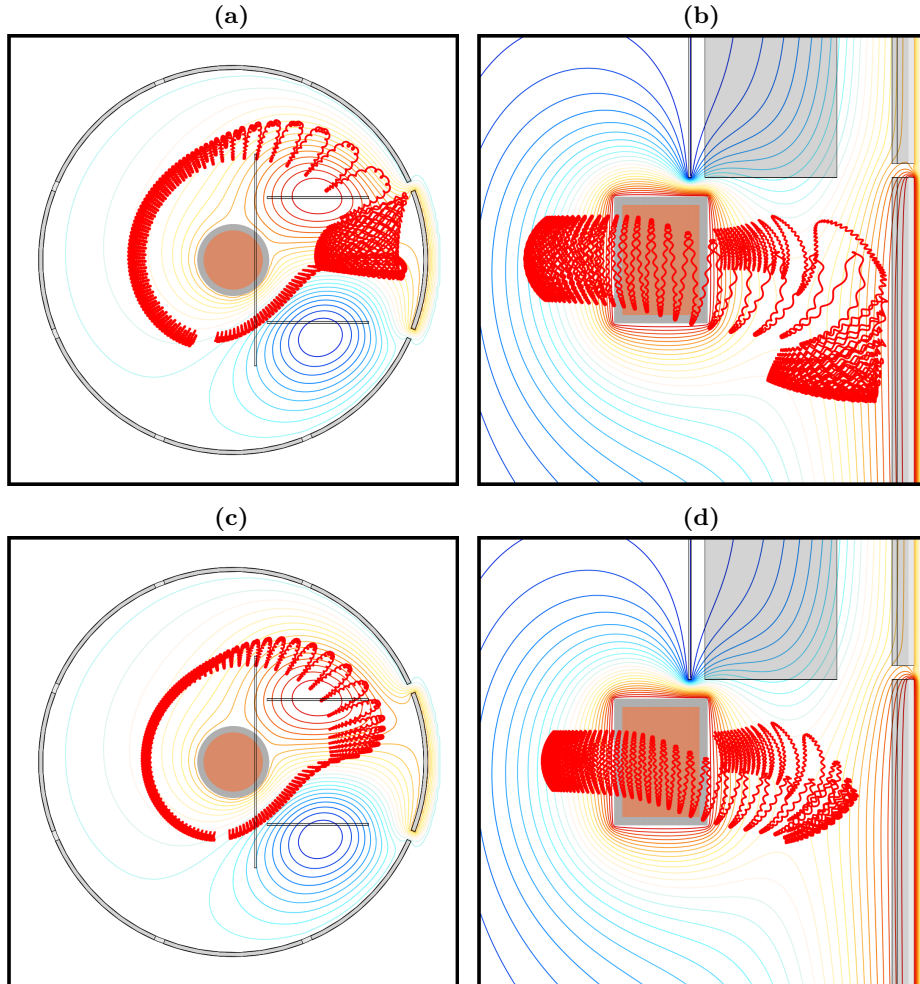
Figure D.2: Trajectory of the long-confined positron with start time 2.37 µs. (a) and (b) is the trajectory before the hold-phase began. (c) and (d) is the trajectory after the dump. at the end of the simulation.

D.3 Confined positron after RW5 dump
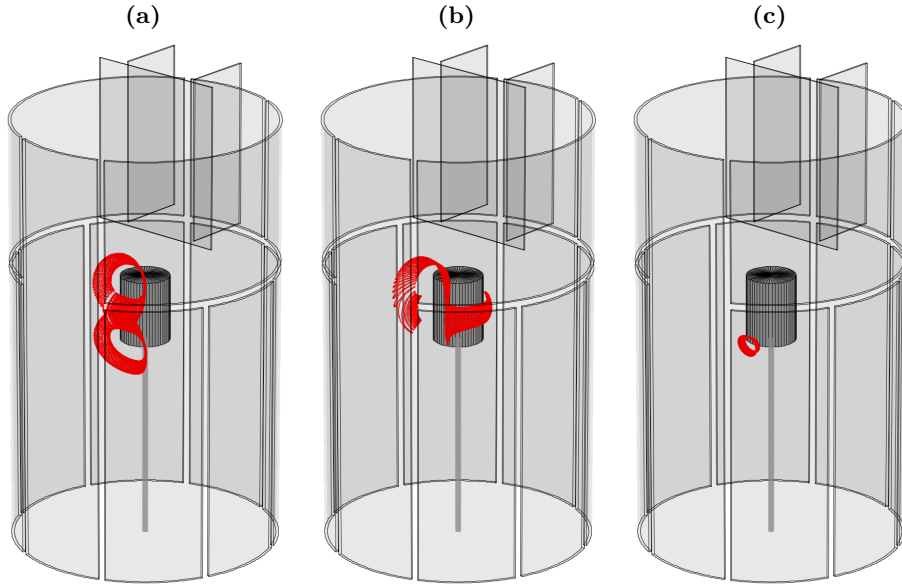
Figure D.3: Illustration of some of the possible trajectories a positron can take when the full-height RW5 electrode is biased to a high voltage. (a) Eight-shaped trajectory. (b) Trajectory very close to the magnet with an arc when being between the magnet and RW5. The arc can switch from being on the top to being on the bottom. (c) Trajectory with just the lower part of the eight-shaped trajectory. This kind of trajectory can also form at the top pole of the magnet.

E   Algorithms

E.1   Multigrid preconditioning

Algorithm 3: SOR iteration for different multigrid levels. The full SOR iteration is performed with $d = 1$.

Input: The potential array: Φ

The multigrid preconditioning level: d (1,2,4,...)

The successive over-relaxation parameter: ω

Function: ITERATESOR(Φ, d, ω)

for $x \leftarrow d$ to $\text{length}_x(\Phi) - d$ by d do

for $y \leftarrow d$ to $\text{length}_y(\Phi) - d$ by d do

for $z \leftarrow d$ to $\text{length}_z(\Phi) - d$ by d do

if ISGEOMETRY(x, y, z) is false then

$\Phi_{x,y,z} \leftarrow (1 - \omega) \cdot \Phi_{x,y,z} +$
$$\frac{\omega}{6} \cdot (\Phi_{x-d,y,z} + \Phi_{x+d,y,z} +$$
$$\Phi_{x,y-d,z} + \Phi_{x,y+d,z} +$$
$$\Phi_{x,y,z-d} + \Phi_{x,y,z+d})$$

Algorithm 4: Prolongation of values from a coarse grid to the next finer grid.

Input: The potential array: Φ

The multigrid preconditioning level: d (2,4,...)

Function: PROLONGATE(Φ, d)

$\quad e \leftarrow d/2$

$\quad$ **for** $x \leftarrow e$ **to** $\text{length}_x(\Phi) - e$ **by** e **do**

$\quad\quad$ **for** $y \leftarrow e$ **to** $\text{length}_y(\Phi) - e$ **by** e **do**

$\quad\quad\quad$ **for** $z \leftarrow e$ **to** $\text{length}_z(\Phi) - e$ **by** e **do**

$\quad\quad\quad\quad$ **if** ISGEOMETRY(x,y,z) is false **then**

$\quad\quad\quad\quad\quad$ **if** $x \,\%\, d = 0$ **and** $y \,\%\, d = 0$ **and** $z \,\%\, d \neq 0$ **then**

$\quad\quad\quad\quad\quad\quad$ $\Phi_{x,y,z} \leftarrow 1/2 \cdot (\Phi_{x,y,z-e} + \Phi_{x,y,z+e})$

$\quad\quad\quad\quad\quad$ **else if** $x \,\%\, d = 0$ **and** $y \,\%\, d \neq 0$ **and** $z \,\%\, d = 0$ **then**

$\quad\quad\quad\quad\quad\quad$ $\Phi_{x,y,z} \leftarrow 1/2 \cdot (\Phi_{x,y-e,z} + \Phi_{x,y+e,z})$

$\quad\quad\quad\quad\quad$ **else if** $x \,\%\, d \neq 0$ **and** $y \,\%\, d = 0$ **and** $z \,\%\, d = 0$ **then**

$\quad\quad\quad\quad\quad\quad$ $\Phi_{x,y,z} \leftarrow 1/2 \cdot (\Phi_{x-e,y,z} + \Phi_{x+e,y,z})$

$\quad\quad\quad\quad\quad$ **else if** $x \,\%\, d = 0$ **and** $y \,\%\, d \neq 0$ **and** $z \,\%\, d \neq 0$ **then**

$\quad\quad\quad\quad\quad\quad$ $\Phi_{x,y,z} \leftarrow 1/4 \cdot (\Phi_{x,y-e,z-e} + \Phi_{x,y-e,z+e} +$

$\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad \Phi_{x,y+e,z-e} + \Phi_{x,y+e,z+e})$

$\quad\quad\quad\quad\quad$ **else if** $x \,\%\, d \neq 0$ **and** $y \,\%\, d = 0$ **and** $z \,\%\, d \neq 0$ **then**

$\quad\quad\quad\quad\quad\quad$ $\Phi_{x,y,z} \leftarrow 1/4 \cdot (\Phi_{x-e,y,z-e} + \Phi_{x-e,y,z+e} +$

$\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad \Phi_{x+e,y,z-e} + \Phi_{x+e,y,z+e})$

$\quad\quad\quad\quad\quad$ **else if** $x \,\%\, d \neq 0$ **and** $y \,\%\, d \neq 0$ **and** $z \,\%\, d = 0$ **then**

$\quad\quad\quad\quad\quad\quad$ $\Phi_{x,y,z} \leftarrow 1/4 \cdot (\Phi_{x-e,y-e,z} + \Phi_{x-e,y+e,z} +$

$\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad \Phi_{x+e,y-e,z} + \Phi_{x+e,y+e,z})$

$\quad\quad\quad\quad\quad$ **else if** $x \,\%\, d \neq 0$ **and** $y \,\%\, d \neq 0$ **and** $z \,\%\, d \neq 0$ **then**

$\quad\quad\quad\quad\quad\quad$ $\Phi_{x,y,z} \leftarrow 1/8 \cdot (\Phi_{x-e,y-e,z-e} + \Phi_{x-e,y-e,z+e} +$

$\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad \Phi_{x-e,y+e,z-e} + \Phi_{x+e,y-e,z-e} +$

$\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad \Phi_{x-e,y+e,z+e} + \Phi_{x+e,y-e,z+e} +$

$\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad \Phi_{x+e,y+e,z-e} + \Phi_{x+e,y+e,z+e})$

E.2   Symmetric electric potentials

Algorithm 5: SOR iteration at the symmetry boundary.

Input: The potential array: Φ
The size of the domain: nx, ny, nz
The multigrid preconditioning level: d (1,2,4,...)
The successive over-relaxation parameter: ω

Function: IterateBoundarySymmetricX$(\Phi, nx, ny, nz, d, \omega)$

> $l_x \leftarrow \text{length}_x(\Phi)$
> // actual length of Φ in x-direction $= \lceil nx/2 \rceil$
> $\delta_x \leftarrow (l_x - 1) \bmod d$
> $x \leftarrow l_x - 1 - \delta_x$
>
> **for** $y \leftarrow d$ **to** $ny - d$ **by** d **do**
> > **for** $z \leftarrow d$ **to** $nz - d$ **by** d **do**
> > > **if** isGeometry(x, y, z) is false **then**
> > > > **if** $nx/2$ is even **then**
> > > > > $\Phi_{x,y,z} \leftarrow (1 - \omega) \cdot \Phi_{x,y,z} +$
> > > > > $\qquad \frac{\omega}{6} \cdot (\Phi_{x-d,y,z} + \Phi_{x,y,z} +$
> > > > > $\qquad\qquad \Phi_{x,y-d,z} + \Phi_{x,y+d,z} +$
> > > > > $\qquad\qquad \Phi_{x,y,z-d} + \Phi_{x,y,z+d})$
> > > >
> > > > **else**
> > > > > $\Phi_{x,y,z} \leftarrow (1 - \omega) \cdot \Phi_{x,y,z} +$
> > > > > $\qquad \frac{\omega}{6} \cdot (2 \cdot \Phi_{x-d,y,z} +$
> > > > > $\qquad\qquad \Phi_{x,y-d,z} + \Phi_{x,y+d,z} +$
> > > > > $\qquad\qquad \Phi_{x,y,z-d} + \Phi_{x,y,z+d})$

Function: IterateBoundaryAntiSymmetricX$(\Phi, nx, ny, nz, d, \omega)$

> $l_x \leftarrow \text{length}_x(\Phi)$
> $\delta_x \leftarrow (l_x - 1) \bmod d$
> $x \leftarrow l_x - 1 - \delta_x$
>
> **if** $nx/2$ is even **then**
> > $f \leftarrow (1 + 2 \cdot \delta_x)/(1 + 2 \cdot (d + \delta_x))$
>
> **else**
> > $f \leftarrow \delta_x/(d + \delta_x)$
>
> **if** $f \neq 0$ **then**
> > **for** $y \leftarrow d$ **to** $ny - d$ **by** d **do**
> > > **for** $z \leftarrow d$ **to** $nz - d$ **by** d **do**
> > > > **if** isGeometry(x, y, z) is false **then**
> > > > > $\Phi_{x,y,z} \leftarrow f \cdot \Phi_{x-d,y,z}$

Algorithm 6: Prolongation of values from a coarse grid to the next finer grid.

Input: The potential array: Φ
        The multigrid preconditioning level: d (2,4,...)

Function: PROLONGATEBOUNDARYSYMMETRICX(Φ, d)

   $e \leftarrow d/2$
   for $x \leftarrow e$ **to** $\text{length}_x(\Phi) - e$ **by** e **do**
      for $y \leftarrow e$ **to** $\text{length}_y(\Phi) - e$ **by** e **do**
         for $z \leftarrow e$ **to** $\text{length}_z(\Phi) - e$ **by** e **do**
            if ISGEOMETRY(x, y, z) is false **then**
               if $x \% d = 0$ **and** $y \% d = 0$ **and** $z \% d \neq 0$ **then**
                 | $\Phi_{x,y,z} \leftarrow 1/2 \cdot (\Phi_{x,y,z-e} + \Phi_{x,y,z+e})$
               else if $x \% d = 0$ **and** $y \% d \neq 0$ **and** $z \% d = 0$ **then**
                 | $\Phi_{x,y,z} \leftarrow 1/2 \cdot (\Phi_{x,y-e,z} + \Phi_{x,y+e,z})$
               else if $x \% d \neq 0$ **and** $y \% d = 0$ **and** $z \% d = 0$ **then**
                  $\Phi_{x,y,z} \leftarrow \Phi_{x-e,y,z}$

               else if $x \% d = 0$ **and** $y \% d \neq 0$ **and** $z \% d \neq 0$ **then**
                  $\Phi_{x,y,z} \leftarrow 1/4 \cdot (\Phi_{x,y-e,z-e} + \Phi_{x,y-e,z+e} +$
                              $\Phi_{x,y+e,z-e} + \Phi_{x,y+e,z+e})$
               else if $x \% d \neq 0$ **and** $y \% d = 0$ **and** $z \% d \neq 0$ **then**
                 | $\Phi_{x,y,z} \leftarrow 1/2 \cdot (\Phi_{x-e,y,z-e} + \Phi_{x-e,y,z+e})$
               else if $x \% d \neq 0$ **and** $y \% d \neq 0$ **and** $z \% d = 0$ **then**
                 | $\Phi_{x,y,z} \leftarrow 1/2 \cdot (\Phi_{x-e,y-e,z} + \Phi_{x-e,y+e,z})$

               else if $x \% d \neq 0$ **and** $y \% d \neq 0$ **and** $z \% d \neq 0$ **then**
                  $\Phi_{x,y,z} \leftarrow 1/4 \cdot (\Phi_{x-e,y-e,z-e} + \Phi_{x-e,y-e,z+e} +$
                              $\Phi_{x-e,y+e,z-e} + \Phi_{x-e,y+e,z+e})$

Algorithm 7: Retrieving the value at a grid point of a potential that is symmetric or antisymmetric in x-direction

---

Input: The potential array: Φ
The size of the domain: nx, ny, nz
The indices of the desired potential value: x, y, z

Ensure: $0 \leq x < nx$
$0 \leq y < ny$
$0 \leq z < nz$

Function: $\text{GETPOTENTIALSYMMETRICX}(\Phi, nx, ny, nz, x, y, z)$

$\quad$ **if** $x \geq \text{length}_x(\Phi)$ **then**
$\qquad$ **return** $\Phi_{nx-1-x,y,z}$
$\quad$ **else**
$\qquad$ **return** $\Phi_{x,y,z}$

Function: $\text{GETPOTENTIALANTISYMMETRICX}(\Phi, nx, ny, nz, x, y, z)$

$\quad$ **if** $x \geq \text{length}_x(\Phi)$ **then**
$\qquad$ **return** $-\Phi_{nx-1-x,y,z}$
$\quad$ **else**
$\qquad$ **return** $\Phi_{x,y,z}$

E.3   Image based sample generation

Algorithm 8: Accumulation of pixel brightnesses into an array.

Input: The source image: I
  The width of the source image: w
  The height of the source image: h

Result: An array with the normalized cumulated brightness values of
  the source image: C

Function: COMPUTECUMULATIVE(I, w, h)

  $s \leftarrow 0$
  maxBrightness $\leftarrow 2^{I.\text{GETBITSPERCHANNEL}()} - 1$ `// 255 for 8 bit`
  for $y \leftarrow 0$ **to** h **do**
    for $x \leftarrow 0$ **to** w **do**
      $s \leftarrow s + I.\text{GETPIXEL}(x, y).\text{GETBRIGHTNESS}()$ / maxBrightness
      $C[y \cdot w + x] \leftarrow s$

  for $i \leftarrow 0$ **to** $w \cdot h$ **do**
    $C[i] \leftarrow C[i]/s$ `// Normalization to 1`
  return C

Algorithm 9: Generation of random samples from a cumulated array.

Input: An array of cumulated, normalized brightness values: C
  The width of the source image: w
  RANDOM(): A random floating point value between 0 and 1
  SEARCH(ARRAY,KEY): Returns index of the key, otherwise
  (-(insertion point) -1).

Result: A random position within a pixel

Function: GETSAMPLE(C, w)
  $i \leftarrow$ SEARCH$(C, \text{RANDOM}())$

  if $i < 0$ **then**
    $i \leftarrow -i - 1$
  return $(i \% w + \text{RANDOM}(), \lfloor i/w \rfloor + \text{RANDOM}())$

E.4   Octree

Algorithm 10: Potential interpolation in an octree.

Input: An array with the eight surrounding cells: C
The location of the interpolation relative to the central vertex of
these cells: x, y, z
The root cell grid spacing: r

Result: An interpolated potential value at (x,y,z): Φ

Function: INTERPOLATE(x, y, z, C)

$\quad m \leftarrow \text{level}(C_0)$ `// Find lowest level`
$\quad$**for** $i \leftarrow 1$ **to** 7 **do**
$\quad\quad$**if** $\text{level}(C_i) < m$ **then**
$\quad\quad\quad m \leftarrow \text{level}(C_i)$

$\quad t \leftarrow 0$ `// Calculate type(binary) with bit operations`
$\quad$**for** $i \leftarrow 0$ **to** 7 **do**
$\quad\quad t \leftarrow t \mid \big((\text{level}(C_i) - m) \ll (7 - i)\big)$

$\quad l \leftarrow r/(2 \ll m)$ `// Normalize coordinates`
$\quad x \leftarrow 0.5 \cdot (x/l + 1)$ `//  `$x, y, z \in [0, 1)$
$\quad y \leftarrow 0.5 \cdot (y/l + 1)$
$\quad z \leftarrow 0.5 \cdot (z/l + 1)$

$\quad$**switch** t **do**
$\quad\quad$**case** 00000000 **do**   `// default trilinear interpolation`
$\quad\quad\quad \Phi \leftarrow$ INTERPOLATE$(x, y, z, a, b, c, d, e, f, g, h, C)$
$\quad\quad$**case** 00000001 **do**   `// `$Rot_x^{180°}$
$\quad\quad\quad \Phi \leftarrow$ INTERPOLATEA$(x, 1 - y, 1 - z, h, g, f, e, d, c, b, a, C)$
$\quad\quad\quad\vdots$
$\quad\quad$**case** 00100000 **do**   `// Figure 6.7, `$Rot_z^{180°}$
$\quad\quad\quad \Phi \leftarrow$ INTERPOLATEA$(1 - x, 1 - y, z, c, d, a, b, g, h, e, f, C)$
$\quad\quad\quad\vdots$
$\quad\quad$**case** 01010101 **do**   `// `$Rot_z^{-90°}$
$\quad\quad\quad \Phi \leftarrow$ INTERPOLATEACEG$(y, 1 - x, z, b, c, d, a, f, g, h, e, C)$
$\quad\quad\quad\vdots$
$\quad\quad$**case** 11111110 **do**   `// No transform`
$\quad\quad\quad \Phi \leftarrow$ INTERPOLATEABCDEFG$(x, y, z, a, b, c, d, e, f, g, h, C)$

$\quad$**return** Φ

Function: INTERPOLATEX$(x, y, z, a, b, c, d, e, f, g, h, C)$

$\quad$`// Finds the cuboid (as illustrated in Figure 6.7) in`
$\quad\quad$`which the point (x,y,z) is located and performs the`
$\quad\quad$`trilinear interpolation. Only the default case has to`
$\quad\quad$`be implemented as the transform is already given via`
$\quad\quad$`the method parameters (a-h are array indices for C).`

Algorithm 11: Adaptive 7-point stencil method for retrieving an electric field vector within an octree.

Input: The location of the field vector evaluation: x, y, z
The cell at (x, y, z): C
The normalized values of x, y, z in C: $u, v, w \in [0, 1]$
Half the size of C: d

Result: The electric field vector at (x,y,z): E

Function: GETFIELDVECTOR(x, y, z, u, v, w, C)

> $d_x^+ \leftarrow 1$
> $d_x^- \leftarrow 1$
>
> **if** $u > 0.5$ **then**
>> $C_x^+ \leftarrow$ GETNEIGHBORINGCELL$_x^+(C)$
>> **if** level$(C_x^+) >$ level(C) **then**
>>> $d_x^+ \leftarrow 1.5 - x$
>>
>> **else if** level$(C_x^+) <$ level(C) **then**
>>> $d_x^+ \leftarrow 2 \cdot x$
>
> **else**
>> $C_x^- \leftarrow$ GETNEIGHBORINGCELL$_x^-(C)$
>> **if** level$(C_x^-) >$ level(C) **then**
>>> $d_x^- \leftarrow x + 0.5$
>>
>> **else if** level$(C_x^-) <$ level(C) **then**
>>> $d_x^- \leftarrow 2 - 2 \cdot x$
>
> $E_x \leftarrow \big($INTERPOLATE$(x - d_x^- \cdot d, y, z)-$
> $\qquad$ INTERPOLATE$(x + d_x^+ \cdot d, y, z)\big)/\big((d_x^+ + d_x^-) \cdot d\big)$
> // E_y and E_z are calculated equivalently.
> **return** E

F Loss channels for all 2D parameter scans

This section provides an overview of the different loss channels for all possible parameter scan combinations. The color indicates the percentage of launched positrons that are lost at a specific location named in the diagram title. The values of the non-scanned parameters are given in Appendix C.

F.1 I_r - I_θ
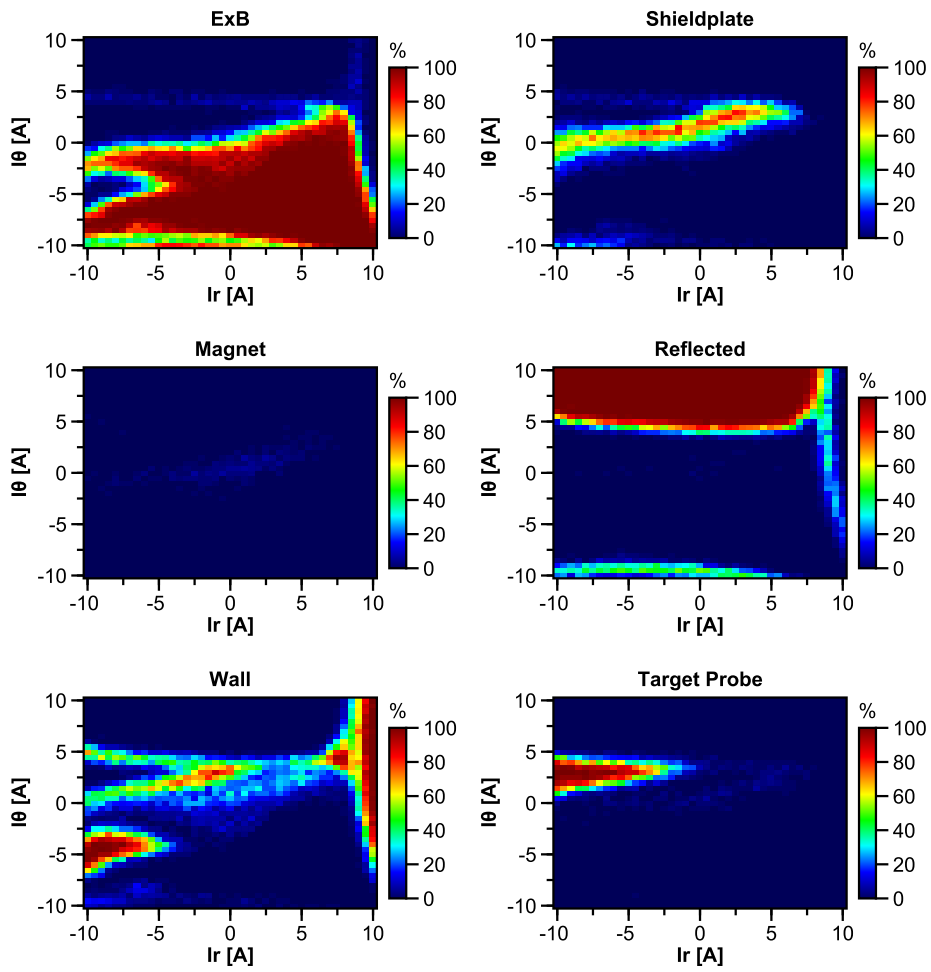
Figure F.1: Steering coil curent I_r vs steering coil curent I_θ.

F.2 I_r - RW1
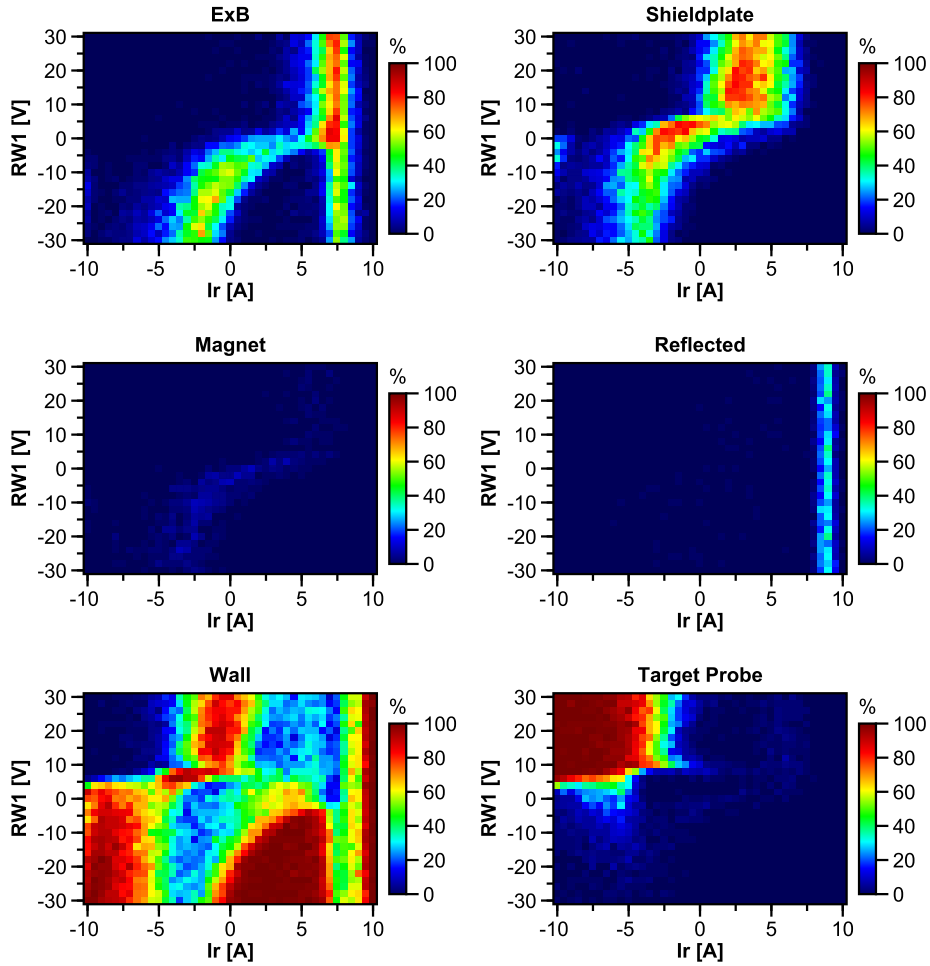
Figure F.2: Steering coil curent I_r vs RW1 bias.

F.3 I_r - Top1
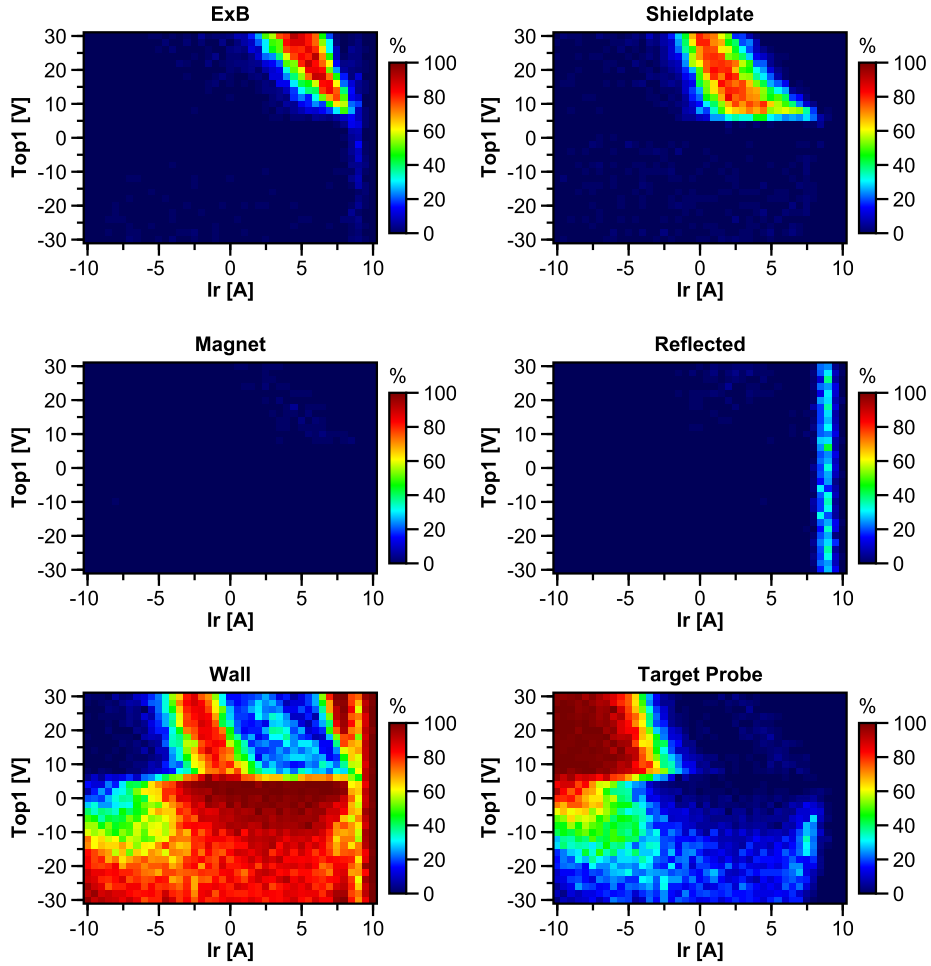
Figure F.3: Steering coil curent I_r vs Top1 bias.

F.4 I_θ - RW1
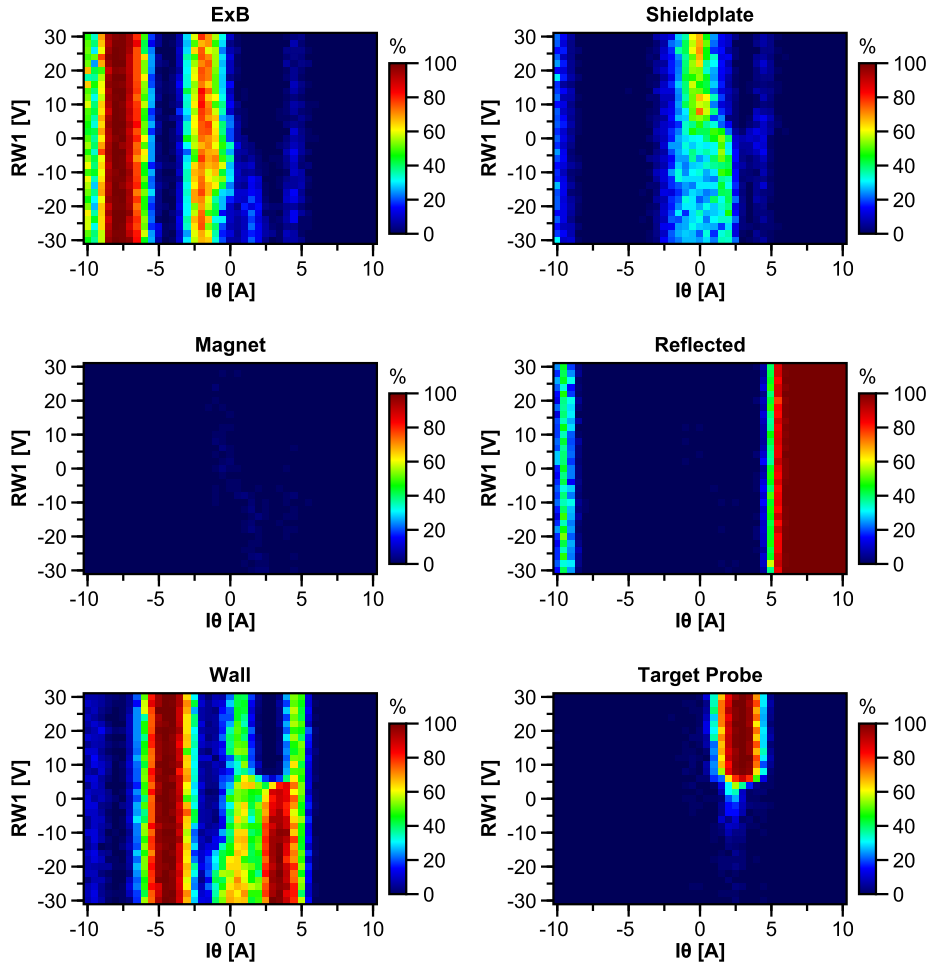
Figure F.4: Steering coil curent I_θ vs RW1 bias.

F.5 I_θ - Top1
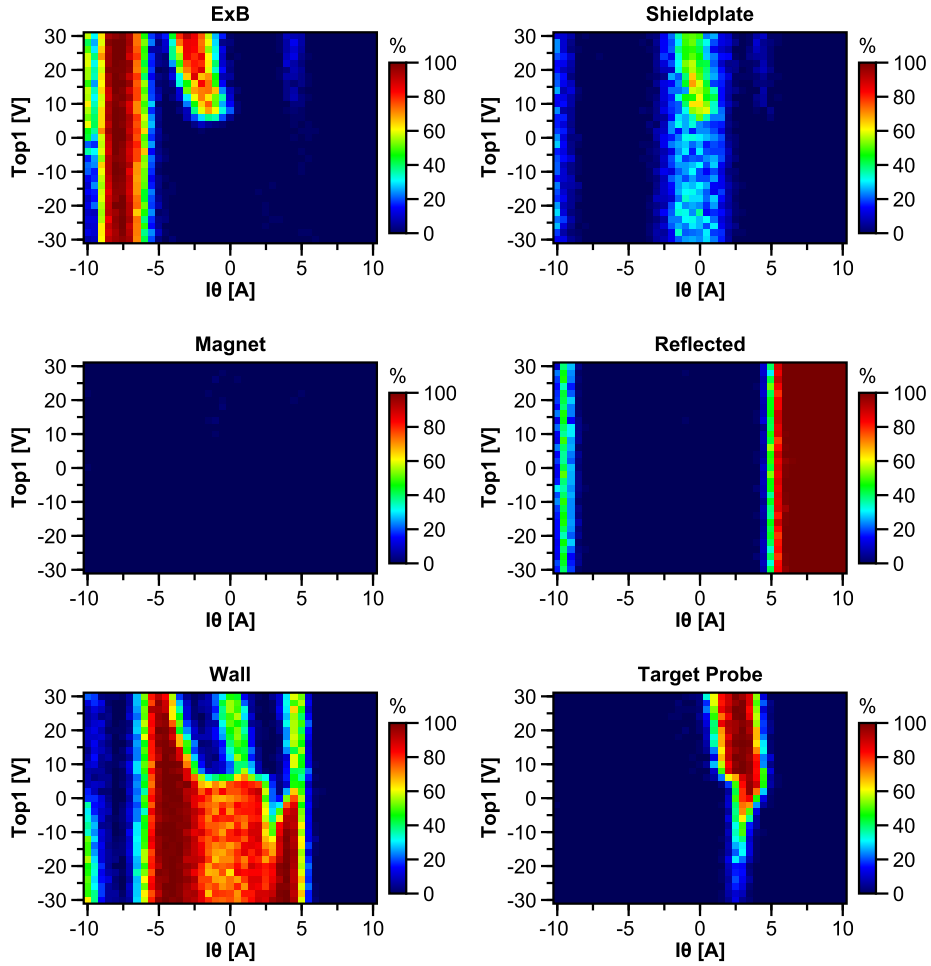
Figure F.5: Steering coil curent I_θ vs Top1 bias.

F.6 E×B - I_r
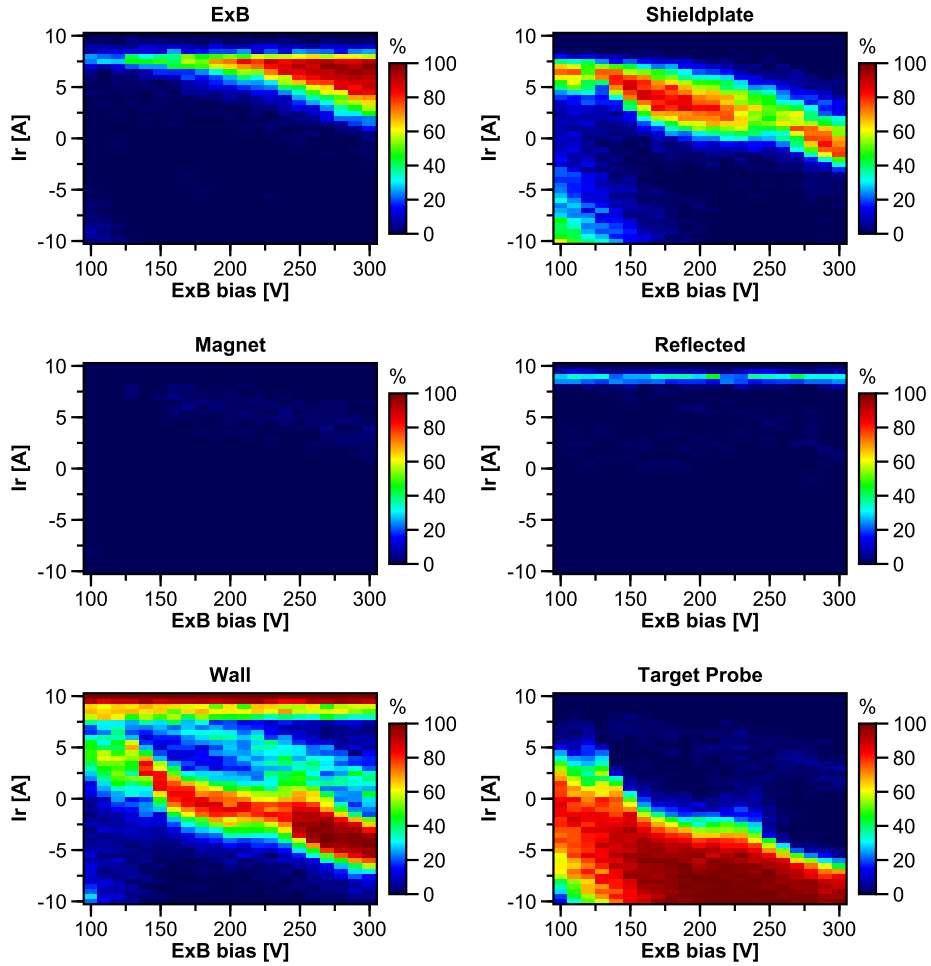
Figure F.6: Bias of the E×B plates vs steering coil curent I_r.

F.7   E×B - I_θ
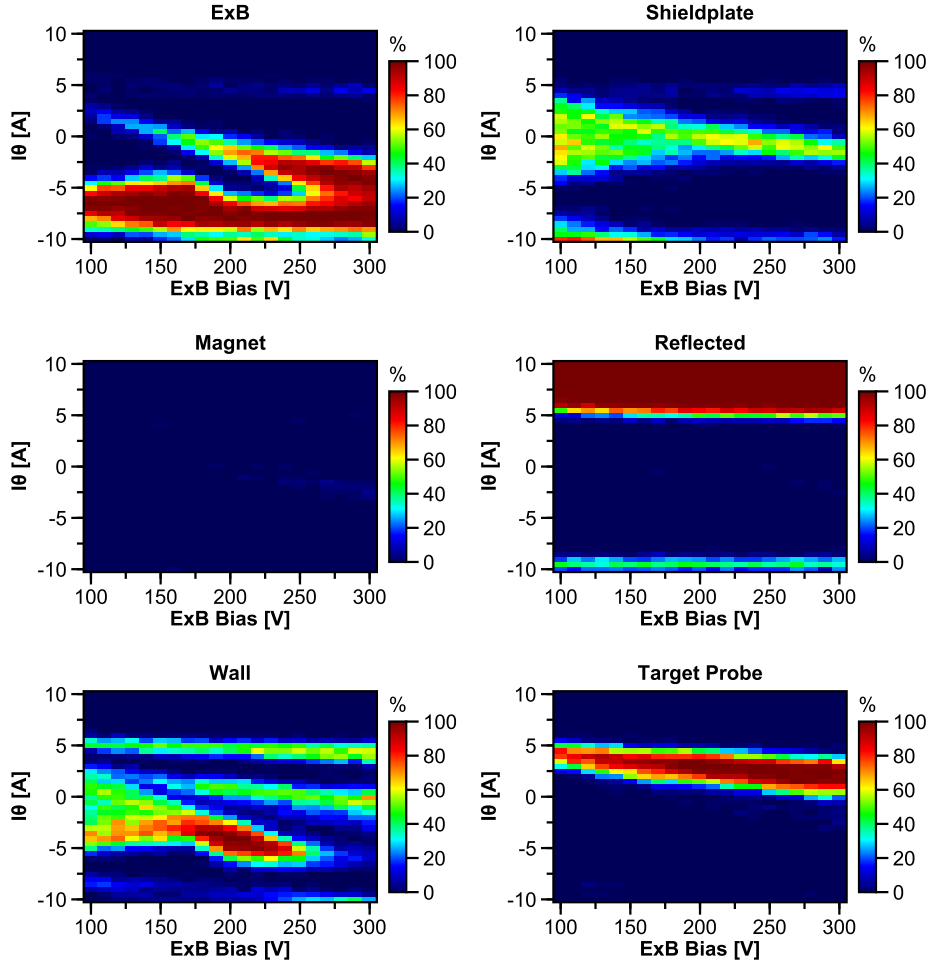
Figure F.7: Bias of the E×B plates vs steering coil curent I_θ.

F.8   E×B - RW1
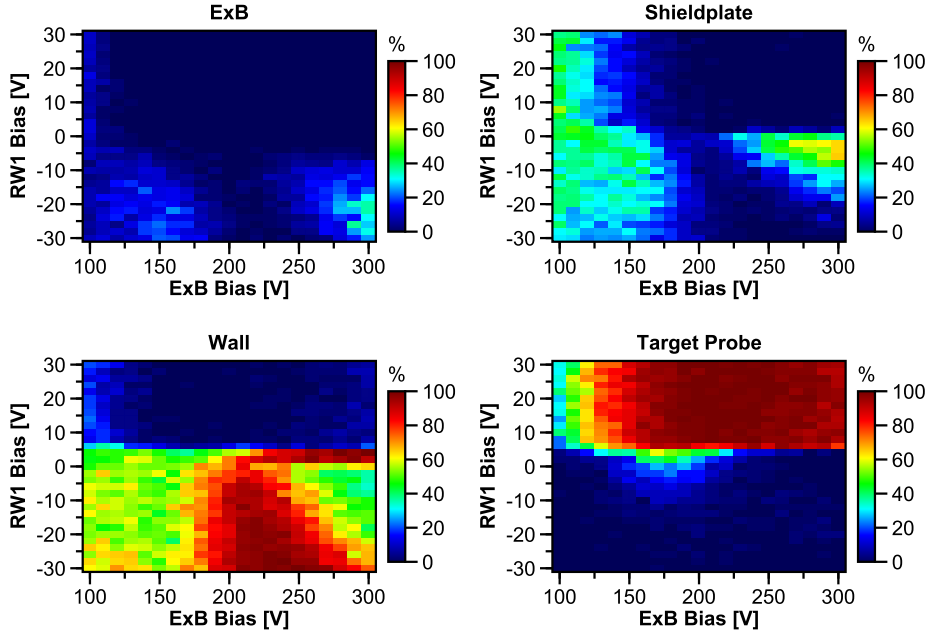
Figure F.8: Bias of the E×B plates vs RW1 bias.

F.9   E×B - Top1
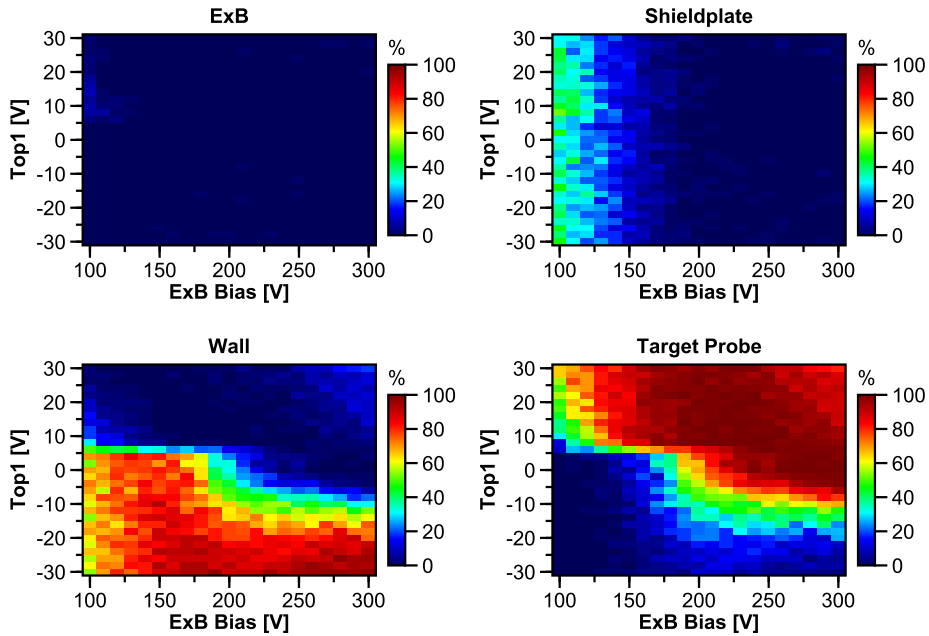
Figure F.9: Bias of the E×B plates vs Top1 bias.

F.10 Magnet case - I_r
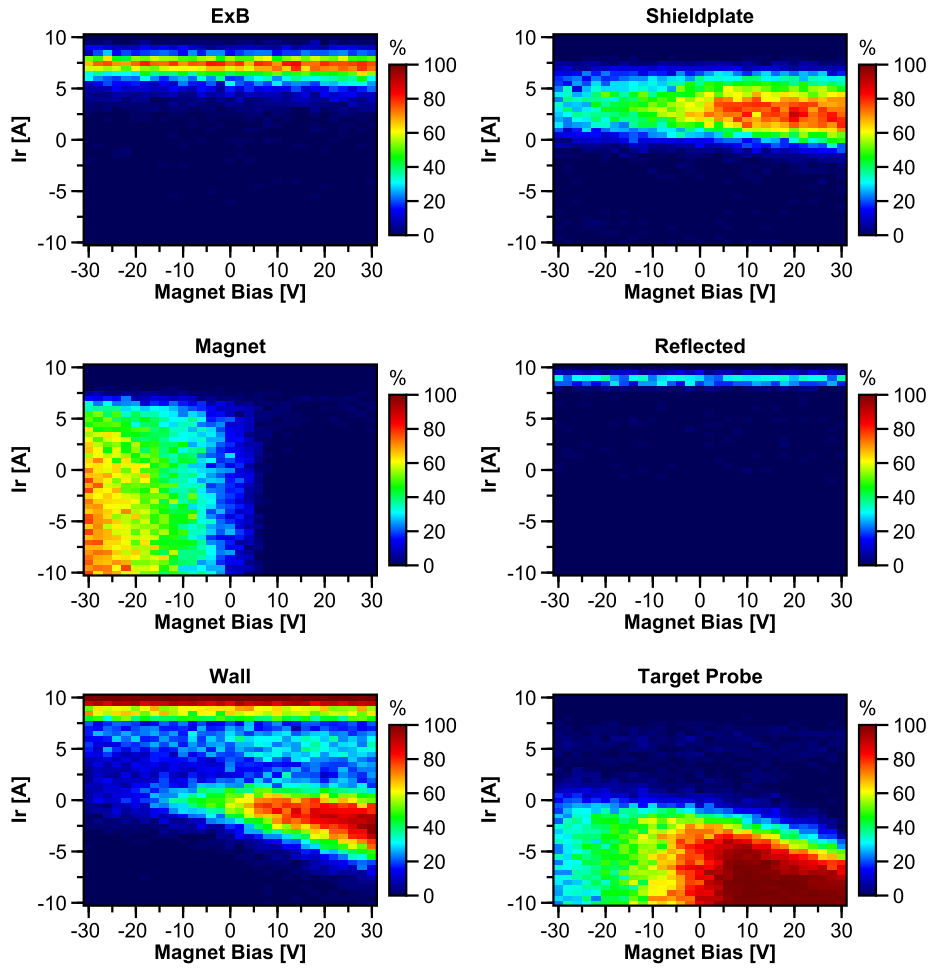
Figure F.10: Magnet case bias vs steering coil curent I_r.

F.11 Magnet case - I_θ
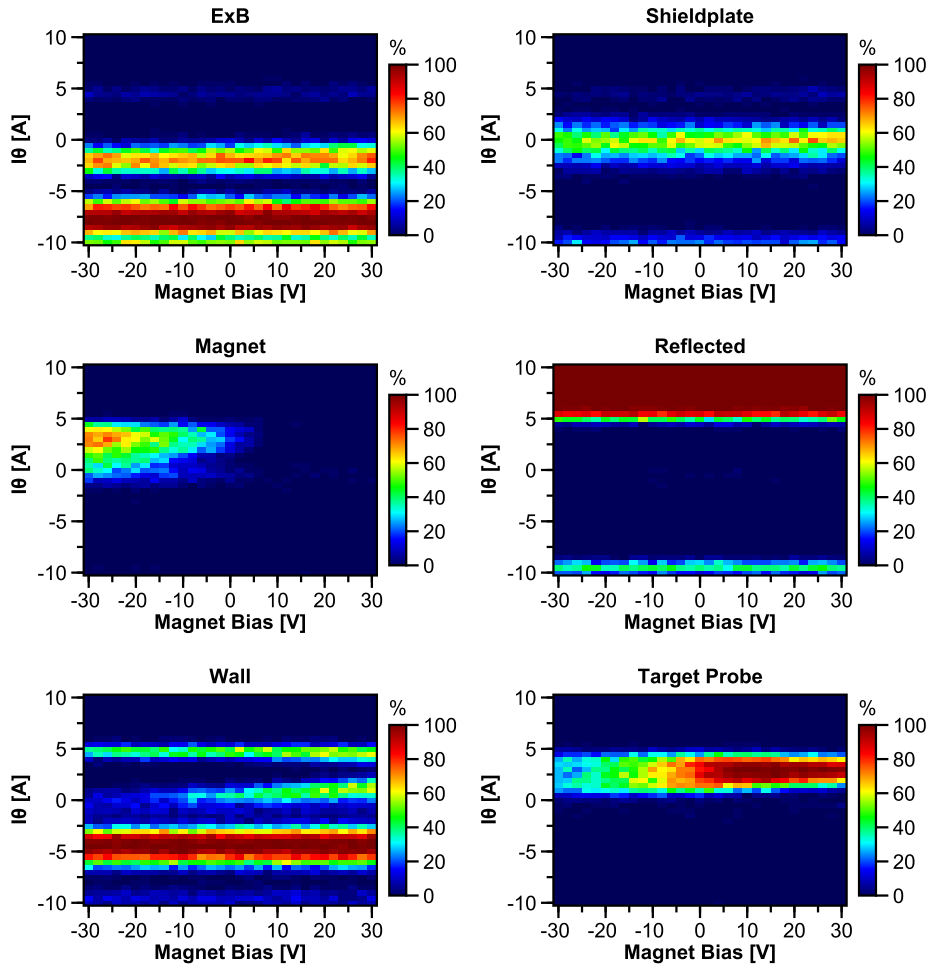
Figure F.11: Magnet case bias vs steering coil curent I_θ.

F.12   Magnet case - RW1
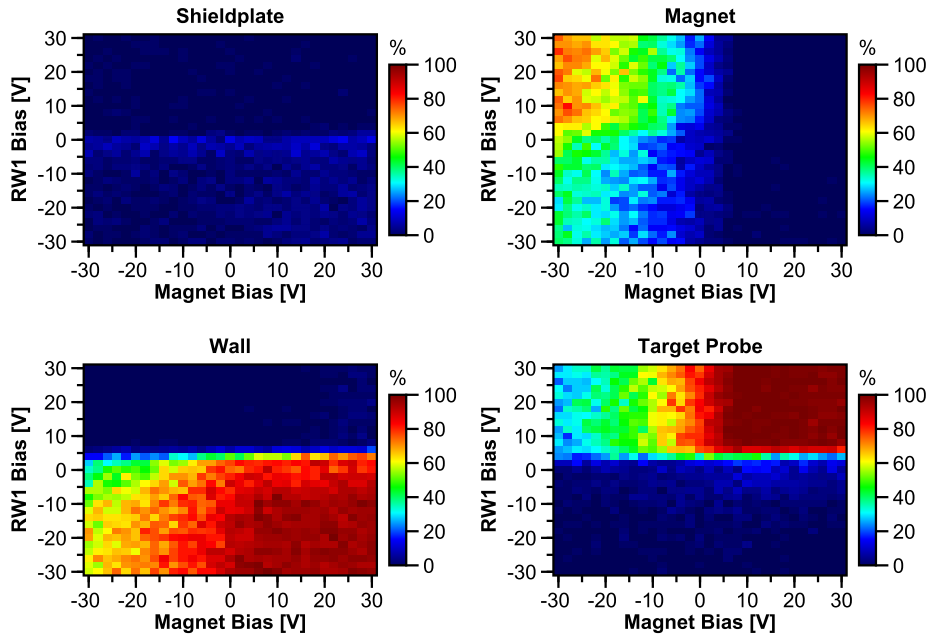
Figure F.12: Magnet case bias vs RW1 bias.

F.13   Magnet case - Top1
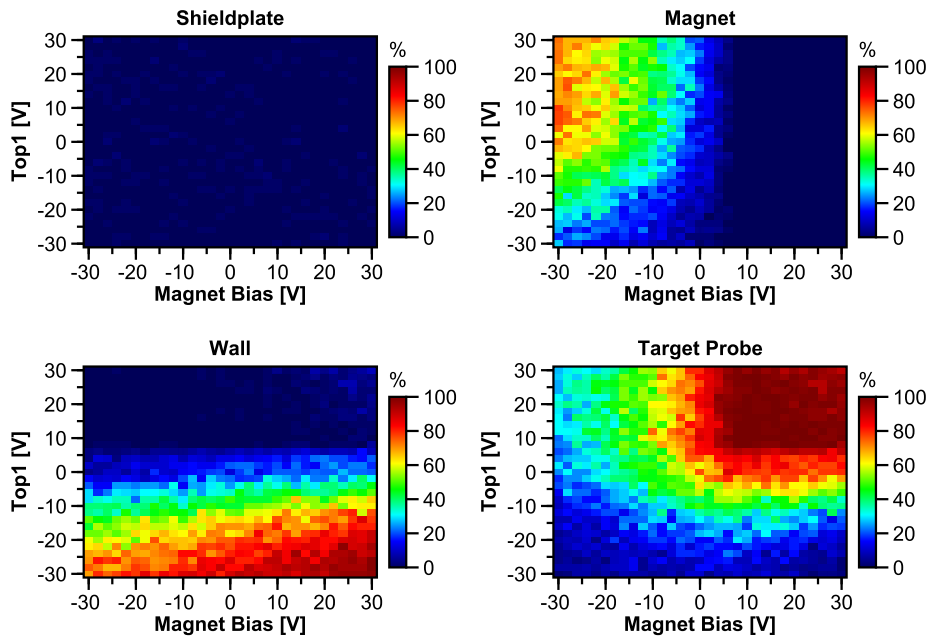
Figure F.13: Magnet case bias vs Top1 bias.
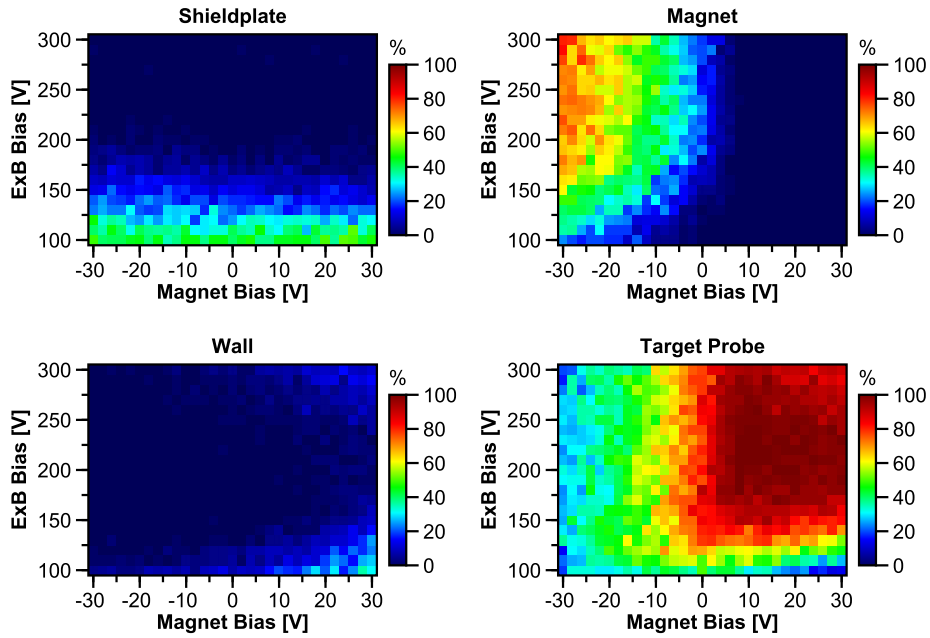
F.14 Magnet case - E×B

Figure F.14: Magnet case bias vs bias of the E×B plates.
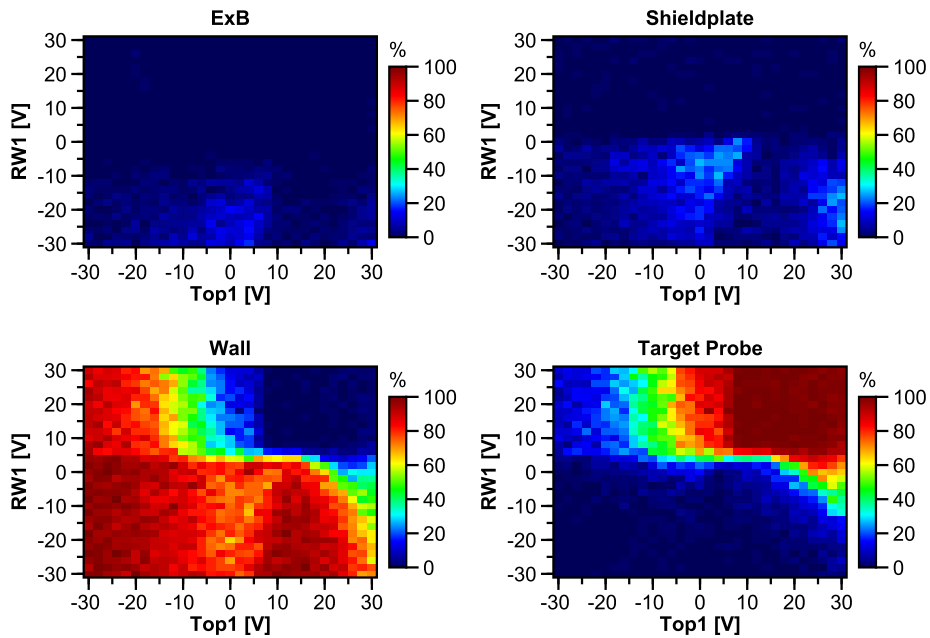
F.15 Top1 - RW1

Figure F.15: Top1 bias vs RW1 bias.
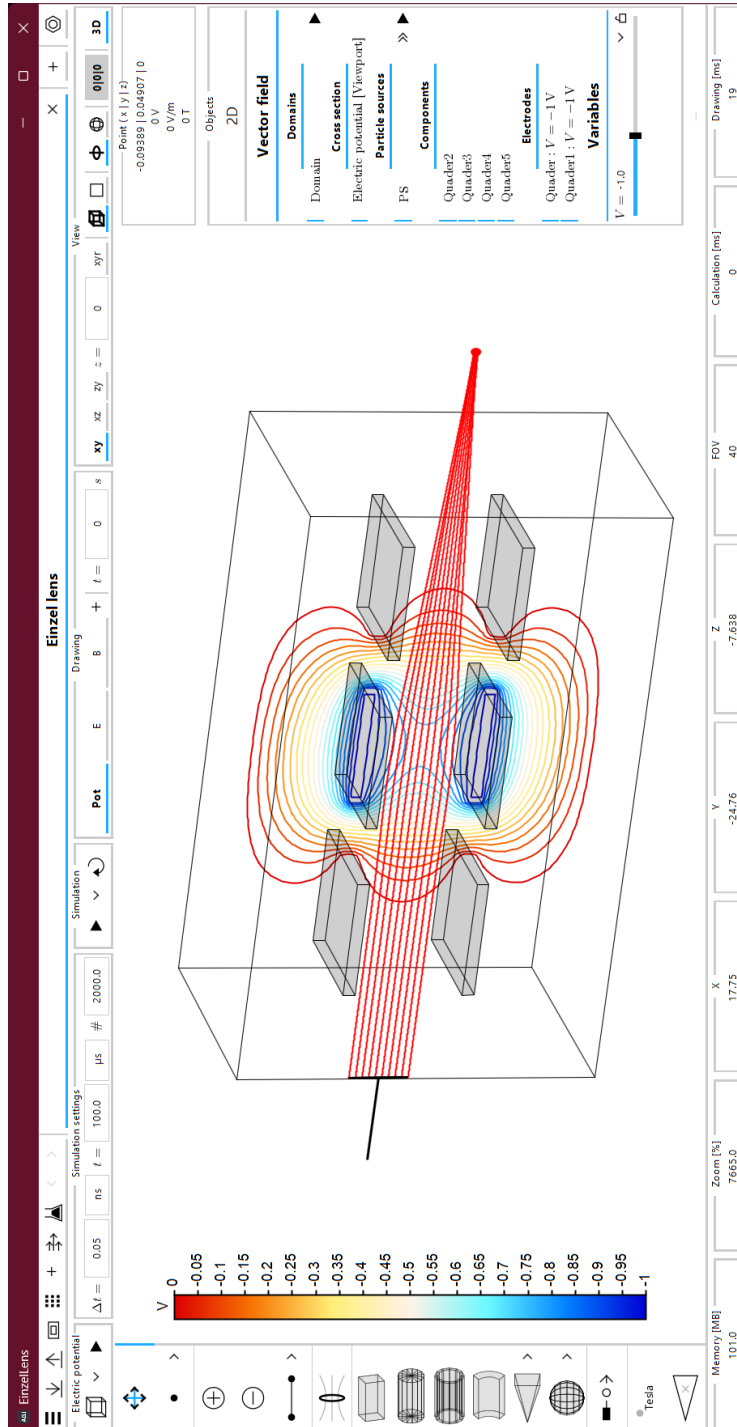
G   User interface of AlGeoJ

Figure G.1: Main user interface of AlGeoJ. On the left is the toolbar. After selection, the object (coil, electrode, particle source,...) can be placed directly in the central viewport. On the right is the list of all created objects and variables. The first row at the top contains the general menu bar and the tab list. The second row is tab-specific and in this case contains simulation settings and view options. The bottom row shows the current zoom factor, among other statistics.
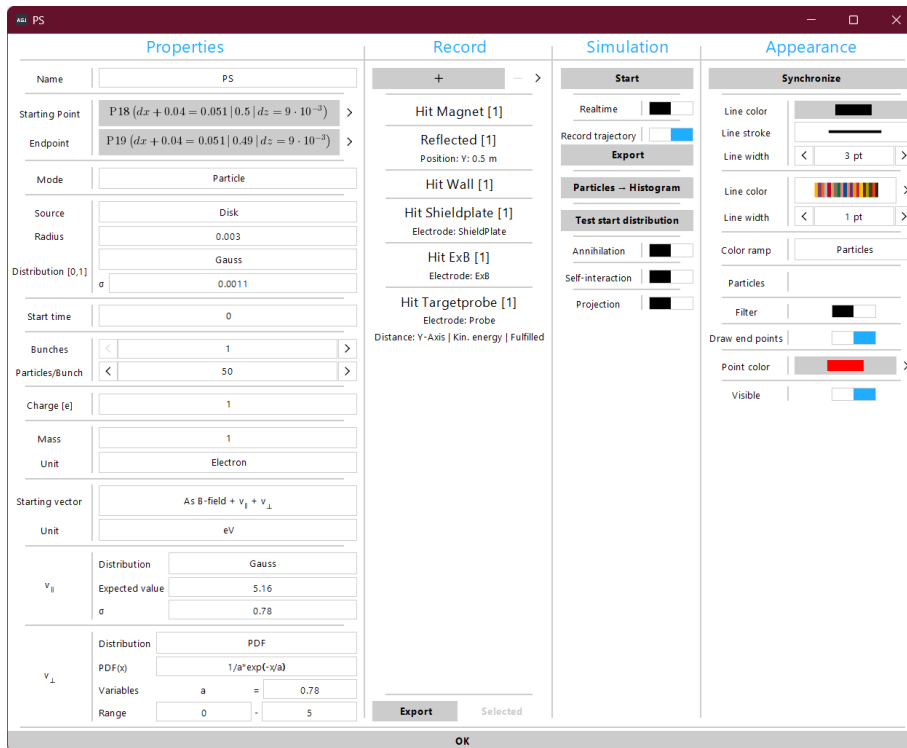
Figure G.2: Properties dialog of a particle source. The left column contains all the settings to define the initial properties of the particles to be launched. This includes the type of particle (particle, field line, photon), the spatial distribution (point, line, disk, image (see Section 5.6),...), the start time distribution, the number of particles, the charge distribtion, the mass distribution, and the starting vector and velocity/energy distributions. Distributions can be specified in a variety of ways, including a single value, a range of random values, Gaussian, or as a probability/cumulative density function. The second column labeled "Record" contains the list of conditions when a particle should record data that can later be used to create plots (the dialog is shown in Figure G.3). The "'Simulation'" column contains the option to enable the drawing of the trajectories, the possibility to plot the recorded data as well as an option to enable particle annihilation. The right column contains all the settings for the appearance of the particles and their trajectories. Setting the colors to a color ramp automatically enables a variety of ways to directly visualize the recorded data in the viewport.
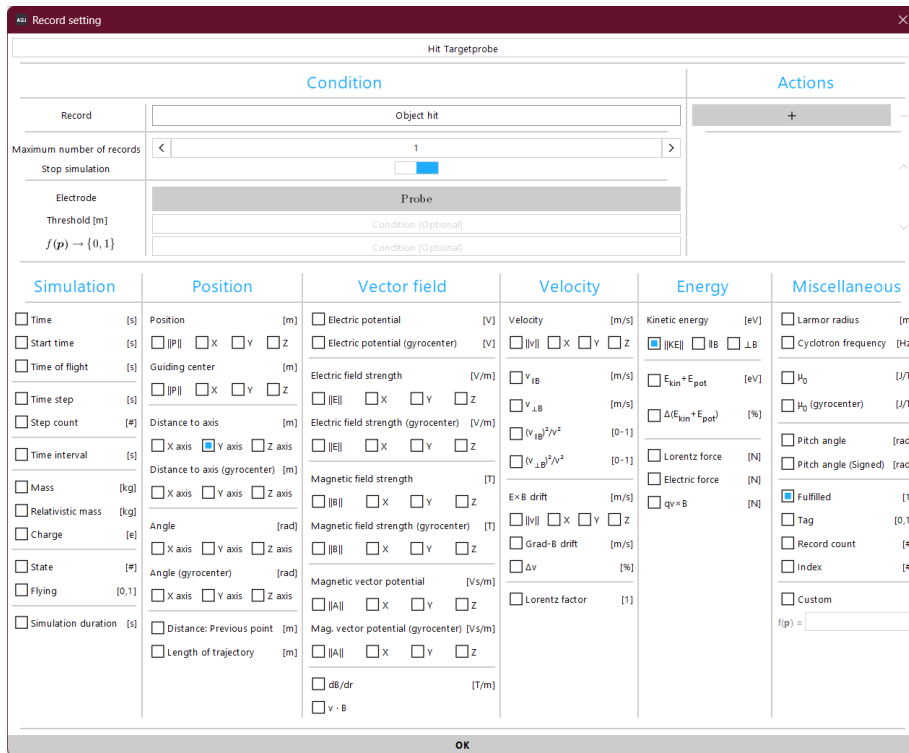
Figure G.3: Dialog of a record setting. First, the type of condition (time, plane intersection, object hit,...) is specified. Then, the maximum number of times this condition can be fulfilled (0=infinite) and whether the particle should be stopped after reaching the limit can be set. Actions can be defined that are executed when the condition is met, such as tagging the particle or randomizing its velocity vector. After setting mandatory values for the selected condition type, the condition can be refined, for example, by specifying that the condition can only be met if the particle was previously tagged. Finally, all parameters to be recorded can be selected.
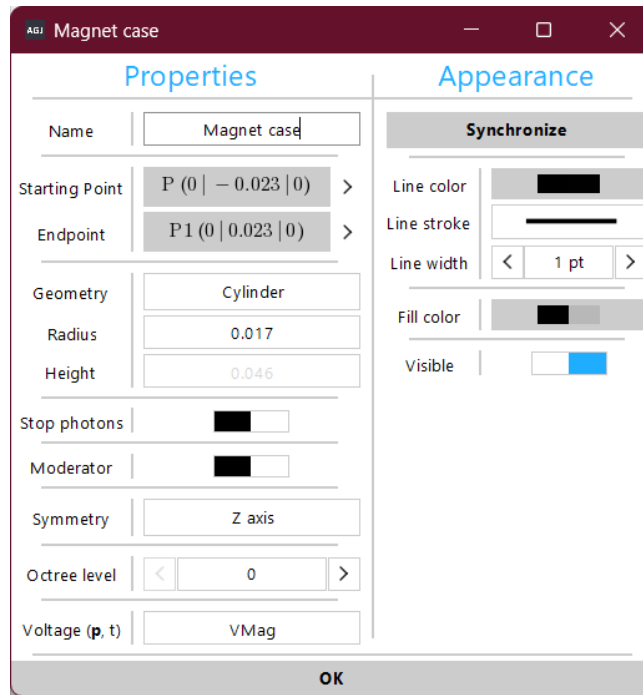
Figure G.4: Properties dialog of an electrode. Each electrode is defined by two points. After specifying the general type of the geometry (cuboid, cylinder, hollow cylinder, STL (see Section 5.5),...), further properties can be defined, such as the radius in this instance (the height is grayed out because it is already defined by the start and end points). Enabling "Stop photons" adds this electrode to the list of objects, that is checked for intersections if particle annihilation is enabled (see Figure G.2). When "Moderation" is activated, further options are shown (moderation efficiency, energy distribution after moderation, and emission angle) to allow this object to act as a moderator (an "Object hit" record setting has to be added for this object, like shown in Figure G.3, and "Moderation" has to be added as action). Setting an appropriate symmetry with respect to the potential domain speeds up the computation of its electric potential, as discussed in Section 5.3.5. Furthermore, the maximum octree refinement level can be set, see Chapter 6. Finally, the applied voltage can be defined. This voltage can also be time-dependent and can even depend on any property of a particle. If no voltage is set, the electrode is assumed to be grounded.

References

[1] D. A. Gurnett and A. Bhattacharjee, *Introduction to plasma physics: with space and laboratory applications.* Cambridge university press, Mar 2017.
Doi: 10.1017/9781139226059.002

[2] J. F. C. Wardle, D. C. Homan, R. Ojha, and D. H. Roberts, *Electron-positron jets associated with the quasar 3C279*, Nature, 395(6701):457–461, Oct 1998.
Doi: 10.1038/26675

[3] K. Hirotani, S. Iguchi, M. Kimura, and K. Wajima, *Pair plasma dominance in the parsec-scale relativistic jet of 3C 345*, The Astrophysical Journal, 545(1):100, Dec 2000.
Doi: 10.1086/317769

[4] D. Homan, M. Lister, H. Aller, M. Aller, and J. Wardle, *Full polarization spectra of 3C 279*, The Astrophysical Journal, 696(1):328, Apr 2009.
Doi: 10.1088/0004-637X/696/1/328

[5] P. Helander, *Microstability of magnetically confined electron-positron plasmas*, Phys. Rev. Lett., 113:135003, Sep 2014.
Doi: 10.1103/PhysRevLett.113.135003

[6] P. Helander and J. Connor, *Gyrokinetic stability theory of electron–positron plasmas*, Journal of Plasma Physics, 82(3), May 2016.
Doi: 10.1017/S0022377816000490

[7] E. V. Stenson, *Plans for EPOS: A Tabletopsized, Superconducting, Optimized Stellarator For Matter/Antimatter Pair Plasmas*, Stellarator News, 167, Oct 2019.
URL: https://stelnews.info//sites/default/files/pdf/sn167.pdf

[8] E. V. Stenson, J. Horn-Stanja, M. R. Stoneking, and T. S. Pedersen, *Debye length and plasma skin depth: two length scales of interest in the creation and diagnosis of laboratory pair plasmas*, J. Plasma Phys., 83(1), Feb 2017.
Doi: 10.1017/S0022377817000022

[9] T. S. Pedersen, J. R. Danielson, C. Hugenschmidt, G. Marx, X. Sarasola, F. Schauer, L. Schweikhard, C. M. Surko, and E. Winkler, *Plans for the creation and studies of electron-positron plasmas in a stellarator*, New J. Phys., 14(3):035010, Mar 2012.
Doi: 10.1088/1367-2630/14/3/035010

[10] M. Stoneking, T. S. Pedersen, P. Helander, H. Chen, U. Hergenhahn, E. Stenson, G. Fiksel, J. von der Linden, H. Saitoh, C. Surko, et al., *A new frontier in laboratory physics: magnetized electron–positron plasmas*, Journal of Plasma Physics, 86(6), Nov 2020.
Doi: 10.1017/S0022377820001385

[11] S. Mukherjee, P. Panchal, J. S. Mishra, R. Gangradey, P. Nayak, and V. Gupta, *Hydrogen outgassing and permeation in stainless steel and its reduction for UHV applications*, Materials Today: Proceedings, 44:968–974, Apr 2021.
Doi: 10.1016/j.matpr.2020.11.007

[12] E. V. Stenson, S. Nißl, U. Hergenhahn, J. Horn-Stanja, M. Singer, H. Saitoh, T. S. Pedersen, J. R. Danielson, M. R. Stoneking, and M. Dickmann, *Lossless positron injection into a magnetic dipole trap*, Phys. Rev. Lett., 121(23):235005, Dec 2018.
Doi: 10.1103/PhysRevLett.121.235005

[13] J. Horn-Stanja, S. Nißl, U. Hergenhahn, T. S. Pedersen, H. Saitoh, E. V. Stenson, M. Dickmann, C. Hugenschmidt, M. Singer, and M. R. Stoneking, *Confinement of positrons exceeding 1 s in a supported magnetic dipole trap*, Phys. Rev. Lett., 121(23):235003, Dec 2018.
Doi: 10.1103/PhysRevLett.121.235003

[14] M. Singer, M. Stoneking, E. Stenson, S. Nißl, A. Deller, A. Card, J. Horn-Stanja, T. Sunn Pedersen, H. Saitoh, and C. Hugenschmidt, *Injection of positrons into a dense electron cloud in a magnetic dipole trap*, Physics of Plasmas, 28(6):062506, Jun 2021.
Doi: 10.1063/5.0050881

[15] M. Singer, *Towards the Creation of a Pair Plasma–Simultaneous Injection of the NEPOMUC Positron Beam and an Electron Beam into the APEX Dipole Trap*, PhD thesis, Technische Universität München, Sep 2021.

[16] S. Nißl, *Numerical Investigations into Injection and Confinement of Single Particles in a Magnetic Dipole Trap*, Master's thesis, Mar 2018.

[17] J. R. Danielson, D. H. E. Dubin, R. G. Greaves, and C. M. Surko, *Plasma and trap-based techniques for science with positrons*, Rev. Mod. Phys., 87:247–306, Mar 2015.
Doi: 10.1103/RevModPhys.87.247

[18] A. P. Mills Jr and E. Gullikson, *Solid neon moderator for producing slow positrons*, Applied Physics Letters, 49(17):1121–1123, 1986.
Doi: 10.1063/1.97441

[19] R. Ley, *Positron production using accelerators*, Hyperfine Interactions, 109(1):167–180, Jan 1997.
Doi: 10.1023/A:1012605500940

[20] B. E. O'Rourke, N. Oshima, A. Kinomura, T. Ohdaira, and R. Suzuki, *Recent developments and future plans for the accelerator based slow positron facilities at AIST*, In *Materials Science Forum*, volume 733, pages 285–290. Trans Tech Publ, Jan 2013.
Doi: 10.4028/www.scientific.net/MSF.733.285

[21] H. Schut, A. van Veen, C. Falub, J. de Roode, and F. Labohm, *Performance of an intense nuclear-reactor based positron beam*, In *Materials Science Forum*, volume 363, pages 430–432. Trans Tech Publications Ltd., Zurich-Uetikon, Switzerland, Apr 2001.
Doi: 10.4028/www.scientific.net/MSF.363-365.430

[22] C. Hugenschmidt, C. Piochacz, M. Reiner, and K. Schreckenbach, *The NEPOMUC upgrade and advanced positron beam experiments*, New Journal of Physics, 14(5):055027, May 2012.
Doi: 10.1088/1367-2630/14/5/055027

[23] B. Krusche and K. Schreckenbach, *Intense positron sources by pair creation with neutron capture γ-rays*, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 295(1-2):155–171, Oct 1990.
Doi: 10.1016/0168-9002(90)90435-9

[24] C. Hugenschmidt, G. Kögel, R. Repper, K. Schreckenbach, P. Sperr, and W. Triftshäuser, *First platinum moderated positron beam based on neutron capture*, Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, 198(3-4):220–229, Dec 2002.
Doi: 10.1016/S0168-583X(02)01527-6

[25] C. Hugenschmidt, G. Kögel, R. Repper, K. Schreckenbach, P. Sperr, B. Strasser, and W. Triftshäuser, *NEPOMUC-the new positron beam facility at FRM II*, In *Materials Science Forum*, volume 445, pages 480–482. Trans Tech Publ, Jan 2004.
Doi: 10.4028/www.scientific.net/MSF.445-446.480

[26] C. Hugenschmidt, H. Ceeh, T. Gigl, F. Lippert, C. Piochacz, P. Pikart, M. Reiner, J. Weber, and S. Zimnik, *The upgrade of the neutron induced positron source NEPOMUC*, In *Journal of Physics: Conference Series*, volume 443, page 012079. IOP Publishing, Aug 2013.
Doi: 10.1088/1742-6596/443/1/012079

[27] C. Hugenschmidt, B. Straßer, and K. Schreckenbach, *Investigation of positron work function and moderation efficiency of Ni, Ta, Pt and W(1 0 0)*, Applied Surface Science, 194(1):283–286, 9th International Workshop on Slow Positron Beam Techniques for Solids and Surfaces, Jun 2002.
Doi: 10.1016/S0169-4332(02)00135-6

[28] C. Piochacz, G. Kögel, W. Egger, C. Hugenschmidt, J. Mayer, K. Schreckenbach, P. Sperr, M. Stadlbauer, and G. Dollinger, *A positron remoderator for the high intensity positron source NEPOMUC*, Applied surface science, 255(1):98–100, Oct 2008.
Doi: 10.1016/j.apsusc.2008.05.286

[29] J. Stanja, U. Hergenhahn, H. Niemann, N. Paschkowski, T. S. Pedersen, H. Saitoh, E. V. Stenson, M. R. Stoneking, C. Hugenschmidt, and C. Piochacz, *Characterization of the NEPOMUC primary and remoderated positron beams at different energies*, Nucl. Instrum. Methods Phys. Res., Sect. A, 827:52–62, Aug 2016.
Doi: 10.1016/j.nima.2016.04.093

[30] F. Wysocki, M. Leventhal, A. Passner, and C. Surko, *Accumulation and storage of low energy positrons*, Hyperfine Interactions, 44:185–200, Mar 1989.
Doi: 10.1007/BF02398669

[31] J. Marler and C. Surko, *Positron-impact ionization, positronium formation, and electronic excitation cross sections for diatomic molecules*, Physical Review A, 72(6):062713, Dec 2005.
Doi: 10.1103/PhysRevA.72.062713

[32] J. Marler and C. Surko, *Systematic comparison of positron-and electron-impact excitation of the ν_3 vibrational mode of CF_4*, Physical Review A, 72(6):062702, Dec 2005.
Doi: 10.1103/PhysRevA.72.062702

[33] A. Deller, M. Stoneking, T. S. Pedersen, E. Stenson, J. Horn-Stanja, U. Hergenhahn, S. Nißl, A. Card, C. Hugenschmidt, M. Singer, et al., *A buffer-gas trap for the NEPOMUC high-intensity low-energy positron beam*, In *62nd Annual Meeting of the APS Division of Plasma Physics*, 2020

[34] X.-P. Huang, F. Anderegg, E. Hollmann, C. Driscoll, and T. O'neil, *Steady-state confinement of non-neutral plasmas by rotating electric fields*, Physical Review Letters, 78(5):875, Feb 1997.
Doi: 10.1103/PhysRevLett.78.875

[35] C. Surko and R. Greaves, *A multicell trap to confine large numbers of positrons*, Radiation Physics and Chemistry, 68(3-4):419–425, 2003.
Doi: 10.1016/S0969-806X(03)00194-4

[36] J. Danielson, T. Weber, and C. Surko, *Plasma manipulation techniques for positron storage in a multicell trap*, Physics of Plasmas, 13(12):123502, Dec 2006.
Doi: 10.1063/1.2390690

[37] M. Singer, S. König, M. Stoneking, P. Steinbrunner, J. Danielson, L. Schweikhard, and T. S. Pedersen, *Non-neutral plasma manipulation techniques in development of a high-capacity positron trap*, Review of Scientific Instruments, 92(12):123504, Dec 2021.
Doi: 10.1063/5.0067666

[38] U. Stroth, *Plasmaphysik*. Springer, 2011.
Doi: 10.1007/978-3-662-55236-0

[39] B. Durand de Gevigney, T. Sunn Pedersen, and A. H. Boozer, *Debye screening and injection of positrons across the magnetic surfaces of a pure electron plasma in a stellarator*, Physics of Plasmas, 18(1):013508, Jan 2011.
Doi: 10.1063/1.3543756

[40] H. Saitoh, J. Stanja, E. V. Stenson, U. Hergenhahn, H. Niemann, T. S. Pedersen, M. R. Stoneking, C. Piochacz, and C. Hugenschmidt, *Efficient injection of an intense positron beam into a dipole magnetic field*, New J. Phys., 17(10):103038, Oct 2015.
Doi: 10.1088/1367-2630/17/10/103038

[41] J. Horn-Stanja, E. V. Stenson, M. R. Stoneking, M. Singer, U. Hergenhahn, S. Nißl, H. Saitoh, T. S. Pedersen, M. Dickmann, and C. Hugenschmidt, *Injection of intense low-energy reactor-based positron beams into a supported magnetic dipole trap*, Plasma Res. Express, 2(1):015006, Jan 2020.
Doi: 10.1088/2516-1067/ab6f44

[42] M. R. Stoneking, H. Saitoh, M. Singer, E. V. Stenson, J. Horn-Stanja, T. S. Pedersen, S. Nißl, U. Hergenhahn, N. Yanagi, C. Hugenschmidt, M. Dickmann, J. R. Danielson, and C. M. Surko, *Toward a compact levitated superconducting dipole for positron-electron plasma confinement*, AIP Conference Proceedings, 1928(1):020015, Jan 2018.
Doi: 10.1063/1.5021580

[43] H. Saitoh, M. Stoneking, and T. S. Pedersen, *A levitated magnetic dipole configuration as a compact charged particle trap*, Review of Scientific Instruments, 91(4):043507, Apr 2020.
Doi: 10.1063/1.5142863

[44] CAEN, *16 Channel 14 bit 500 MS/s Digitizer V1730*, Jun 2016, Rev. 2.

[45] K. Iwata, R. Greaves, and C. Surko, *γ-ray spectra from positron annihilation on atoms and molecules*, Physical Review A, 55(5):3586, May 1997.
Doi: 10.1103/PhysRevA.55.3586

[46] P. M. Bellan, *Fundamentals of plasma physics.* Cambridge University Press, 2008.
Doi: 10.1017/CBO9780511807183

[47] P. O. J. Scherer, *Computational Physics.* Springer, 2010.
Doi: 10.1007/978-3-642-13990-1

[48] J. C. Simpson, J. E. Lane, C. D. Immer, and R. C. Youngquist, *Simple analytic expressions for the magnetic field of a circular current loop*, NASA Technical Report No. 20010038494, Jan 2001.

[49] T. L. Tang, *Finite length solenoid potential and field*, 2011.
URL: https://nukephysik101.files.wordpress.com/2011/07/finite-length-solenoid-potential-and-field.pdf

[50] M. Misakian, *Equations for the magnetic field produced by one or more rectangular loops of wire in the same plane*, J. Res. Nat. Inst. Stand. Technol., 105, Jul 2000.
Doi: 10.6028/jres.105.045

[51] D. Zhang and C. S. Koh, *An Efficient Semianalytic Computation Method of Magnetic Field for a Circular Coil With Rectangular Cross Section*, IEEE Transactions on Magnetics, 48(1):62–68, Jan 2012.
Doi: 10.1109/TMAG.2011.2167981

[52] P. Porazik, J. R. Johnson, I. Kaganovich, and E. Sanchez, *Modification of the loss cone for energetic particles*, Geophys. Res. Lett., 41(22):8107–8113, Nov 2014.
DOI: 10.1002/2014GL061869

[53] R. H. Landau, J. Paez, and C. C. Bordeianu, *A survey of computational physics.* Princeton University Press, 2011.
DOI: 10.1515/9781400841189

[54] T. Möller and B. Trumbore, *Fast, minimum storage ray/triangle intersection*, In *ACM SIGGRAPH 2005 Courses*, pages 7–es, Association for Computing Machinery, Jul 2005.
DOI: 10.1145/1198555.1198746.

[55] J. Cleary, B. Wyvill, G. Birtwistle, and R. Vatti, *A Parallel Ray Tracing Computer*, In *In Proceedings of the Association of Simulat Users Conference*, pages 77–80, 1983.

[56] I. Wald, T. Ize, A. Kensler, A. Knoll, and S. G. Parker, *Ray tracing animated scenes using coherent grid traversal*, In *ACM SIGGRAPH 2006 Papers*, pages 485–493, Jul 2006.
DOI: 10.1145/1179352.1141913.

[57] T. Akenine-Möller, *Fast 3D triangle-box overlap testing*, In *ACM SIGGRAPH 2005 Courses*, pages 8–es, Jul 2005.
DOI: 10.1145/1198555.1198747.

[58] J. Amanatides, A. Woo, et al., *A fast voxel traversal algorithm for ray tracing*, In *Eurographics*, volume 87, pages 3–10, 1987

[59] A. S. Glassner, *An introduction to ray tracing.* Morgan Kaufmann, 1989

[60] A. K. Singh and B. S. Bhadauria, *Finite Difference Formulae for Unequal Sub-Intervals Using Lagrange's Interpolation Formula*, Journal of Math. Analysis, 3:815–827, Jan 2009

[61] D. Geier, *Advanced octrees 2: Node representations*, Aug 2014.
URL: https://geidav.wordpress.com/2014/08/18/advanced-octrees-2-node-representations/

[62] B. Goetz et al., *Project Valhalla.*
URL: https://openjdk.org/projects/valhalla/

[63] S. Nißl, E. V. Stenson, U. Hergenhahn, J. Horn-Stanja, T. Sunn Pedersen, H. Saitoh, C. Hugenschmidt, M. Singer, M. Stoneking, and J. Danielson, *Positron orbit effects during injection and confinement in a magnetic dipole trap*, Physics of Plasmas, 27(5):052107, May 2020.
DOI: 10.1063/5.0007252

[64] A. H. Boozer, *Time-dependent drift Hamiltonian*, The Physics of Fluids, 27(10):2441–2445, 1984.
DOI: 10.1063/1.864525

[65] E. W. Weisstein, *Sphere point picking*, 2002

[66] Stahl-Electronics, *High Voltages Switch HS-500*, Jan 2022, Rev. 1.45.

[67] Stanford Research Systems, *Digital Delay Generator DG645*, Sep 2008, Rev. 1.1.

[68] R. Kennke et al., *Project Lilliput*.
URL: `https://wiki.openjdk.org/display/lilliput`

[69] J. R. Rose et al., *Project Panama*.
URL: `https://openjdk.org/projects/panama/`

[70] *NIST Digital Library of Mathematical Functions*, `http://dlmf.nist.gov/` `19.8`, Release 1.0.17 of 2017-12-22, F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller and B. V. Saunders, eds., 2018.
URL: `http://dlmf.nist.gov/`

Acknowledgments

First, I would like to thank Ulrich Stroth and Thomas Sunn Pedersen for giving me the opportunity to do this work. I would also like to give a special thanks to my mentor Eve Stenson, who was always there to help and advise me. Furthermore, I would like to express my gratitude to all current and former members of the APEX Collaboration for all their help and support. Zu guter Letzt möchte ich mich ganz herzlich bei meinen Eltern für ihre bedingungslose Unterstützung bedanken und dafür, dass sie sich fast jeden Tag die Frage verkniffen haben, wann ich endlich fertig bin.