COMET Phase-I Track Reconstruction using Machine Learning and Computer Vision

Ewen Lawson Gillies

High Energy Physics Group

Department of Physics
Imperial College London

A dissertation submitted to Imperial College London
for the degree of Doctor of Philosophy

Abstract

The COMET experiment will measure Charged Lepton Flavour Violation by searching for the neutrinoless decay of a muon into an electron while the muon is electromagnetically bound to an atomic nucleus. This μ-e conversion process is not allowed in the Standard Model of particle physics which makes it an excellent probe for Beyond the Standard Model physics.

The first phase of the experiment will improve the current sensitivity limit on μ-e conversion from the 7.0×10^{-13} at 90% C.L. to 3.0×10^{-15} at 90% C.L. To achieve this, COMET will utilize several novel design elements to produce the world's high-intensity muon beam to maximise the number of observed muonic atoms.

The high-intensity design of this experiment poses significant challenges to both the tracking and triggering systems. A novel algorithm called the CDC Hit Filter is designed to alleviate these challenges. The algorithm utilizes machine learning classification and a circular Hough Transform to identify and remove background hits. When applied offline, it can remove 98% of background hits while retaining 98% of signal hits. When adapted to the online environment, it can remove 89% of the background hits considered by the trigger while retaining 89% of signal hits. Both of these mark a significant improvement over the more traditional cut-based approach, which can remove 75% of background hits while retaining 75% of signal hits.

Declaration

This dissertation is the result of my own work, except where explicit reference is made to the work of others.

Ewen Lawson Gillies

Acknowledgements

First and foremost, I would like to thank my supervisor Yoshi Uchida. Your continued support of my pursuit of physics and your guidance throughout the years has made this Ph.D. possible. In doing so, it has given shape to the bright future I now see ahead of me. Thank you.

Similarly, I'd like to thank Ben Krikler. In addition to all the hours you spent teaching me, your work set the stage for mine and from it, I've harvested countless lessons, best practices, and inspiration that I will carry for the rest of my career.

To Yuki Fujii and Alex Rogozhnikov, thank you for welcoming me to your country as your student. I greatly value the expertise and culture you shared with me during the periods we spent working together. Most of all, thank you for contributing significantly to the algorithms presented in this work. They would not be possible without your input.

To my other Imperial colleagues: Ajit, Jordan, Paul, Peter, Per, and Phill, thank you for all the feedback and wisdom you shared with me over the years. To Yoshi Kuno and the rest of the COMET collaboration, thank you for all your hard work and thank you for all the good times we shared.

Moving on to institutions, I'd like to thank the Schrödinger Scholarship Scheme at Imperial College London for giving me the opportunity to pursue my Ph.D. at my dream university. Additionally, I'd like to thank Yandex Data Factory, the Japanese Society for the Promotion of Science, the Institute of High Energy Physics, and the Nvidia Corporation for their support of this research.

Last but not least, I'd like to thank my high school physics teacher, Raphael Uziel. You were my first physics teacher and now, over ten years later, you're still the best lecturer I've ever had. Thank you so much for introducing me to physics and for starting me on a path that turned out to be an excellent adventure.

Contents

Chapter 1.

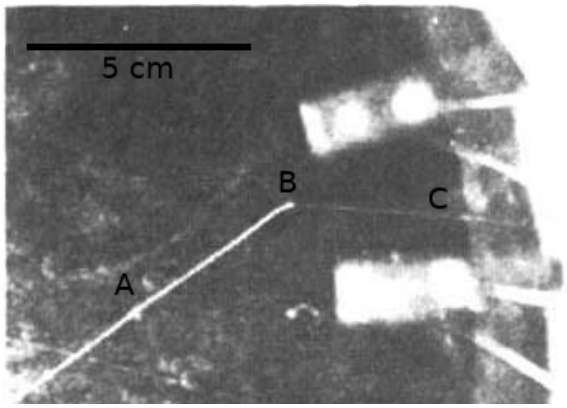
The History of Muon Mysteries

The muon has a rich history of refusing to behave as physicists expect. Its discovery baffled physicists at the time, most of whom were content with a universe that consisted of electrons, neutrons, protons, positrons and photons. During the ten years following its discovery, it was thought to be a negative-pion. Even once it was correctly identified, its behaviour motivated theoretically dubious Lepton Flavour Conservation laws. It is currently responsible for some of the largest disagreements between measurement and the Standard Model.

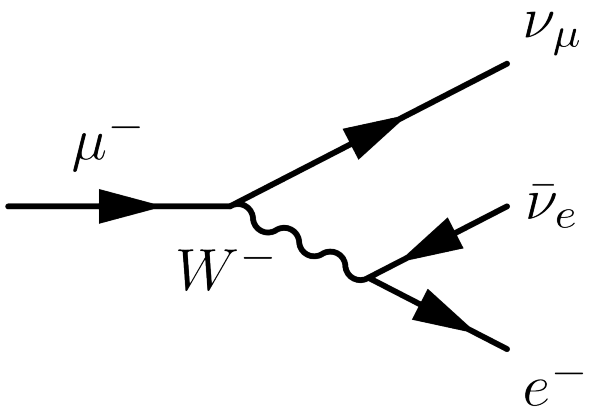
The early history of muon research exhibits a noteworthy pattern: the theory surrounding the particle raced to keep up with the relevant experimental measurements. This chapter discusses where and when the gaps between muon theory and experiment have occurred, how some of these gaps were solved, and details the ones that still exist. It concludes with a summary of the ongoing research that is relevant to the COMET experiment.

1.1. The Discovery of the Muon

In 1937, the first muons were detected in the US and Japan [1, 2]. These new, alien particles were first detected as cosmic rays and identified as "less massive than protons but more penetrating than electrons." Two years previous to their discovery, Hideki Yukawa had hypothesized a massive, charged particle to mediate the force between nucleons, which is now known as the strong force [3]. He described this particle as roughly "2×10^2 times as large as the electron mass", which agreed with the measured mass of the cosmic ray muons. This red herring was reinforced in 1940, muons were first observed to decay into electrons by Williams and Roberts, as seen in Figure 1.1a. This too agreed with Yukawa's hypothesised particle.



(a) This image from 1940 [2] shows a bubble chamber with a muon trajectory originating bottom left corner as marked by A. The muon decays at B, resulting in a thinner electron track as marked by C.



(b) A Feynman diagram of the muon decay seen in Figure 1.1a as described in the Standard Model. The inner workings of this diagram were a topic of much debate when the muon was first discovered.

Figure 1.1.: These images compare the first observation of muon decay to the modern understanding of the process. By 1940, physicists were confident that the muon decayed into an electron, as seen in Figure 1.1a. They were unsure of what other particles were involved. Figure 1.1b shows the modern understanding of muon decay in the Standard Model.

The observation of muon decay only created more questions. Four-momentum conservation dictates that the daughter electron must be accompanied by some other particle. This agreed with experimental data, which observed that this electron carried away a fraction of the available energy. The three competing explanations were that the muon was:

- a spin-1 particle that decayed into an electron and one neutrino,
- a spin-$\frac{1}{2}$ particle that decayed into an electron and a photon, or
- a spin-$\frac{1}{2}$ particle that decayed into an electron and two neutrinos.

The first case agreed well with Yukawa's prediction of a spin-1 particle. However, in 1941 Nordheim studied the missing energy of the decay electrons and noted it was too high to fit with either of the first two cases. This led him to speculate that muon decay may result in the "simultaneous creation of a number of neutrinos" [4].

In 1947, measurements of the decay modes of the mysterious cosmic rays identified two, distinct new particles. The parent cosmic ray was identified as Yukawa's "pi-meson", which we now know as the pion [5], while the decay particle was named the "muon". This discovery meant that they hypothesis that muons were spin-1 could abandoned, which leads to a number of searches for muons decaying into electrons and photons, all of which

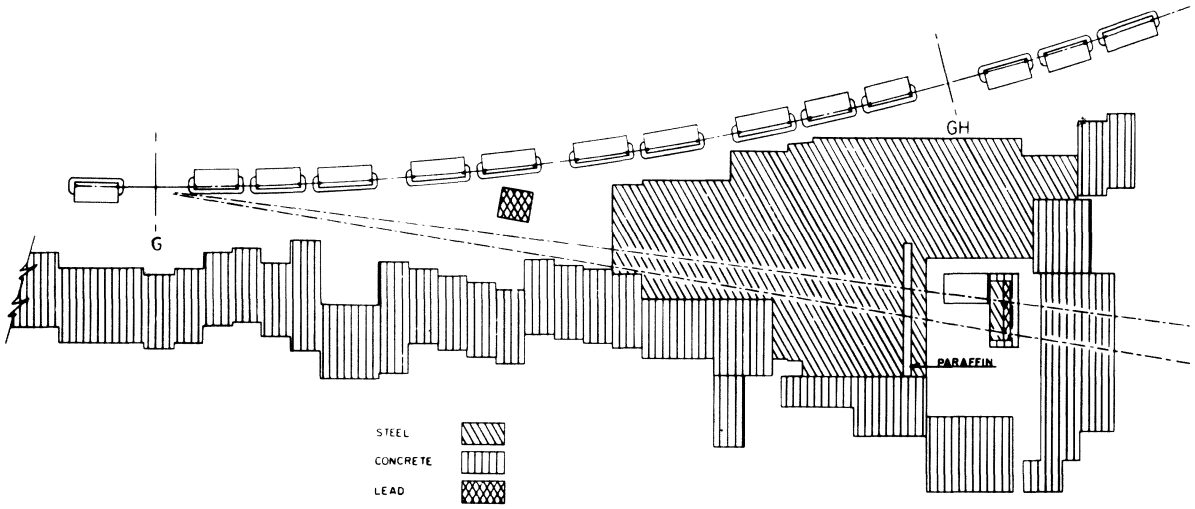


Figure 1.2.: The beamline schematic for the 1962 Brookhaven muon-neutrino discovery experiment [7]. In this setup, 15 GeV protons were fired at a fixed beryllium target to create a beam of secondary pions. These pions decay to create muon and muon-neutrino pairs. Some of the weakly-interacting neutrinos are able to then pass through over 13 meters of steel shielding to hit the 10-ton aluminium spark chamber. The detector watches for the appearance of either muons or electrons resulting from neutrino interactions.

come back with negative results [6]. This is implied that the muon decayed into an electron and two neutrinos.

If this were true, another natural conclusion could be reached: the two decay neutrinos could not annihilate with each other to produce a photon. This motivated the existence of conserved quantum numbers that distinguished the two neutrinos. This is now referred to as Lepton Flavour Conservation. In muon decay, this means that the decay electron is accompanied by an anti electron-neutrino and a muon-neutrino so that the muon-flavour and electron-flavour numbers are balanced between the initial and final states. This can be seen in the Feynman diagram shown in Figure 1.1b.

1.2. Muon Decay and Lepton Flavour Conservation

Lepton Flavour Conservation was not experimentally validated until 1962 when a group at Brookhaven National Laboratory created a neutrino beam and probed its lepton flavour content. The setup is discussed in Figure 1.2. This experiment found that only muon-like particles appear in the detector from the neutrino beam, thus proving that the beam must only contain muon-neutrinos. This observation was taken as proof of both lepton flavour and lepton flavour conservation [7].

Lepton flavour Conservation has since become a part of the Standard Model. Most conservation laws in the Standard Model are supported by Noether's theorem, which states that all symmetries in a physical system correspond to a conserved quantity [8]. Such conservation laws can be derived theoretically by finding the corresponding symmetry in the Standard Model Lagrangian. With that said, Lepton flavour Conservation has no corresponding symmetry, which plants a theoretical seed of doubt in Lepton flavour Conservation.

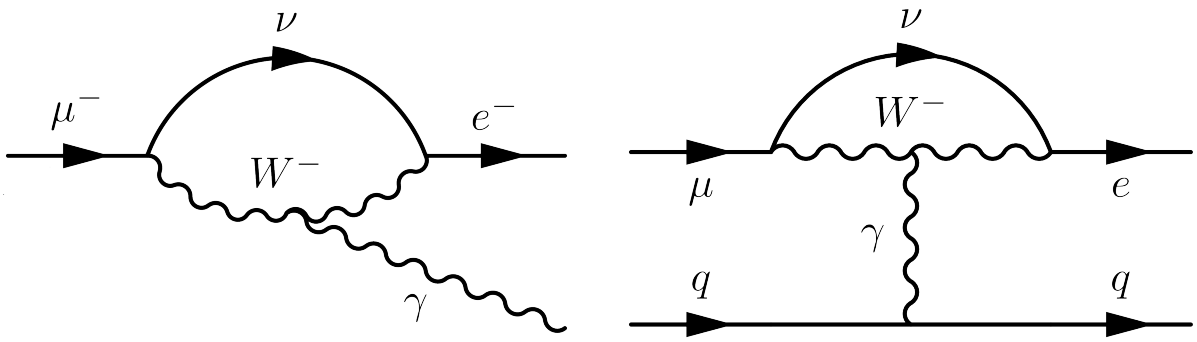
1.3. Neutrino Oscillation and Lepton Flavour Violation

This doubt was reinforced by experimental observation in 1968, when a team at Brookhaven measured the neutrino flux coming from the Sun [9]. They found that the electron neutrino flux was around a third of what they expected. This exact problem had been anticipated a year earlier by Pontecorvo, who predicted that neutrinos may oscillate between the two known neutrino flavours and cause "the flux of observable solar neutrinos [to] be half as large as the total flux of solar neutrinos" [10]. The process he described is now known as neutrino oscillation and violates Neutral Lepton flavour Conservation.

The first evidence of neutrino oscillation was observed by the Super-Kamiokande experiment in 1998, 30 years after the solar neutrino problem was posed [11]. This discovery proved that lepton flavour was not conserved by neutrinos! Instead, the theory of neutrino oscillation states that the flavour eigenstates of the neutrino are formed from mixtures of its mass eigenstates. While all neutrinos are created as one of the flavour eigenstates, they propagate a combination of the three mass eigenstates. Each of the mass eigenstates propagate differently, which means the flavour eigenstate of the neutrino evolves as it propagates [12]. By demonstrating and explaining Neutral Lepton Flavour Violation (NLFV), these results naturally call the laws of Charged Lepton Flavour Violation (CLFV) into question.

1.4. Motivating CLFV Searches

Searches for Charged Lepton Flavour Violation (CLFV) are primarily motivated by the fact that CLFV is not allowed in the Standard Model. Any signal of CLFV would provide great insight into Beyond the Standard Model physics while a negative result would be a powerful benchmark measurement for existing and future theories.



(a) A Feynman diagram of $\mu^- \to e^-\gamma$ caused through neutrino oscillation.

(b) A similar diagram to Figure 1.3a in the presence of nuclear matter.

Figure 1.3.: These diagrams compare two CLFV channels that arise from NLFV. In both cases, the neutrino flavour must then oscillate from a muon-neutrino to an electron-neutrino inside the loop it forms with the W boson.

With that said, the discovery of neutrino oscillation is already a Beyond Standard Model process which can give rise to some CLFV. Two such processes can be seen in Figure 1.3. Fortunately, the predicted rates of CLFV via NLFV are many orders of magnitude lower than current experimental limits. The easiest way to show this is to consider the $\mu \to e\gamma$ process shown in Figure 1.3a. This process has a branching ratio equal to:

$$\mathcal{B}(\mu \to e\gamma) = \frac{3\alpha}{32\pi}\left|\sum_{i=2,3} U^*_{\mu i} U_{ei} \frac{\Delta m^2_{i1}}{M^2_W}\right|^2 \simeq \mathcal{O}\left(10^{-54}\right) \tag{1.1}$$

where $U_{\alpha i}$ an element of the Pontecorvo-Maki-Nakagawa-Sakata (PMNS) matrix that describes the amplitude of the mass eigenstate i in flavour state α, Δm_{ij} is the mass difference between the i-th and j-th eigenstates, M_W is the mass of the W boson, and the leading α factor is the fine structure constant [13, 14]. With current experimental limits at $\mathcal{O}\left(10^{-13}\right)$, this calculation shows that μ-e gamma searches incur no observable intrinsic backgrounds from NLFV. Similar arguments can be made for other CLFV processes like coherent μ-e conversion, $\mu^- N \to e^- N$, an example of which is shown in Figure 1.3b.

CLFV searches are further motivated by existing tensions between lepton physics measurements and the Standard Model. There has been a 3σ disagreement between the SM prediction of the muon anomalous magnetic dipole moment and the measured value since 2001 [15, 16]. More recently, studies demonstrate a 7σ tension between measurements of the proton radius and its measured value when using the hyperfine structure of muonic hydrogen [17]. Finally, a number of collider experiments have shown tensions that call lepton universality into question [18, 19, 20, 21, 22, 23], which in turn can be shown to imply CLFV [24].

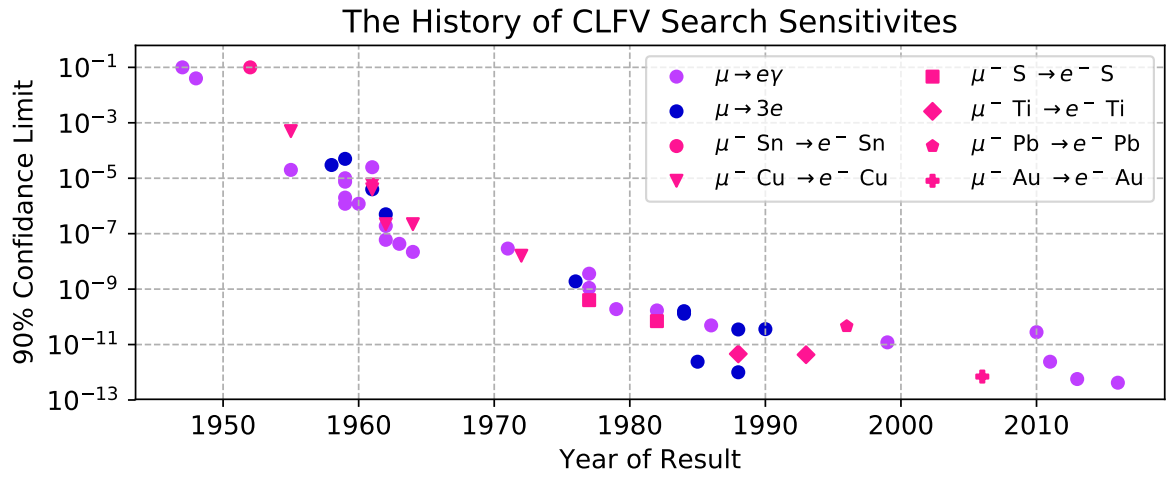


Figure 1.4.: Since the first study of muon decay in 1948, the sensitivity of muonic CLFV searches have improved by 12 orders of magnitude [14, 25]. Note that this plot also shows the evolution of the host nucleus material used in the $\mu^- N \to e^- N$ searches.

1.5. Modern Charged Lepton Flavour Violation Searches

The earliest search for muonic CLFV dates back to the initial studies of muon decay in 1948 [6]. Since then, experiments continued to increase the sensitivity of this search in three main channels:

- "μ-e gamma": $\mu^\pm \to e^\pm \gamma$
- "μ to three e": $\mu^+ \to e^+ e^- e^+$
- "μ-e conversion": $\mu^- N \to e^- N$

The evolution of the CLFV search sensitivities[1] in these channels can be seen in Figure 1.4. The MEG experiment is responsible for the most sensitive μ-e gamma measurement, while the SINDRUM collaboration is responsible for the most sensitive $\mu^+ \to e^+ e^- e^+$ and μ-e conversion measurements. The COMET and Mu2e experiments are both aiming to set the new μ-e conversion sensitivity limit.

1.5.1. The MEG Experiment

The MEG Experiment completed its first data-taking period at the Paul Scherrer Institute (PSI) in Switzerland in 2009 and published its final results in 2016. This result set an

[1] All exclusion measurements from here on are taken at 90% C.L., although the notation is dropped for brevity.

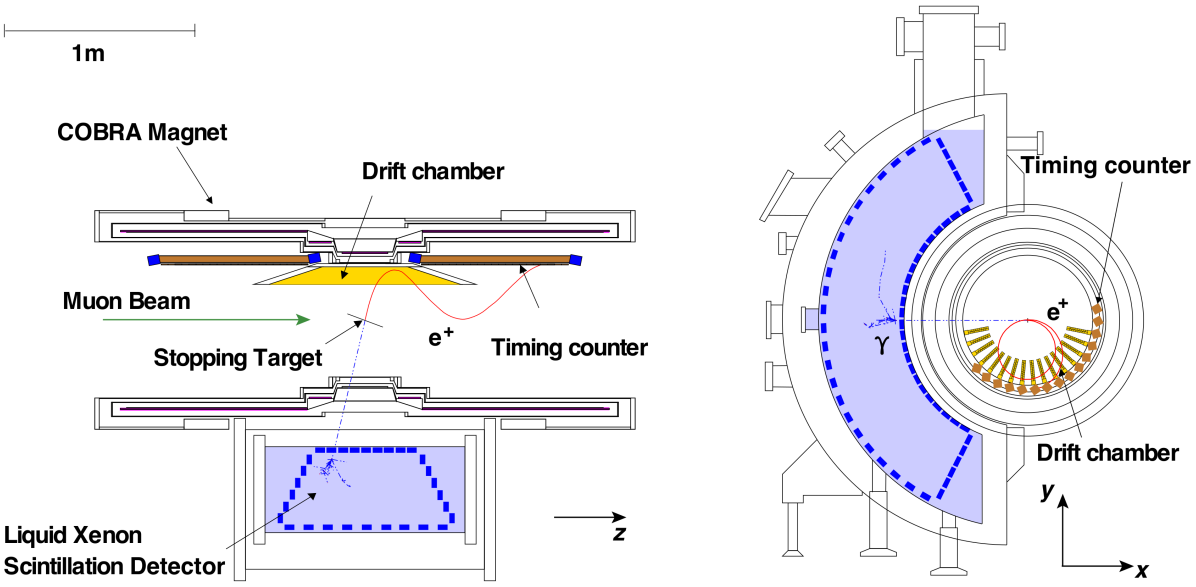


Figure 1.5.: The MEG detector system measures the decay products of stopped μ^+ in a fixed target. It is composed of an liquid Xenon electromagnetic calorimeter and a tracking system that respectively search for photons and positrons from the $\mu^+ \to e^+\gamma$ decays. It operates at PSI, which houses the world's most intense continuous μ^+ beam [25].

upper limit of 4.2×10^{-13} on the $\mu^+ \to e^+\gamma$ process[2], which is a factor 30 improvement over the previous result [25]. This is the most recent result across all three channels, as seen in Figure 1.4. The experimental setup used to make this measurement is discussed in Figure 1.5.

1.5.2. The SINDRUM Experiment

The SINDRUM experiment published the current sensitivity limit for the $\mu^+ \to e^+ e^- e^+$ in 1988 at 1.0×10^{-12} [27]. Of the three channels, this is the oldest world-leading measurement. The next-generation of this search will be carried out at PSI by the Mu3e collaboration. This experiment is in its final stages of construction as PSI, with plans to start data taking in 2019 [28].

The SINDRUM collaboration went on construct the SINDRUM-II experiment at PSI, which set world-leading sensitivity limit for the $\mu^- N \to e^- N$ in 2006 at 7.0×10^{-13} [28]. The detector setup for this experiment and a simulated signal event are shown in Figure 1.6.

[2] Searches for $\mu^+ \to e^+\gamma$ are favoured over $\mu^- \to e^-\gamma$ since they avoid the muon capture related backgrounds, which are discussed in Section 2.3.4

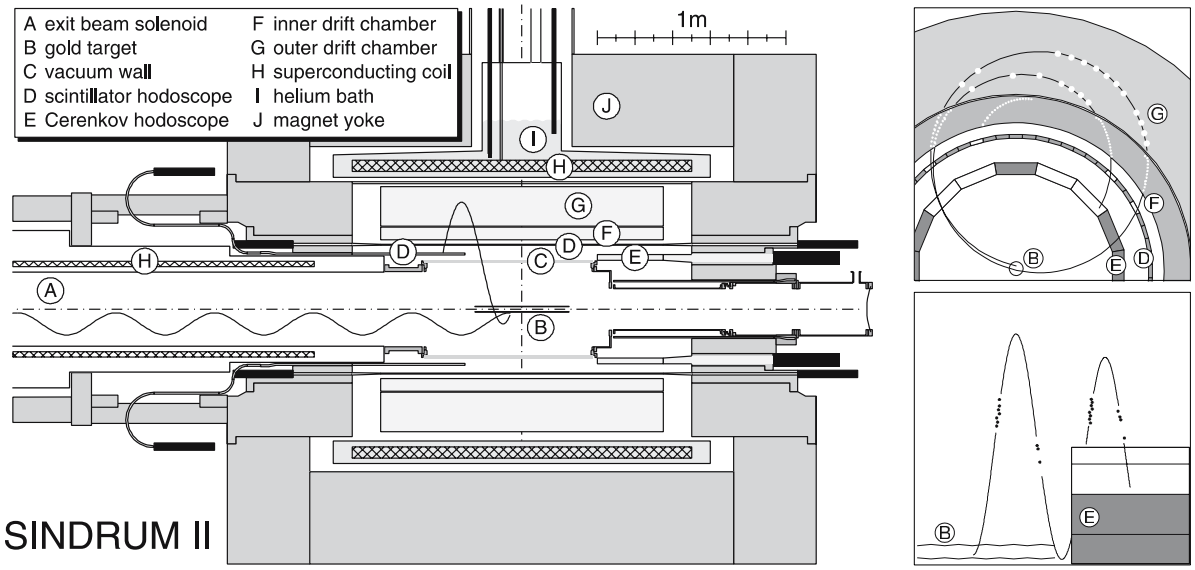


Figure 1.6.: The SINDRUM-II detector system probed for $\mu^- N \rightarrow e^- N$ on a fixed gold target [26]. A sample signal event from GEANT simulations is shown in the images to the right.

1.5.3. The COMET and Mu2e Experiments

While SINDRUM was underway in Switzerland, similar research was being carried out at the Moscow Meson Factory in what was then the USSR. In 1989, Soviet researchers proposed the MELC experiment, which was to be the next generation μ-e conversion experiment [29, 30]. The project was given the green light in 1992, but unfortunately had to be shut down in 1995 due to the political and economic turbulence that was caused by the dissolution of the USSR. In 1997, researchers at Brookhaven National Laboratory proposed a new experiment called MECO, which was heavily inspired by MELC. It was designed to probe for μ-e conversion down to a limit of 2×10^{-17} [31, 32]. It was approved in 2001, only to be cancelled in 2005 when funding was pulled from Brookhaven's particle physics program [33].

In 2009, a new collaboration called Mu2e was formed in the US from the ashes of the MECO collaboration. The detector hall is currently under construction at Fermilab. Mu2e's beamline design (which echoes MECO's) can be seen in Figure 1.7. Mu2e hopes to achieve a sensitivity of 3×10^{-17} over a three-year data-taking period on the same time scale as COMET [34]. As such, Mu2e is COMET's competition.

In 2009, the COMET collaboration was also formed. Like MECO, the COMET experiment borrows many design principles first seen in the MELC experiment. COMET is currently under construction in Tokai, Japan. Unlike Mu2e, COMET will be built in two phases, as is

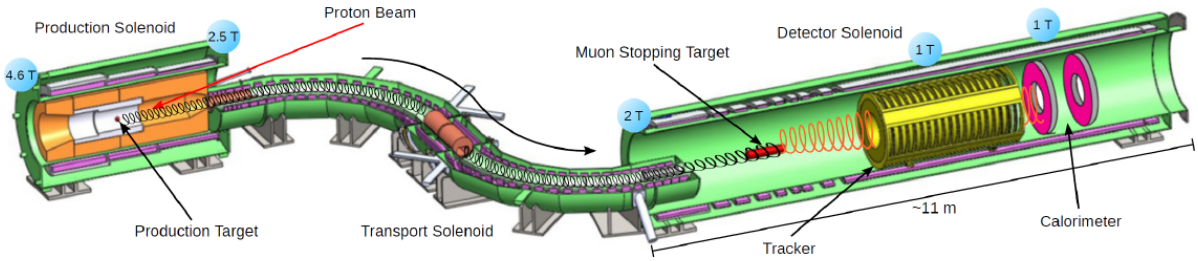


Figure 1.7.: The Mu2e experiment is designed to probe for $\mu^- N \rightarrow e^- N$ [34]. Its design was heavily inspired by the MECO and MELC experiments. Mu2e is directly competing with COMET to make a world-leading sensitivity measurement of $\mu^- N \rightarrow e^- N$.

discussed in Chapter 3. In its first phase, COMET will achieve a sensitivity of 3×10^{-15} over a 180-day data-taking period. In the full design in the second phase, COMET will achieve a sensitivity of 3×10^{-17}, again over a 180-day data-taking period. This phased design will help illuminate the underlying physics involved in the COMET experiment, which is discussed in the next chapter.

Chapter 2.

The Physics of Muon to Electron Conversion

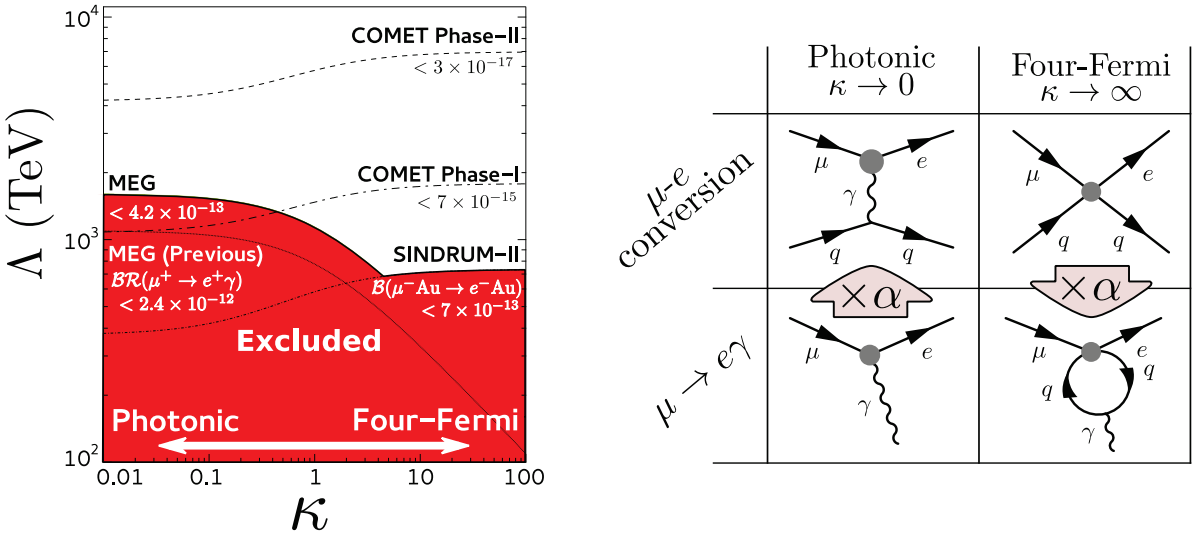
Muon to electron conversion refers to the neutrino-less decay of a muon into an electron while the muon is electromagnetically bound to an atomic nucleus. It can be formalised as:

$$\mu^- + N(A,Z) \rightarrow e^- + N(A,Z) \tag{2.1}$$

While this process is not allowed by the Standard Model, most extensions to the Standard Model predict μ-e conversion at some level [35, 36, 37]. Even neutrino oscillation can give rise to this process, as discussed in Section 1.4. With that said, the landscape of Beyond Standard Model physics is constantly shifting as theories are discarded, created, or adapted to account for new measurements.

The beauty of searching for μ-e conversion experimentally is that it is largely immune to the shifts in theoretical predictions. In fact, it does not require any hypothesis about which extension to the Standard Model is correct. Instead, these searches only seek to measure the initial bound muon and resulting decay electron of whatever Beyond Standard Model interaction mediates this process. This property is commonly referred to as "model independence".

With model independence in mind, this chapter will start by discussing a simplified effective field theory that highlights how different search channels are affected by the underlying CLFV process. It then focuses on the Standard Model physics that describes the properties of the signal electron and the critical backgrounds.



(a) The coverage of Λ-κ space by modern CLFV experiments.

(b) A comparison of how the two CLFV channels can mimic one another, albeit at a suppressed rate.

Figure 2.1.: A comparison of the μ-e gamma and μ-e conversion search channels. Figure 2.1a demonstrates the complementary coverage of the two channels in the κ-Λ phase space, as introduced in equation (2.2). Figure 2.2b illustrates how one channel may be suppressed over the other depending on the nature of the New Physics [38].

2.1. Model-Independent Effective Field Theory for Muon CLFV

The COMET experiment is almost completely model-independent. The only requirement is that the new physics should manifest at low energies as a four-Fermi contact interaction or in a photonic interaction, and that these interactions are mediated by physics processes that are best described at a very high energy scale. From this, one can imagine resumming the new physics by interaction type to yield the following effective Lagrangian[1]:

$$\mathcal{L}_{\mathrm{CLFV}} = \frac{1}{\kappa+1}\frac{m_\mu}{\Lambda^2}\bar{\mu}_R\sigma_{\mu\nu}e_L F^{\mu\nu} + \frac{\kappa}{\kappa+1}\frac{1}{\Lambda^2}\,\bar{\mu}_L\gamma_\mu e_L\left(\,\bar{u}_L\gamma^\mu u_L + \bar{d}_L\gamma^\mu d_L\right) + h.c. \qquad (2.2)$$

where κ and Λ are parameters, the (L, R) subscripts denote the chirality of the fermion fields, $F^{\mu\nu}$ is the photon field strength, and m_μ is the muon mass.

The first term describes a photonic CLFV interaction, while the second describes the four-Fermi interaction. The parameter Λ has units of energy and characterizes the energy-

[1] This treatment is first discussed in [39] in 2013. A review of the current status of such treatments is discussed in [40].

scale of the new physics processes. The dimensionless parameter κ determines the extent to which the new physics is photonic ($\kappa \to 0$) or four-Fermi-like ($\kappa \to \infty$). The limits on the κ-Λ parameter space from experimental searches is shown in Figure 2.1a. This figure highlights that these channels can probe much higher energy scales than are currently available in direct searches.

If the new physics is photonic, the μ-e gamma searches will be more sensitive. The μ-e conversion could still detect the new physics if the outgoing photon interacts with the host nucleus. This incurs a factor of α, the fine structure constant, and hence reduces the sensitivity by two orders of magnitude. If the new physics is four-Fermi-like, then the μ-e conversion searches will be more sensitive. In this case, the μ-e gamma searches could only see a signal if the quark lines reconnect and a photon is emitted, which would significantly impact the sensitivity of the search. Diagrams corresponding to these scenarios can be seen in Figure 2.1b. Taken together, these searchers are complementary. A signal in both channels could determine a value of κ, which could help illuminate the underlying nature of the new physics.

2.2. Constraining Theoretical Physics

A high-sensitivity measurement of μ-e conversion would heavily constrain a number of new physics models. For example, consider the Grand Unification Theories based around the E_6 Lie group [41]. These theories hypothesise that all known interactions, with the exception of gravity, can be embedded into the E_6 gauge group. This group admits a complex or chiral representation, which is needed to construct the Standard Model [42]. After symmetry breaking, this group can give rise to the $SU(3) \times SO(2) \times U(1)$ gauge group of the Standard Model.

These models predict the existence a new, neutral Z' boson. This particle is expected to admit naturally large flavour changing vertices. Given that these interactions have not been observed by past $\mu^- N \to e^- N$ experiments, the Z' boson is hypotheised to have a large mass. To satisfy limits on $\mu^+ \to e^+ e^- e^+$ and naturalness assumptions on the fermion mass matrix, the additional gauge boston "should not be much lighter than $\mathcal{O}$(TeV)" [43]. If COMET detects $\mu^- N \to e^- N$, the experiment could provide the first insight into Z' interactions. If it does not, then E_6 theories are forced to consider unnaturally small femion mixing levels or unnaturally massive Z' bosons [43].

2.3. Measuring Muon to Electron Conversion

Searches for μ-e conversion rely on a firm understanding of the initial-state muonic atom, the properties the final-state signal electron, and all background processes that can mimic this final state. This requires an interesting mixture of Standard Model theory, nuclear physics models, and phenomenological models.

2.3.1. The Bound Muon

As muons travel through atomic matter, they lose energy due to ionization. When the kinetic energy of the muon reaches a lower threshold, typically a few keV, an atom in the material is able to capture the muon in its Coulomb potential. The bound state of a muon and an atomic nucleus is referred to as a "muonic atom".

Once bound to the nucleus, the muon transitions down through the atomic orbitals, typically reaching the ground state after $\mathcal{O}(100$ fs). Each transition incurs an X-ray emission with an energy that characterizes the given transition [44]. This is used in experiments to determine the total number of muonic atoms formed in a target material. From this point, the bound muon can either undergo the signal process, decay while orbiting the nucleus, or be captured by the nucleus.

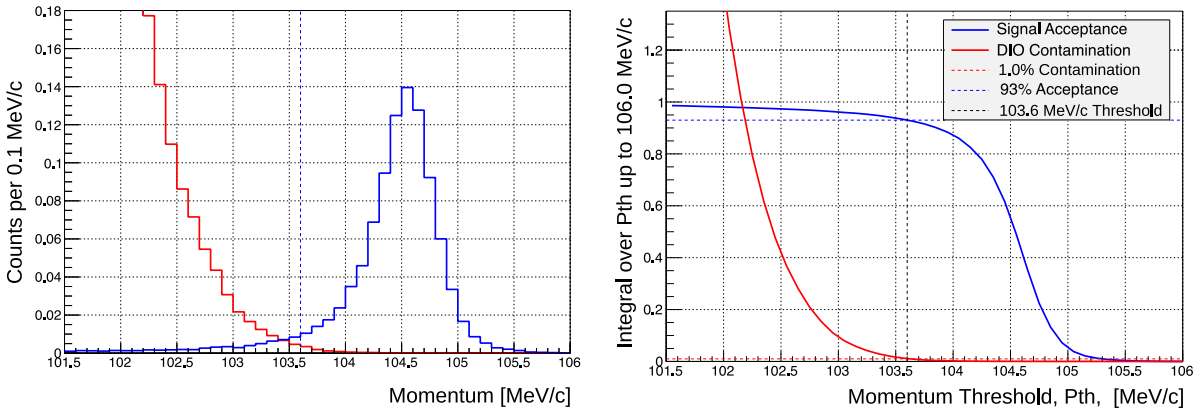
2.3.2. The Signal Electron

All μ-e conversion searches hypothesise that the ground-state muon may interact with the quarks of the nucleus via some new physics process. This process would yield an electron and an unchanged nucleus, which violates lepton flavour conservation, but leaves the other quantum numbers intact. This results in a well-defined signal, despite the electron's mysterious origin.

The kinematics of the signal is determined by the conservation of four-momentum. Since the nucleus is unchanged, all of the energy is transferred to the electron and to the recoil of the nucleus. The electron energy, E_e, is calculated as:

$$E_e = m_\mu c^2 - E_B - E_R \tag{2.3}$$

where $m_\mu = 105.66$ MeV/c^2 is the mass of the muon [45], E_B is the binding energy of the decay electron to the nucleus, and E_R is the energy of the recoiling nucleus. Both E_B and E_R vary from element to element.



(a) The reconstructed momentum distribution of simulated signal and DIO electrons.

(b) The acceptance of signal and DIO electrons as a function of the momentum threshold.

Figure 2.2.: A comparison of signal and DIO electron momenta and acceptances. This data is simulated assuming a rate of 3×10^{-15} signal electrons per muonic atom and the DIO electrons were modelled using the momentum spectrum presented by Czarnecki in [46]. The vertical scale in Figure 2.2a has been set so that the integral of the signal curve is one.

This means the signal as a lone, mono-energetic electron. As such it is inherently immune to combinatorial backgrounds that limit the other two search channels, which makes this process an excellent candidate for high-intensity searches. Best of all, the energy of this electron separates it from the electrons generated by background processes.

2.3.3. Muon Decay in Orbit

Muon decay-in-orbit (DIO) is one of the main background sources in modern μ-e conversion searches. It occurs when the bound muon undergoes its flavour conserving decay mode:

$$\mu^- + N(A,Z) \rightarrow e^- + \overline{\nu}_e + \nu_\mu + N(A,Z) \tag{2.4}$$

This process looks similar to free muon decay with one notable difference: the presence of the nucleus affects the allowed phase space of the electron energy.

In free muon decay, the maximum decay electron energy is half of mass-energy of the muon, i.e. 52.83 MeV/c^2. This occurs when the electron recoils back-to-back with the neutrinos. In the bound case, the nucleus provides the decay with an additional degree of kinematic freedom. This allows the decay neutrinos to recoil off of each other while the electron to recoils off of the nucleus. Nearly all of the mass-energy of the muon can be transferred to the electron in this configuration, which perfectly mimics the signal in μ-e conversion searches.

Fortunately, this process is heavily suppressed near the signal energy, so much so that it has only become relevant in modern searches [26, 38]. Figure 2.2a illustrates how the high-momentum tail events from DIO electrons encroaches on the distribution of signal electron momenta. A lower bound is selected for the signal momentum window to remove this background, such that any electrons below this threshold are not considered to be signal electrons. Figure 2.2b shows the acceptance of both signal and DIO electrons as a function of this threshold.

2.3.4. Nuclear Muon Capture

Nuclear muon capture is another potentially problematic background in μ-e conversion searches. It is an analogous process to electron capture:

$$\mu^- + N(A, Z) \rightarrow \nu_\mu + N(A, Z-1) \tag{2.5}$$

This process leaves the nucleus in an excited state. As it de-excites, it can emit a number of particles, including protons, neutrons, gamma rays, alpha particles, deuteron, and triton.

Although this cannot produce any signal-like electrons, the sheer volume of these decay particles can be dangerous in high-intensity μ-e conversion searches. The charged particles can flood the detectors, drowning out any signal electrons. The neutral particles can pierce through shielding material to damage the electronics. To make matters worse, the underlying nuclear processes extremely hard to model in general and can change depending on the original nucleus. Instead, experimental measurements of the decay products are used to construct simple phenomenological models. Most notably, the AlCAP Experiment at PSI aims to measure nuclear muon decay spectrum for aluminium in preparation for the COMET and Mu2e experiments [47].

2.3.5. The Host Nucleus

The properties of the signal and background processes depends greatly on the atomic number, Z, of the host nucleus. The maximum energy available to the both signal and DIO electrons is impacted by the binding energy and recoil energy of the nucleus, as given by equation (2.3). For heavier elements, the binding energy is the dominant effect, while for lighter elements, the recoil energy becomes more significant. The SINDRUM-II experiment opted for a heavy host nucleus, gold, which corresponds to a maximal energy of 95.56 MeV/c^2 [26], which is significantly lower than the total 105.66 MeV/c^2 available. In

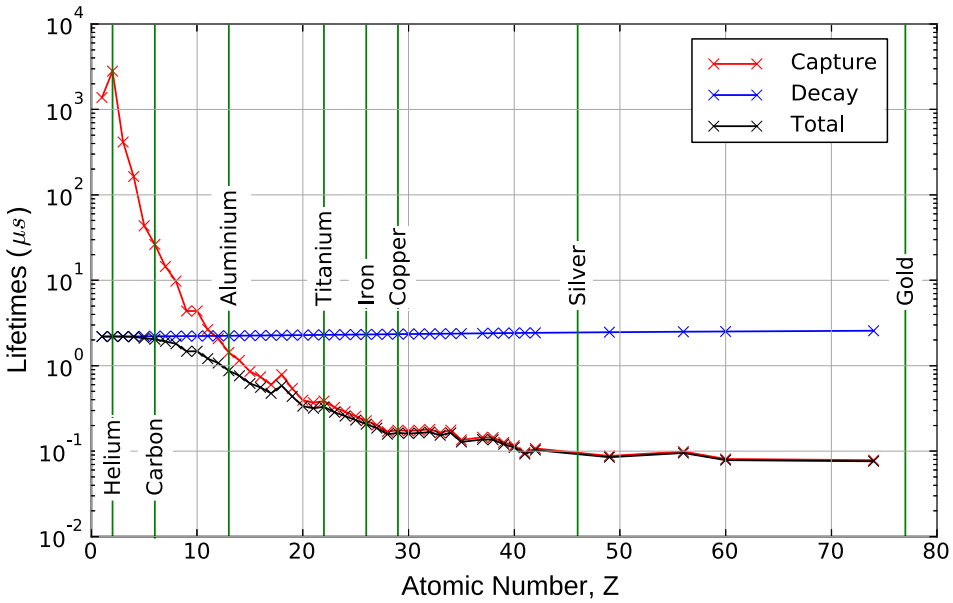


Figure 2.3.: The bound muon lifetime as a function of Z of the host nucleus. The red line shows the partial lifetime of nuclear muon capture, the blue line shows the partial lifetime of muon decay-in-orbit, while the black line shows the total lifetime. The plot is taken from [38], which read the data from Geant4 [48]. Note that the partial decay lifetime is assumed to be the same for all elements in this plot.

general, higher energy signals are easier to distinguish from electrons from other potential background sources, like beam-electrons or cosmic rays. This favours lighter host nuclei for μ-e conversion searches.

As the atomic number of the element increases, the muon becomes more tightly bound to the nucleus. This increases the overlap between the muon wavefunction and the nuclear wavefunction, which increases the interaction probability of the signal process. Furthermore, since a bound muon is never truly at rest, time dilation increases the partial lifetime of muon decay. This effect scales with the atomic number of the nucleus. Both of these properties favour heavier host nuclei for μ-e conversion searches.

Most importantly for COMET, the lifetime of the bound muon varies greatly as a function of Z. Since the signal process is extremely rare, it has no measurable effect on the bound muon lifetime. Instead, this lifetime is determined a shifting balance between the muon's DIO and nuclear capture rates. The nuclear capture rate dominates for heavier elements due to the increased number of protons that can participate and the increased overlap between the muon waveform and the nucleus. For elements lighter than $Z \approx 12$, this shifts so that the DIO rates dominate [38]. A plot of this relationship can be seen in Figure 2.3.

The COMET experiment will use aluminium as the host nuclei. This element corresponds to a to a signal electron energy of 104.96 MeV/c^2, meaning only 0.7 MeV/c^2 is lost to the aluminium nuclei. The bound muon lifetime for this element is 864 ns, which is long enough to allow for a clean separation between beam-related backgrounds and signal particles. Once the muons bind to the aluminium nucleus, 61% will be captured by the nucleus, leaving the remaining 29% to decay-in-orbit or convert to an electron [38]. The COMET experiment is designed to heavily exploit both of these properties,as discussed in the next chapter.

Chapter 3.

The COMET Experiment

The COMET experiment will take place in two phases: Phase-I and Phase-II. The first phase has two main goals:

- to probe μ-e conversion at 100 times the sensitivity limit set by SINDRUM-II [26], and
- to gain an understanding of the beamline in preparation for Phase-II.

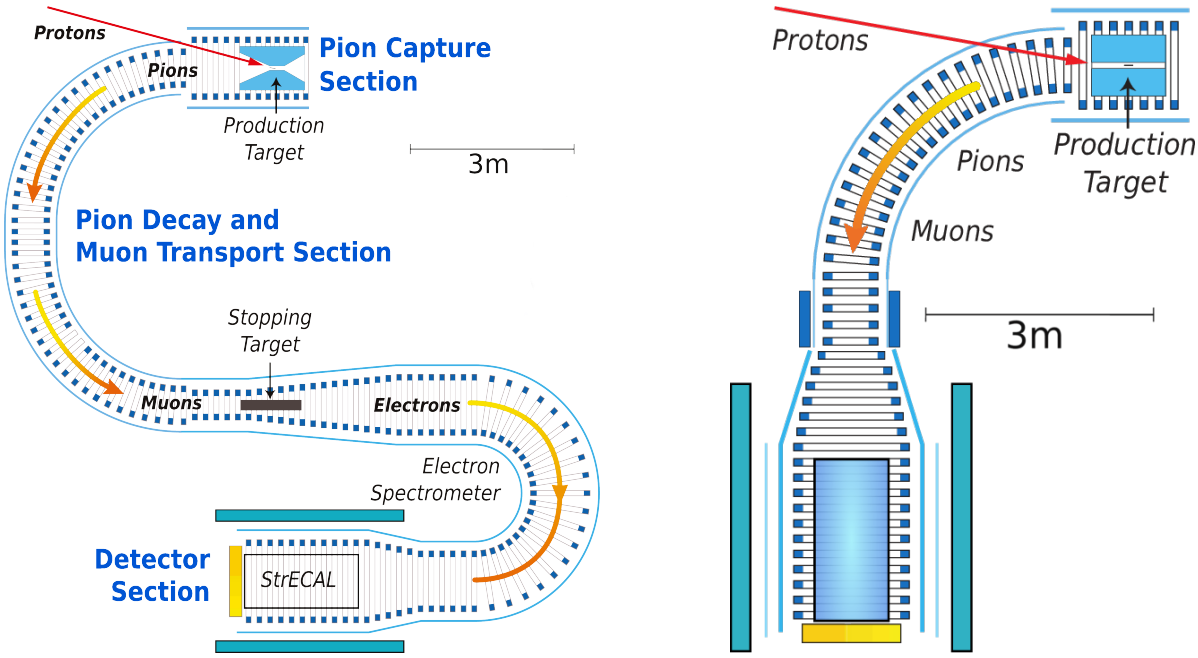
The first goal is the focus of this thesis. To achieve this goal, the COMET beamline is designed to create the initial-state bound muons in the target material. The collaboration has built the Cylindrical Detector (CyDet) for Phase-I to measure the momentum of the resulting particles in search of the signal electron. Together, the experiment is designed to take full advantage of the signal properties, introduced in Section 2.3.

3.1. The COMET Beamline

The COMET experiment will achieve a world-leading sensitivity measurement of μ-e conversion. To this end, both Phase-I and Phase-II aim to:

- Remove all sources of high-momentum background electrons in the beam.
- Create as many muonic atoms in the target material as possible.
- Avoid backgrounds from low-momentum and beam-prompt particles.
- Remain sensitive to the high-momentum, delayed signal electrons.

These goals have inspired the striking Phase-II beamline design seen in Figure 3.1a. Moving in the direction of the arrows in this figure, each component is summarized:



(a) The Phase-II design.

(b) The Phase-I design.

Figure 3.1.: The COMET Phase-II beamline compared to the Phase-I beamline. The Pion Capture section, the first 90-degree bend of the Muon Transport Section, and the Detector Solenoid appear in both phases [38].

1. **The Proton Beam**: A pulsed, high-intensity beam of 8 GeV protons strikes the stationary Pion Production Target.
2. **The Pion Capture Section**: A superconducting solenoid with a 5 T field captures the low-momentum pions and directs them into a curved solenoidal section.
3. **The Muon Transport Section**: The curved solenoids are used in conjunction with collimators create a low-momentum muon beam from the pion beam.
4. **The Stopping Target Section**: The muon beam is directed at the Muon Stopping Target, where the muonic atoms are created.
5. **The Electron Spectrometer**: Particles from the stopped muon processes are steered into another curved solenoid that is designed to only accept signal electrons.
6. **The Detector Solenoid**: The detector system sits inside a solenoid providing a 1 T magnetic field and watches for the signal electrons.

The Phase-I beamline, seen in Figure 3.1b, is a truncated version of this full design that achieves the goals outlined above. It incorporates the Proton Beam, the Pion Capture Section, and half of the Muon Transport Section. It does not have the Stopping Target

Section or Electron Spectrometer. Instead, the Muon Stopping Target sits inside the Detector Solenoid, with the CyDet system surrounding it. This section will detail the Phase-I beamline components, while a detailed discussion of Phase-II can be found in [38].

3.1.1. The Proton Beamline

The COMET experiment is the future flagship experiment for the Japan Proton Accelerator Research Complex (J-PARC). This laboratory is currently in the process of a major upgrade to their proton beamline to deliver the high-intensity, pulsed proton beam to the COMET experiment [49]. These protons strike the fixed target inside the Pion Capture Section to create a high yield of low-momentum pions, which decay in the beamline to create the muon beam.

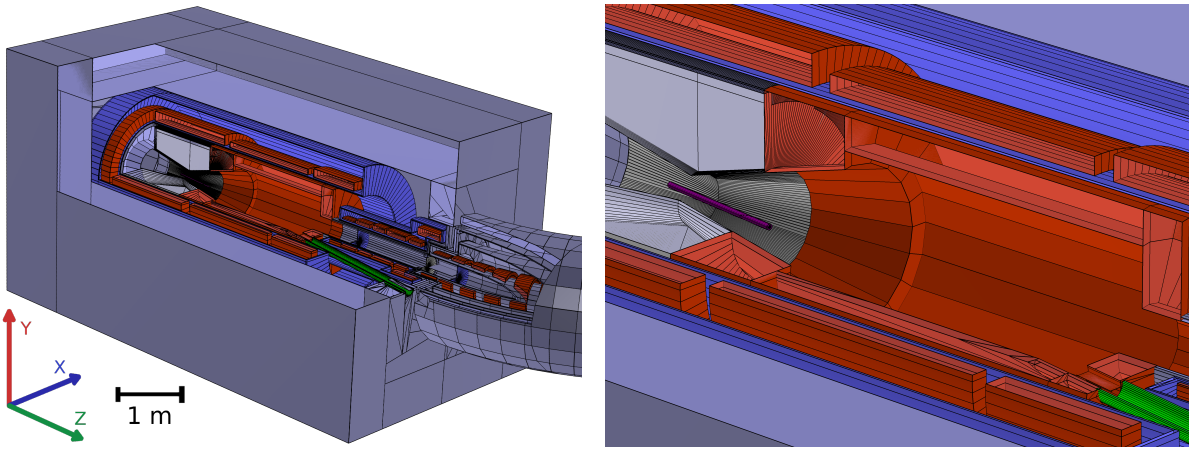
An 8 GeV proton beam will be used to maximise the pion yield while suppressing dangerous backgrounds from antiprotons. This beam will be pulsed to create high-intensity proton bunches that are well separated in time. The resulting muon beam mirrors this structure such that the muonic atoms are created as the muon bunches reach the Muon Stopping Target. The gaps between the bunches have a much lower background beam intensity, but maintain a large signal acceptance due to the 864 ns lifetime of bound muons in aluminium. This is discussed in more detail in Chapter 6. The key parameters for the proton beam are:

- **Energy**: 8 GeV
- **Protons Per Bunch**: 1.6×10^7
- **Bunch width**: 100 ns
- **Bunch separation**: 1.17 μs or 1.75 μs

3.1.2. The Pion Capture Section

Once inside the Pion Capture Section, the bunched protons hit the Pion Production Target to create an explosion of particles. This section is surrounded by a rectangular iron yoke, as shown in Figure 3.2a. The target itself can be seen in Figure 3.2b.

The target sits inside a solenoidal 5 T magnetic field that is designed to capture the backscattered muons and pions, as implied in Figure 3.1. This reduces the overall momentum of the captured beam, which reduces the chance of high-momentum electrons beam electrons. This also provides the pions with more time to decay to muons in-flight. The resulting muons are easier to stop, which increases the muonic atom yield in the Muon Stopping Target. Once the pions and muons are captured, they are directed through a



(a) The Pion Capture Section

(b) The Pion Production Target.

Figure 3.2.: The Pion Capture region, half-sectioned. These incoming protons travel through the bisected beampipe, shown in green, to hit the Pion Production Target, shown in pink. The target is surrounded by both the Pion Capture Solenoid, shown in red, and a considerable amount of shielding, shown in grey.

matching solenoid that brings the field strength down to 3 T and feeds the particles into the Muon Transport section. The important dimensions for the Pion Capture Section are:

- **Target Length**: 70 cm
- **Target Diameter**: 2.6 cm
- **Target Orientation**: -10° around y and -10° around z.
- **Yoke Length**: 642 cm
- **Yoke Width**: 350 cm
- **Yoke Height**: 345 cm

3.1.3. Curved Solenoids and Collimation

The Muon Transport Section is designed to select low-momentum muons and pions using curved solenoids, dipole magnets, and collimators. This is achieved through a *dispersion* effect. Beam dispersion, D, is defined as the amount a relative change in momentum, $\Delta p/p$, affects the trajectory of a particle, Δx:

$$D = \frac{\Delta x}{\Delta p/p} \tag{3.1}$$

Particles that are transported through curved solenoids disperse perpendicularly to the plane of the solenoid curvature. The magnitude of the dispersion, D, for a particle of charge q, transverse momentum $p_\perp$, longitudinal momentum $p_\parallel$, that travels a distance s through

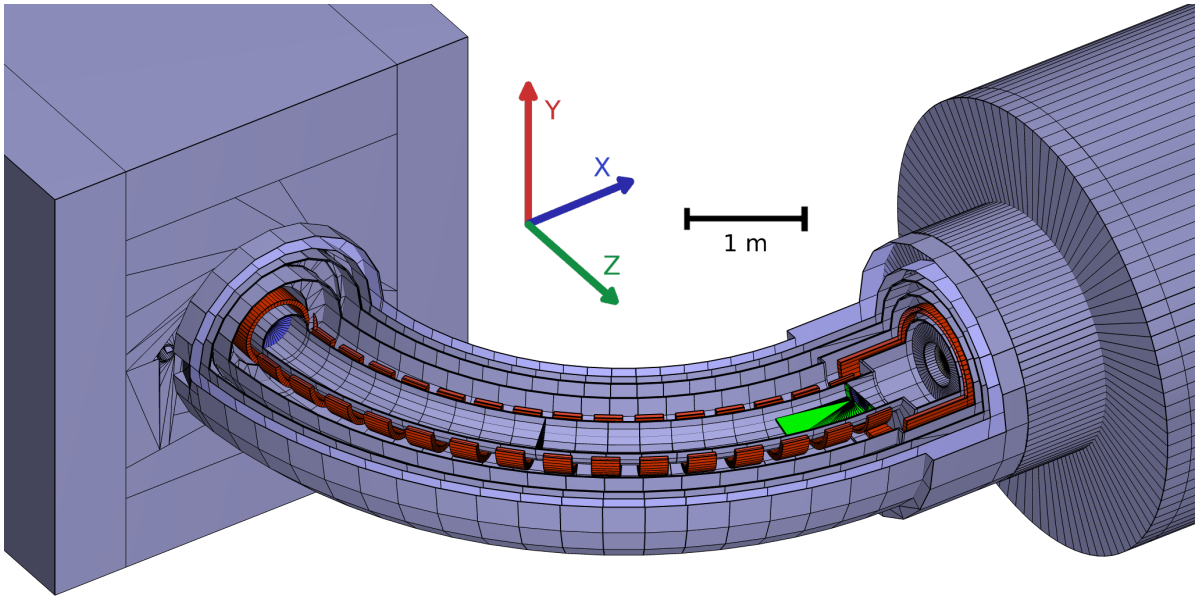


Figure 3.3.: The Muon Transport Section. This region is composed of solenoids, shown in red, and dipole magnets, which are not shown here. The collimator is placed before the Detector Solenoid and is shown in green. It has a symmetrically placed plane in the top half of this region, which is not shown here.

a solenoid with radius of curvature R and magnetic field strength B is:

$$D = \frac{1}{qB}\left(\frac{s}{R}\right)\frac{2{p_\parallel}^2 + {p_\perp}^2}{2p_\parallel} \tag{3.2}$$

The dispersion in the Muon Transport Section allows for separation of low and high-momentum particles. The plane of curvature is the x-z plane, meaning the higher momenta particles in the beam drift vertically in y. The beam is then collimated to reduce the high-momentum particles. To tune this selection, a constant dipole field is applied to evenly shifts the whole beam vertically. Changing the dipole fields allows different cross sections of the dispersed beam to be accepted by the collimators. The dipole field strength used in simulation is 50 mT, which results in the efficient transport of muons with $|\mathbf{p}| \approx 40$ MeV/c.

The Muon Transport Section for Phase-I can be seen in Figure 3.3. The curved section has a bending radius of 3 m. The 15 solenoids shown there keep the magnetic field strength at 3 T along the core of the curved beamline. At the end of this section sits a collimator formed of two horizontal plates so that the collimation occurs at the point of maximal dispersion.

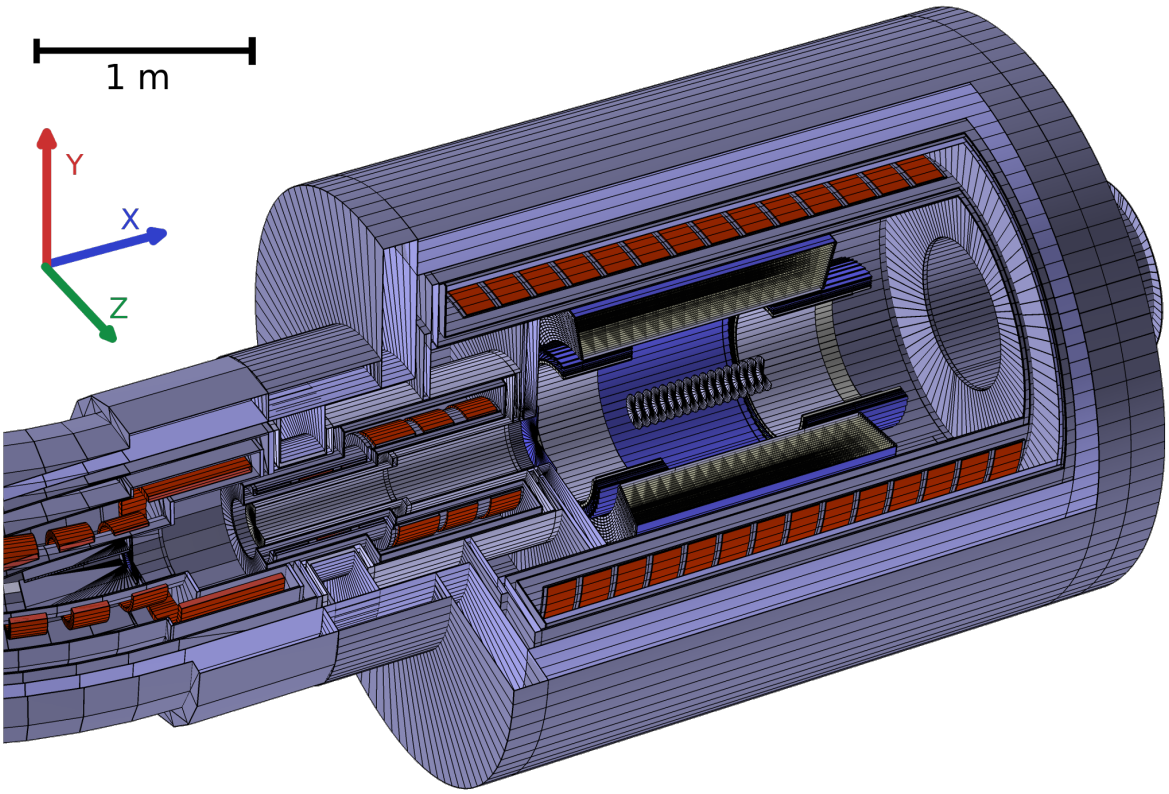


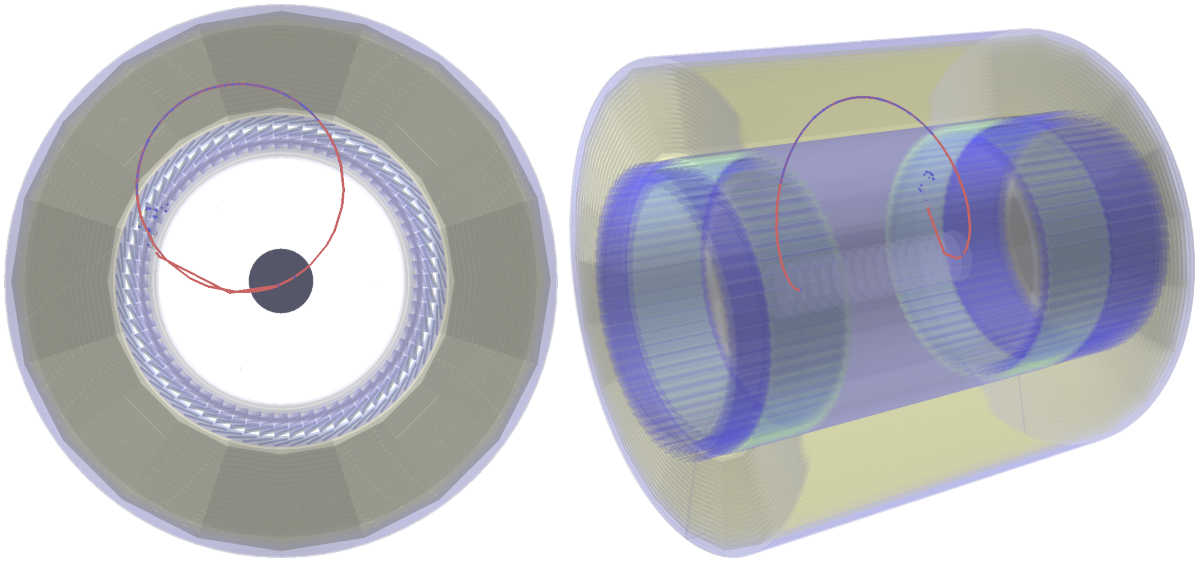
Figure 3.4.: The Cylindrical Detector System. The Muon Stopping Target disks are placed at the very center of the system, lie parallel to the x-axis, and are surrounded by the CDC and CTH.

3.2. The Cylindrical Detector

The Cylindrical Detector (CyDet) System will be used in Phase-I for the μ-e conversion measurement. It is shown in Figure 3.4. This system sits inside the Detector Solenoid and surrounds the Muon Stopping Target. The target and detector volumes share a common central axis, which is parallel to the incoming beam.

3.2.1. A Signal Event

Once inside the Detector Solenoid, the beam hits the Muon Stopping Target. This target sits inside the central cavity of the aptly named Cylindrical Drift Chamber (CDC). The CDC is the tracking detector for the CyDet and forms the focus of the track reconstruction algorithms in this thesis. This detector is accompanied by the Cherenkov Trigger Hodoscope (CTH). The CTH is made from two stations, one placed just inside the inner radius of the upstream end of the CDC and a similar station at the downstream end. These stations are



(a) Projected view onto zy-axes. **(b)** Rotated view.

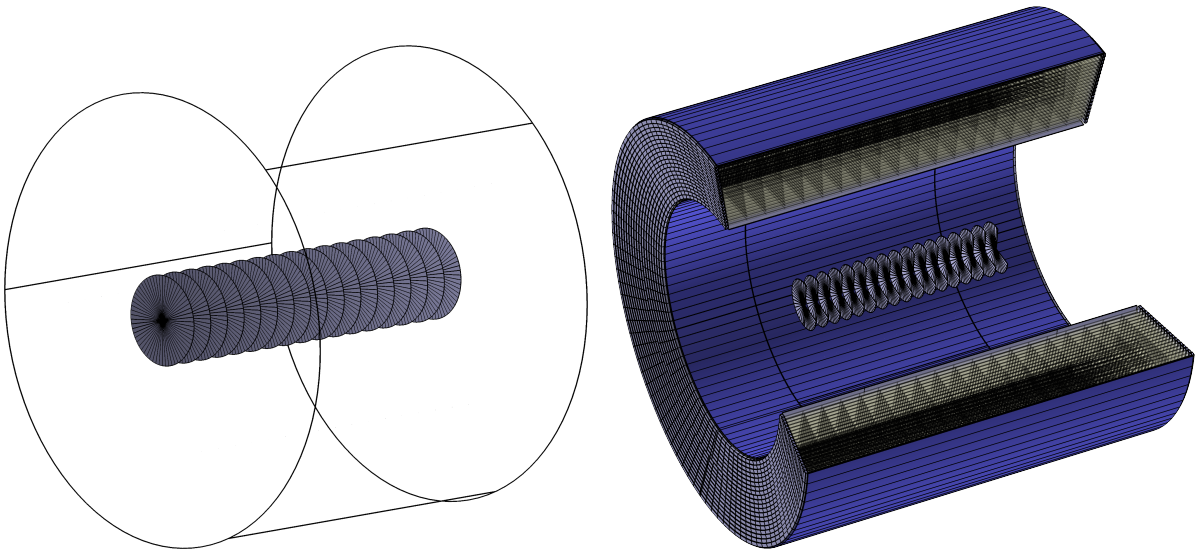
Figure 3.5.: An event display of a signal event in the CyDet. The signal electron track, shown in red, originates from one of the Muon Stopping Target disks. It passes through the CDC and causes a number of hits, shown in blue. It terminates in the downstream CTH station and causes a trigger signal.

designed to detect signal electrons after they pass through the CDC and exit the CyDet system, as seen in the ample event in Figure 3.5. As implied by the name, they form the primary trigger system for the CyDet.

3.2.2. The Muon Stopping Target

The Muon Stopping Target lies at the heart of the CyDet. This target seeks to strike a balance between two competing effects: maximising the number of stopped muons and minimising the multiple scattering of particles in the CyDet. The former prefers a higher-mass target, while the latter prefers less material inside the CDC cavity. A series of thin disks are currently used to satisfy both of these criteria, as seen in Figure 3.6a. Listed here are the key parameters for this region:

- **Number of Disks**: 17
- **Disk Radius**: 10 cm
- **Spacing Between Disks**: 5 cm
- **Disk Thickness**: 200 μm



(a) The Muon Stopping Target **(b)** The Cylindrical Drift Chamber

Figure 3.6.: The Muon Stopping Target. Figure 3.6a shows both the target disks and support frame that sit in the centre of the CDC, as seen in Figure 3.6b. Note each disk has three wires connecting it to the wireframe support that are too thin to be rendered at this scale.

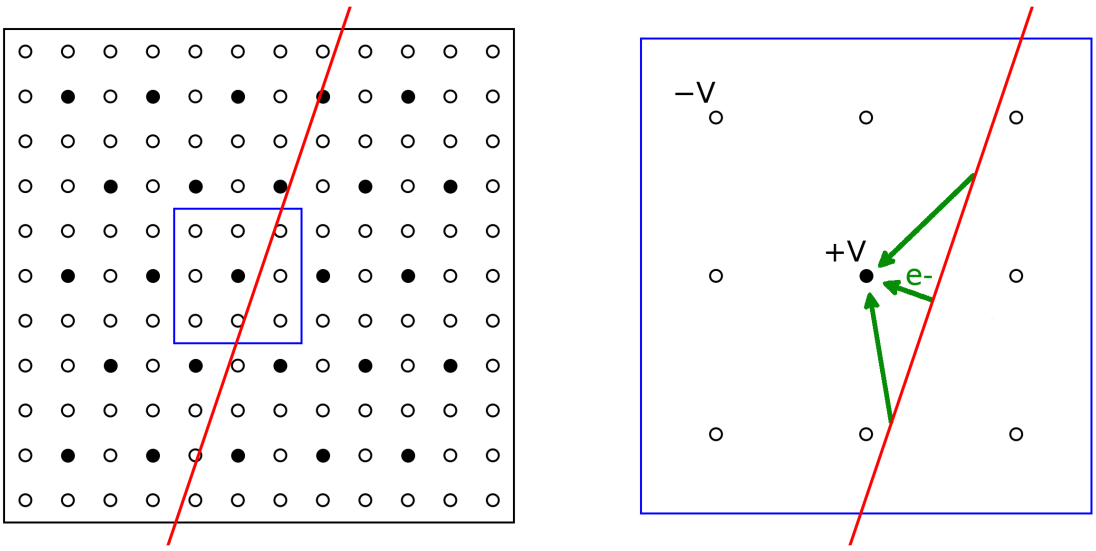
3.2.3. Cylindrical Drift Chamber

The CDC is a cylindrically symmetric wire drift chamber with a hollowed out cavity along its central axis that surrounds the Muon Stopping Target, as seen in Figure 3.6b. This section outlines the operating principles of wire drift chambers and details the relevant design elements of the CDC geometry.

Operating Principles

Wire drift chambers are constructed using an enclosed gaseous volume, cathode wires, and anode wires. The wires are placed in the gas so that each anode is surrounded by cathodes and hence forms its own local maximum in the electric field generated by the wires. This maximum attracts all electrons in the region between the anode and surrounding cathodes. This region is referred to as the "cell" of the anode wire. The anode wires are commonly referred to as sense wires, while the cathodes are referred to as field wires. This geometry is illustrated in Figure 3.7. The following steps lead to a readout hit in a given cell:

1. A charged particle passes through the gas volume, ionizing some of the gas molecules.
2. The freed electrons are accelerated towards the closest sense wire by the electric field in the cell, as shown in Figure 3.7b.



(a) An example of a wire drift chamber.

(b) The highlighted cell in the chamber.

Figure 3.7.: A schematic that highlights the operating principles of a wire drift chamber. Figure 3.7a illustrates the layout of anode and cathode wires across a simplified wire drift chamber. The anode wires are shown as the filled circles, while the cathodes are the open circles. The particle trajectory is shown in red. A single cell is highlighted in the blue box, and show in detail in Figure 3.7b. Inside the cell, the particle ionizes the gas to create the electrons, shown in green, that are accelerated towards the anode wire.

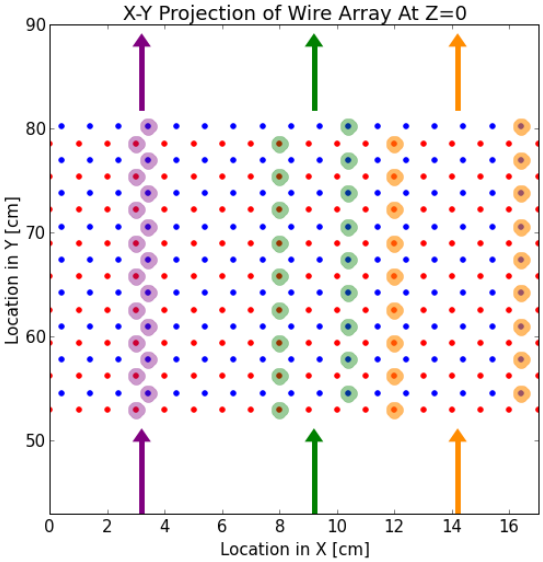
3. As the electrons approach the wire, they gain enough energy to ionize the gas to create more electrons. This causes an avalanche that amplifies the original signal.

4. As the avalanche approaches the wire, it induces a pulse of current in the wire which flows towards the ground.

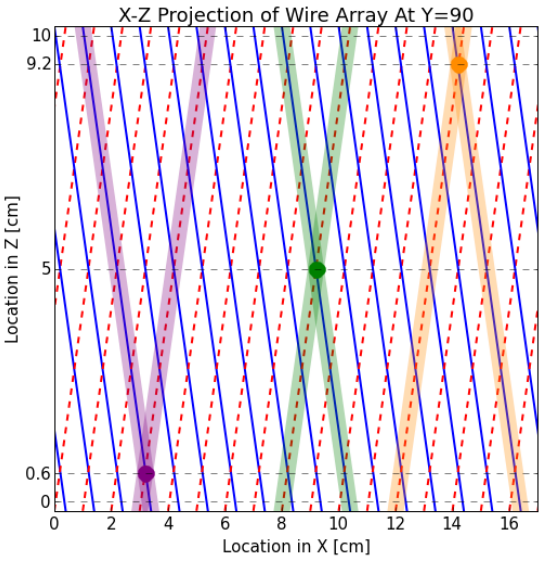
5. This induced pulse is detected by readout hardware at the end of the wire.

Hits are defined by the position of the sense wire and the time that the pulse was readout, t_{RO}. The hit position where the charge track first ionized the gas is inferred using these two pieces of information. This requires determining the distance between the hit position and the wire, which is referred to as the "drift distance". This can be calculated as a function of the drift time, t_{D}, which is the time it takes for the ionization electrons to drift from the initial hit position to the sense wire. The drift time can be recovered using the recorded time of the induced pulse t_{RO}, the time the wire passes through the cell t_{Hit}, and the time it takes for the signal to pass through the data acquisition (DAQ) hardware t_{DAQ}:

$$t_{\mathrm{D}} = t_{\mathrm{RO}} - t_{\mathrm{Hit}} - t_{\mathrm{DAQ}} \tag{3.3}$$



(a) The readout projection of the wire chamber.



(b) The "hidden" projection of the wire chamber.

Figure 3.8.: A demonstration of how stereo angles can recover 3D hit positions. In both figures, the blue layers are rotated around y by a stereo angle θ, while the red layers are rotated around y by -θ. This example contains no super-layers. Both images show three colour-coded particles passing through the chamber that leave colour coded hits.

Typically, t_{Hit} is measured using a detector with a fast timing response like a scintillator strip, where the time of flight between the wire cell and the scintillator is negligible. The DAQ time t_{DAQ} can be measured and accounted for during calibration. Finally, t_{RO} is recorded for each hit.

With the drift time calculated, the resulting values for the drift distance are provided to the track fitting algorithm. This fitting algorithm then finds a track that passes through each cell at the appropriate drift distance from the sense wire. In the presence of a magnetic field, the momentum of the particle transverse to the beam can be recovered from this fit:

$$p_{\perp} = qBR \tag{3.4}$$

where B is the magnitude of the magnetic field, q is the charge of the particle, and R is the radius of curvature of the trajectory in the transverse plane.

Stereo Angles

Wire chambers can naturally describe the locations of hits in the perpendicular plane to the wire direction, but this can only provide a $p_{\perp}$ measurement in the CDC. Stereo layers are

used to provide information about the third dimension of the hit location, which is used to measure $p_{\parallel}$. For the purposes of this explanation, consider the x-y plane to be parallel to the readout plane and z to be perpendicular to the x-y plane.

In a wire drift chamber, stereo layers are layers where the orientation of the wires deviates slightly from the z-direction. The angle of deviation is referred to as the "stereo angle" and is typically a few degrees. These layers are normally used in conjunction with "super layers", however, COMET has opted to use only stereo layers in the CDC to improve the $p_{\parallel}$ resolution. The COMET design resembles Figure 3.8, where the sign of the stereo angle alternates by layer.

Figure 3.8 uses a model drift chamber to demonstrate how stereo layers are used to recover information about the z-position of the hit. Figure 3.8a shows the readout projection of this chamber. While the y position of the hits is unaffected by the stereometry, the x positions of the hits alternate by stereo layer for a given track. The disagreement between the two sets of layers encodes information about the z-position of the hit, as shown in Figure 3.8b. Here, we can see that the purple track intersects the drift chamber at $z = 0.6$ cm, which is close to the readout plane at $z = 0$ cm. This means that the readout plane sees a correspondingly small shift in x between the two set of layers. The green track intercepts at $z = 5$ cm, causing a larger disagreement in x at the readout plane, while the orange track at $z = 9.2$ cm causes the largest disagreement in x.

The particles in this demonstration have $p_{\parallel} = p_z = 0$, which means the disagreement in x between layers in different y positions is constant. For $p_{\parallel} \neq 0$, this disagreement will shrink or grow. By quantifying the change in disagreement between layers, a measurement for $p_{\parallel}$ can be achieved.

Geometry Description

The CDC is designed to measure the total momentum of 105 MeV electron tracks with a resolution of 200 keV or better. It is composed of 19,548 wires in total across 39 layers. The outer two most layers are guard layers with both anode and cathode wires. While they are similar to the sense layers, they are not read out. Instead, these layers collect any space charge that builds upon the inner and outer walls of the CDC. The remaining 37 layers alternate between field and sense layers, with 19 field layers and 18 sense layers.

The 18 sense layers contain 4,482 sensitive channels. All of these layers are exhibit a stereo angle with a magnitude between 64 and 75 mrad. These angles are achieved by

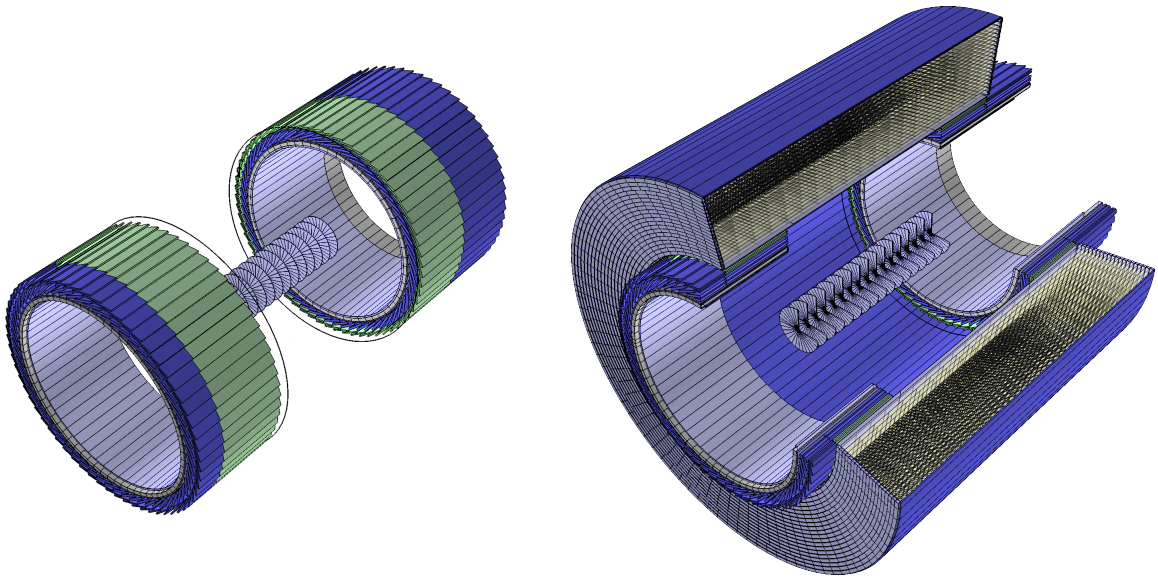
angularly displaying the origin and the end point of each wire so that the wires are no longer coaxial with the CDC. The sign of stereo angle, or angular displacement, alternates between each layer, causing two stereo projections to arise from any given track. This configuration is chosen to ensure a longitudinal spatial resolution of 3 mm, which is needed to ensure the overall resolution of 200 keV. The CDC is also designed to ignore low momentum background particles. This is achieved by placing the CDC in a 1 T field and offsetting the instrumented volume from the target so that the CDC has an inner radius of 49.65 cm. Noting that the Muon Stopping Target disks have a radius of 10 cm, equation (3.4) can be used to calculate the minimum $p_\perp$ needed for a decay electron reach the CDC as $p_\perp \sim 65$ MeV. A summary of the CDC parameters is found in Table 3.1 for reference.

3.2.4. Cherenkov Trigger Hodoscope

The Cherenkov Trigger Hodoscope (CTH) provides the trigger signal for the CyDet system. It is composed of two cylindrically symmetric stations that line the inside of the upstream and downstream ends of the CDC, as seen in Figure 3.9. Each station is formed from an inner layer of Cherenkov counters and an outer layer of scintillator strips. Together, these layers are used to provide a trigger signal. This signal is formed from the tight-coincidence of four modules, as explored later in Section 6.1.

Table 3.1.: Key parameters of the CDC.

Inner wall	Length	1495.5 mm
	Radius	496.0 ∼ 496.5 mm
Outer wall	Length	1577.3 mm
	Radius	835.0 ∼ 840.0 mm
Sense wires	Material	Au plated W
	Diameter	25 μm
	Number of active channels	4482
	Number of guard channels	504
Field wires	Material	Al
	Diameter	126 μm
	Number of wires	14562



(a) CTH with Muon Stopping Target. **(b)** CTH with Muon Stopping Target and CDC.

Figure 3.9.: The Cherenkov Trigger Hodoscope for Phase-I. The CTH stations surround the upstream and downstream ends of the Muon Stopping Target, as shown in Figure 3.9a. The stations sit inside the CDC, as shown in Figure 3.9b.

Operating Principles

When radiation passes through a scintillator, it excites the scintillator's component atoms which then deexcite and emit scintillation light. This light is guided to a photomultiplier tube where it is converted to an electric charge pulse, which is then further amplified by the electronics system. Scintillators have several attractive properties as detectors:

- They have a fast response and recovery time relative to other detectors, which offers superior timing resolution.
- Above a certain threshold, they give a linear response to the amount of energy deposited.

Due to these properties, scintillators are some of the most widely detectors in particle physics. [50].

Cherenkov counters rely on Cherenkov radiation to detect incident particles. Cherenkov radiation occurs when a particle moves through a medium faster than light moves in the same medium. This relationship is given by:

$$\beta = \frac{v}{c} > \frac{1}{n} \Longrightarrow v > \frac{c}{n} \tag{3.5}$$

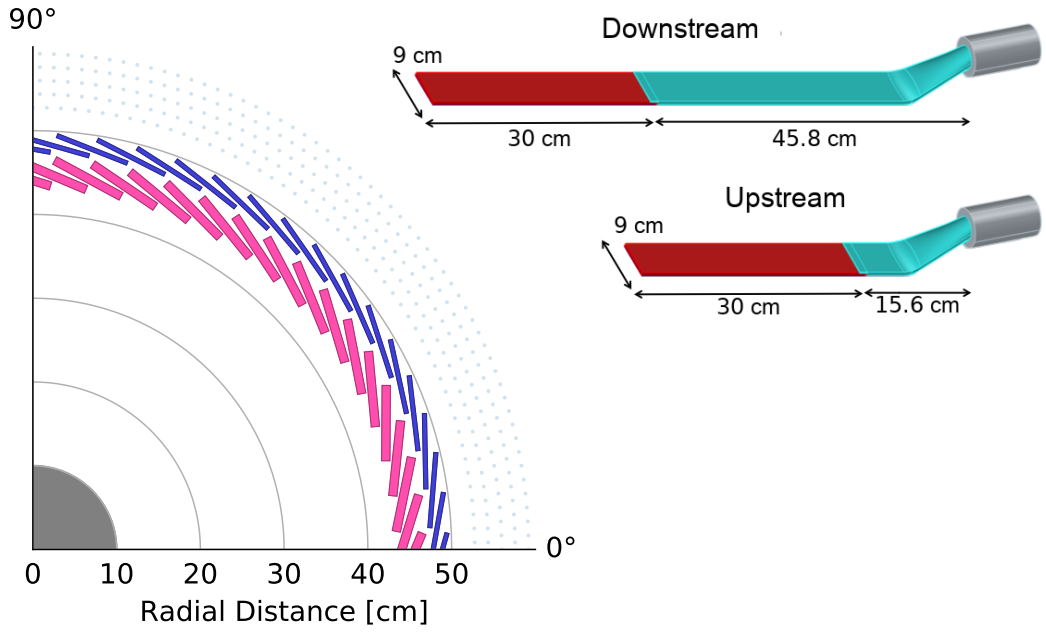


Figure 3.10.: The Cherenkov Trigger Hodoscope schematics. The left image shows the projection the CyDet down the beamline axis, but cuts the CDC off at the fifth layer. The cross-section of the scintillators are shown in blue, while the Cherenkov counters are shown in pink. This projection hides the fact that there are two CTH stations, best seen in Figure 3.9. The right image shows the different dimensions of the CTH modules at the upstream and downstream stations, with the red parts showing the active volumes and the blue part showing the light guides.

where v is the speed of the particle, n is the real part of the material's refraction index, and c is the speed of light in a vacuum [51]. When this criterion is satisfied, the particle generates an electromagnetic shock wave, similar to sonic shock waves created by objects travelling faster than the speed of sound. Similar to the scintillator case, the photons generated in this process are guided to a photomultiplier tube where they are converted to an electric current, which is then amplified and readout. Cherenkov counters have several attractive properties as detectors:

- They have a fast response and recovery time, just like scintillators
- They are able to ignore non-relativistic particles and only detect particles based on their overall velocity. This makes them more sensitive to electrons than protons for a fixed energy.

Geometry Description

The CTH is designed to detect the signal electrons as they exit the CDC. It is composed of two distinct stations, as seen in Figure 3.9. Each station is formed from an inner layer of 64 Cherenkov counters and an outer layer of 64 scintillators. Within each layer, the detector modules are tilted slightly and shifted by a half-length so that neighbouring modules overlap. This configuration and be seen into the left in Figure 3.10.

The Cherenkov counters and scintillators modules are right rectangular prisms of similar dimension. The Cherenkov counters are 9 cm × 30 cm × 1 cm, while the scintillators are 9 cm × 30 cm × 0.5 cm. As seen in both Figure 3.9 and Figure 3.10, the longest side of these modules are aligned with the central axis of the CDC. The scintillators are made from a plastic called polyvinyltoluene (PVT). The Cherenkov counters are made from acrylic.

Both the scintillators and Cherenkov counter modules are optically coupled to a photo-multiplier tube (PMT) via an acrylic light guide. This configuration can be seen to the right in Figure 3.10. To maximise the acceptance, the upstream light guides are 15.6 cm long, while the downstream light guides are 45.8 cm light guides long.

Chapter 4.

Simulating the COMET Experiment

Over 1.09×10^{12} proton-on-target (POT) events were simulated to develop the novel algorithms introduced in this thesis. This equates to around 68,000 Phase-I bunch-events. These data are used to model the beam related background and the stopped muons that create the signal particles. This chapter outlines the software used to simulate the experiment, describes physics processes included in the simulation and explains the resampling techniques used to improve the statistical significance of the data sample.

4.1. The ICEDUST Software Framework

The Integrated COMET Experiment Data User Software Toolkit (ICEDUST) is the COMET experiment's official software framework. While questionably named, it has been rigorously designed to handle and analyse COMET data. At its core, it is designed to handle simulation data and real data using the exact same software algorithms. The code base itself was copied over from the existing framework of the T2K experiment's off-axis near detector, ND280 [52], and modified to suit COMET needs. The overall data flow can be seen in Figure 4.1.

Simulations of the hadronic physics inside the Pion Production Target can be handled by a host of simulation packages, including PHITS, MARS, Fluka, and Geant4 [53, 54, 48]. The simulation starts with the equivalent of one POT event. Details of the resulting particle flux out of the Pion Production Target are read into a common data format defined by the oaRooTracker package. The SimG4 package then reads these data and tracks the particles through the beamline components. SimHitMerger is used to combine the data from many events into one event and impose the expected pulsed structure. The data are then read by

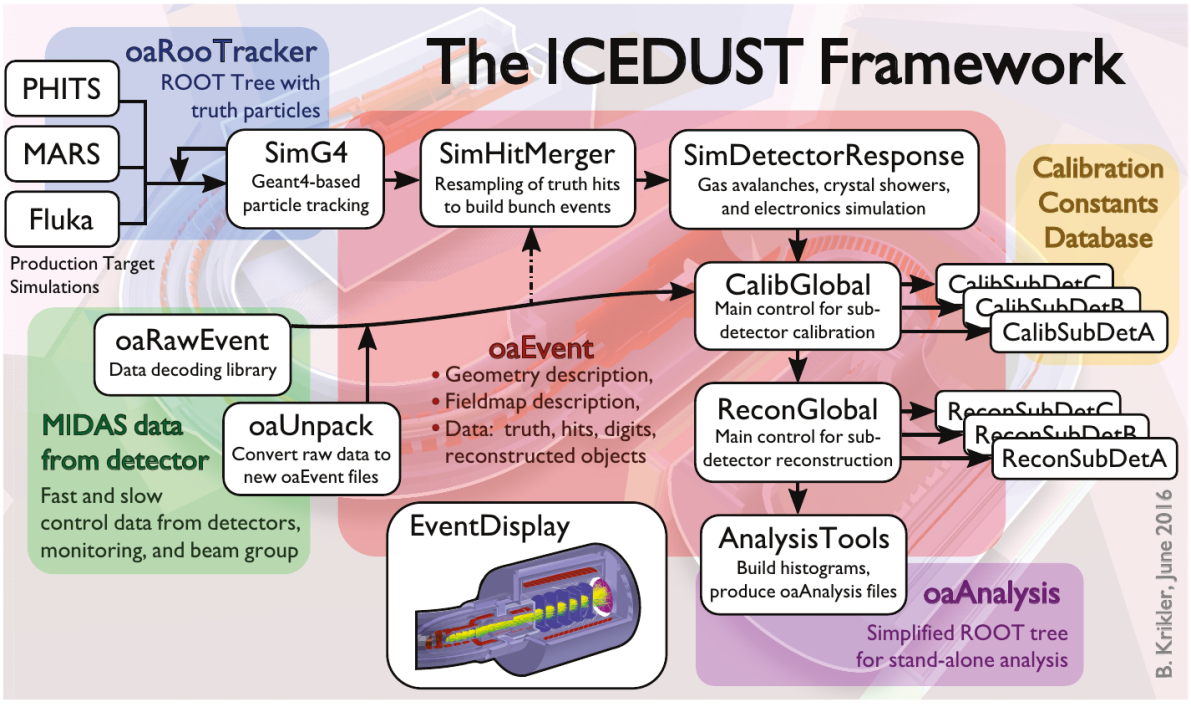


Figure 4.1.: The ICEDUST data flow model, including both real data and simulated data. This work focuses on the SimG4, SimHitMerger, and SimDetectorResponse stages [38].

SimDetectorResponse, which creates data structures that mimic those produced during the real experiment.

Real data will be stored in the MIDAS file format. This data will be converted using the oaRawData and oaUnpack packages to create the data structures mimicked by SimDetectorResponse. From here, both real data and simulation data are calibrated, reconstructed, and analysed.

The data flow in ICEDUST is built around the oaEvent data format. This file format is based around a ROOT tree with one branch, where each entry stores the data from one COMET event. Each COMET event can store an arbitrary amount of POT events. At each stage simulation, reconstruction, and analysis, relevant data from the input oaEvent file can be copied into the output oaEvent file, which ensures the full legacy of the data can be accessed at any point.

In an effort to ensure that the mass-produced simulation has no critical bugs, a thorough testing suite has been added to ICEDUST. This testing suite builds upon existing tests from ND280 for the oaEvent package and adds tests for oaRooTracker, SimG4, oaGeomInfo[1],

[1]This package is used to store and access information about the COMET geometry, most notably about subdetector systems like the CDC.

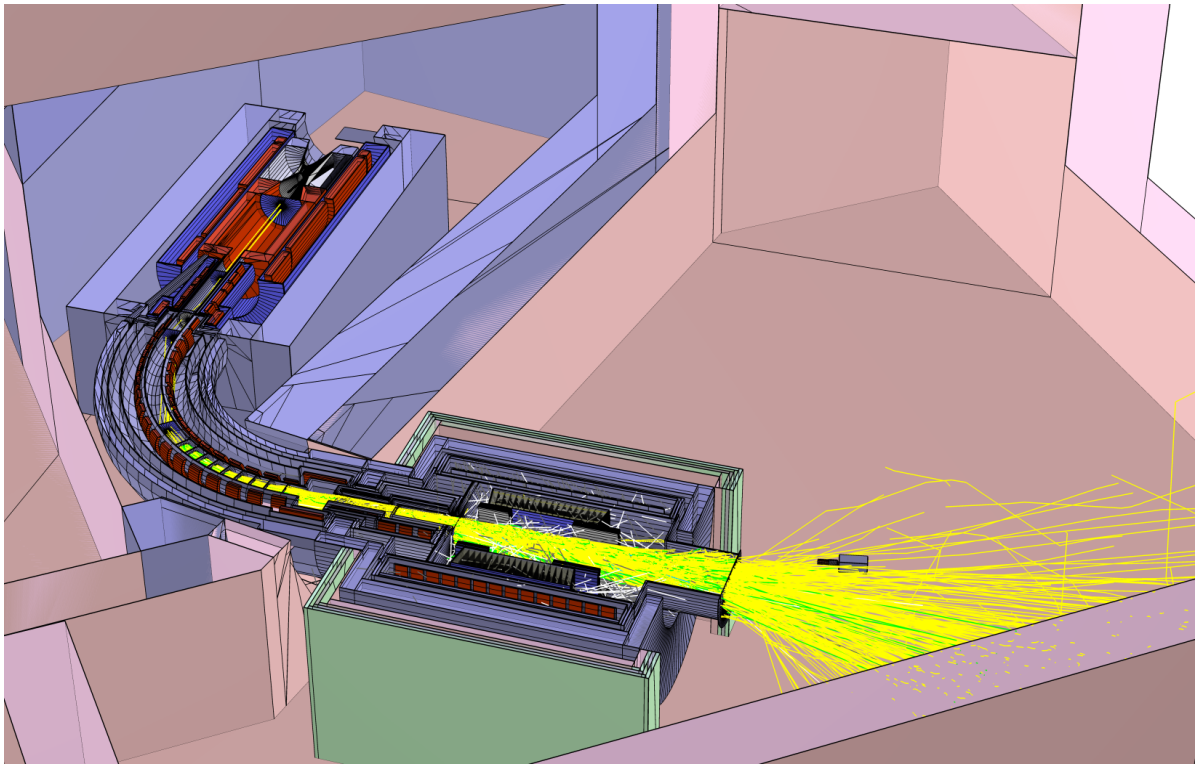


Figure 4.2.: Phase-I event display of 8e6 proton-on-target events, one half of a bunch, in the CyDet system. This image also includes the experimental hall in which the Phase-I beamline is placed. Only the trajectories that reach the Detector Solenoid are rendered.

SimHitMerger, and SimDetectorResponse. These tests run automatically inside the existing development workflow whenever changes to the official copy of the code are under review. For more information on this workflow, see Appendix A.

4.2. Running the Simulations

Simulating the COMET experiment is challenging. At the core of this challenge lies the fact that COMET is a high-intensity experiment that is designed to generate many more particles than it will detect. The particles that are detected must first traverse a gauntlet of complicated beamline components through an equally complicated magnetic field. Even worse, these particles are otherwise indistinguishable from their less fortunate counterparts that do not make it to the detector region. Protons that miss the production target entirely to instead hit some element of the shielding can still create particles that are later detected.

This forces the simulation to mirror the experiment's high-intensity environment and simulate far more particles than actually create hits in the detector. Despite efforts, no cuts in the simulated particle phase space could be found that would improve simulation time

without sacrificing a potential source of background, as demonstrated later in Section 6.2. To ensure a high-quality data sample for the tracking studies, all particles resulting from the initial proton events were simulated. A visualization of the simulation can be seen in Figure 4.2.

The samples created for this study are the first large scale production simulation samples produced for COMET Phase-I using ICEDUST. As such, they prioritise model accuracy over computational performance. While expensive, this means they can serve as a benchmark for later computational optimization. Furthermore, the machine-learning elements of this work require high statistics samples that are statistically independent of each other. This section outlines the simulation flow that was designed to meet both of these requirements. Summaries of the resulting samples are presented in Appendix C.

4.2.1. The Physics Models

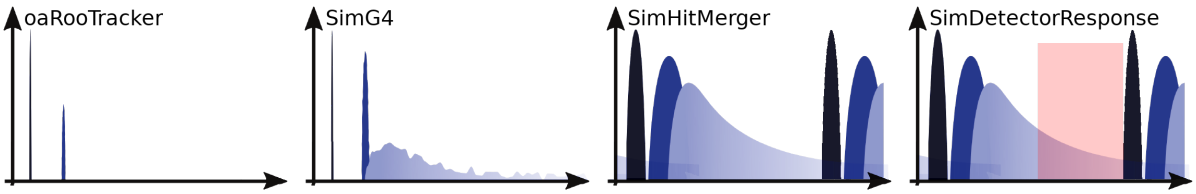
These samples were created using Geant4 to simulate both the interactions inside the Pion Production Target and the transportation of the particles through the rest of the experiment. When using Geant4, the user is responsible for registering all relevant physics processes that will be simulated in the "physics list". For ICEDUST, the QGSP_BERT_HP list was taken as a starting point. This model is intended to give high accuracy for low-energy physics involving neutrons [48]. The list was adapted to include all relevant and potentially relevant phenomenology. Custom physics models were implemented for both bound muon decay and muon capture. Further details for both of these are provided in Appendix B.

4.2.2. From Initial Proton to Detector Response

There are four main steps to ICEDUST simulations:

- Reading the input using oaRooTracker,
- Simulating the events in SimG4,
- Combining the results into a bunch-like timing structure in SimHitMerger,
- Simulating the detector response in SimDetectorResponse.

The simulation starts by propagating the 8 GeV protons through the proton beamline that feeds the COMET experiment using the TURTLE software framework [55]. The results of this simulation are sampled just as the protons enter the Pion Production section and saved in an oaRooTracker file, as depicted by the black peak in Figure 4.3a.



(a) Primary particles for one event are created.

(b) The particles create hits in detectors.

(c) Many POT events are combined into bunch-trains.

(d) Detector hits in the time window are simulated.

Figure 4.3.: These diagrams illustrate the evolution of simulation. All these cartoons are histograms with time as their x-axis. The black peaks represent the proton beam, the dark blue peaks represent the muon beam, the light blue peaks represent the hits generated in the detector, and the red highlights the fiducial time window.

SimG4 reads the description of the input protons from this file to continue the simulation. In this section, each event corresponds to one input proton. The protons can do one of two things:

- scatter off of the pion-production target or,
- miss the pion-production target and collide with the beam dump.

The resulting secondaries from either case are simulated in full by Geant4 through the Muon Transport section and into the detector region. The hits generated in SimG4 describe the time and magnitude of all energy deposits in the detectors. They are referred to as "truth" hits. The timing structure of these hits is shown in Figure 4.3b.

SimHitMerger combines the individual POT events simulated by SimG4 to form the bunch-like timing structure discussed in Section 3.1.1. First, bunch-like events are created by shifting the timing of each of the 1.6e7 POT events in the bunch by a random value in the range $[-50\text{ns},\ 50\text{ns}]$ and combining the results. The pulsed beam is simulated using a handful of bunch-events, where each is separated by 1170 ns to form “bunch-train” events. The timing structure can be seen in Figure 4.3c and discussed in detail in Section 6.3.

SimDetectorResponse simulates the physics processes inside the detectors that result in detected hits. This includes gas avalanche simulations within the CDC and gain processes in the CTH. These processes are only simulated within a fiducial time window. This section outputs both the ADC waveforms and descriptions of calibrated detector hits. The relevant fiducial time window is depicted in Figure 4.3d.

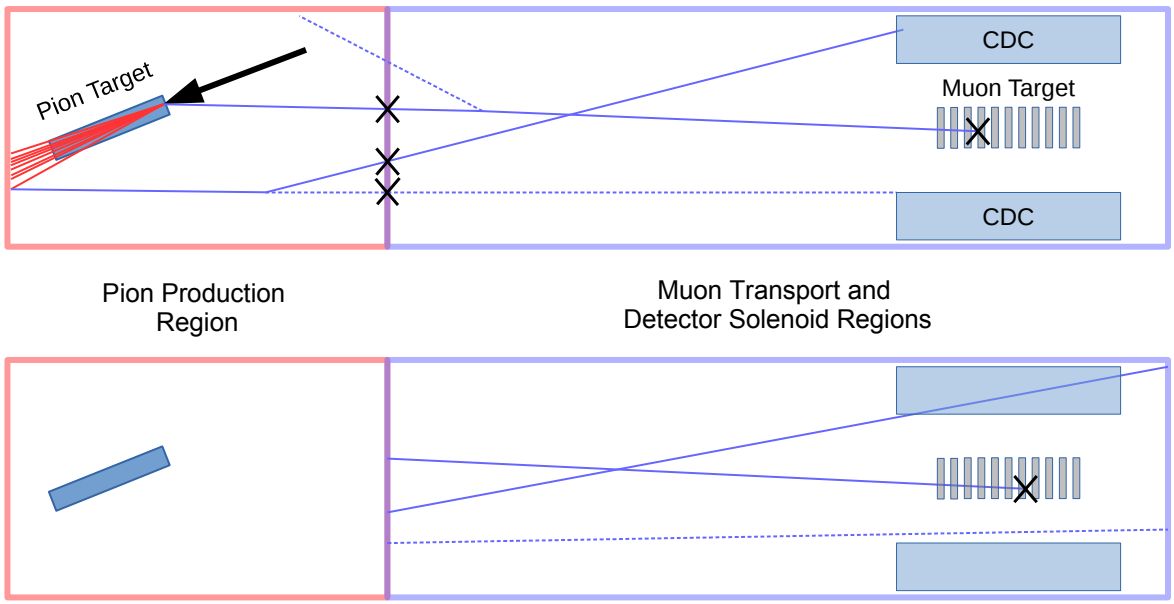


Figure 4.4.: A schematic that illustrates the resampling technique used in the simulation. The upper image shows an event where the RooTracker histogram is generated. The primary proton is represented by the arrow, the secondaries that hit the beam dump are shown in red, and the particles that make it into the region of interest are shown in blue. The sampling points for the histogram is shown by the black X's. The lower image shows the resampling of the same event.

4.2.3. Resampling Techniques

The detailed physics list and full simulation of beam dump particles cause over 99% of the simulation time to be spent in the Pion Production section of the experiment. The simulation spends the other 1% of the time simulating particles in the Muon Transport and Detector Solenoid sections, which is the "region of interest" for track reconstruction.

To correct this imbalance, particle four-momenta and four-position are recorded as they enter this region of interest in Geant4. This information is used to make a RooTrackerHistogram file, which contains two histograms for each particle type:

- an 8D histogram which records the four-momentum and four-position information for the given particle type, and
- a 1D frequency histogram which records how many of the given particle type occurred in each event.

The simulation can then be restarted from this file to only simulate the particles in the regions of interest. First, each frequency histogram in the file is randomly sampled to

determine the number of each particle needed to build a POT-equivalent event. For each of these particles, the corresponding 8D histogram is randomly sampled to determine its initial properties.

This resampling technique gives the particles that reach the region of interest multiple chances to reach the detector, which provides fundamentally new information while skipping the first 99% of the simulation. Figure 4.4 illustrates this process, highlighting the fact that sampling for the histogram carefully avoids double counting.

4.2.4. Signal Simulation

Since the signal is extremely rare, there is no chance of it occurring "naturally" in the simulation. Instead, the signal simulation uses a similar resampling technique to the background simulations. First, the four-positions of all stopped muons in the Muon Stopping Target are recorded in a RooTrackerHistogram. This process is represented by the black X's on the Muon Stopping Target in Figure 4.4. The resulting RooTrackerHistogram is sampled to determine the initial position of the signal electrons.

The initial time of the signal electron is then smeared by the distribution of the bound muon lifetime in aluminium. This distribution is described by an exponential decay with a characteristic lifetime of 864 ns. With the time established, these electrons are given a momentum with a magnitude of 105 MeV and a random initial direction. The signal tracks are then overlaid on background bunch-train events to create a combined signal and background event.

Chapter 5.

Machine Learning and Computer Vision Methods

The last decade has seen a dramatic revival of machine learning across both academia and industry. This revival has been catalysed by several key developments, including:

- the exponentially growing amount of data generated in everyday life,
- the ability to distribute this information over the internet, and
- the analytical power offered by modern computers.

In this context, machine learning is often used to extract meaningful information or results from vast collections of data. Experimental particle physics has a long history of extracting physics principles by measuring rare events. In doing so, particle physicists are constantly pushing the boundaries of data collection and analysis. Machine learning methods are gaining traction in the field, as summarised in [56, 57]. This chapter will introduce some of the core principles of machine learning, as reviewed in [58]. From here, the algorithms that form the basis of the track reconstruction in this thesis are introduced.

5.1. Fundamentals

The aim of the machine learning discussed in this thesis is to build a function f that can predict the target property of a given datum, y, using a vector of that datum's other properties, $\mathbf{x}$. This can be formalised as:

$$f(\mathbf{x}, \boldsymbol{\theta}) = \hat{y} \tag{5.1}$$

The known properties that make up the vector $\mathbf{x}$ are referred to as the "input features." The function f is constructed by the machine learning algorithm and referred to as the "predictor" in this section. This function is parametrised by a vector of parameters, $\boldsymbol{\theta}$, and evaluates to a prediction, $\hat{y}$, of the true value of the target property, y. Constructing the predictor normally means determining the optimal set of parameters $\boldsymbol{\theta}$.

5.1.1. Training and Testing a Supervised Learning Algorithm

In general, there are two families of learning algorithms:

- *Supervised Learning*: The value of the target variable, y, is available for all elements in the data. This means that the true value of y can be used to guide the algorithm to build an accurate predictor.
- *Unsupervised Learning*: The value of the target variable, y, is not available for elements in the data. These learning models are usually used to cluster or organise data.

The algorithms discussed here supervised algorithms that are trained and tested on simulation data, where the target y value is readily available[1]. This process can normally be divided into two stages:

1. *Training*: The machine learning algorithm fits the function f to a data sample called the "training" data. It is given both $\mathbf{x}$ and y to determine the values of $\boldsymbol{\theta}$ in f that best describe the relationship between $\mathbf{x}$ and y.
2. *Testing*: The algorithm is provided with a new data set called the "testing" data. The predictor is evaluated with the $\boldsymbol{\theta}$ values determined in training to make a prediction for each element in the testing data set. The accuracy of these predictions is used to judge the performance of the predictor.

From this structure arises one of the first tenets of machine learning: training and testing data must be statistically independent. This is to ensure that the predictor can generalise what it learns from the training set to accurately predict properties of as-of-yet unseen data members in the testing set.

[1] With that said, unsupervised learning is a very attractive prospect for real data, which often come as an abundance of unlabelled data.

5.1.2. The Loss Function, Regularisation, and Gradient Descent

Both the training and testing stages require some measure of how well the algorithm is performing. This is determined using a "loss" function, which describes the disagreement between the value of the target variable y and its predicted values $\hat{y}$ across the whole data set as a single number. A prime example of a loss function is the mean squared error:

$$E(Y,\hat{y}) = \frac{1}{2N}\sum_{i=1}^{N}\left(\hat{y}^{(i)} - y^{(i)}\right)^2$$
$$E(Y,X,\boldsymbol{\theta}) = \frac{1}{2N}\sum_{i=1}^{N}\left[f\left(\mathbf{x}^{(i)},\boldsymbol{\theta}\right) - y^{(i)}\right]^2 \tag{5.2}$$

On the left hand side, the function E is the loss function and the capital Y and $\hat{Y}$ refer to the target y values and their corresponding $\hat{y}$ predictions across all elements in the data set. The capital X is the matrix of all features for all data elements. The right hand side averages the mean squared error from all N elements in the data set, where the superscript denotes the i-th data element[2].

The parameters $\boldsymbol{\theta}$ are iteratively updated using the training set. This process utilizes a method called "gradient descent", which uses the differential of the loss function with respect to each element in $\boldsymbol{\theta}$ to update $\boldsymbol{\theta}$ so that the loss is minimized. Using mean squared error as an example:

$$\theta_j := \theta_j - \alpha\frac{\partial}{\partial\theta_j}E(Y,X,\boldsymbol{\theta}) \tag{5.3}$$
$$:= \theta_j - \alpha\frac{\partial}{\partial\theta_j}\frac{1}{2N}\sum_{i=1}^{N}\left[f\left(\mathbf{x}^{(i)},\boldsymbol{\theta}\right) - y^{(i)}\right]^2$$
$$\theta_j := \theta_j - \frac{\alpha}{N}\sum_{i=1}^{N}\left[f\left(\mathbf{x}^{(i)}\boldsymbol{\theta}\right) - y^{(i)}\right]\frac{\partial}{\partial\theta_j}f\left(\mathbf{x}^{(i)},\boldsymbol{\theta}\right) \tag{5.4}$$

In these equations, α is referred to as the learning rate. It controls the speed of convergence to help the loss function reach the global minimum. This update rule recycles the residuals needed to calculate the loss, which makes the mean square error attractive loss function.

As with all optimisation problems, local minima must carefully be avoided. In machine learning, this often occurs when an algorithm finds a minimum that is unique to the training set. This problem is referred to as "overfitting". To combat overfitting, a technique called

[2]The leading value of $1/2$ is a convention used in machine learning to ease computational cost of training.

"regularisation" is often used. A simple example below demonstrates how this can be achieved:

$$J(Y,X,\boldsymbol{\theta}) = E(Y,X,\boldsymbol{\theta}) + \frac{\lambda}{2}\sum_{j=1}^{M}{\theta_j}^2 \tag{5.5}$$

This equation introduces the loss function J which is the sum of the training loss E and the new regularisation term. This new term penalises the model for relying too heavily on any one parameter θ_j, which encourages the model to have a smoother response overall. The parameter λ is introduced to control the strength of this penalty. The update rule can then be defined using the function J in place of E in equation (5.3) to recover:

$$\theta_j := \theta_j - \alpha\frac{\partial}{\partial\theta_j}J(Y,X,\boldsymbol{\theta}) \tag{5.6}$$

$$\theta_j := \theta_j(1+\lambda) - \frac{\alpha}{N}\sum_{i=1}^{N}\left[f\left(\mathbf{x}^{(i)},\boldsymbol{\theta}\right) - y^{(i)}\right]\frac{\partial}{\partial\theta_j}f\left(\mathbf{x}^{(i)},\boldsymbol{\theta}\right) \tag{5.7}$$

5.1.3. Classification and the Logistic Function

Supervised learning problems can be divided into two main categories: "regression" and "classification". Regression refers to cases where y is a continuous variable, like particle momentum. Classification refers to cases where y is a discrete variable. The simplest case is called binary classification, where y takes a boolean value, like "signal" or "background".

Binary classification, which is sometimes called logistic regression, can be treated as an extension of the normal regression example discussed so far. In this case, an additional function, g, is applied to the outcome of the predictor:

$$h\left(\mathbf{x}^{(i)},\boldsymbol{\theta}\right) = g\left(f\left(\mathbf{x}^{(i)},\boldsymbol{\theta}\right)\right) = \frac{1}{1+e^{-f\left(\mathbf{x}^{(i)},\boldsymbol{\theta}\right)}} = \hat{y}^{(i)} \tag{5.8}$$

In this example, g is the "logistic" function, which always returns a value between 0 and 1. A cut is then normally taken to reduce this to a binary value.

For binary classification, the logarithmic loss function is favoured over the mean squared error. This function has the following form:

$$E(Y,X,\boldsymbol{\theta}) = -\frac{1}{N}\sum_{i=1}^{N} y^{(i)}\log\left(h(\mathbf{x}^{(i)},\boldsymbol{\theta})\right) + \left(1-y^{(i)}\right)\log\left(1-h(\mathbf{x}^{(i)},\boldsymbol{\theta})\right) \tag{5.9}$$

Since all y values are 0 or 1, only one of these terms survives for any given data member. The leading negative sign accounts for the fact that the logarithmic factors are always negative for $0 < h(\mathbf{x}^{(i)}, \boldsymbol{\theta}) < 1$. This loss function has the distinct advantage of being convex, which means it has an accessible global minimum. As with the mean squared error, the update rule for $\boldsymbol{\theta}$ can be calculated from equation (5.6). Interestingly enough, one recovers nearly the same update equation as in equation (5.7), except that the logistic function remains in the residual term:

$$\theta_j := \theta_j(1+\lambda) - \frac{\alpha}{N}\sum_{i=1}^{N}\left[h\left(\mathbf{x}^{(i)}, \boldsymbol{\theta}\right) - y^{(i)}\right]\frac{\partial}{\partial \theta_j} f\left(\mathbf{x}^{(i)}, \boldsymbol{\theta}\right) \tag{5.10}$$

5.2. Algorithms

The discussion about machine learning so far has been in general terms, without any discussion of the form of $f(\mathbf{x}, \boldsymbol{\theta})$. This section will build on these principles and explore the two machine learning algorithms used in this work: Neural Networks and Gradient Boosted Decision Trees. For more details, see [58].

5.2.1. Neural Networks

Neural Networks (NN) are arguably the richest paradigm of machine learning models. While they are conceptually difficult, the mathematics behind them is surprisingly elegant. The following example explains how neural networks are evaluated once they are trained. For simplicity, it considers an input vector with three features and a classifier that yields a single output value. In general, neural networks can have an arbitrary number of input features and output values.

The basic structure of a neural network is a graph. The graph consists of three main types of nodes: input nodes, hidden nodes, and output nodes. The graph starts by placing the value of each feature onto its own input node, as in Figure 5.1. The first step is to pass a weighted linear sum of these features to each node in the first hidden layer. This operation can be formalized as a matrix multiplication:

$$z_i^{(1)} = \sum_{j=1}^{3} \Theta_{ij}^{(0)} x_j \quad \text{or equivalently} \quad \mathbf{z}^{(1)} = \Theta^{(0)}\mathbf{x} \tag{5.11}$$

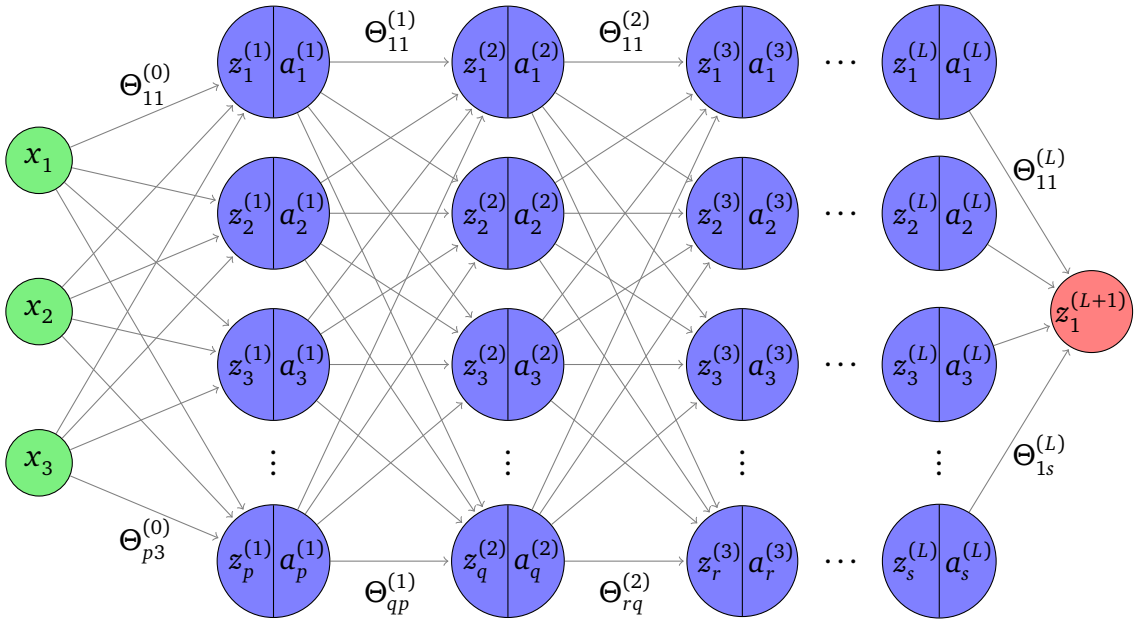


Figure 5.1.: A diagram of the structure of a neural network, illustrating the input layer in green, the hidden layers in blue, and the output layer in red. The ellipses indicate that neural networks can have a variable number of hidden layers and a variable number of nodes in each layer. The number of input and output nodes are three and one respectively. Only the outer edges are labelled by their corresponding values from the $\Theta^{(k)}$ matrices.

This equation introduces some key notation and concepts. The vector $\mathbf{z}^{(1)}$ is the input to the first hidden layer, where each element corresponds to a node in the layer. This vector has p elements in it according to Figure 5.1. It is calculated as the product of a matrix of weights $\Theta^{(0)}$ and the input vector $\mathbf{x}$. In this notation, the superscript on $z_i^{(l)}$ denotes which layer it is in, while the subscript denotes the node position in the layer.

In this example, $\Theta^{(0)}$ defines a $(p \times 3)$ matrix of free parameters. We can generalize to define $\Theta^{(l)}$ as the matrix that controls the mapping from the layer l to layer $l+1$. This matrix will have a shape of $(n \times m)$, where m is the number of nodes in layer l and n is the number of nodes in $l+1$. In general, each layer can have an arbitrary number of nodes. This defines the free parameters in NN models as a vector of matrices, all of which can vary in shape. This vector is denoted by Θ.

Inside the node, a non-linear "activation" or "response" function is used on the input to generate the output of the node. A popular choice is the logistic function:

$$a_i^{(1)} = g\left(z_i^{(1)}\right) = \frac{1}{1 + \exp\left(-\sum_{j=1}^{3} \Theta_{ij}^{(0)} x_j\right)} \tag{5.12}$$

This defines the output of the first hidden layer as a function of the three input features. This can readily be extended to define the flow of information between hidden layers:

$$z_i^{(l+1)} = \sum_{j=1}^{m} \Theta_{ij}^{(l)} a_j^{(l)} \tag{5.13}$$

$$a_i^{(l+1)} = \frac{1}{1+\exp\left(-z_j^{(l+1)}\right)} = \frac{1}{1+\exp\left(-\sum_{j=1}^{m} \Theta_{ij}^{(l)} a_j^{(l)}\right)} \tag{5.14}$$

Note that in general, the activation function g **must** be non-linear, otherwise the system could be reduced to a single linear transformation of the input variables.

Finally, Figure 5.1 shows that the output node is calculated according to equation (5.13). This value is unbounded, which works well for most regression problems. For classification problems, it is normal to apply another logistic function or something similar, as in equation (5.8). This can be explicitly written as:

$$\text{Regression:} \quad \hat{y}_k = z_k^{(L+1)} = f(\mathbf{x}, \Theta)_k \tag{5.15}$$

$$\text{Classification:} \quad \hat{y}_k = h\left(z_k^{(L+1)}\right) = h\left(f(\mathbf{x}, \Theta)_k\right) \tag{5.16}$$

In these equations, the output has been subscripted by k, which only has a value of one in the example. As stated previously, neural networks can return an arbitrary number of values, meaning k would index the output vector of values.

Neural networks are trained through gradient descent using the "back propagation" algorithm. The derivation of this technique is explored in Appendix D.1. Those who are unfamiliar with this technique and dubious about neural networks in general are urged to read it.

5.2.2. Gradient Boosted Decision Trees

Gradient Boosted Decision Tree (GBDT) classifiers are conceptually simpler than neural networks, but less mathematically pleasing. They fall under the umbrella category of "ensemble" learning methods, which combine many weak learning algorithms into a far stronger one. Unsurprisingly, GBDTs combine the output of many Decision Trees using a Gradient Boosting method. These models typically use Classification and Regression Trees (CARTs) as the underlying tree model [59].

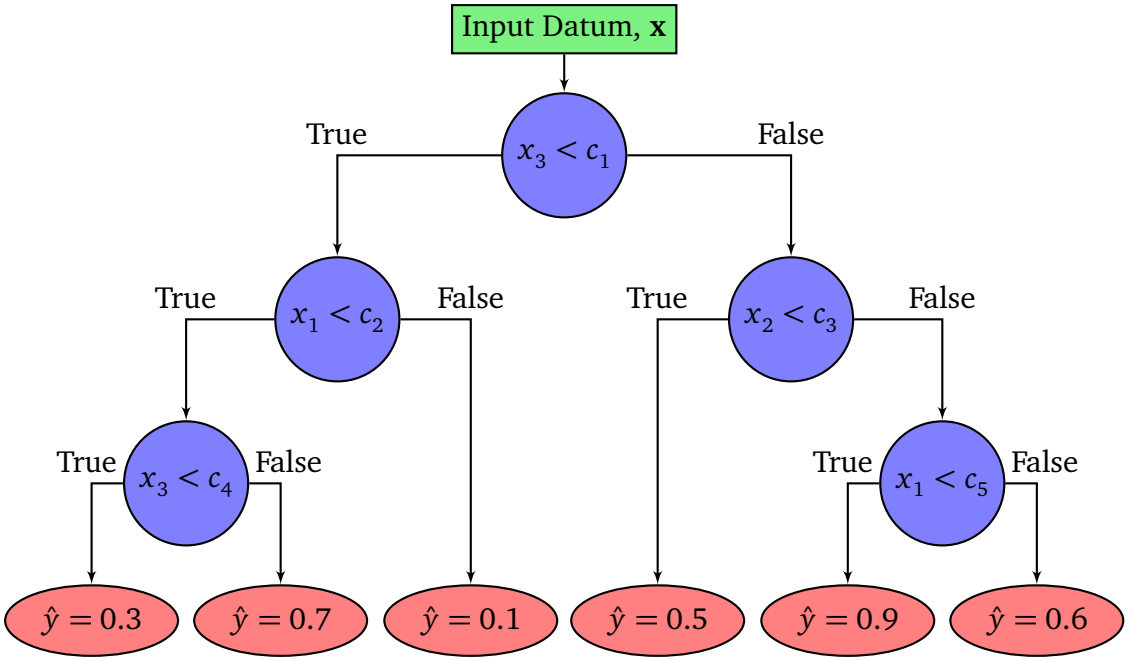


Figure 5.2.: A diagram of a fully grown Decision Tree. The input data is represented as a green block, the splitting nodes are drawn in blue, and the leaf nodes are drawn in red with their associated values.

Evaluating a single CART is very simple. These trees work on the principle of splitting the feature space into smaller subspaces and assigning an output value to each one. Figure 5.2 illustrates the trees' structure. In this figure, the input datum with feature set $\mathbf{x} = (x_1, x_2, x_3)$ is passed into the tree. From here, the datum is passed through splitting nodes of the tree, which represent a sequence of cuts on the feature space. These cuts are made at thresholds c_k, which have been left in general terms in the diagram. Each leaf node has a corresponding $\hat{y}$ value which is assigned to each datum that arrives in the node. These values are explicitly provided in the example for clarity.

Gradient Boosting in general is an example of an additive model, which is a subset of ensemble methods. Additive models admit the following form:

$$F_M(\mathbf{x}) = \sum_{i=1}^{M} \gamma_i h_i(\mathbf{x}) \tag{5.17}$$

This defines the output of an additive ensemble of M weak predictors as F_M. Each predictor, or tree in this case, is indexed by i and has a corresponding output of h_i. The weight of each predictor in the sum is denoted γ_i. Evaluating the ensemble is then a question of evaluating each tree and taking the weighted sum of their outputs [60].

Gradient Boosted algorithms are built so that each new predictor corrects for the mistakes of the existing predictors in the model. This means that each tree is built independently and sequentially. The full details of CART building and Gradient Boosting can be found in Appendix D.2 and Appendix D.3, respectively.

5.2.3. The Circular Hough Transform

Particle physics track reconstruction was one of the first disciplines to utilize computer vision algorithms. Of note is the Hough Transform, which was developed by Paul Hough in 1962 to automate track detection in bubble chamber experiments [61]. While the original algorithm was developed to detect straight lines, this work utilizes the circular Hough transform [62].

All (x, y) points, or hits[3], on a circle centred around the point (a, b) with a radius r satisfy the following equation:

$$(x-a)^2 + (y-b)^2 = r^2 \tag{5.18}$$

This identifies the three parameters $(a,\ b,\ r)$ that can be used to describe any circle in (x, y) space. The circular Hough transform finds the best circle for a set of cocyclic hits by fixing r and determining the best (a, b) candidate.

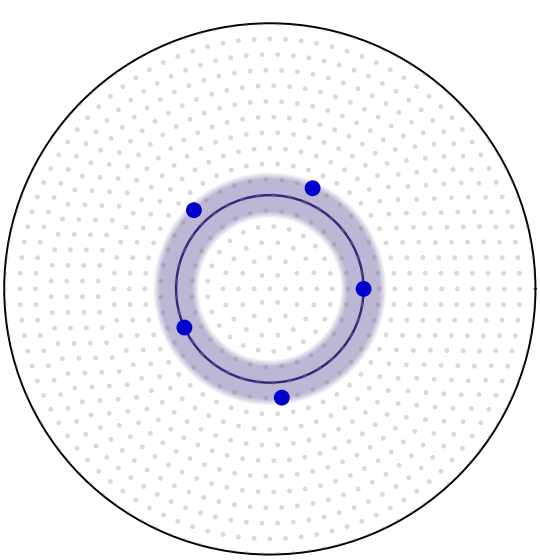
It starts with the list of hits in (x, y). For each hit, it generates a list of candidate (a, b) points that lie a distance r from the hit. It then selects the most popular (a, b) value-set from all candidates from all hits. This can be represented graphically as drawing a circle of radius r around each of the hits and checking where these circles overlap.

An obvious shortcoming of this algorithm is that a fixed r is required. In real applications, circle finding normally requires finding if a set of hits that lie roughly the same distance from a point (a, b). To achieve this, circles drawn around the cocyclic hits can be smeared to accept a wider range of (a, b) candidates. This smearing is best described in the context of a discrete space, in which all the position vectors $\mathbf{x} = (x, y)$ for the hits are part of some finite set. The same can be said for the potential circle-centre position vectors, $\mathbf{a} = (a, b)$. Formally, one can write:

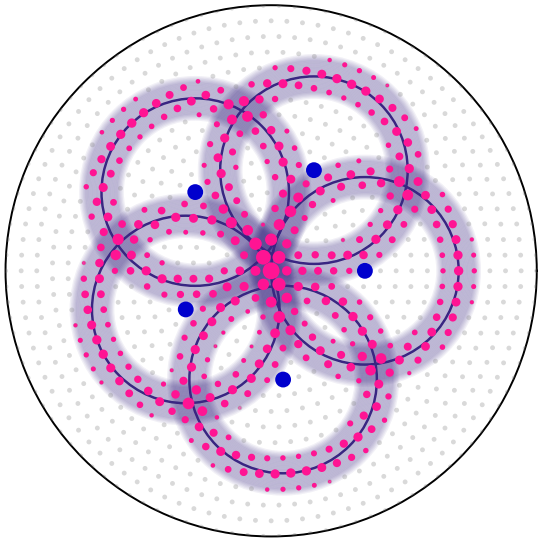
$$\mathbf{x}_j \in X \quad \text{for} \quad j \in [0, N-1], \quad N = |X| \tag{5.19}$$

$$\mathbf{a}_i \in A \quad \text{for} \quad i \in [0, M-1], \quad M = |A| \tag{5.20}$$

[3]Points in (x, y) will be referred to as hits to match the language used in later sections.



(a) The input to the algorithm hypothesises that each of the blue points lies somewhere within the circular shaded blue area.



(b) Each hit point votes on its favoured track centres. Each hit is given equal voting weight. The score for each potential centre is shown by the size of the pink dot.

Figure 5.3.: These two images visually summarize equation (5.23). In this example, $A = X$, which are defined in equation (5.19) and equation (5.20). Furthermore, T_{ij} is defined such that centres at the target radius are favoured, while points towards the edge are favoured less. Despite being in a discrete space with points that are not perfectly cocyclic, the transformation still finds the centre of the hypothesised red circle.

where $|S|$ denotes the cardinality of the set S. These definitions highlight the fact that X and A do not have to be the same in general. From this, a function can be defined that determines if centre $\mathbf{a}_i$ lies within some radial band of point $\mathbf{x}_j$:

$$H(\mathbf{a}_i, \mathbf{x}_j) = \begin{cases} 1 & : \ r_{\min} < |\mathbf{a}_i - \mathbf{x}_j| < r_{\max} \\ 0 & : \ \text{else} \end{cases} \tag{5.21}$$

where $r_{\min}$ and $r_{\max}$ define the range of acceptable radial of candidates. This is the simplest example of smearing, but the concept can be extended to allow for a family of smearing functions.

The circular Hough transform readily admits another two properties:

$$\begin{aligned} H(\mathbf{a}_i, \mathbf{x}_j) &= F(|\mathbf{a}_i - \mathbf{x}_j|) \\ H(\mathbf{a}_i, \mathbf{x}_j) &= T_{ij} \equiv F(R_{ij}) \end{aligned} \tag{5.22}$$

The first line simply states that any general transformation H between hits and circle-centres must be radially symmetric. H can then be phrased as a function F that only cares about the distance between the hit point at $\mathbf{x}_j$ and the potential centre at $\mathbf{a}_i$. The second line builds on this to observe that in a discretised space, the pairwise distances between all points $\{\mathbf{x}_j \in X\}$ and $\{\mathbf{a}_i \in A\}$ can be described by a matrix R where $R_{ij} = |\mathbf{a}_i - \mathbf{x}_j|$ From here, the transformation itself readily admits its own matrix formulation, where T_{ij} is defined from $F(R_{ij})$. Note that this in an $M \times N$ matrix, as implied directly from equation (5.19) and equation (5.20). This discrete transform is visualized in Figure 5.3.

This leads to the full matrix form of the circular Hough transform:

$$c_i = \sum_{j=0}^{N-1} T_{ij} w_j \tag{5.23}$$

This has introduced two new vectors, $\mathbf{w}$ and $\mathbf{c}$, which are best explained by first exploring a simple example. Define T_{ij} using equation (5.21) and define w_j to equal one if there is a hit at $\mathbf{x}_j$ and zero otherwise. This implicitly defines the element c_i as the number of hit points that lie between $r_{\min}$ and $r_{\max}$ of the circle-centre at $\mathbf{a}_i$. This means the most likely circle centre corresponds to the maximal element in $\mathbf{c}$.

Three key generalizations can be made by lifting the assumptions made for the example case:

- The hit property vector $\mathbf{w}$ can store continuous values. This allows hits to be weighted to adjust their relative contributions to the centre property vector, $\mathbf{c}$.
- The function F can return continuous values. This means F can be selected so that it matches the expected radial distribution of the hits around the centre.
- As a natural extension of these two, $\mathbf{c}$ can also store continuous values. Furthermore, when considering the most likely centre, a set of maximal points can be considered instead just one.

In conclusion, the circular Hough transform can be thought of as a voting scheme. Each hit point, $\mathbf{x}_j$, votes on its favourite circle-centres. The amount which it favours a given centre $\mathbf{a}_i$ is described in the matrix element $T_{ij} = F(R_{ij})$. The amount which it gets to vote is controlled by its weight w_j. The output scores each potential circle centre by how many votes it receives, which is stored in the $\mathbf{c}$ vector. The maximal element(s) of $\mathbf{c}$ are the most popular circle centre candidate(s).

Chapter 6.

Signal and Background Acceptance

To introduce concepts seen throughout the analysis work, this chapter starts by illustrating how the CyDet detects a signal electron. It the then uses the simulation to show how the beam evolves from an 8 GeV proton beam to a high intensity, low-momentum muon beam. It finishes with an analysis of three groups of parameters that are key to tracking:

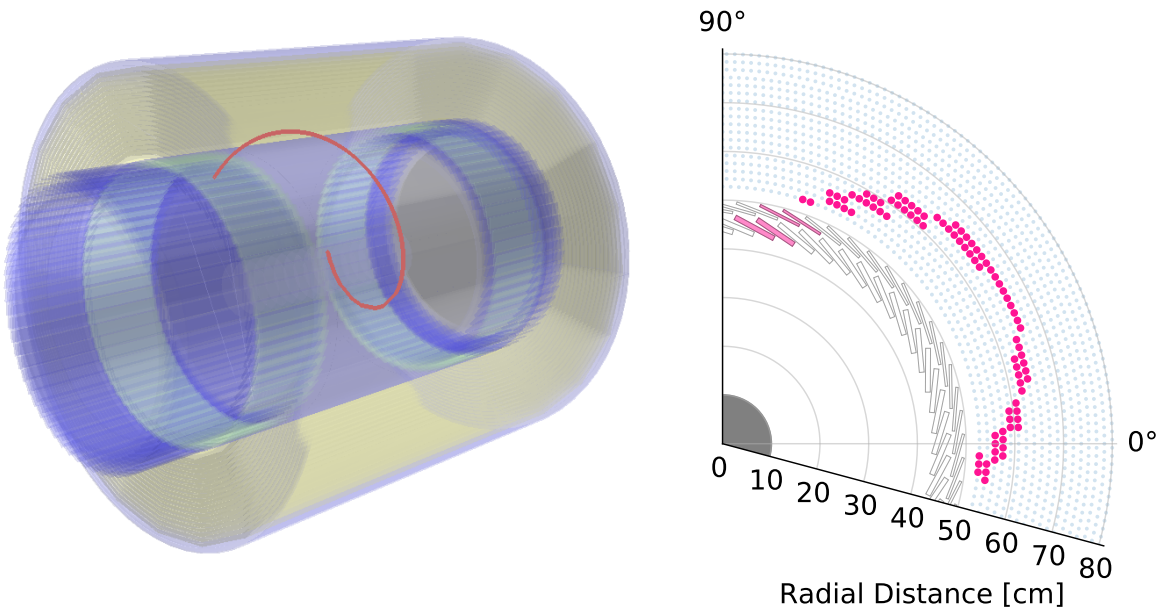
- Bunch size: The number of proton-on-target (POT) events per bunch, which determines the intensity of the tracking environment.
- Time window: The size and placement of the fiducial time window relative to the bunch arrival affects both signal and background acceptance.
- Trigger and Track Quality: All signal tracks must pass a minimum standard before any reconstruction can be attempted.

6.1. Hits in the Cylindrical Detector

A signal track is show in simulation event display in Figure 6.1a, where the CyDet components are transparent. Figure 6.1b shows this same track in terms of CDC and CTH hits in the readout plane.

6.1.1. CTH Hits

The CTH is composed of scintillators and Cherenkov counters. To model their 10 ns sampling rate, the CTH hits are defined by the sum of all energy deposition within 10 ns bins for each CTH module. The ADC values are set to this sum, which ignores any non-linear scaling between the deposited energy and the detected ADC signal.



(a) A signal electron inside the CyDet. **(b)** The readout projection of the track.

Figure 6.1.: An example of a signal event in the CyDet detector. Figure 6.1a shows the electron passing through the CyDet, while Figure 6.1b illustrates what the same track looks like to the detector system at the readout plane. The hits on the CDC wires and CTH modules are shown in pink.

The ADC values are calculated as the deposited energy from Geant4 for the scintillators. For the Cherenkov counters, the ADC values are determined by the number of Cherenkov photons generated by the incident particle. The light guides can also generate Cherenkov photons, such that the guides attached to the Cherenkov counters are treated as part of the module's fiducial volumes. Since scintillation produces around 100 times more light than Cherenkov radiation, the light guides attached to the scintillators are ignored.

The CTH hits in Figure 6.1b exemplify the "four-fold coincidence" trigger pattern. This pattern is formed from a neighbouring pair of modules in the scintillator layer and a corresponding pair in the Cherenkov layer. The module pairs must have at least one overlapping module. All four hits must occur within a 50 ns time window that is inside the event's fiducial time window.

6.1.2. CDC Hits

The CDC hits in Figure 6.1b are from a single pass, but manifest as two distinct semi-circular patterns. This is caused by the alternating stereo angles, as discussed in Section 3.2.3. Each hit is defined as the sum of the charge deposition in a channel ADC in each channel over

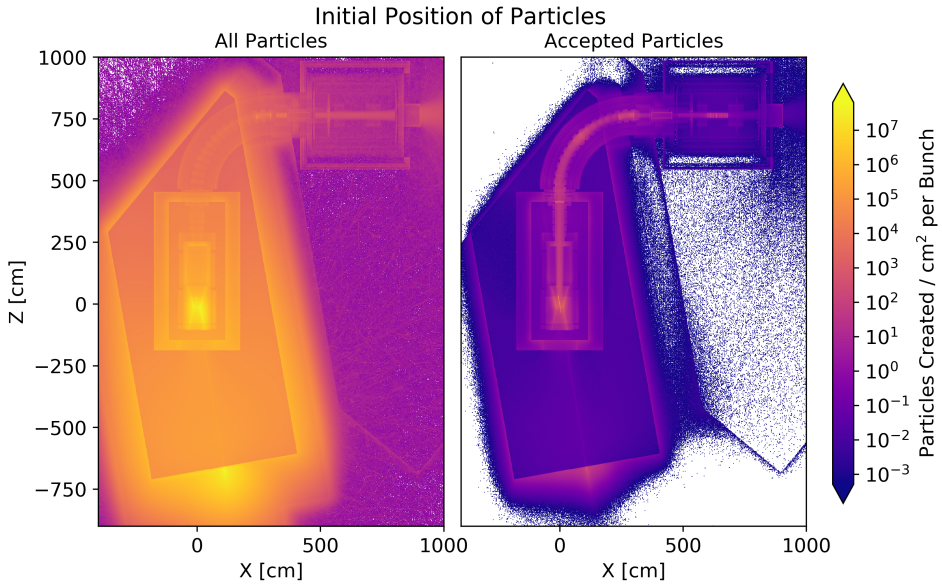


Figure 6.2.: The initial particle distributions projected onto the x-z plane. The left plot shows where all simulated particles are created. The right plot shows where particles that are accepted by the Muon Transport or Detector Solenoid are created, including all parents, grandparents, etc. of accepted particles. Note the colour axis is logarithmic. These plots can be compared to the full 3D geometry, as seen in Figure 4.2.

the whole fiducial time window. This corresponds to the "worst-case" scenario for CDC time resolution. The earliest hit time in the sum is used to describe the CDC hit.

The fiducial time window in the CDC is defined to be longer than the trigger time window to account for the drift time of the hits. Like the CTH window, it is defined relative to the arrival of the proton bunch at the target. The real experiment may define the window relative to the CTH trigger time instead, but this has not been finalized.

6.2. The COMET Beam

The momentum of each initial proton is set at 8 ± 0.1 GeV. They are directed towards the pion production target, although not all are guaranteed to hard-scatter off of the target. The protons that pass through the target or miss it all together typically deposit their energy in the beam dump.

The left plot in Figure 6.2 illustrates that the COMET beamline is very radioactive, with $1.5e11$ particles are simulated per bunch. Of these particles, the plot to the right plots the initial position of the $2.7e6$ particles that make it to the Muon Transport or Detector solenoid sections. This plot demonstrates that even particles that are born in the beam

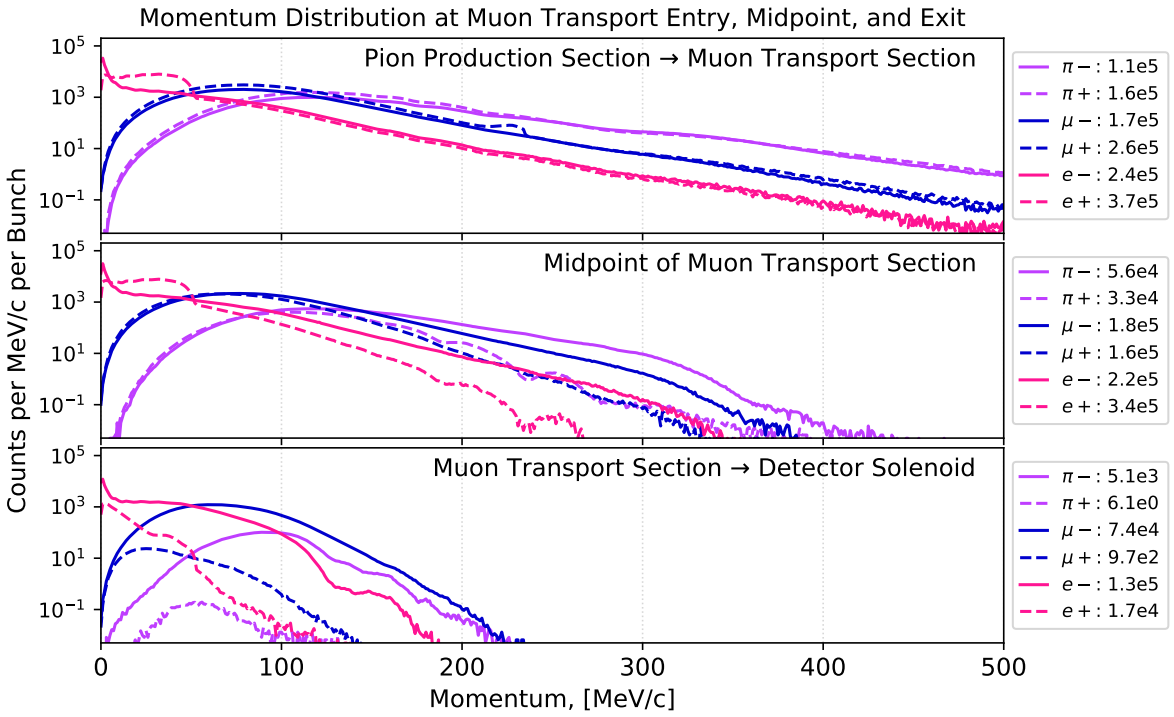


Figure 6.3.: The momenta of pions, muons, electrons, and their antiparticles as they enter the Muon Transport Section, pass through its mid-point, and exit it. The midpoint is defined as the plane at 45 degrees along the full 90-degree bend of the curved solenoid. The legend for each plot provides the integral of the corresponding entry, including any particles in the overflow bins. These plots are normalized to one bunch and are displayed on a logarithmic axis.

dump are accepted by the later stages of the beamline. As delayed particles, they are a potentially dangerous background source, which forces the simulation to consider them.

As discussed in Section 3.1.3, the Muon Transport section is designed to filter out all high-momentum particles and positively charged particles to create the low-momentum muon beam. Figure 6.3 clearly demonstrates that the high-momentum particles that enter the transport section are filtered away, while the low-momentum muons are preserved. It also demonstrates that the positively charged particles are more heavily filtered than their negatively charged counterparts.

The position distribution of stopped muons shown in Figure 6.4 illuminates how the Muon Stopping Target stops as many muons as possible with a minimal material budget, as discussed in Section 3.2.2. In total, 4.675×10^{-4} muons are stopped in the target per POT event. The left plot shows that slightly more muons stop in the earlier disks. The right plot shows that the peak of the transverse distribution is slightly off centre. Since this

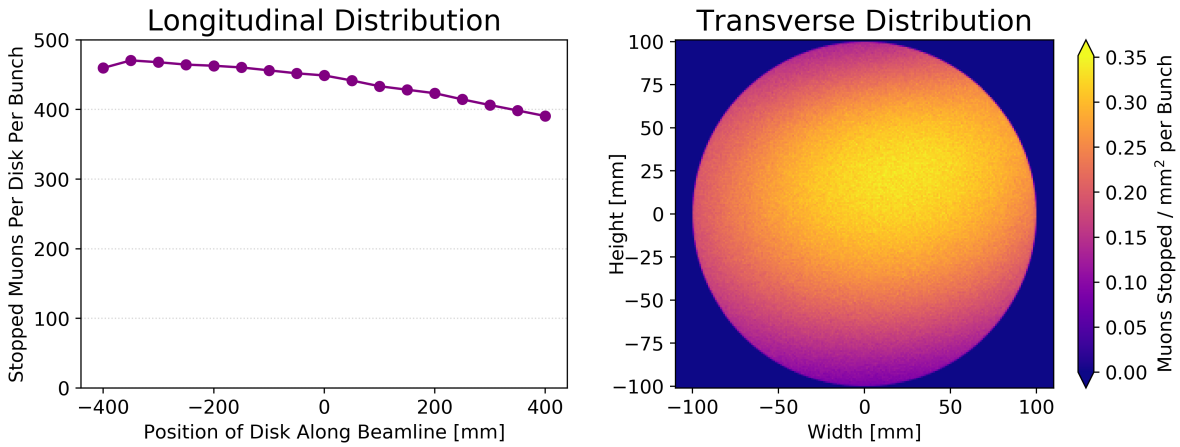


Figure 6.4.: The longitudinal and transverse stopping distribution of muons in the Muon Stopping Target disks. The longitudinal plot shows the number of stopped muons per disk, which are perpendicular to the muon beam. The transverse distribution sums over all disks. For both plots with (0,0,0) corresponds to the centre of the central target disk. Both plots are normalized to one bunch.

distribution determines the initial position the simulated signal electrons, there may be some asymmetries in the detected signal hit patterns.

6.3. Bunch Parameters

The pulsed structure of the COMET beam, as discussed in Section 3.1.1, structure can be summarized by three key parameters:

- Bunch Period, B_T: The time between bunches.
- Bunch Width, B_W: The time between the first and last proton in the bunch
- Bunch Intensity, B_I: The number of protons delivered in each bunch.

The default values for (B_T, B_W, B_I) are $(1170\text{ ns}, 100\text{ ns}, 16\text{e}6\text{ POT})$. The arrival of the centre of the bunch to the Pion Production Target defines $t = 0$. The protons have a flat timing structure within the bunch width, meaning each POT event has an equal probability arriving anywhere between -50 ns and $+50$ ns.

The bunch width affects the timing of CDC hits, CTH hits, and stopped muons, as seen in Figure 6.5. If the bunch width were zero, the upper plots demonstrate the peak from the beam flash that occurs between $[40\text{ ns}, 100\text{ ns}]$ can be separated from the peak from Muon Stopping Target radiation that starts at 100 ns. The slow decay of this second peak in the CDC hits highlights how the bound muon lifetime results in delayed background hits. The

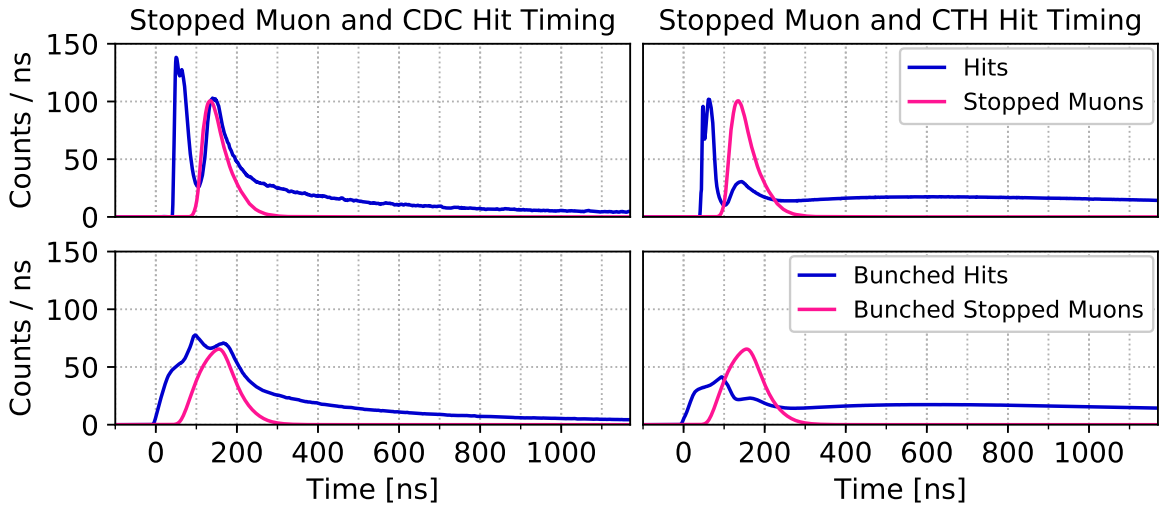


Figure 6.5.: The effects of the bunch width on the background hit and stopped muon timing in the CDC and CTH. The top plot shows the timing distrubitons for $B_W = 0$ ns, while the bottom plots show these distributions for $B_W = 100$ ns. Note that lines are normalized to one bunch and that the CDC hits do not include the drift time.

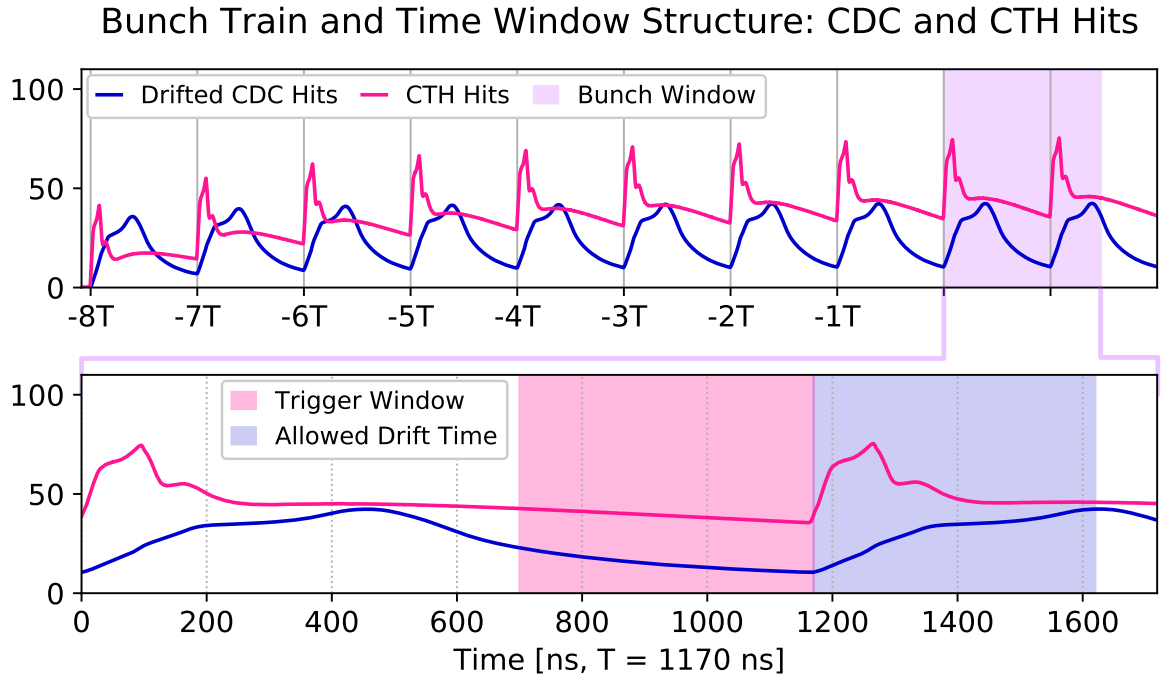


Figure 6.6.: The bunch-train timing structure in the CDC and CTH hits. The upper plot shows the full bunch-train window. The lower plot zooms in on the fiducial time window of the bunch-train event. This plot highlights [700 ns, 1170 ns] as the CTH trigger window. The CDC accepts an additional 450 ns to allow the any signal hits to drift to the wires.

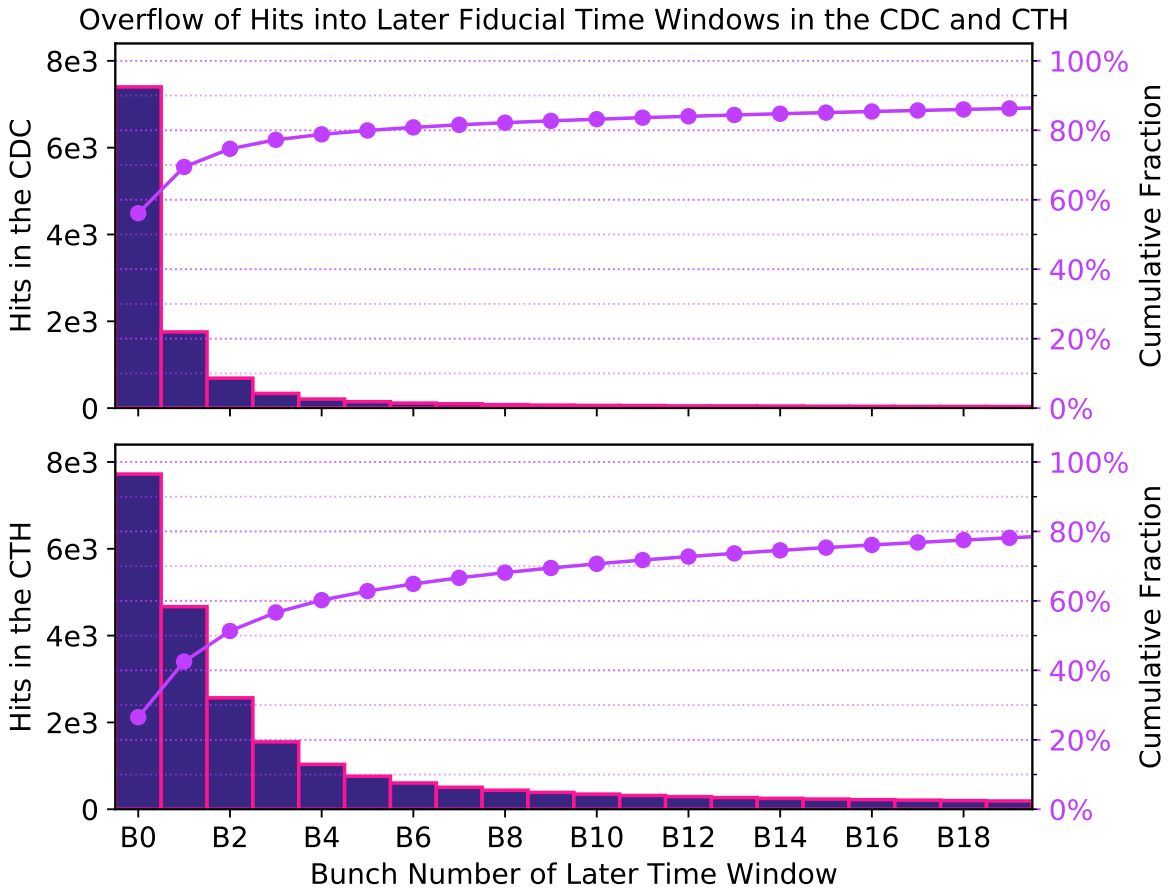


Figure 6.7.: The overflow of background hits into later fiducial time windows for the CDC and the CTH. The histograms, corresponding to the left axes, count the number of hits from a bunch arriving at $t = 0$ occur within the fiducial time window of the N-th bunch, defined as: $[1170 \cdot N \text{ ns}, 700 \text{ ns} + 1170 \cdot (N+1) \text{ ns}]$ and denoted as BN. The lines, corresponding to the right axes, determine the fraction of hits in the first N windows relative to the number of hits in the first 100 windows.

lower plots highlight how the 100 ns bunch width smears these two peaks together and pushes the time of the earliest hits to $t = 0$. The tail in the CTH is caused by the neutron background, which has a larger cross-section solid volumes of the CTH than the gaseous volume of the CDC.

The *bunch-train* timing structure is used to model the beam background. This structure includes the tails of the previous eight bunches and any early beam flash from the next bunch, as seen in the upper plot in Figure 6.6. This plot shows how the tails from the previous bunches form a pedestal of hits in the later time windows. The bottom plot shows the default fiducial time window for the bunch-train events. While the CTH window ends at 1170 ns, the drift time in the CDC requires an additional 450 ns to detect any signal hits.

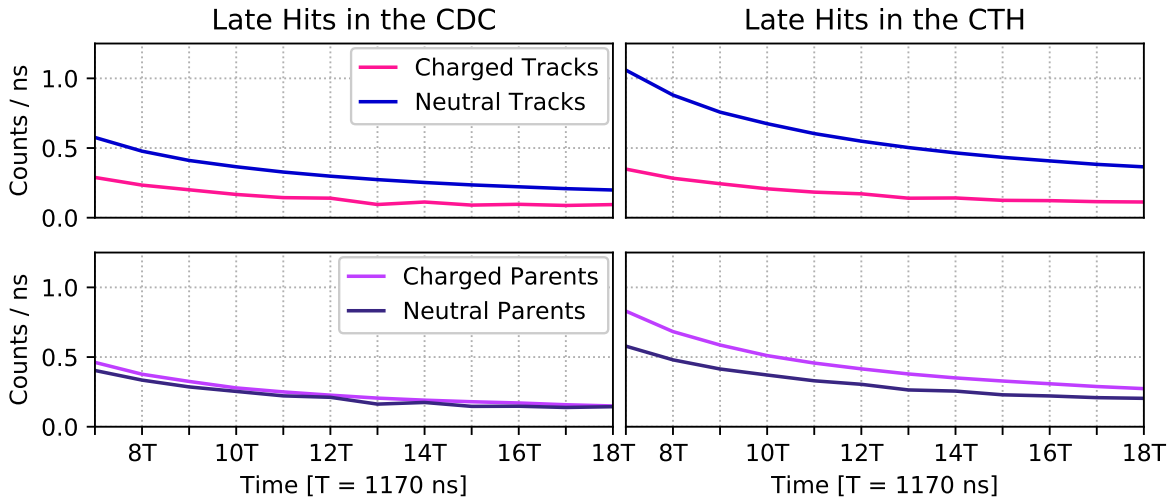


Figure 6.8.: The distribution of the hit rate from late particles. The top plots show the hit rates in the CDC and CTH separated out by the charge of the particle that leaves the hit. Similarly, the bottom plots show the hit rate separated out by the charge of the parent of the particle that made the hit.

This causes the CDC to accept a large portion of hits from the next bunch. Optimizing this time window is explored later in Section 6.4.5.

Figure 6.7 illustrates how many hits from a bunch at $t = 0$ overflow into the fiducial time window of later bunches. This plot illustrates the previous bunches can create backgrounds that take $\mathcal{O}(1\text{ ms})$ to hit the CDC or the CTH. This effect should be verified using other simulation packages like PHITS or FLUKA, but this is outside the scope of this work. Building bunch-train events in ICEDUST requires around 1 GB of memory for each bunch in the train, which limits the bunch-train to 8 tail bunches, the analyzed bunch at $t = 0$, and the next bunch. This "ten bunch-train" accounts for around 70% of CTH hits and 80% of CDC hits from the tail of the previous 100 bunches.

To dive deeper into the late hits, Figure 6.8 inspects the underlying cause. The top plots show that neutral particles are more likely to cause late hits in both the CDC and the CTH. In both detectors, muons and electrons account for 90% of the charged hits while neutrons account for 99% of the neutral hits. The bottom plots explore the charge of the parents of the particles that cause these hits and demonstrates that these parents can be other neutral particles or charged particles. This suggests that this late background is caused by ambient radiation in the COMET experimental hall.

Figure 6.9 shows the effects of the intensity on the CDC hits, the CTH hits, the CDC occupancy and the CTH fake trigger acceptance. Using 16e6 POT events per bunch, the

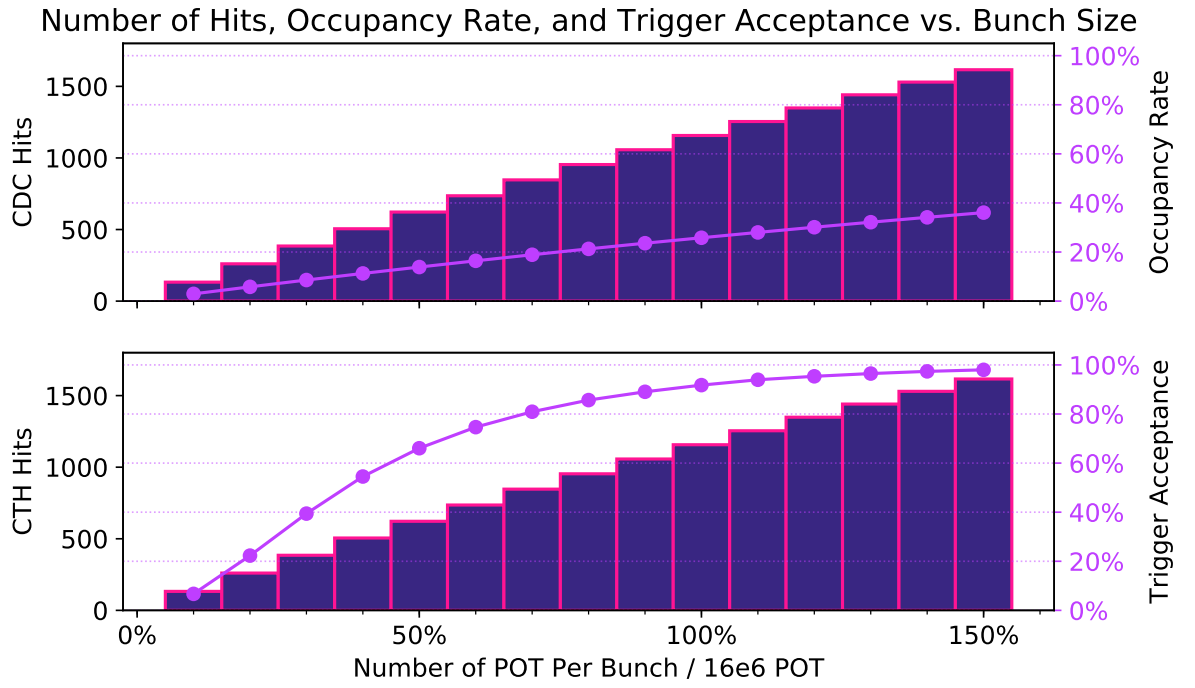


Figure 6.9.: The CDC occupancy and the CTH trigger acceptance as a function of the bunch intensity. The histogram counts the hits in each detector using the left axes. The occupancy and trigger acceptance are plotted with the purple line and correspond to the right axis.

default timing window, and ten bunches in each bunch-train, the trigger fires for 91.7% of events in which 25.8% of the CDC channels are occupied. Both of these values are high for the DAQ systems or tracking fitting algorithms to handle. Reducing both of these numbers is the focus of this work.

6.4. Timing, Triggering, and Tracking Cuts

The signal acceptance can be decomposed into three main components:

- Timing: the size and placement of the fiducial time window.
- Triggering: the acceptance of the CTH geometry and triggering algorithms.
- Track Quality: the acceptance of "quality" track by the CDC geometry, where "quality" denotes a track that passes the minimum requirements for the track fitting algorithm.

The final acceptance comes from tracks that are accepted by all three of these components. The breakdown of these acceptances for the current design values for all thresholds is shown in Table 6.1.

Selection	Acceptance	Comment
Time	30%	Discussed in Section 6.4.2.
Triggering	30%	Discussed in Section 6.4.3.
Track Quality	78.3%	Discussed in Section 6.4.4.
Total	6.6%	Accounts for all of the above.

Table 6.1.: The breakdown of the signal acceptance into timing, triggering, and track quality components.

6.4.1. Timing Parameters

The CDC occupancy, CTH fake trigger acceptance, and overall signal acceptance all depend on the following four time parameters:

- the lower limit for the fiducial time window for the CDC and the CTH, t_{min},
- the upper limit for the CTH fiducial time window, t_{max},
- the upper limit for the CDC fiducial time window, $t_{\text{max}} + t_D$,
- the coincidence time window for the CTH trigger pattern, t_W, and

The optimal parameters would maximal signal acceptance with minimal CDC occupancy and CTH fake triggers. This study attempts to find more favourable sets of $(t_{\text{min}}, t_{\text{max}}, t_D, t_W)$ than the default values of $(700\text{ ns}, 1170\text{ ns}, 50\text{ ns}, 450\text{ ns})$. Note that the fake trigger "rate" is measured as the percentage of background only bunch-train events that contain the trigger pattern.

6.4.2. The Time Distribution of Signal Tracks

The time distribution of signal tracks is determined by the stopped muon time, shown in Figure 6.5 and the lifetime of bound muons in aluminium, as described in Section 4.2.4. The blue line in the left plot in Figure 6.10 shows the stopped muon process rate from one bunch. This long tail on this line implies that signal tracks might be created in a later fiducial time window. To model this effect, signal tracks are simulated initialized by sampling the modulo of the stopped muon process time with 1170 ns, as shown by the pink line in the same plot. This forces all signal tracks to start within the analyzed bunch that starts at $t = 0$. The contour plot to the right in Figure 6.10 shows the fraction of signal tracks that are created within different $[t_{\text{min}}, t_{\text{max}}]$ windows. The flight time of signal tracks

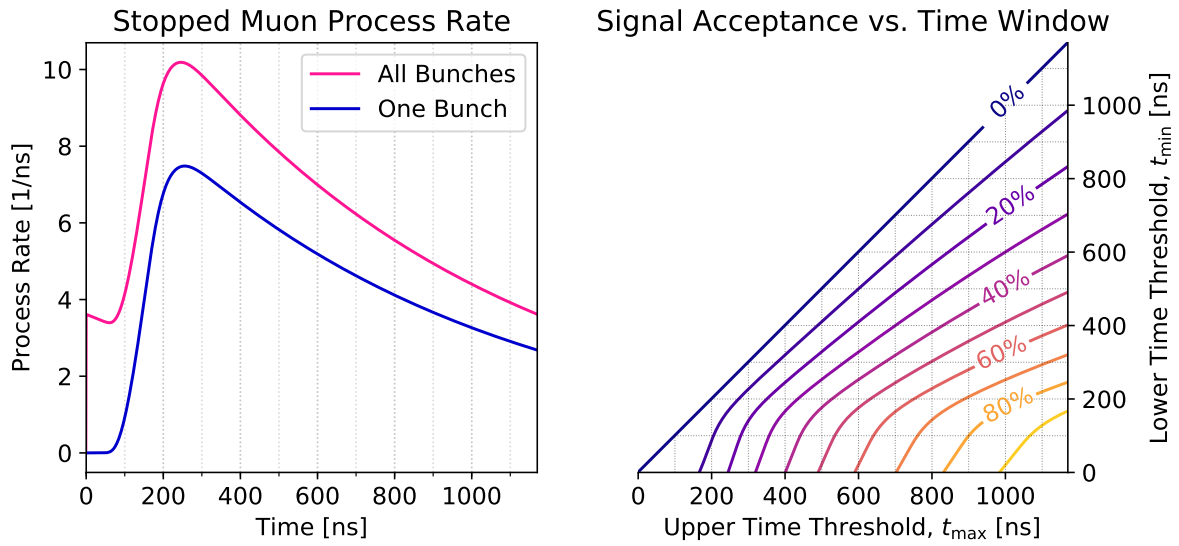


Figure 6.10.: The stopped muon process time and the implied signal acceptance. The left plot shows the stopped muon process rate from one bunch arriving at the production target at $t = 0$ in blue. The blue line includes the stopped muons from all previous bunches. The contours in the right plot show the signal acceptance as a function of the lower and upper time limits.

is negligible, such that this plot represents the overall signal acceptance for all possible fiducial time windows.

6.4.3. The CTH Trigger

A lower threshold on the ADC of the CTH scintillators can reduce the fake trigger acceptance. The left plot in Figure 6.11 shows the loose cut is selected at $E_{\text{min}} = 170$ keV on the scintillator ADC values. This cut keeps 99% of signal trigger hits while removing 60% of background hits. A tighter cut could be made that keeps 95% of signal hits and removes 95% of background hits. Given the simplicity of the ADC modelling, the looser cut was selected. The ADC values of the signal trigger hits in the Cherenkov hodoscope were not separable from the background hits. The right plot shows the fake trigger acceptance after the ADC cut for all $[t_{\text{min}}, t_{\text{max}}]$ windows, where $t_W = 50$ ns. This determines the fake trigger acceptance to be 83.1% for the default CTH parameters.

The plot to the left of Figure 6.12 how the coincidence window, t_W, affects the signal and background acceptance. This plot introduces the signal trigger acceptance at 9.0% for the default time window timing, which includes the overall geometrical acceptance of the CTH. If the ADC cut is used and the t_W is reduced to 10 ns, i.e. one CTH time bin, the fake trigger acceptance is reduced to 31.0%, but only loses 0.2% of the signal acceptance.

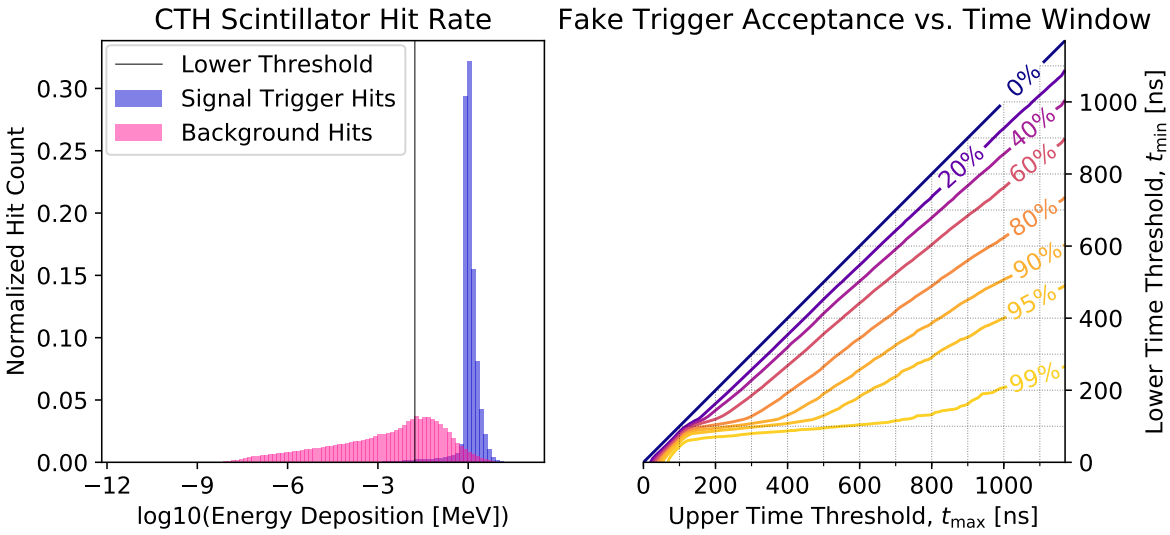


Figure 6.11.: The CTH scintillator ADC distribution and the fake trigger acceptance. The left plot compares the ADC values for signal and background on a logarithmic x-axis with selected cut denoted by a vertical line. The right plot shows the fake trigger acceptance after the ADC cut as a function of the lower and upper fiducial time window limits, where $t_W = 50$ ns.

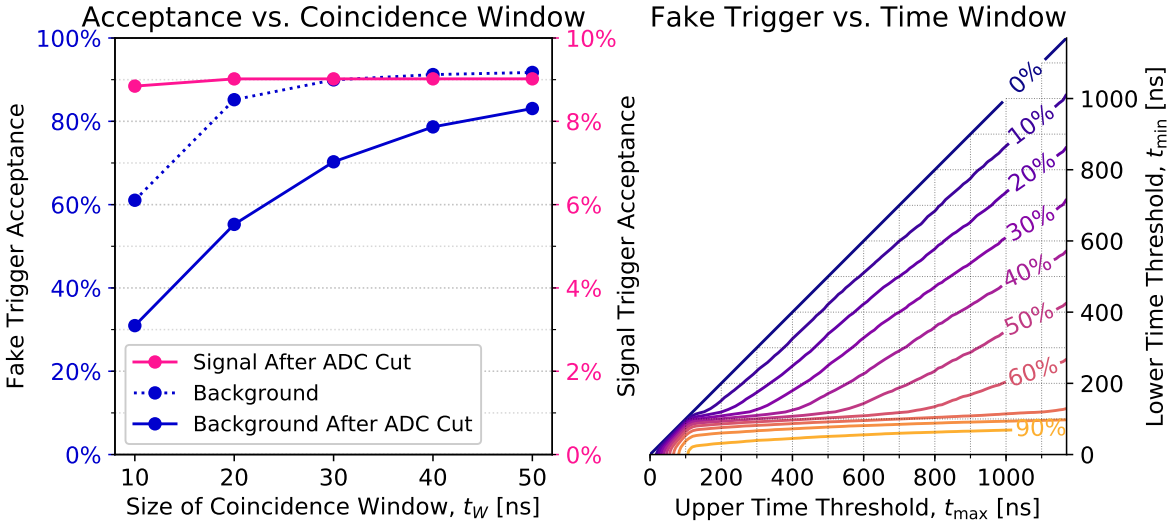


Figure 6.12.: The signal and background trigger acceptance as a function of the coincidence window, t_W. The CTH ADC cut is used for both of these plots. The left plots the fake trigger acceptance the left axis and the signal trigger acceptance on the right axis as a function of the trigger coincidence window size, t_W. This plot uses the default $[700\text{ ns}, 1170\text{ ns}]$ time window. The right plot shows the fake trigger acceptance for the $t_W = 10$ ns for all possible time windows.

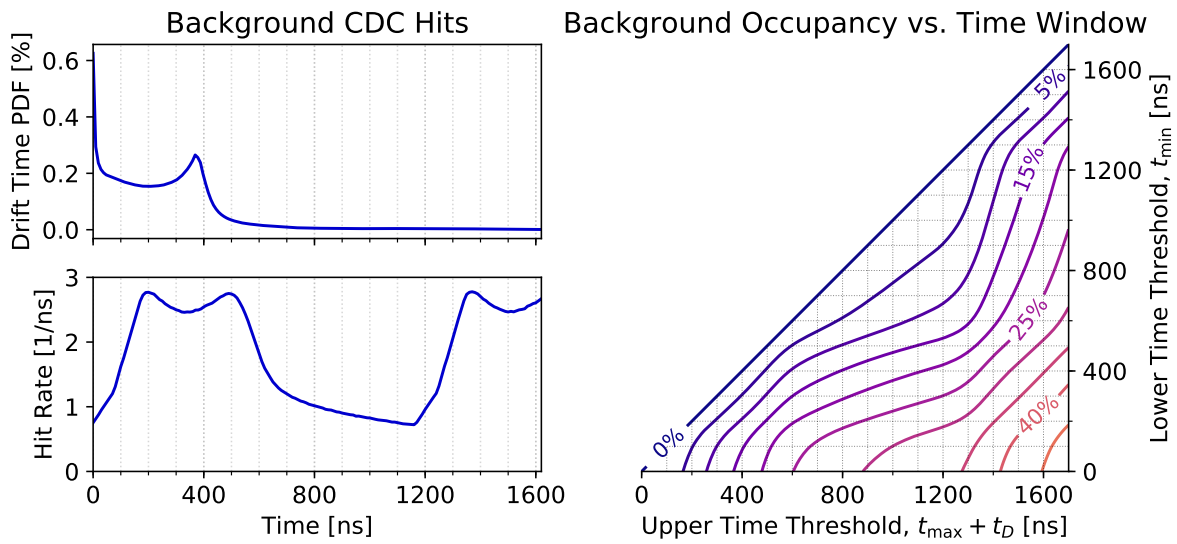


Figure 6.13.: The background drift time, hit rate and occupancy in the CDC. The top left plot shows the probability distribution of drift times for all background CDC hits. The bottom left plot shows the background hit rate in the CDC after this drift time is considered. The right side shows CDC occupancy as a function of the fiducial time window.

The right-hand plot in Figure 6.12 shows the fake trigger acceptance as a function of the fiducial time window for the 10 ns coincidence window and using the ADC cut. The signal trigger acceptance closely follows the signal acceptance plot shown Figure 6.10, albeit at an evenly reduced acceptance due to the geometrical acceptance of the CTH.

6.4.4. Tracking Cuts

The plot thickens once the CDC acceptance is considered. The background CDC hit rate can be seen in the plot to the bottom left in Figure 6.13. This plot features two double peak structures. The double peak is caused by the two peaks in the drift time distribution, which is plotted in the top left plot. The first double peak comes from the bunch arriving at $t = 0$ while the second is from the next bunch that arrives at 1170 ns. The right plot shows the CDC background occupancy as a function of the fiducial time window. The dip between the two double peaks is reflected by the shallower contour lines from 600 ns to 1200 ns. In order to be accepted by the CDC, the signal track must:

- have at least 35 hits from their first turn through the CDC survive the time cut, where
- the maximum layer reached by these hits is at least the fifth layer, and
- at least one hit survives in the first three layers at the track's first turn entry and exit positions.

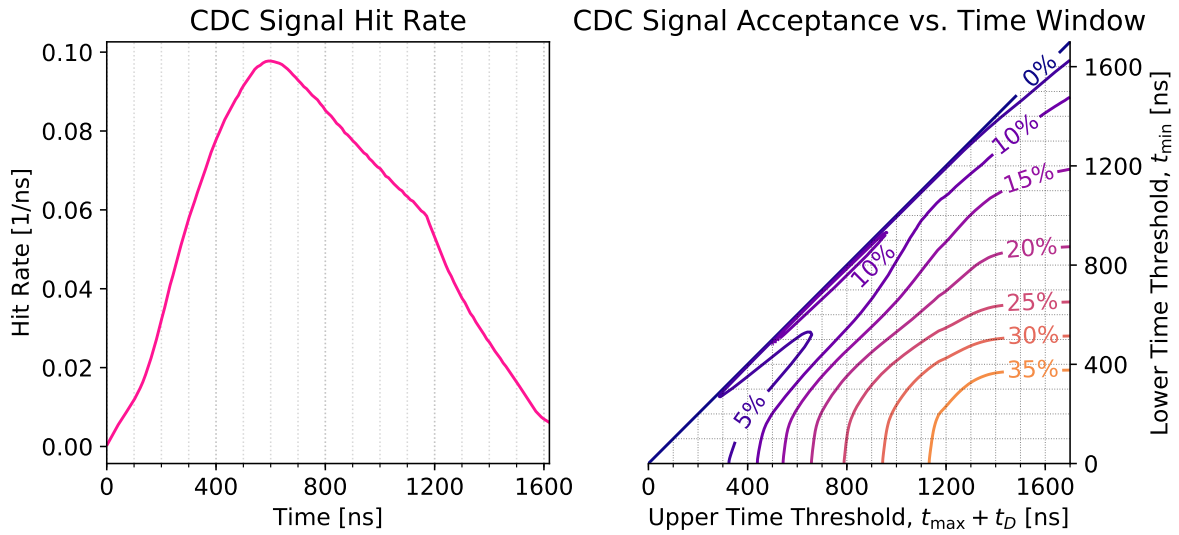


Figure 6.14.: The signal hit rate and occupancy in the CDC. The plot on the left shows the signal hit rate in the CDC after the drift time is considered. The right side shows the signal CDC acceptance, as defined in the text, as a function of the fiducial time window.

The first two requirements ensure there are enough hits to obtain the needed resolution in the final tracking fit. The third ensures that there is a good "seed" to initialize the full track fit.

The signal hit rate is shown in the left plot of Figure 6.14. This distribution has a kink just after 100 ns, which matches the start time for bound muon decay for the bunch arriving at $t = 0$ in Figure 6.10. A second kink kind occurs at 1170 ns, which is the latest creation time for any signal track. All hits occurring after 1170 ns are due to the drift time. The plot to the right of Figure 6.14 shows the signal acceptance in the CDC, as defined above, for different fiducial time windows.

6.4.5. Cut Optimization Study

A parameter scan was used to optimize the default values of $(t_{\text{min}}, t_{\text{max}}, t_W, t_D)$. The scan used a time resolution of 10 ns was used for the scan in the following allowed regions:

- Upper Time Threshold, $t_{\text{max}} \in [10\text{ ns}, 1170\text{ ns}]$
- Lower Time Threshold, $t_{\text{min}} \in [0\text{ ns}, (t_{\text{max}} - 10)\text{ ns}]$
- Allowed Drift Time, $t_D \in [0\text{ ns}, 530\text{ ns}]$
- Trigger Coincidence Window, $t_W = 10\text{ ns}$

For each set, fake CTH trigger acceptance, CTH_B, the background CDC occupancy, CDC_B, and signal overall signal acceptance for both the CTH trigger and CDC track quality, SIG_A, was determined. A metric, T, is constructed from these values. This metric is inspired by the statistical significance that is commonly used in optimization problems in particle physics:

$$T = \frac{\text{SIG}_A * \text{SIG}_A}{\sqrt{\text{SIG}_A + \text{CDC}_B}\sqrt{\text{SIG}_A + \text{CTH}_B}} \tag{6.1}$$

If left unbounded, the scan determines (120 ns, 1700 ns, 180 ns) as the ideal time window. While this triples the signal acceptance at relativly low background CDC and CTH cost, previous studies have shown that there are dangerous, prompt background sources that occur before 500 ns. Constraining the lower time threshold to be above (500 ns) yields (500 ns, 1170 ns, 180 ns) as the optimal set, while constraining it further (700 ns) yields (700 ns, 1170 ns, 180 ns).

These three sets imply that $t_D = 180$ ns the preferred value of t_D when $t_{\text{max}} = 1170$ ns, which lowers the upper threshold of the CDC time window from 1620 ns to 1350 ns. The contour plots in Figure 6.14 and Figure 6.13 support this conclusion. They show that signal acceptance does not improve much as the upper threshold moves past 1400 ns, while the background increases rapidly after 1300 ns.

These two new parameter sets are compared to the default set in Table 6.2. This table shows the resulting background and signal acceptance for the CDC and the CTH, as well as a name used to refer to the set. While not explored in this study, it is worth noting that proton beamline at J-PARC is able to scale the bunch intensity down or increase the bunch spacing to 1755 ns to alleviate the trigger and occupancy rates.

Parameters [ns]				Fake	Occu.	Signal			
t_{min}	t_{max}	t_D	t_W	CTH	CDC	CTH	CDC	Both	Sample Name
500	1170	180	10	44.9%	20.9%	14.4%	29.7%	10.5%	Improved Signal
700	1170	180	10	30.9%	14.0%	8.8%	22.6%	6.4%	Improved Default
		450	50	83.1%	25.8%	9.0%	23.5%	6.6%	Default

Table 6.2.: The selected parameter sets and their resulting CTH and CDC acceptances. For the CDC, background acceptance is taken as the background channel occupancy while the signal acceptance follows the definition in Section 6.4.4.

Chapter 7.

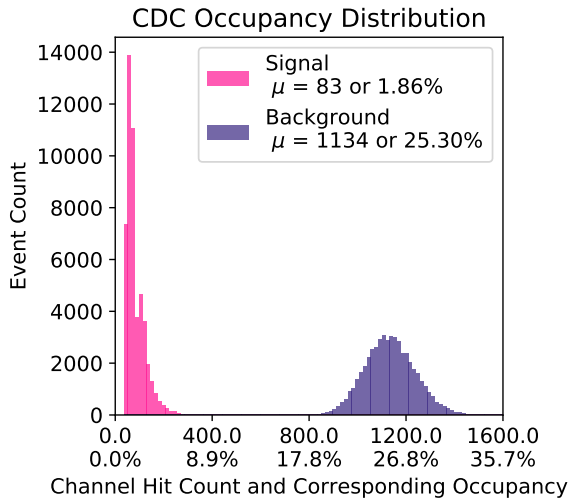
The CDC Hit Filter

The CDC track reconstruction algorithms are responsible for the final physics measurement for COMET Phase-I. These algorithms must be able to handle in the COMET's high-intensity environment while remaining sensitive to the wide range of signal track shapes accepted by the CDC. The first stage of reconstruction is designed to filter out as many background tracks as possible while maintaining a high signal acceptance.

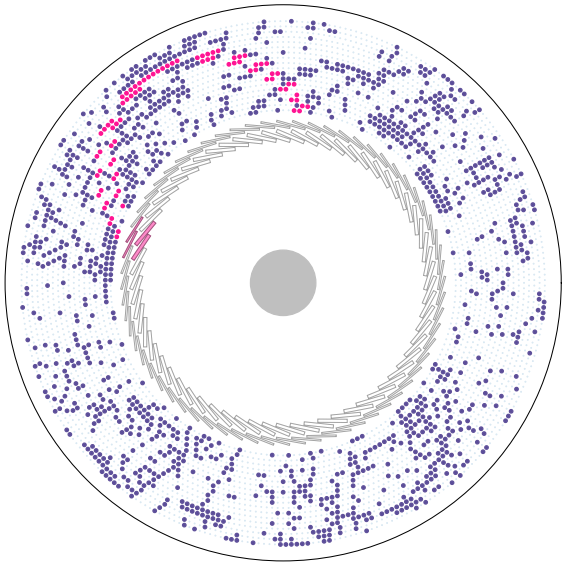
The high CDC occupancy and the stereometry make it very hard to distinguish the different track shapes in the event. To address this, the filtering algorithm operates at the hit-level to describe how likely it is that the hit came from a signal track. This reduces the first stage of track reconstruction to a hit classifier referred to as the CDC Hit Filter (CDCHF).

The full classifier has three main stages:

1. **Local Filtering**: Use as many features as possible to build a classifier that selects signal hits, discussed in Section 7.2
2. **Shape Recognition**: Score all the selected signal hits the previous stage by their ability to form a signal track with the other selected hits, discussed in Section 7.3
3. **Track Filtering**: Build a new classifier that uses the original features from the first stage and the score from the second stage to further separate the signal and background hits, discussing in Section 7.4.



(a) The CDC occupancy distribution from signal and background hits in a mixed sample.



(b) The event display of the CDC coloured by truth label.

Figure 7.1.: The CDC occupancy distribution and an event display from a high occupancy event. Both plots share the colour key, denoted in the legend in the occupancy plot to the left. This legend also provides the average number of hits and occupancy for the corresponding entries.

7.1. Data Sample Description

The simulated signal tracks are added into the background bunch trains to produce mixed events. All of the signal tracks pass the CTH trigger and CDC acceptance cuts, as outlined in Section 6.4. In the event that both background and signal hit the same CDC channel, the ADC values are summed, the hit timing is set to the minimum time of the two hits, and the signal label is kept. The same rules are used to sum CTH hits that occur in the same module within the same 10ns bin. The trigger signal is recalculated for the event after the signal is added so that both signal and/or background can participate in the four-fold coincidence pattern. The CTH trigger time is set to the time of the earliest hit of the earliest instance of the signal pattern. The sample has a total of 8.8×10^{11} POT event, as described in Section C, which results in 50,000 background bunch-train events for the default intensity of 1.6×10^{7} POT per bunch.

The distribution of signal and background occupancy can be seen in Figure 7.1a. The two distinct peaks in the signal distribution correspond to the first turn and second turns through the CDC. The long tail is from events where the signal track circles through the CDC many times. On average, there are 83 channels hit by a signal track per event. The

background occupancy follows a Gaussian distribution with a mean of 1134 channel hits and a standard deviation of 104 hits. The event display in Figure 7.1b provides a more intuitive look at these occupancy levels. This event shows a signal turn electron track in an event with 1278 background hits, meaning this event is 1.5 σ above the mean background occupancy values. Of these background hits, 40.6% comes from protons, 24.0% comes from electrons, 17.8% comes from neutrons, and the remaining 17.6% comes from pions, muons and all other radiation. The event displayed here will be used to demonstrate to the functionality of the CDCHF.

7.2. Local and Neighbouring Classification

The first stage of the CDCHF uses machine learning to classify each hit using all the features that can be defined without considering the full context of the hit in the event. This step is inspired by more traditional hit filter which typically places a lower and/or upper threshold on each feature, or perhaps a series of 2D cuts on the 2D projections of the feature space.

7.2.1. Feature Space

The lowest level features are referred to as the "local-level" features. These features describe each hit in isolation using:

- the magnitude of the ADC value of the hit, q, and
- the radial distance of the hit's channel from the centre of the CDC, $|\mathbf{x}|$.
- the delay between the recorded time of the hit and the trigger signal in the event, t_R.

While easily defined, local-level features fail to consider the hits as a part of a larger hit pattern. To this end, the "neighbour" features consider the properties of the properties of hits on neighbouring channels. These features capture the fact that signal hits in this plot are almost always flanked by other signal hits in the same layer, as is apparent in Figure 7.1b. This defines the neighbour features as:

- the magnitude of the ADC value of any hits on the neighbouring channels, $q^{(L)}$ and $q^{(R)}$, and
- the relative time, t_R, of any hits on the neighbouring channels, $t_R^{(L)}$, and $t_R^{(R)}$.

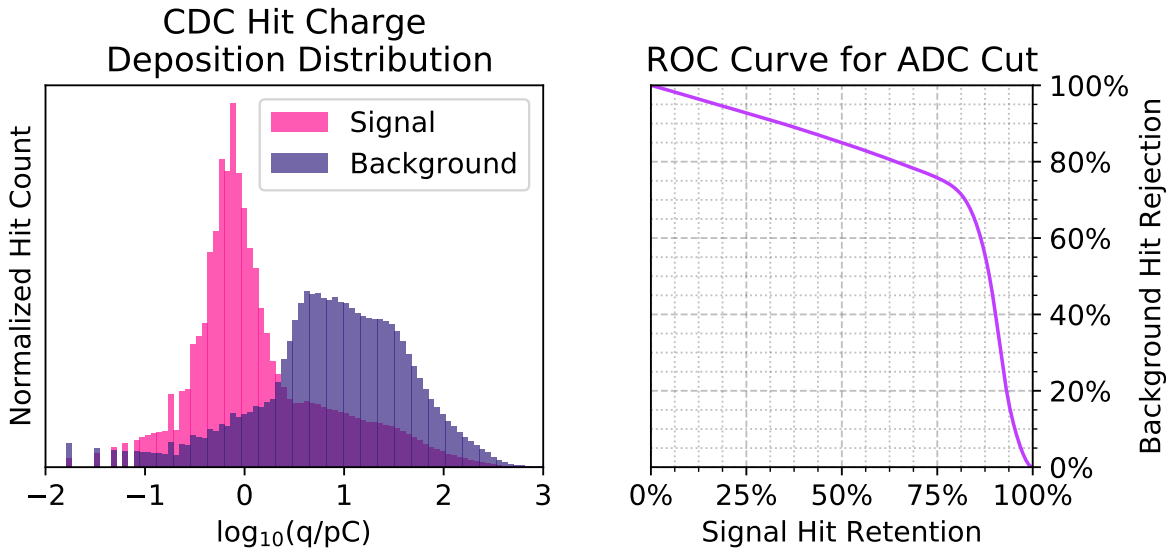


Figure 7.2.: The signal and background hit charge deposition distribution and corresponding ROC curve. To highlight the separation of the two samples, the distributions are normalized and that the x-axis is displayed on a logarithmic scale. While the detector response yields ADC values in units of e, this plot scales these values by assuming a linear signal amplification in the CDC gas, as discussed in Section 3.2.3.

Local Features

Signal electron tracks deposit far less charge on the CDC channels than most of the background tracks, as seen in the left plot in Figure 7.2. The original hit filter for the CDC placed an upper threshold on this distribution to remove background hits. The 1D cut is taken as a baseline for the filtering performance. The signal hit efficiency and background hit rejection for all possible cuts on this feature are summarized in the ROC curve to the right in Figure 7.2. This curve plots the background rejection and signal acceptance of all possible cuts on the distribution to the left. This type of plot will be used throughout this work to judge classification performance.

The "relative hit time" feature, t_R, is defined as the delay between the earliest hit in the earliest CTH trigger pattern in the event, t_T, and the detected time of the hit, t_D, as $t_R = t_H - t_T$. The distributions of this feature is seen in the left plot in Figure 7.3. The limits of this new feature are determined by the fiducial time window parameters as $[t_{\text{min}} - t_{\text{max}}, t_{\text{max}} - t_{\text{min}} + t_D] = [-470 \text{ ns}, 920 \text{ ns}]$. If a signal track causes the CTH trigger, the timing of the signal hits will closely follow the drift time distribution, while the background hits timing will form a delayed peak from the accepted hits from the next bunch. With that said, only 38% of the earliest trigger signals in the mixed events contain signal hits for the default time window parameters. The fake triggers cause the smaller

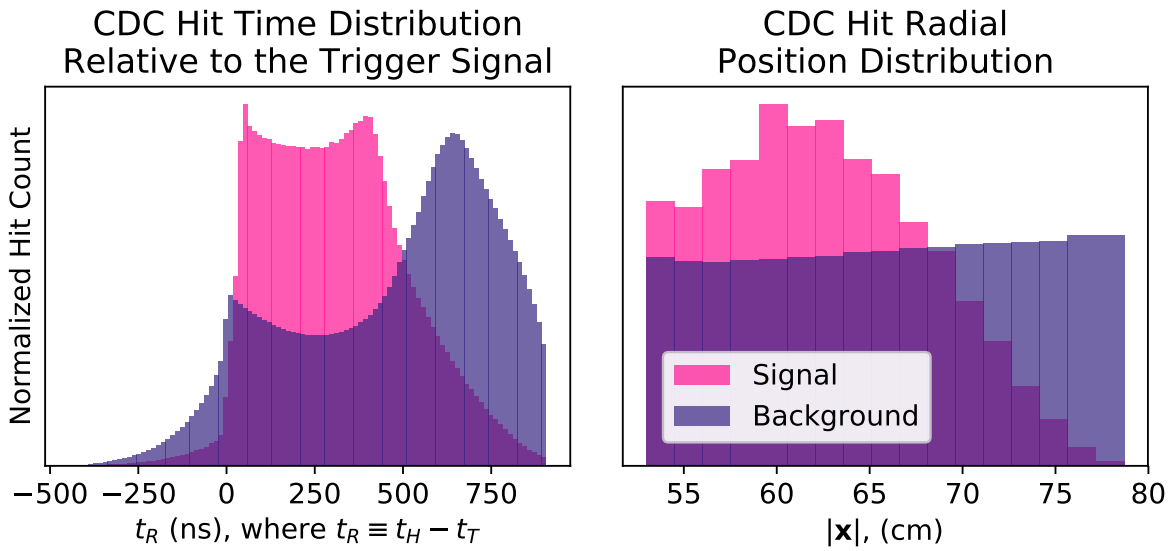


Figure 7.3.: The hit timing and radial distributions for signal and background particles. The timing distribution is taken relative to the time of the earliest CTH trigger signal in the event. The radial distribution is naturally binned to the number of layers in the CDC.

background peak at $t = 0$ seen in the plot and smears the signal hit distribution away from the drift time distribution.

The radial position distribution seen to the right in Figure 7.3 highlights that the CDC is designed to accept the full arc of the signal tracks, such that there are few signal hits in outer most layers. The flat background distribution is the result of two competing effects. The inner layers always get hit by tracks that enter from within the CDC cavity, meaning they get hit more often. The outer layers have more channels overall, meaning they can accommodate more summed CDC hits per event.

Neighbour Features

The local features of any hits on the neighbouring channels of a hit are defined as the "neighbour" features of the hit. For clarity, the hit-to-be-classified is referred to as the "labelled" hit and the hits on the channels anti-clockwise and clockwise to the labelled hit are referred to as the left and right neighbours, respectively.

When the labelled hit has no neighbouring hit, its neighbour features are set to a default "zero-bin" value. This value must fall outside the distribution of the corresponding local feature so that empty neighbour channels are not confused with ones that have a hit. For charge deposition, the zero-bin value is naturally defined as zero. For relative timing, its defined to be just outside the local-level distribution, i.e. less than $t_{\text{min}} - t_{\text{max}}$.

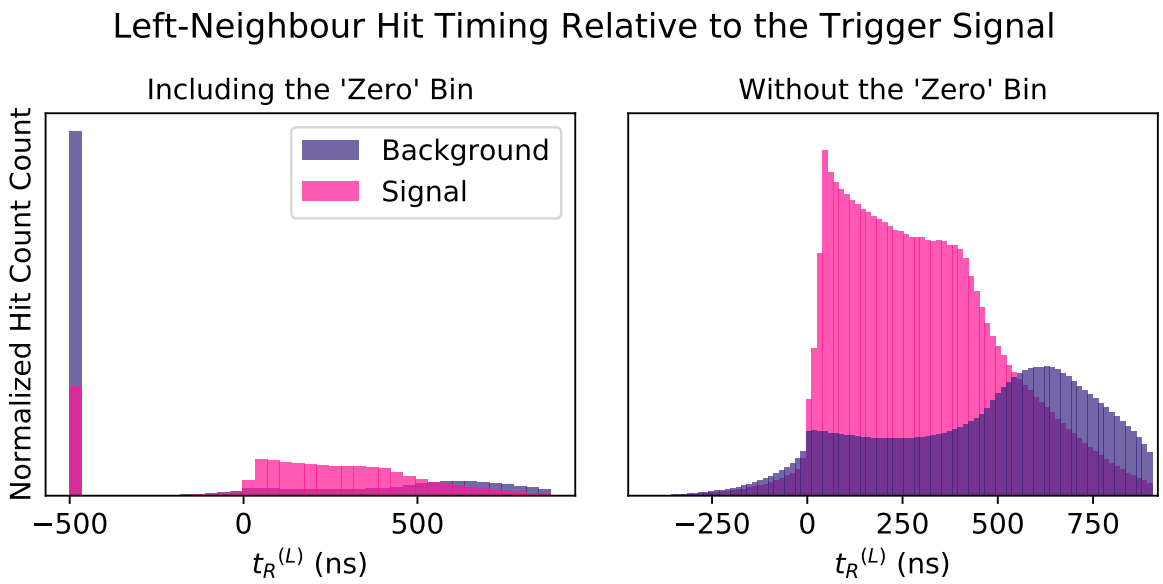


Figure 7.4.: The relative timing of the hits to the left of the labelled hit. The right plot highlights the dominant zero bin, which denotes an empty channel to the left of the labelled hit, while the right shows the distribution in all other bins.

The distribution for $t_R^{(L)}$ is plotted Figure 7.4. For the default window, the zero-bin for this feature is set to −500 ns. The plot to the left shows how this bin dominates the distribution. The zero-bin contents determine that background hits are around three times more likely to have no left-neighbouring hit than signal hits. The plot to the right highlights that the signal hits in the remaining distribution are easier to separate than in the local t_R feature. Given the symmetry of the CDC, the plots for the right-neighbour detected time indistinguishable from these plots. The neighbour charge plots have the same zero-bin values and a similarly improved distribution in the remaining bins.

In summary, the two feature groups are formally defined below. Note that the Neighbour-Level Feature set is defined to include all features in the Local-Level Feature set.

- **Local-Level Features**: $(q, t_R, |\mathbf{x}|)$.
- **Neighbour-Level Features**: $(q, q^{(L)}, q^{(R)}, t_R, t_R^{(R)}, t_R^{(L)}|\mathbf{x}|)$.

7.2.2. Classification Performance

As a first step in the hit filter, a Gradient Boosted Decision Tree (GBDT) was trained on these features to classify the signal hit points. The full algorithm was developed in PYTHON, with extensive use of the SCIPY, PANDAS, and NUMPY libraries [63, 64]. The early implementations used the `GradientBoostingClassifier` from the SCIKIT-LEARN library, which in general

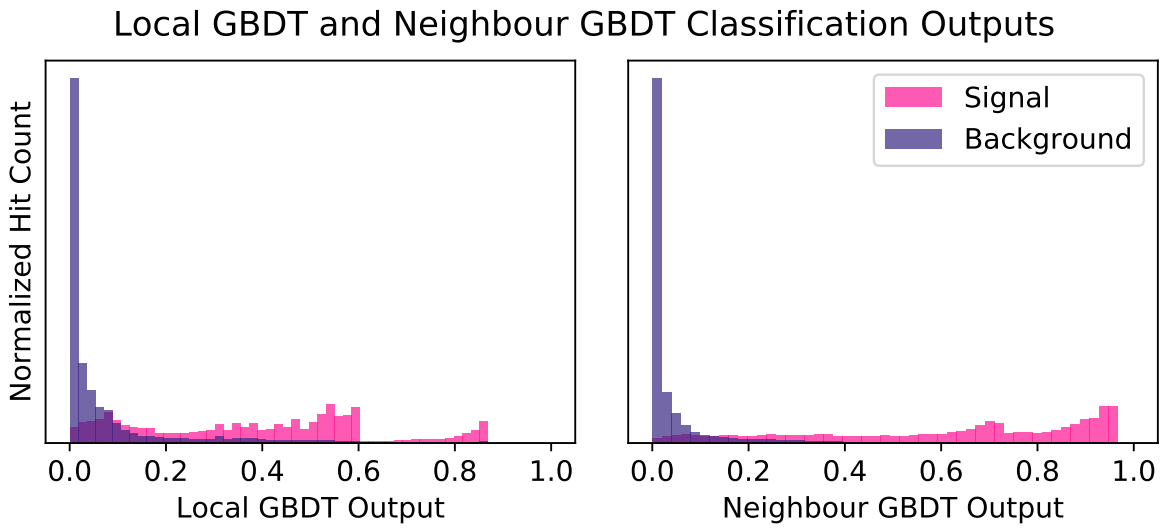


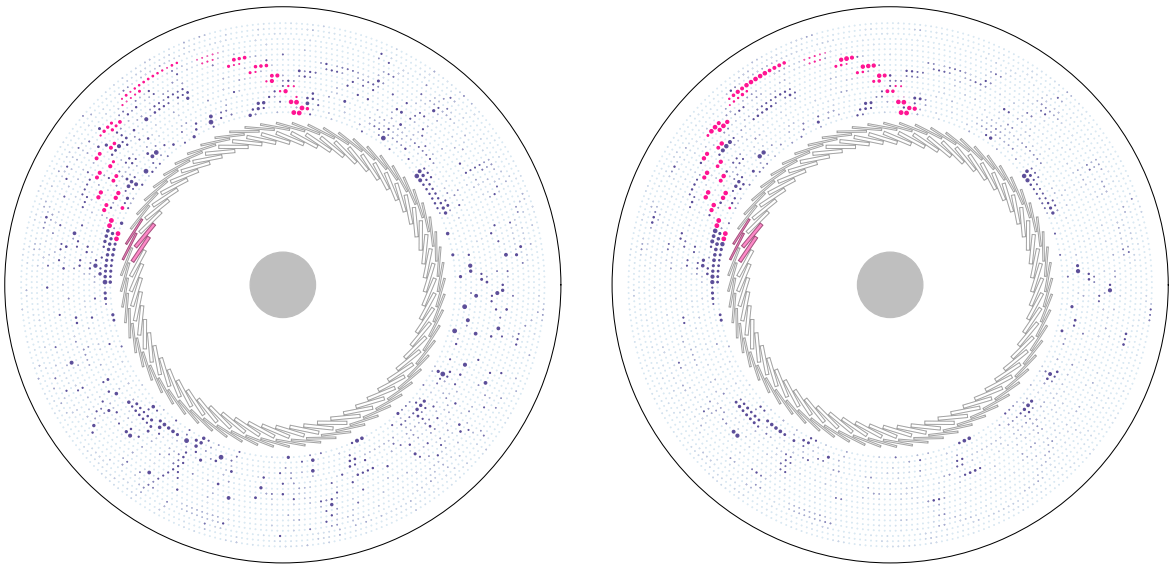
Figure 7.5.: The normalised classification output distribution of the Local GBDT and Neighbour GBDT.

is a great for library for algorithm development [65]. The final implementation uses the `XGBClassifier` from the XGBOOST library, which offers parallelization and GPU support for both training and predicting [66].

The "Local GBDT" was trained using the local-level features, while the "Neighbour GBDT" was trained using the neighbour-level feature set. The default settings of the `XGBClassifier` were used, since no significant gains were found by adjusting its hyper-parameters. As such, each GBDTs are composed of 100 trees where each tree is grown to a maximum depth of 3 layers.

12,500 training events were used for this stage, although previous results from smaller sample sizes suggest that similar performance could be obtained from as few as 2,500 events. The same number of events were used to test the classifiers. To match the definition of an accepted signal track in Section 6.4.4, the signal hits in the first three layers are weighted as ten times more important than the other hits. This penalizes the classifier ten times more for misclassifying a signal hit from the first three layers than misclassifying any other type of hit.

The output distribution of the classifiers shown in Figure 7.5 clearly show that the Neighbour GBDT significantly outperforms the Local GBDT. The output of the Local GBDT, $\hat{y}_{\text{LCL}}$, has several distinct peaks in the signal distribution. The best-separated peak around 0.8 is formed exclusively from the signal hits in the first three layers that do not have any coincident background hits. The most poorly separated signal peak between 0 to 0.2 is



(a) The output of the Local GBDT, $\hat{y}_{\text{LCL}}$.

(b) The output of the Neighbour GBDT, $\hat{y}_{\text{NGH}}$.

Figure 7.6.: The output of the Local and Neighbour GBDTs, $\hat{y}_{\text{LCL}}$ and $\hat{y}_{\text{NGH}}$, for each hit in the sample event. As in Figure 7.1b, the signal hits are colored in pink, while the background hits are blue. The size of the hit is scaled to the output of the corresponding GBDT output, with a full circle denoting a signal-like hit.

formed from signals hits get washed out by an earlier coincident background hit. The signal hits from 0.2 to 0.6 represent all other scenarios. While the Neighbour GBDT also features several peaks, the higher dimensionality of the input features makes the peaks harder to interpret. With that said, the peak from 0.7 to 1 contains the majority of signal hits from the first three layers. The ROC curves for these distributions are compared later to the performance of the full CDCHF in Figure 7.15.

To help visualize the performance of these GBDTs, the size of each hit in sample event in Figure 7.6 is scaled by its output from the corresponding GBDT. The visualized $\hat{y}_{\text{LCL}}$ scores in Figure 7.6a demonstrates a significant reduction in the CDC background occupancy. Unfortunately, some of the signal hits in the outer most layer are lost, while many isolated background hits dot the CDC channels. Figure 7.6b shows that using the $\hat{y}_{\text{NGH}}$ score alleviates these issues, but that there is still room for improvement.

7.3. The Reweighted Circular Hough Transform

The next stage of the CDCHF uses a circular Hough transform to take advantage of the dominant signal track pattern highlighted by the $\hat{y}_{\text{NGH}}$ score Figure 7.6b. Building on the

introduction in Section 5.2.3, this transform for the CDC is introduced in its scalar form as:

$$H\big(\mathbf{a}_i, \mathbf{x}_j\big) = F(|\mathbf{a}_i - \mathbf{x}_j|) \quad \text{for} \quad i \in [0, M-1], \;\; j \in [0, N-1] \tag{7.1}$$

where $\mathbf{x}_j$ denotes the 2D position of the j-th channel in the readout projection and $\mathbf{a}_i$ denotes the 2D position of the i-th point in the the discretised "Hough-space" in the same projection. Each $\mathbf{a}_i$ point is defined as the centre of a signal-like circle formed by some of the CDC channels. The radial symmetry of the transform means that H can be defined by a function F over a single, scalar argument. Defining the matrix $R_{ij} \equiv |\mathbf{a}_i - \mathbf{x}_j|$, an equivalent matrix form is defined as:

$$T_{ij} \equiv F\big(R_{ij}\big) \quad \text{for} \quad i \in [0, M-1], \;\; j \in [0, N-1] \tag{7.2}$$

This stage of the CDCHF breaks down into three steps. First, it transforms the hits into Hough-space to detect all the circular hit patterns in the event. Second, it uses the Hough-space information to identify the strongest hit pattern. Third, it determines which hits in the CDC are responsible for this pattern. This can be formalized by both matrix and scalar equations:

	Scalar Equations	Matrix Equations	
Forward Transform	$c_i = \sum_{j=0}^{N-1} F(\lvert\mathbf{a}_i - \mathbf{x}_j\rvert) \cdot w_j$	$\mathbf{c} = T\mathbf{w}$	(7.3)
Track Selection	$c_i{}' = G(\mathbf{c})_i$	$\mathbf{c}' = G(\mathbf{c})$	(7.4)
Inverse Transform	$w_j{}' = \sum_{i=0}^{M-1} F^{-1}(\lvert\mathbf{a}_i - \mathbf{x}_j\rvert) \cdot c_i{}'$	$\mathbf{w}' = T^{-1}\mathbf{c}'$	(7.5)

where c_i describes the any circular hit pattern centred around $\mathbf{a}_i$ and w_j weights the contribution from channel j to all c_i values. The function G uses all c_i values in $\mathbf{c}$ to reward the best circular patterns in the event with higher scores, which are returned in the $\mathbf{c}'$ vector. Finally, the inverse transform uses $\mathbf{c}'$ to create a new weight $w_j{}'$ that weights the hit at $\mathbf{x}_j$ by its agreement with the best patterns in the event. The $\mathbf{w}'$ vector is used in the last stage of the CDCHF algorithm.

7.3.1. The Forward and Inverse Hough Transform

The input for the forward Hough transform H is a collection of hits in the CDC, as described by the $\mathbf{w}$ vector. As highlighted in equation (7.3), this vector controls how much the j-th channel contributes to the Hough-space scores in the $\mathbf{c}$ vector. Consider the case where $\mathbf{w}$ is perfectly weighted such that $w_j = 1$ if there is a signal hit on channel j and zero otherwise.

The forward Hough transform H calculates a "score vector" $\mathbf{c}$ according to equation (7.3) to identify the any circular signal-track hit patterns. If $\mathbf{w}$ is perfectly weighted, the c_i element describes the sum of F evaluated between the track-centre $\mathbf{a}_i$ and the signal hits in the event. To identify the true track centre at $\mathbf{a}_i$ with the maximal c_i value, $F\big(|\mathbf{a}_i - \mathbf{x}_j|\big)$ should peak when $|\mathbf{a}_i - \mathbf{x}_j|$ is equal to radius of the circular pattern in the event.

Next, the function G reweights the track centre scores in $\mathbf{c}$ values so that only the best values survive in the new $\mathbf{c}'$ vector, as defined in equation (7.4). Now consider the case where the output of G is perfect, but the CDC hits are not labelled. This corresponds to a $\mathbf{c}'$ with $c_i{}' = 1$ if the true-track centre is at $\mathbf{a}_i$, and zero otherwise.

The inverse transform H^{-1} is designed to use $\mathbf{c}'$ and F^{-1} to determine which hits form a circular pattern around the true track-centre. This means $F^{-1}(|\mathbf{x}_j - \mathbf{a}_i|)$ should peak when the distance between a hit at $\mathbf{x}_j$ and the true track-centre at $\mathbf{a}_i$ is the equal to the radius signal-track pattern in the event. If $\mathbf{c}'$ is perfectly weighted, equation (7.5) defines $w_j{}'$ as F^{-1} evaluated between the true track-centre and the hit on the j-th CDC channel.

This highlights the symmetry of the circular Hough transform, where both $F\big(|\mathbf{a}_i - \mathbf{x}_j|\big)$ and $F^{-1}\big(|\mathbf{a}_i - \mathbf{x}_j|\big)$ should both peak when $|\mathbf{a}_i - \mathbf{x}_j|$ is equal to the signal radius. So long as $\mathbf{w}$ and $\mathbf{c}'$ approximate these "perfectly weighting" cases, F and F^{-1} can be approximated by the same transformation. This significantly reduces the parameter space and removes the need to approximate the left-inverse of T. The inverse equation and its matrix are then:

Scalar Equations	Matrix Equations	
$w_j{}' = \sum_{i=0}^{M-1} F(\lvert\mathbf{a}_i - \mathbf{x}_j\rvert) \cdot c_i{}'$	$\mathbf{w}' = T^T \mathbf{c}'$	(7.6)

7.3.2. The Hough Transform In 1D

The function F must accommodate the the radii of signal tracks in all events, denoted as $R^{(S)}$. The first iteration of the CDCHF fit $F\big(|\mathbf{a}_i - \mathbf{x}_j|\big)$ directly the $R^{(S)}$ distribution. This

defines F to reward a $\{\mathbf{a}_i, \mathbf{x}_j\}$ pair by the relative frequency of $|\mathbf{a}_i - \mathbf{x}_j|$ in $R^{(S)}$. This fit determined the functional for of $F\big(|\mathbf{a}_i - \mathbf{x}_j|\big)$ as:

$$F(|\mathbf{a}_i - \mathbf{x}_j|) \equiv \begin{cases} \gamma_i \exp\left(\dfrac{\big[|\mathbf{a}_i - \mathbf{x}_j| - r_{\text{peak}}\big]^2}{2{\sigma_{\text{r}}}^2}\right) + \kappa_1 & : \; r_{\text{min}} \leq |\mathbf{a}_i - \mathbf{x}_j| < r_{\text{peak}} \\ \gamma_i \left[1 + \kappa_1 - \dfrac{|\mathbf{a}_i - \mathbf{x}_j| - r_{\text{peak}}}{r_{\text{max}} - r_{\text{peak}} + \kappa_2}\right] & : \; r_{\text{peak}} < |\mathbf{a}_i - \mathbf{x}_j| \leq r_{\text{max}} \\ 0 & : \; \text{else} \end{cases} \tag{7.7}$$

$$\gamma_i = \frac{1}{\lambda + \sum_{j=0}^{N-1} F(|\mathbf{a}_i - \mathbf{x}_j|)} \tag{7.8}$$

This form is determined emperically. Between r_{min} and r_{peak}, F is a Gaussian curve with a width of σ_{r} an a peak at r_{peak}. The whole curve is raised by the constant κ_1, which is fixed at $\kappa_1 = 0.05$. This parameter lets σ_{r} focus on modelling the shape of the rising edge of $R^{(S)}$ as it approaches r_{peak} instead of the long tail of $R^{(S)}$. After r_{peak}, the function drops off linearly until it reaches zero at r_{max}. The parameter κ_2 allows r_{peak} to equal r_{max} without risking a divide-by-zero error. Its value is fixed at $\kappa_2 = 0.1$ cm, which has a negligible affect for typical r_{max} and r_{peak} values. The factor γ_i normalizes all c_i by their maximum possible values. This is normalization is controlled by λ, which is discussed later in Section 7.3.4, and is only important full 2D transform.

The optimal parameters values were determined by a parameter scan that randomly sampled the $(r_{\text{min}}, r_{\text{peak}}, r_{\text{max}})$ between $[25\text{ cm}, 37\text{ cm}]$ and σ_{r} between $[0.1\text{ cm}, 4.1\text{cm}]$ at a resolution of 10 mm. These values were blindly selected on the basis of the resulting performance of the full CDCHF. The optimal values are determined as:

$$\big(r_{\text{min}}, r_{\text{peak}}, r_{\text{max}}, \sigma_{\text{r}}\big) = (28\text{ cm}, 33.5\text{ cm}, 35\text{ cm}, 1\text{ cm}) \tag{7.9}$$

The top plot of Figure 7.7 shows that the optimal parameters for F are displaced from the $R^{(S)}$ distribution by 2.5 cm. The middle plot shows that this causes the distribution of the $\mathbf{c}$ values to "miss" the true track centres at $|\mathbf{a}_i - \mathbf{x}_j| = 0$. The bottom applies the inverse hough transform to the $\mathbf{c}'$ values to recover the $\mathbf{w}'$ distribution. The $\mathbf{w}'$ distribution agrees well with the original $R^{(S)}$ distribution, such that the initial 2.5 cm disagreement cancels out.

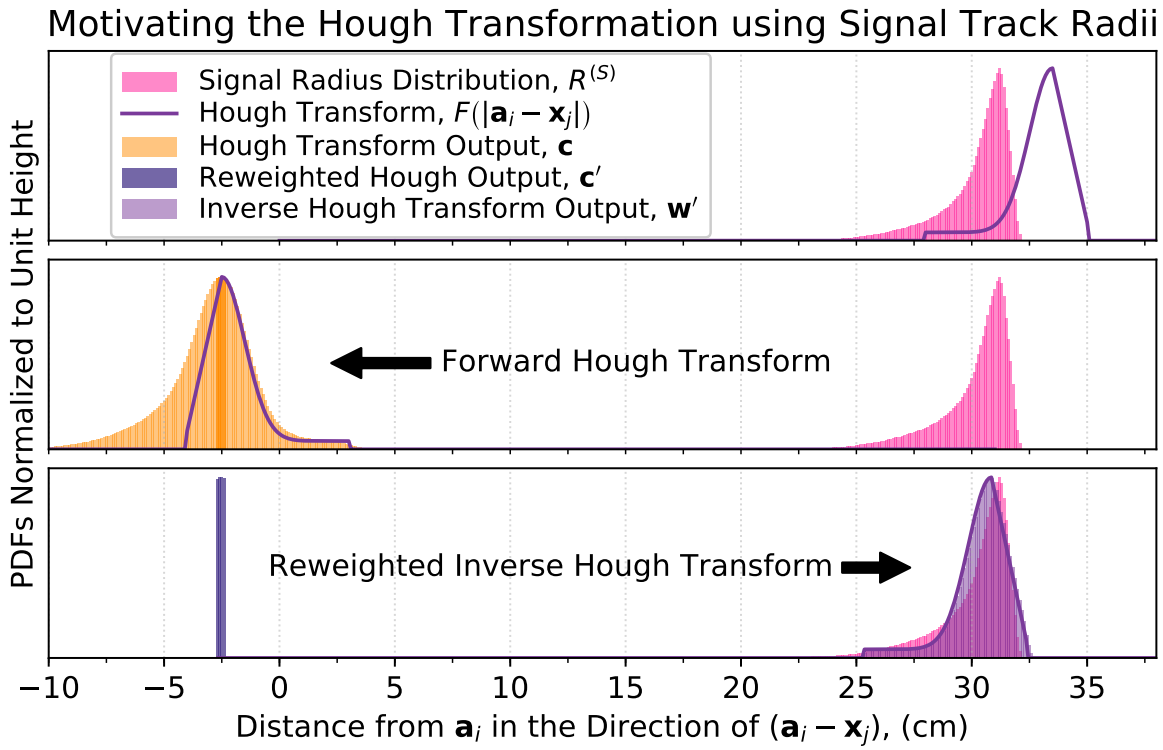


Figure 7.7.: The Circular Hough Transform in one dimension. The top plot compares the shape of the $R^{(S)}$ in pink to F in purple. The middle applies the forward transform to $R^{(S)}$ and shows the distribution of $\mathbf{c}$ values in orange, where the purple line shows the contributions from the peak of the $R^{(S)}$. The last plot shows the values of $\mathbf{c}'$ in blue and compares the the resulting $\mathbf{w}'$ distribution in purple to the original $R^{(S)}$ distribution. Note that all the distributions are normalized to unit height.

Noting that the $R^{(S)}$ is not provided to the CDCHF algorithm directly, this agreement is an excellent cross-check for the Hough transform stage. It demonstrates that of the 2,000 randomly selected parameter sets for F, the CDCHF performs best with a $(r_{\text{min}}, r_{\text{peak}}, r_{\text{max}}, \sigma_{\text{r}})$ set that is able to map $R^{(S)}$ back on to itself. This implies that the algorithm is able to recover $R^{(S)}$ implicitly from the CDC hit positions from the mixed background and signal events and use this information to better improve its classification abilities.

With this said, the functions F and G are defined such that any overall shift of the radial parameters $(r_{\text{min}}, r_{\text{peak}}, r_{\text{max}})$ will always cancel out. To prove this, define the unshifted parameters that would fit F to $R^{(S)}$ perfectly as $(r^{(S)}_{\text{min}}, r^{(S)}_{\text{peak}}, r^{(S)}_{\text{max}})$. Denote theses parameters explicitly as arguments of F and define d to describe a uniform shift. The function F admits:

$$F\left(r^{(S)}_{\text{min}} + d, r^{(S)}_{\text{peak}} + d, r^{(S)}_{\text{max}} + d, |\mathbf{a}_i - \mathbf{x}_j|\right) = F\left(r^{(S)}_{\text{min}}, r^{(S)}_{\text{peak}}, r^{(S)}_{\text{max}}, |\mathbf{a}_i - \mathbf{x}_j| - d\right) \tag{7.10}$$

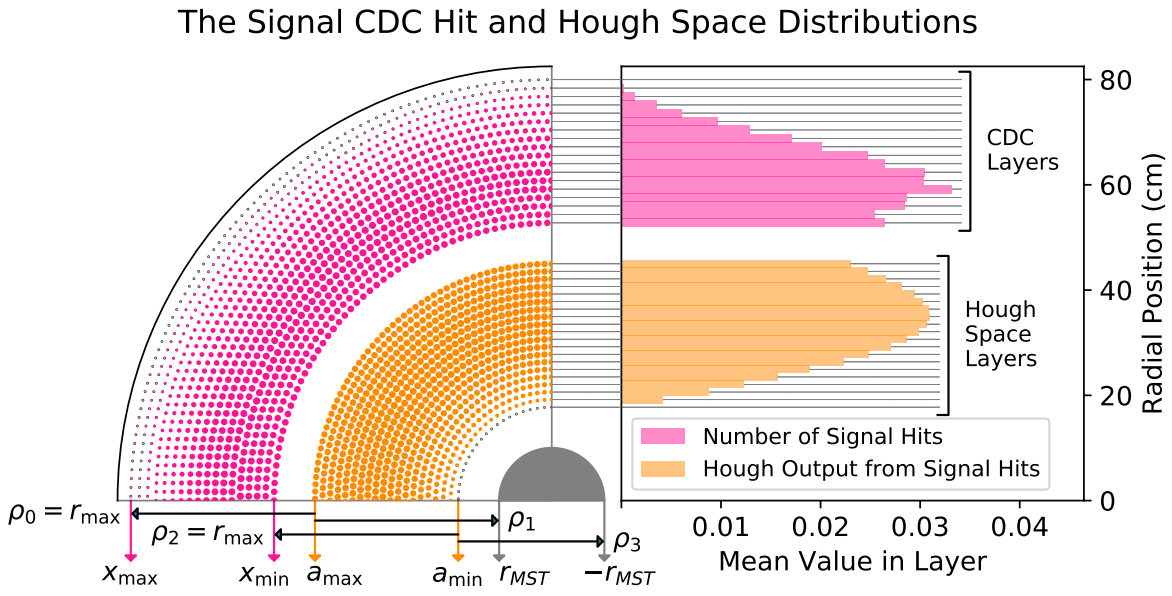


Figure 7.8.: The CDC signal hit and Hough space distributions in 2D and their normalized radial projections. The left plot shows all CDC channels $\{\mathbf{x}_j\}$ and all track-centres $\{\mathbf{a}_i\}$. The size of the pink dots denotes the 2D distribution of signal hits in all events. The size of the orange dots denotes the distribution of all c_i values from signal hits using F as defined by equation (7.9). The right plot shows the average values in the corresponding layers. The bottom of the left plot highlights how the limits of $\mathbf{a}_i$ are defined.

such that shifting the parameters of F by d is the same as shifting the argument by $-d$. The peak selected by G is shifted to $-d$, after which F is reapplied to recover a $\mathbf{w}'$ distribution that peaks at $\left[\left(r^{(S)}_{\text{peak}} - d\right) + d = r^{(S)}_{\text{peak}}\right]$. To answer why $d = 2.5$ cm is preferable to $d = 0$, the 2D discretised Hough space must be defined.

7.3.3. The 2D Discretised Hough Space

The symmetry of the CDC causes all $\mathbf{a}_i$ track centres to lay between two radial limits, a_{max} and a_{min} such that:

$$a_{\text{min}} \leq \left|\mathbf{a}_i\right| \leq a_{\text{max}} \qquad \forall i \in [0, M-1] \tag{7.11}$$

The $\mathbf{a}_i$ points are defined to have 20 radial layers and 80 points in the inner-most layer. The rest of the layers are given enough points to match the angular resolution of the first layer. The optimal limits are determined to be $[18\text{ cm}, 35\text{ cm}]$, as outlined in the next section.

The distribution of signal hits across $\mathbf{x}_j$ and their contributions to c_i at each $\mathbf{a}_i$ are visualized to the left in Figure 7.8. This plot scales each pink point by the fraction of signal

hits in the corresponding CDC channel in all events to the total number of signal hits in all events. This fraction is denoted as $\bar{w}_j$. The $\bar{w}_j$ distribution is transformed into Hough-space using F as determined from the parameter scan. The resulting c_i values, denoted $\bar{c}_i$, are mapped onto their corresponding $\mathbf{a}_i$ points. Since the $\bar{w}_j$ and $\bar{c}_i$ distributions are radially symmetric, the right plot in Figure 7.8 shows the same information as the left plot by taking the average values by layer. Note that the normalization imposed by γ_i scales the c_i values to be the same order of magnitude as $\bar{w}_j$.

The bottom axis on the left plot in Figure 7.8 denotes the radial limits of the CDC channels as $\left[x_{\text{min}}, x_{\text{max}}\right]$, the Muon Stopping Target radius as r_{MST}, and plots the four ρ parameters. The ρ parameters are useful when discussing the a_{min} and a_{max} limits. They are defined as:

$$\rho_0 \equiv x_{\text{max}} - a_{\text{max}} \qquad \rho_1 \equiv a_{\text{max}} - r_{\text{MST}} \qquad \rho_2 \equiv x_{\text{min}} - a_{\text{min}} \qquad \rho_3 \equiv a_{\text{min}} + r_{\text{MST}}$$

Determining the a_{min} and a_{max} Limits

The limits of the Hough space limits can be defined by the fact that signal tracks must originate from the Muon Stopping Target and hit the CDC. In this case, a_{min} is centre of a track that starts at $\left(-r_{\text{MST}}\right)$, loops inwards towards x_{max}, but only has enough $p_\perp$ to reach x_{min}. The B field in the CDC constrains all signal hits to occur within x_{max} such that a_{max} corresponds to the centre of a track that starts at r_{MST}, loops outwards towards x_{max}, and has enough $p_\perp$ to reach x_{max}. "These geometrical" considerations define $\rho_0 \equiv \rho_1$ and $\rho_2 \equiv \rho_3$.

The limits can also be defined by requiring that F best describes all possible signal radii. This means F must be able to reach all the CDC channels, which means a_{min} and a_{max} should be x_{min} and x_{max} shifted by the maximum possible signal radius, r_{max}. These "functional" considerations defines $\rho_0 \equiv \rho_2$. Since r_{MST} is not considered, no relations are implied for ρ_1 and ρ_3.

The values for $(r_{\text{MST}}, x_{\text{min}}, x_{\text{max}})$ are $(10\text{ cm}, 53\text{ cm}, 80\text{ cm})$. Taking $r_{\text{max}} = 35$ cm as in the parameter scan, these two sets of definitions can be formalized and evaluated:

Geometrical Limits	Functional Limits	
$a_{\text{min}}^{(G)} \equiv \dfrac{x_{\text{min}} - r_{\text{MST}}}{2} = 21.5\text{ cm}$	$a_{\text{min}}^{(F)} \equiv x_{\text{min}} - r_{\text{max}} = 18\text{ cm}$	(7.12)
$a_{\text{max}}^{(G)} \equiv \dfrac{x_{\text{max}} + r_{\text{MST}}}{2} = 45\text{ cm}$	$a_{\text{max}}^{(F)} \equiv x_{\text{max}} - r_{\text{max}} = 45\text{ cm}$	(7.13)

The key difference between the two cases is that the first assumes that the maximal $p_\perp$ is defined by distance between x_{max} and the $\left(+r_{\text{MST}}\right)$, while the second implicitly defines $\left(+r_{\text{MST}} = x_{\text{max}} - 2r_{\text{max}}\right)$ by assuming that r_{max} is needed to reach x_{max}.

Evaluating the Parameter Scan

The functional limits $(a_{\text{min}}^{(F)}, a_{\text{max}}^{(F)})$ were used during the previously mentioned parameter scan used to optimize F. This gave the algorithm as much freedom as possible when exploring this parameter space to find the optimal F and $\mathbf{a}_i$ definitions.

Its encouraging that the optimal parameter set found $a_{\text{max}}^{(F)} = a_{\text{max}}^{(G)}$. This implies the CDCHF worked best when $\rho_0 = \rho_1$, such that it was able to recover information about the target geometry without forcing this relationship. Given that $R^{(S)}$ has a maximum value of $r_{\text{max}}^{(S)} = 32.5$ cm, this also explains the 2.5 cm as the difference between $a_{\text{max}}^{(G)} - r_{\text{MST}} = \rho_1$ and $r_{\text{max}}^{(S)}$.

The selected value of $a_{\text{min}}^{(F)}$ means that the inner three Hough layers are less than the $a_{\text{min}}^{(G)}$ value implied by the geometry. While technically non-zero, the $\bar{c}_i$ values on the innermost layer are two orders of magnitude less than the next layer up. This is because F is non-zero for one or two $\mathbf{x}_j$ for each $\mathbf{a}_i$ position in the innermost layer, which can hardly be called a circular pattern. The next two layers have a substantial $\bar{c}_i$ values. Taken together, the optimal definition for a_{min} may lie somewhere between $a_{\text{min}}^{(G)}$ and $a_{\text{min}}^{(F)}$. With that said, the inner $\mathbf{a}$ layers have minimal impact on the relevant $\mathbf{w}'$ values, especially given the track quality cuts require at least 5 layers to be hit by the signal track.

7.3.4. Track Candidate Selection in Hough Space

The final element of the full Hough transform G breaks down into three steps, which defined in terms of the full Hough transform stage:

$$\begin{aligned}
&\text{Forward Transform} && c_i = T_{ij} w_j \\
&\text{Candidate Filter} && c_i^{(1)} = \frac{c_i}{c_{\text{max}}} \cdot H\left(c_i - c_{LT}\right) && (7.14) \\
&\text{Local Sharpening} && c_i^{(2)} = \frac{\exp\left(\alpha c_i^{(1)}\right)}{\Gamma_i}, \quad \Gamma = \frac{\sum_{k \in S_i} \exp\left(\alpha c_k^{(1)}\right)}{\left|S_i\right|} && (7.15) \\
&\text{Maximum Selection} && c_i{}' = \exp\left(\beta c_i^{(2)}\right) && (7.16) \\
&\text{Inverse Transform} && w_j{}' = \left(T_{ij}\right)^T c_i{}'
\end{aligned}$$

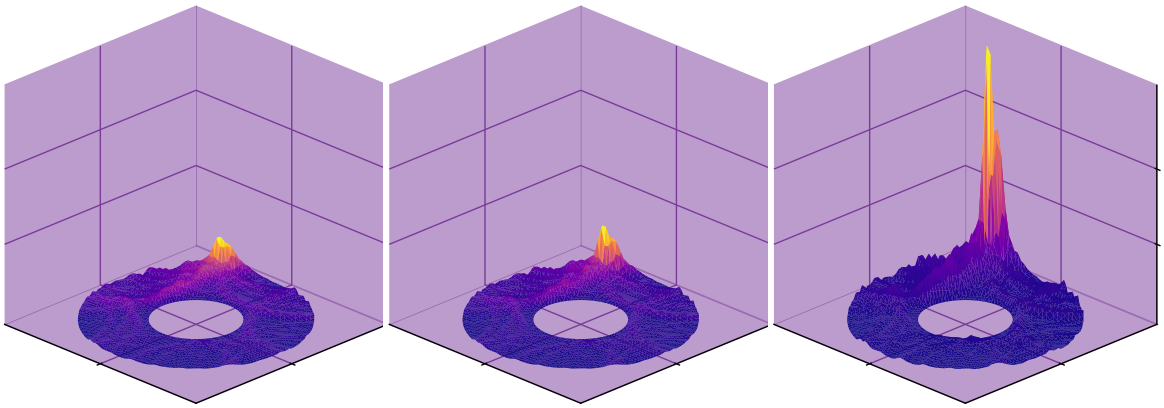
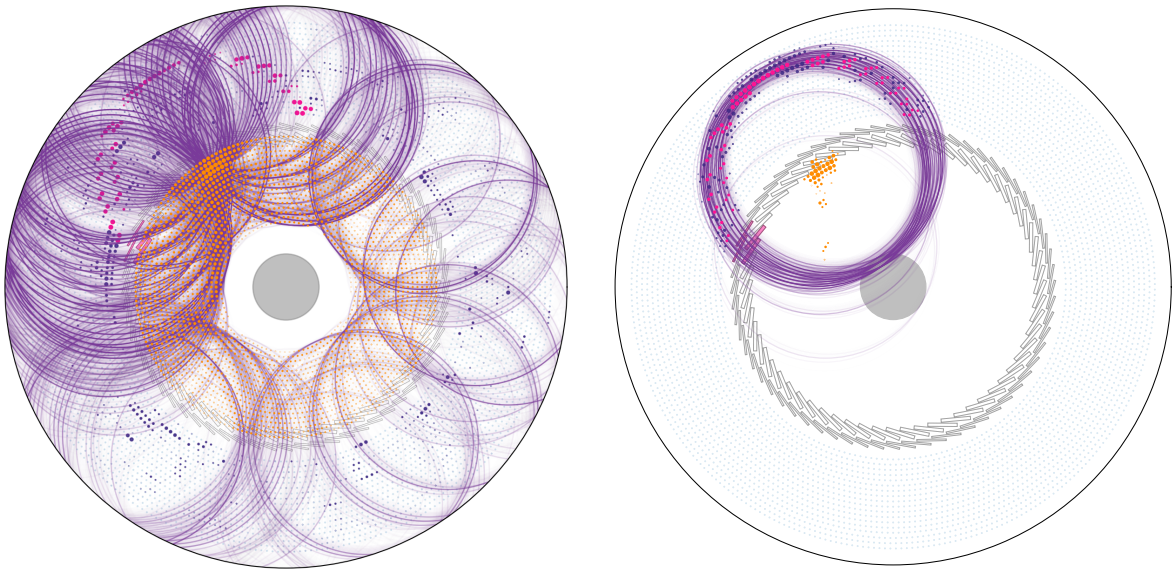


Figure 7.9.: The reweighting steps in Hough-space visualized in 3D. The x-y axes map to (a, b) respectively, while the z and colour axes are used to demonstrate the reweighting of c_i. The colour axis is normalized to the maximum in the plot, while the z axis is normalized to the maximum across all three plots. The left most plot shows the initial c_i values from the sample event. To better demonstrate the exponential weighting, c_{LT} is set to zero in equation (7.14). The second plot shows the values $c_i^{(2)}$ values and the third plot shows the $c_i{}'$ values.

The first step of the track selection defined by equation (7.14) uses a Heaviside step function to remove all c_i values below a lower threshold, defined by the c_{LT} parameter. The remaining values are normalized by $c_{\max}$, which is the maximum c_i value found in the training sample, such that the returned $c_i^{(1)}$ value is between $[0, 1]$. In practice, c_{LT} is reparametrised as the minimum percentile of accepted c_i values, $c_{MP} = c_{LT}/c_{\max}$.

The second step aims to select and reward local maxima to allow for multiple peaks in Hough space in a given event. This stage defines the set S_i to describe the points $\mathbf{a}_k | k \in S_i$ as the nearest neighbours to $\mathbf{a}_i$. For the central layers, this set is the 8 points surrounding $\mathbf{a}_i$. For the inner and outer layers, this set has 5 elements. Denoting the cardinality of the set as $|S_i|$, equation (7.15) multiplies all $c_i^{(1)}$ by a new parameter α, exponentiates the product, and normalizes this exponential at each $\mathbf{a}_i$ by its sum across the neighbours defined by $|S_i|$. This results in a new score $c_i^{(2)}$. In short, this punishes $c_i^{(1)}$ values that have a larger neighbouring value. In the limit $(\alpha \to \infty)$, this sets the $c_i^{(2)}$ to one if it is the maximum in S_i and to zero otherwise. The effects of this local sharpening can be seen by comparing the first two plots in Figure 7.9.

The third step multiplies the $c_i^{(2)}$ by a final parameter β and exponentiates the product to determine $c_i{}'$. This can be thought of as an "inverse-log" scaling, such that separations between $c_i^{(2)}$ values on a linear space are magnified to separate the $c_i{}'$ by orders of magnitude. The strength of this magnification is parametrised by β. This process kills all non-peak values in $c_i^{(2)}$ while allowing an arbitrary number of competing peak values to survive.



(a) The forward Hough Transform **(b)** The inverse Hough transform

Figure 7.10.: The forward and inverse Hough transform event displays. Figure 7.10a scales each the point on each CDC channel by the Neighbour GBDT output, $w_j = \hat{y}_{\text{NGH}}$. The pink and blue colouring denotes signal and background, respectively. The forward Hough transform for each hit is visualized using the purple circles, where the transparency of the circle scales with w_j. The orange dots in the left plot are scaled by the c_i scores and by $c_i{}'$ in the right plot, meaning they can be matched to the peaks shown in Figure 7.9. The right plot shows the inverse transform are plotted as purples circles, where the transparency scales with $c_i{}'$. Likewise, the dots on the CDC channels are scaled by $w_j{}'$.

Multiple peaks are very common in the CDC due to the dual stereo projections, making this a valuable trait.

Together with λ, defined in equation (7.8), the Hough-space reweighting parameters are $(\alpha, \beta, c_{MP}, \lambda)$. These values were included in the blind parameter scan. The full 8D set of selected values is:

$$\left(r_{\text{min}}, r_{\text{peak}}, r_{\text{max}}, \sigma_{\text{r}}\right) = (28 \text{ cm}, 33.5 \text{ cm}, 35 \text{ cm}, 1 \text{ cm}) \tag{7.17}$$

$$\left(\alpha, \beta, c_{MP}, \lambda\right) = (1.1, 20, 0.966, 5) \tag{7.18}$$

The scan chose a much higher value of β than α, implying that the local sharpening stage is not as important as the maximum selection stage.

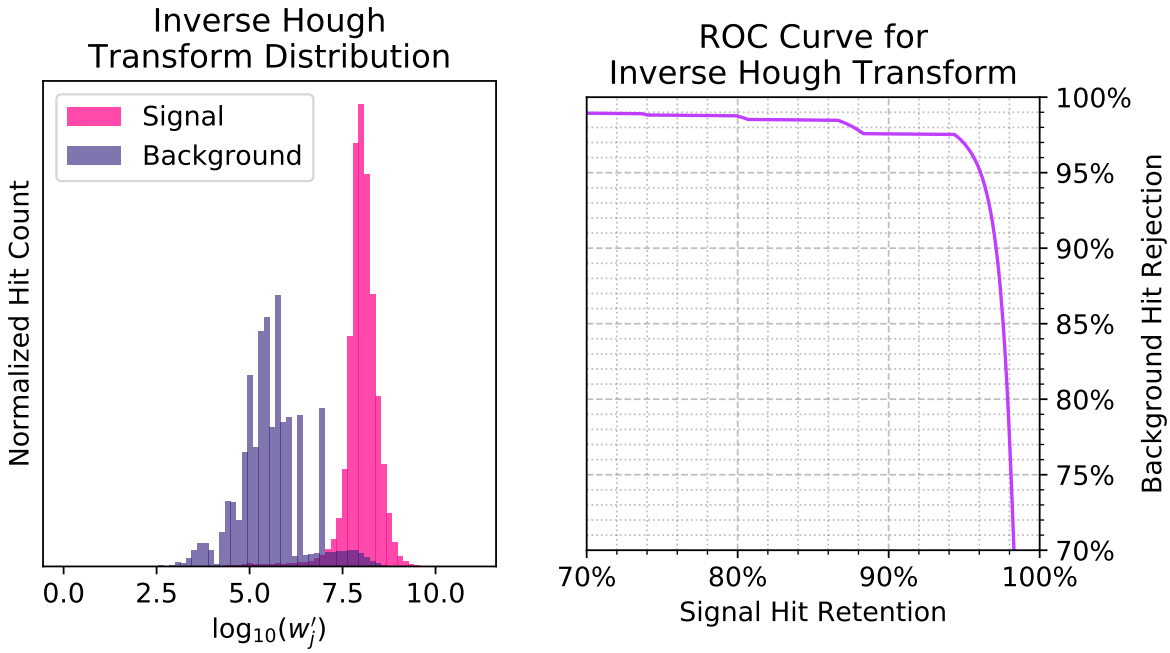


Figure 7.11.: The signal and background inverse hough transform output, $w_j{}'$, scores and the resulting ROC curve. This distribution is plotted against $\log(w_j{}')$ scale.

7.3.5. The Track-Level Feature

The forward and inverse Hough transform are displayed for the sample event in Figure 7.10. These plots highlight the simplicity of the central concepts of the transformation. Setting $w_j = \hat{y}_{\text{NGH}}$ allows the most signal-like hits to determine the highest c_i score. The true signal-hits in pink with high w_j form a circular hit pattern, while the scatted background hits with a high w_j in blue value do not. This suppresses c_i scores from the background hits in the $c_i{}'$ vector, as seen by the orange dots in both plots. This suppression is also seen in Figure 7.9, which described the same event.

The distribution of the new $w_j{}'$ feature is compared for signal and background hits in the left plot in Figure 7.11. The selected $\beta = 20$ value returns very high scores for $w_j{}'$, so this distribution is shown against $\log(w_j{}')$. Moving from right to left, this distribution demonstrates a smooth signal curve that is well separated from most of the background. The portion of background that overlaps the signal corresponds to the background hits that flank the signal hits such that they are rewarded for agreeing with the signal track shape.

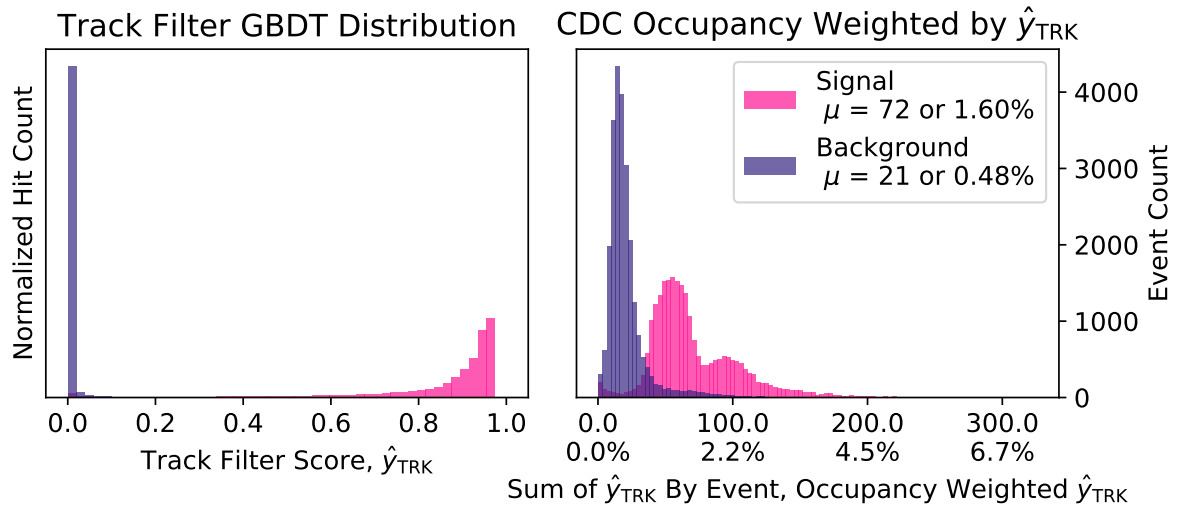


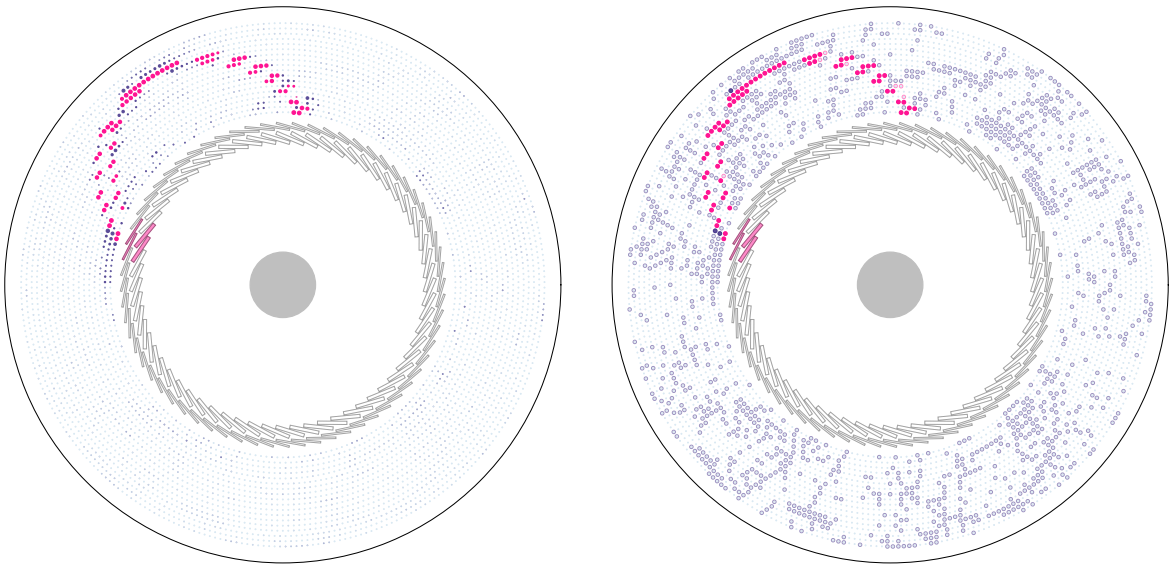
Figure 7.12.: The signal and background distribution for the CDCHF output, $\hat{y}_{\text{TRK}}$, and occupancy of all events. The weighted occupancy is defined by the sum of the $\hat{y}_{\text{TRK}}$ scores by event divided by the total number of CDC channels, i.e. 4482.

7.4. The CDC Hit Filter GBDT

Each hit can now be described by three local features, four neighbouring features, and the inverse Hough output score w_j'. Together, these eight features define the Track-Level feature set, which is used to train the Tracking GBDT, which defines the output of the CDCHF. The output scores of this GBDT, $\hat{y}_{\text{TRK}}$, can be seen in left plot in Figure 7.12. The right plot shows the distribution of the signal and background $\hat{y}_{\text{TRK}}$ scores summed by event. Dividing this sum by the number of channels defines the "weighted occupancy" of the event. Comparing this plot to the true occupancy plot in Figure 7.1a illustrates that the CDCHF is able to reduce significantly reduce the number of background hits in all events, although the tail demonstrates that there are some events that maintain high background levels.

7.4.1. Event Displays

The results of this classification are displayed for the sample event in Figure 7.13. The left plot scales each hit by its $\hat{y}_{\text{TRK}}$ values. The $\hat{y}_{\text{TRK}}$ scores for most of the background hits are too low to show up on this plot, but there is a small cluster in the first layer that can still be seen. The right plot sets a lower threshold of k on $\hat{y}_{\text{TRK}}$ such that all hits with $\hat{y}_{\text{TRK}} > k$ are labelled as signal hits. The cut value k is taken such that 85% of signal hits survive the cut.



(a) CDCHF output **(b)** CDCHF output after threshold cut

Figure 7.13.: The output of the CDCHF for each hit in the sample event, where the signal hits are pink and the background hits are blue. The left plot scales the size the hit by its CDCHF output, $\hat{y}_{\text{TRK}}$. The right shows the output of the GBDT after a lower threshold is used to remove all hits where $\hat{y}_{\text{TRK}} < k$. The hits that pass this threshold have filled circles while those that do not have empty circles

7.4.2. Feature Evaluation

Three metrics can be used to summarize the relative importance of each feature in the GBDT:

- **Gain**: the improvement in separation when a feature is used to split a node.
- **Coverage**: the number of samples affected by the feature.
- **Weight**: the number of times a feature was used to split a node in the GBDT forest.

The plot to the left of Figure 7.14 shows these metrics for each feature in the CDCHF. Each metric normalized to their maximum across all features so that they can share a common axis. Unsurprisingly, $w_j{}'$ is the most important feature by all three metrics.

After $w_j{}'$, the weight determines the local feature group as more important than the neighbour feature group. This means the track-level feature incorporates all the needed shape information for signal hit patterns, which reduces the utility of checking for any neighbour-level patterns. The gain and coverage tell a different story, ranking q, $t_R^{(R)}$ and $|\mathbf{x}|$ as the most important features after $w_j{}'$. While it is not immediately obvious why $t_R^{(R)}$

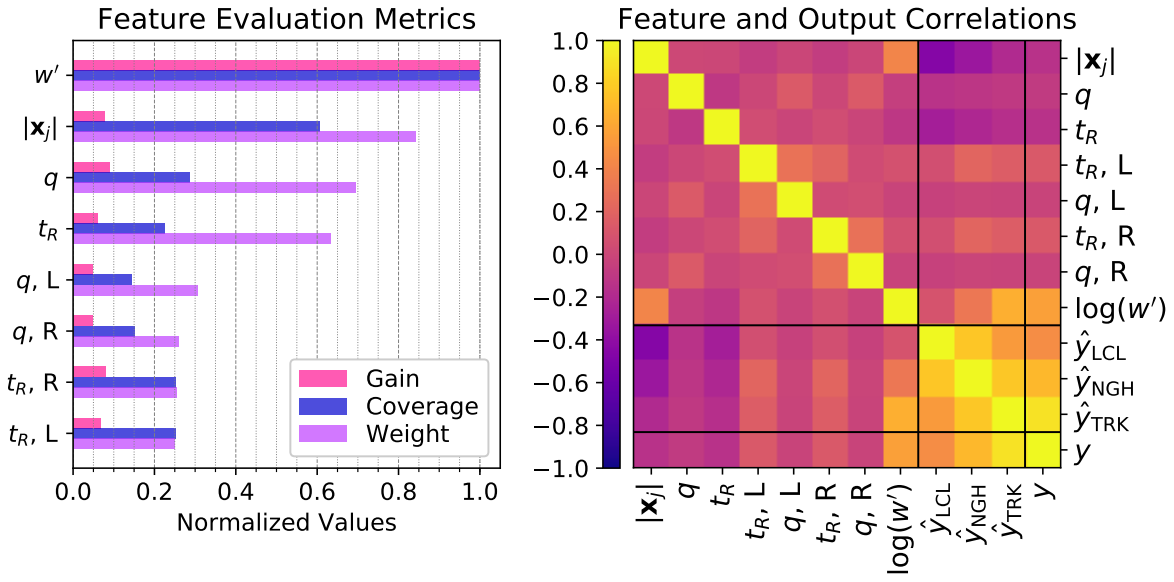


Figure 7.14.: The feature correlation matrix and evaluation metrics. The left plot shows the gain, coverage, and weight of each feature used in the CDC Track Filter. Each metric is normalized to its maximum value across all features. These metrics themselves are defined in the text. The right plot shows the correlations between all features and GBDT scores.

is preferred to t_R, it can be noted that this group contains one charge feature, one time feature, and the radial feature. This highlights the importance of all three fundamental features of the hit and implies that each one brings valuable information when classifying the hit.

Along similar lines, the right plot in Figure 7.14 shows the feature and score correlations. This plot considers the logarithm of ${w_j}'$ to move it back to a linear scale so that it can be compared with the other features. The strong negative correlation between $|\mathbf{x}|$ and $\hat{y}_{\text{LCL}}$ implies local-level classifier prefers to keep any hits in the inner layers, which highlights the effects of weighting the signal hits in the first three layers as ten times more important than the other hits. As more features are included in the classifier, the Neighbour and CDCHF scores show that this correlation is controlled.

7.4.3. Performance Evaluation

The CDCHF introduces machine learning to that are novel to tracking to significantly reduce the background occupancy of the CDC events. In doing so, it aims to complement the more traditional track reconstruction techniques that work well for a limited number of hits, but do not scale well in high occupancy environments.

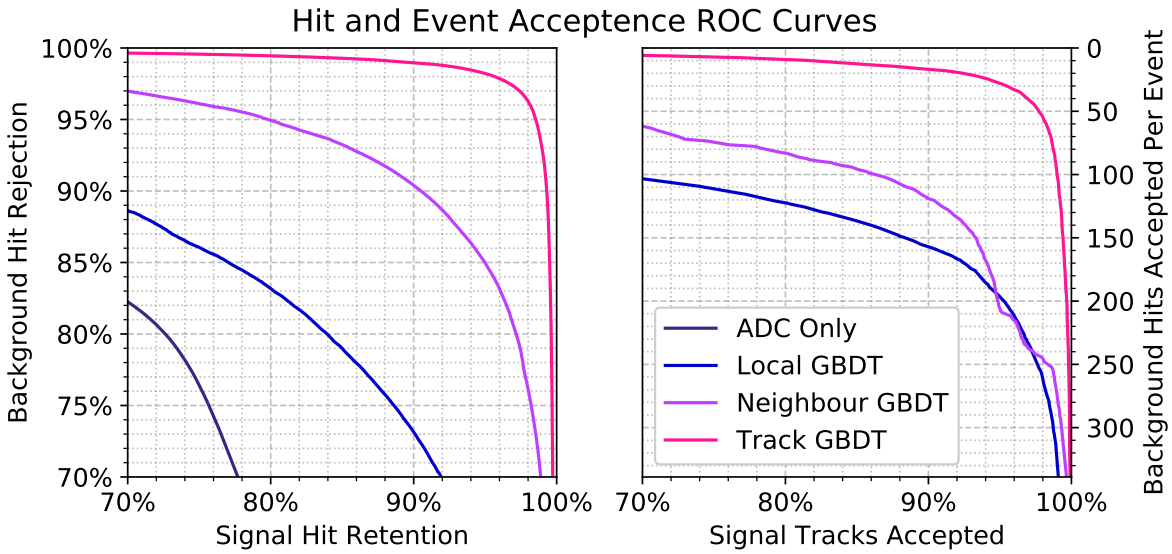


Figure 7.15.: The final output ROC curves for the baseline ADC classifier, Local GBDT, Neighbour GBDT, and CDCHF. The left plot measures the performance of the algorithms in terms of their ability to classify signal and background hits. For each implied cut on the left ROC curve, the right ROC curve plots the fraction of signal events that still pass the CDC quality cuts to the average number of background hits that remain in these accepted events.

The main strength of this approach is its ability to filter hits by simultaneously consider their features, their immediate neighbours, and their agreement with track-level patterns. This is best demonstrated by the ROC curves in Figure 7.15. The left plot shows the performance of the algorithm at the hit level. Given that this is simulated data, these curves are likely an optimistic estimate of a similar algorithm's performance on data. With that said, they clearly demonstrate that hit filtering improves as more features are considered. This inspires the strongest conclusions from this study: using more features to filter hits results in better performance. This conclusion should extend to real data.

This algorithm can also be treated as a track-finding algorithm instead of the hit filtering stage of a larger algorithm. To do so, all hits in an event that pass the $\hat{y}_{\text{TRK}}$ threshold cut k define the signal track, while the rest are removed. Figure 7.15 compares the number of signal tracks that still pass the quality cut for a given k value to the average number of background hits remaining in the accepted events. The left region of the plot corresponds to high purity cuts, while the right region corresponds to high signal acceptance.

These curves determine the CDCHF as the best track finding algorithm of the four considered. Note that the baseline ADC-cut classifier cannot be seen on this axis scale. Sampling this curve, cuts on this GBDT would allow for the signal track acceptance and

background hit acceptance to be (90%, 17 hits), (95%, 28 hits), or (99%, 100 hits). If this were truly the only step in track finding, a higher purity is preferred. With that said, most track fitting methods incorporate some outlier detection abilities, which means a higher acceptance cut could be used.

Its worth noting that the Neighbour GBDT dips below the Local GBDT in the high acceptance region. This dip is caused by the track quality cuts, where the minimum layer requirement prefers tracks that enter the CDC in a perpendicular direction to the inner CDC wall. These tracks leave isolated signal hits in the inner layers and long chains of signal hits in the outer layers. The Neighbour GBDT is trained to identify isolated hits as background, which means it struggles to satisfy the track quality cut that requires signal hits in the first three layers. The ten-fold reweighting of signal hits in the first three layers aims corrects for this somewhat, but not completely.

7.4.4. Hardware and Computation Timing Metrics

As a final note, the timing metrics for this algorithm are discussed. The full algorithm requires the Neighbour GBDT, the inverse reweighted Hough transform, and the final Tracking GBDT. The Neighbour GBDT and Track GBDT are run on an NVIDIA Titan Xp GPU and respectively take 153 μs and 181 μs per event. The transform stage runs on an AMD 1920X and takes 1.8 ms per event, although would most likely run an order of magnitude faster on a GPU. Together, each event takes 2.13 ms per event so that the algorithm runs processes 468 events per second.

Chapter 8.

Online CDC Hit Filtering for the Track Trigger

The CDC will implement a tracking-trigger to complement the CTH trigger. The CDC trigger must operate within the bunch time window of 1170 ns, or equivalently at a rate of around 850 kHz. This chapter outlines how concepts from the CDC Hit Filter can be adapted to an online environment to form a new algorithm called the Online CDC Hit Filter (OCDCHF). This filter can serve as the first level of the full CDC track-trigger.

8.1. The Trigger Environment

While not finalized, the DAQ systems are aiming to accommodate an overall trigger rate of around 30 kHz, which roughly means a fake trigger rate of 30 kHz/850 kHz ≈ 0.035, or 3.5% of events. For the default timing window and a loose cut on the scintillator ADC values, the CTH triggers on 30%–80% of background events, as seen in Figure 6.12. This means the CDC track-trigger must be able to reject between 88% and 96% of the fake trigger accepted by the CTH to achieve the overall 3.5% fake trigger rate.

To form a trigger decision, the CDC data flows through several levels of electronics. The first level is the CDC data acquisition system which is made up of 104 "RECBE" boards [67]. Each board is wired to an average of 43 channels. These boards convert the analogue pulses from the drift chamber wires to digital signals to determine the ADC values of the hits. To meet the rate requirement of the trigger, the RECBE boards reduce the ADC resolution before passing it to the dedicated trigger electronics.

The first level of the dedicated trigger electronics is made from 10–20 “COTTRI” boards, where the precise number has yet to be determined. This level processes the low-level ADC information to produce the higher-level data needed for the CDC trigger. This data is transferred to the master trigger board, which provides the overall CDC trigger decision.

8.2. Bonsai Boosted Decision Tree

Borrowing from the offline version, the OCDCHF considers the integrated ADC value of each hit, q, and the integrated ADC value on its left and right neighbours, $q^{(L)}$ and $q^{(R)}$ to classify each hit as signal or background. To satisfy the data rate requirements, the trigger system considers these values at a significantly reduced resolution. This can be leveraged to create a light-weight version of a GBDT called a Bonsai Boosted Decision Trees (BBDT).

The BBDT classifier was originally developed for the LHCb trigger [68], although similar algorithms are found in most of the popular gradient boosting libraries. This method starts by reducing the resolution of the ADC values by histogramming them into n_b bins. Each hit is described by the three bin indexes (b, b_L, b_R) that correspond to its original $(q, q^{(L)}, q^{(R)})$ values. This contains the the feature space of all possible hits to $(N_b \equiv n_b \times n_b \times n_b)$ possible bin index combinations.

Next, a GBDT is trained over the (b, b_L, b_R) features and its output for each of the possible N_b bin index combinations cached in a lookup table. This lookup table defines the BBDT classifier, such that new hits can be classified by using their (b, b_L, b_R) values to fetch the cached output from the BBDT lookup table. This procedure works well on FPGAs, which often implement lookup tables to describe complex functions.

By binning the features, the BBDT may lose some performance compared to a GBDT trained over unbinned features. This performance loss will shrink as n_b grows, but the size of the lookup table grows as n_b^3. The goal is to select an n_b that minimizes both the performance loss and size of the lookup table.

8.3. Parallelizing the Classification

Naively, any CDC trigger algorithm must analyse all 4,482 channels to reach a decision. The problem can be parallelized for the BBDT since each channel only needs information about its immediate neighbours to be classified. Implementing the BBDT lookup tables on the RECBE boards would by breaking each event into 104 pieces, resulting in an overall

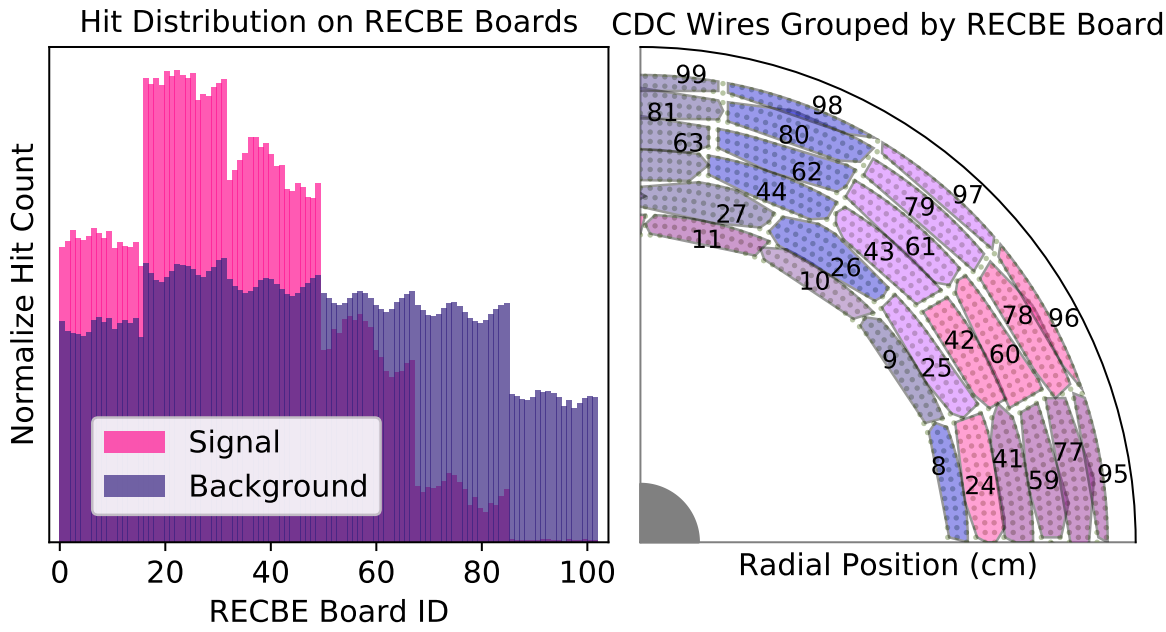


Figure 8.1.: The hit distribution across the RECBE boards, accompanied by an event display of the RECBE wire grouping. The left plot shows the number of signal and background hits in each RECBE board, where the x-axis denotes the board ID. The right plot grouping wires by RECBE boards ID in a section of the CDC, and overlays the corresponding board ID.

speed-up of around two orders of magnitude. Similarly, implementing the BBDT on the on the COTTRI FE boards would provide a factor 10–20 speed up.

Parallelizing on the RECBEs also offers another feature to the classifier: the "board ID" of the RECBE board, B_{ID}, which enumerates the RECBEs from 0 to 103. The board IDs provide some positional information since channels readout by the same board are naturally close to each other in (x, y), as demonstrated in the right-side plot Figure 8.1. If the BBDTs are implemented on the RECBE boards, this feature is "free" in the sense it does not increase the size of the lookup table on each RECBE board. Instead, each implemented BBDT would be trained to classify the wires on the corresponding board such that each BBDT would be unique.

The left side plot in Figure 8.1 shows that some separation can be achieved using the board ID. This distribution captures the radial distribution of signal and background hits reduced to 6 bins. There is also a notable variation within each bin for the background distribution, implying a slight angular asymmetry in the background distribution. This mirrors the asymmetric stopped muon profile, seen previous in Section 6.2, Figure 6.4.

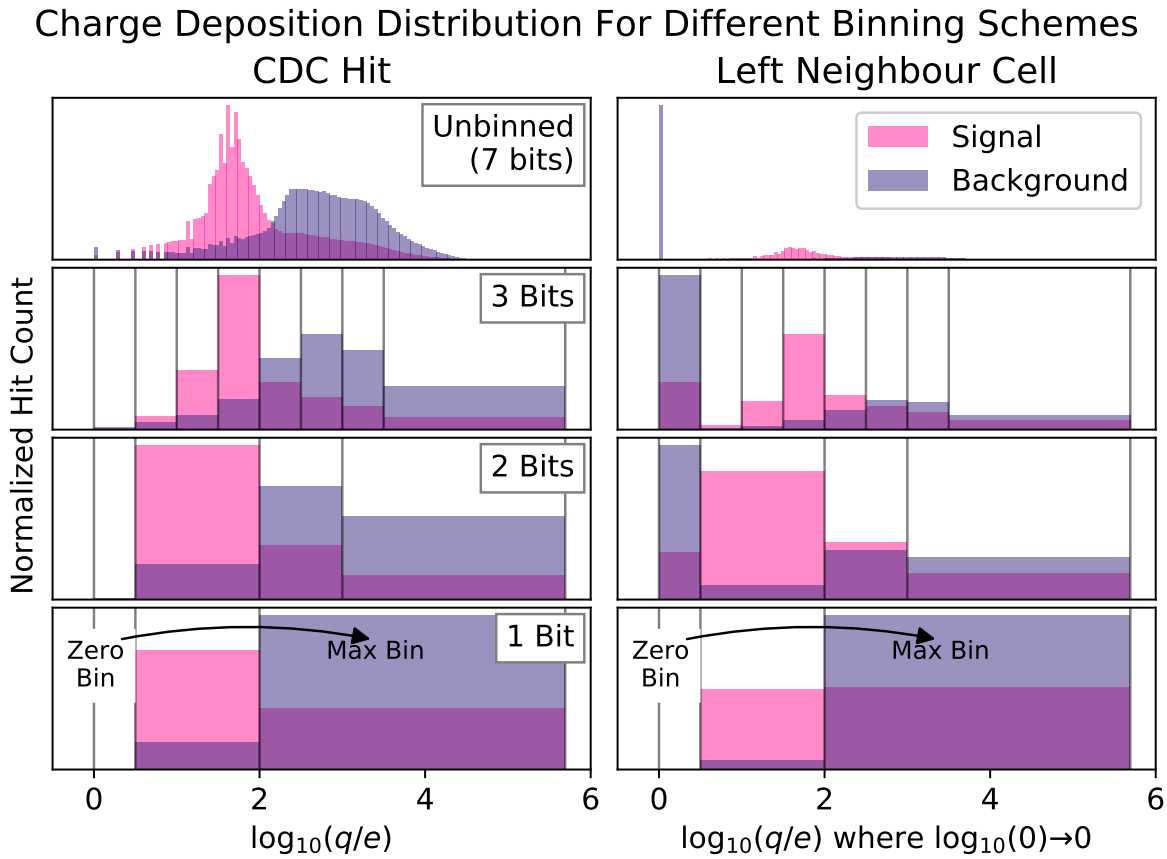


Figure 8.2.: The ADC distributions of the hits, shown in the left column, and their left neighbours, shown in the right column, for different binning schemes. The ADC values are left in the unscaled units of e returned by the detector simulation. The top row represents the unbinned feature distribution, although the histogram in the plot itself has $2^7 = 128$ bins. The vertical lines in the remaining plots demonstrate the binning scheme defined for the q values in the trigger electronics. Noting the log scaling of the axis, any $q = 0$ neighbouring values are mapped to the zero bin. To maintain only two bins for the 1-bit case, any values in the zero bin is mapped to the maximum bin.

8.4. The Binned Energy Deposition Features

The CDC trigger algorithm aims to use as few bits as possible to represent the ADC values. This study considers the minimal 1, 2, and 3-bit representations. The number of bins available doubles for each added bit, such that these cases correspond to $n_b = 2$, $n_b = 4$ and $n_b = 8$. The unbinned feature space is compared to three potential binning schemes in Figure 8.2. The left plots shows the rebinned q distribution and the right plots show the rebinned $q^{(L)}$ distributions, which also represent the $q^{(R)}$ distributions due to the symmetry of the CDC. Noting the logarithmic x-axis, all $q = 0$ values in b_L are plotted at $x = 0$.

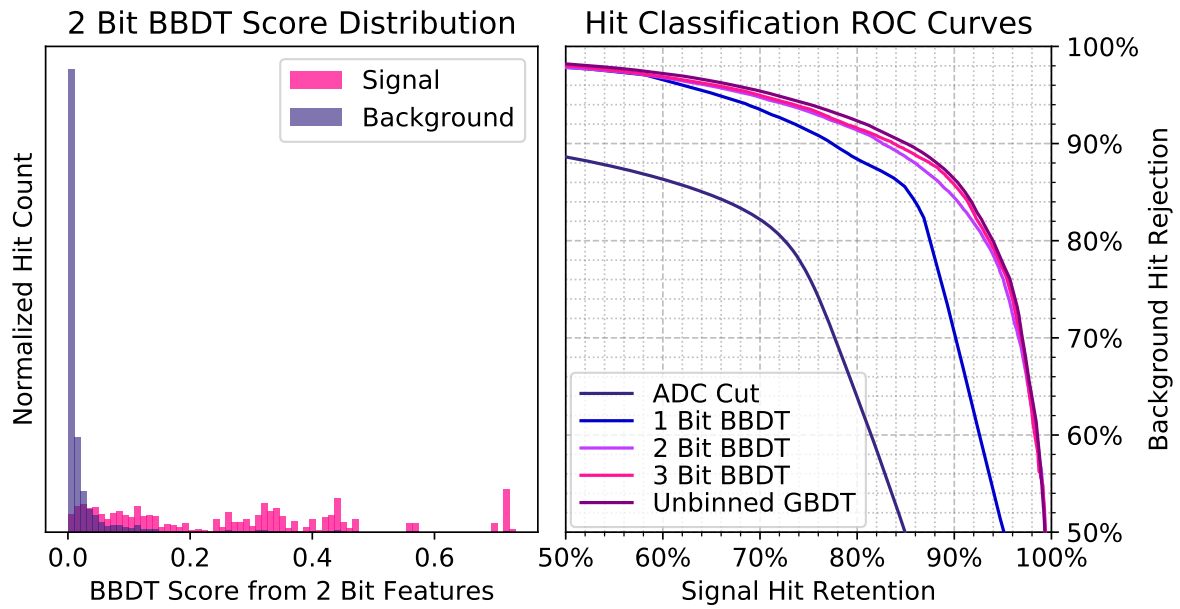


Figure 8.3.: The performance of the BBDT hit classifiers method. The left plot shows the output from a BBDT trained over the 2 bit ADC features. The right plot compares the performance of the BBDT for all three binning schemes. It also shows the performance of a GBDT classifier trained over the unbinned ADC features, which represents the upper limit for the BBDT performance. Similarly, the baseline, ADC cut classifier shown on the plot represents a lower limit of hit classification performance.

The locations of the bin edges for the three are chosen using this plot. Formally, the bin features and edges are defined as:

$$\text{Bin Features} \quad b^{(N)} \equiv i \quad \text{such that} \quad E_i^{(N)} \leq q < E_{i+1}^{(N)} \tag{8.1}$$

$$\text{1 Bit Edges} \quad E^{(1)} \in \left\{0,\ 3e,\ 100e,\ q_{\max}\right\} \tag{8.2}$$

$$\text{2 Bits Edges} \quad E^{(2)} \in \left\{0,\ 3e,\ 100e,\ 1000e,\ q_{\max}\right\} \tag{8.3}$$

$$\text{3 Bits Edges} \quad E^{(3)} \in \left\{0,\ 3e,\ 10e,\ 31e,\ 100e,\ 310e,\ 1000e,\ 3100e,\ q_{\max}\right\} \tag{8.4}$$

where $q_{\max}$ is defined at infinity. For the one bit case, all $b^{(1)} = 0$ and $b^{(1)} = 2$ values are put in the same bin, as shown in Figure 8.2.

8.5. Classification Performance

Five classifiers were created to determine the optimal number of ADC bits needed for the BBDT hit filter. Each classifier is defined by its feature set:

- "ADC Cut": a cut-based classifier on the (q) distribution.

- "1 Bit BBDT": a BBDT trained on $(b^{(1)}, {b_L}^{(1)}, {b_R}^{(1)}, B_{\mathrm{ID}})$
- "2 Bit BBDT": a BBDT trained on $(b^{(2)}, {b_L}^{(2)}, {b_R}^{(2)}, B_{\mathrm{ID}})$
- "3 Bit BBDT": a BBDT trained on $(b^{(3)}, {b_L}^{(3)}, {b_R}^{(3)}, B_{\mathrm{ID}})$
- "Unbinned GBDT": a GBDT trained on $(q, q^{(L)}, q^{(R)}, B_{\mathrm{ID}})$

The ADC cut classifier determines the lower limit of online hit filtering. The unbinned GBDT represents the upper limit of the ADC, neighbouring ADC, and board ID feature set. The BBDT and GBDT classifiers assume that the all neighbour information is available for all hits, which ignores the fact that each RECBE/COTTRI board reads out subset of the total wires, as highlighted to the right plot of Figure 8.1, meaning that some neighbor information will be missing.

The ROC curves in the plot to the right of Figure 8.3 demonstrate the performance of the three proposed binning schemes for the default timing window. The 2 Bit BBDT, 3 Bit BBDT, and Unbinned GBDT curves show that very little is gained by including more than 2 bits. While the 1 Bit BBDT classifier does not perform as well, its definition maintains the simplicity of a 1D cut-based classifier cut while capturing the performance of the neighbour features. Essentially, the 1 Bit BBDT would apply the 1D cut to each hit, and then consider if the hit and its neighbours pass the cut. With this in mind, its relative improvement over the ADC Cut classifier shows that considering neighbouring hits at any resolution provides an inexpensive avenue to improve performance.

8.6. Event Triggering

The hit-level classification provides a strong starting point for the CDC track trigger. This classifier is designed for the RECBE or COTTRI boards, which could use the filtered hits to perform some shape-based triggering algorithm. A potential candidate is:

- Calculate the BBDT score for each wire on the RECBE boards.
- Use the COTTRI boards to calculate the average BBDT score for each RECBE board to create a new "board score" so that RECBE boards with more signal hits have a higher score.
- Pass all RECBE scores to the master trigger board, which computes a light-weight circular Hough transform using the RECBE board scores, in analogy to the forward transform in the offline case.

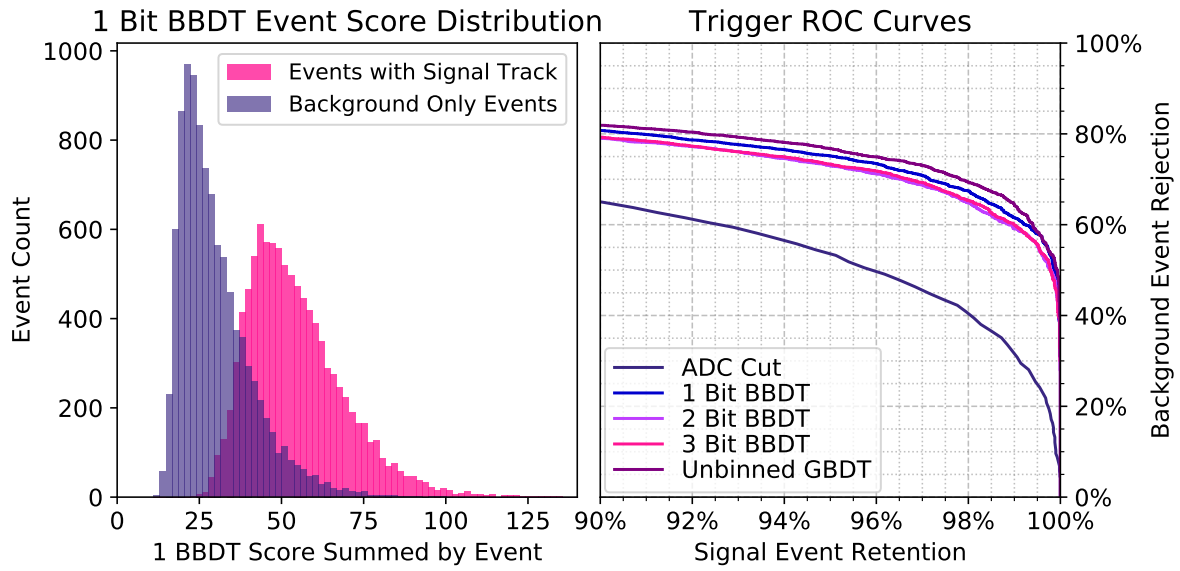


Figure 8.4.: The summed BBDT trigger performance. The left plot shows the distribution sum of the 1 Bit BBDT scores for background only events and for events with both background and a signal track. The right plot shows the ROC curves for a CDC trigger algorithm that only considers the sum of the BBDT scores. Note that the axes have different limits to highlight the high signal acceptance region.

- Accept events that show a peak in the Hough space.

This approach is still in development.

For now, a more naive approach is considered that classifies each event by the sum of all of the BBDT scores in the event. While this approach does not meet the needed CDC trigger performance, it aims to highlight the improvement of the BBDTs over the baseline ADC cut using event-level metrics. To this end, the classifiers were evaluated over 10,000 background-only events and 10,000 mixed background and signal events. The resulting classifier scores were summed for each event.

The plot to the right of Figure 8.4 shows the resulting trigger performance. The summed ADC Cut classifier counts the number of hits in each event with $q < 100e$ to classify an event. The rest take the sum of scores of each hit to classify the event. As expected, the ADC Cut and unbinned GBDT classifiers define the upper and lower limits of performance. The 2 Bit and 3 Bit curves are indistinguishable. Interestingly, the summed 1Bit BBDT score performs best in this scenario, even though it uses less information than the 2 and 3 Bit cases. This implies the threshold used for the 1 Bit BBDT is effective at removing noise for this naive triggering algorithm. With that said, this is not guaranteed to be true for a more complex algorithm.

Chapter 9.

Conclusion

COMET Phase-I is part of the next-generation of high-sensitivity particle physics experiments. Its novel design will allow it to push the frontier of high-intensity physics in search of the Charged Lepton Flavour Violating μ-e conversion process. The search is an excellent probe for Beyond the Standard Model physics and can indirectly measure energy scales that are orders of magnitude higher than those available to direct searches.

The ambitious intensity of the COMET is part of a growing trend in particle physics. The LHC has similarly ambitious plans to increase its intensity, with plans to triple the overall luminosity in the next decade [69]. As the intensity increases, the experiments will need to improve their data acquisition and analysis systems at every level, especially at the track reconstruction level. These challenges are well met by the recent machine learning renaissance, where algorithms are designed to improve in performance as more data becomes available.

This work provides a proof-of-concept algorithm that integrates machine learning into the lowest level of track reconstruction for the COMET experiment. It is designed to complement traditional track reconstruction algorithms that perform well in lower intensity environments. This work aims to encourage the larger particle physics community to adopt similar approaches to meet the coming high-intensity challenges. With that said, there are still several open questions that need to be addressed before this algorithm can be used on real data.

9.1. Summary of Thesis Achievements

The primary achievement of this thesis is the CDC Hit Filtering algorithm. This algorithm mixes the Hough transform, which is the first automated track identifier used in particle

physics, with similarly well-established classification techniques found in machine learning. By filtering out undesirable hits, it can reduce the intensity of the tracking environment so that more traditional tracking algorithm can be used. As such, it aims to build on the success of existing algorithms by leveraging all the information available at the lowest level of track reconstruction. Ultimately, a successful implementation of this algorithm will allow COMET to push its intensity much further than the baseline ADC cut-based hit filter would allow.

In this summary, it is important to note that while this algorithm is more complex than traditional, cut-based approaches, it is far simpler than the cutting-edge of computer vision in machine learning, i.e. Convolutional Neural Networks. Its balance between simplicity and performance aims to ease the burden of applying it to real data.

Elements of this algorithm are reformulated to define the Online CDC Hit Filter. This algorithm is designed to work well on an FPGA so that it can form the basis of the CDC Track Trigger. Both the filtering algorithm and triggering algorithm are still in the early stages of development. Even at this early stage, the Online CDC Hit Filter demonstrates it can significantly out perform more traditional filtering techniques without introducing unwieldy levels of complexity.

Finally, this work established the large-scale simulation pipeline for the COMET collaboration. This produced the first production-scale simulation data samples, which provide four orders of magnitude more statistics than were previously available. These samples are used groups throughout the COMET collaboration to support numerous research and development avenues. This work also introduces an automated testing pipeline for ICEDUST, which helps ensure the stability and accuracy of all future simulation and analysis efforts.

9.2. Impact on the COMET Experiment

The single event sensitivity of the COMET experiment is determined as [38]:

$$\text{S.E.S} = \frac{1}{t_{\text{run}} R_{\mu/p} \mathscr{B}_{\text{capture}} \left(I_p/e\right) A_{\mu\to e}} \tag{9.1}$$

where t_{run} is the total time of the run, $R_{\mu/p}$ is the muon stopping rate per POT, $\mathscr{B}_{\text{capture}}$ is the branching ratio for nuclear muon capture in Aluminium, I_p is the current of the proton beam, and $A_{\mu\to e}$ is the overall signal acceptance. Assuming a 6 month run time, the first four of these parameters for Phase-I are taken to be $t_{\text{run}} = 1.5\times10^{7}$ s, $R_{\mu/p} = 4.675\times10^{-4}$,

Selection	Acceptance	Comment
Time + Geometry	**6.6**%	Accounts for the three below.
Time	30%	Discussed in Section 6.4.2.
Triggering	30%	Discussed in Section 6.4.3.
Track Quality	78.3%	Discussed in Section 6.4.4.
Online Track Trigger Efficiency	**90**%	Target design value.
Data Aquision Efficiency	**90**%	Target design value.
Tracking Efficiency	**99**%	Target design value.
Signal Momumtum Window	**93**%	Discussed in section Section 2.3.3, 103.6 MeV $< p_e <$ 106 MeV
Total	**4.9**%	Accounts for all above values in bold.

Table 9.1.: The breakdown of the overall signal acceptance for COMET Phase-I. Note that p_e referes to the reconstructed momentum of a signal electron. The "target design values" are taken from the COMET TDR.

$\mathscr{B}_{\text{capture}} = 0.61$, and $I_p = 0.4$ μA. Extending Table 6.1, the signal acceptance for the whole experiment can be broken down into its components, as seen in Table 9.1.

Taken together, all these values imply a total single event sensitivity of $\mathscr{O}\left(10^{-15}\right)$. This thesis aims to aide the COMET experiment in realising this sensitivity by reducing the background as much as possible while maintaining a high signal-acceptance. Section 6.4.5 concludes that the timing cuts may be optimized to improve the "Time + Geometry" acceptance from 6.6 % to 10.5 % with little impact on the background acceptance. If the backgrounds are tolerable, this could improve the overall signal acceptance from 4.9 % to 7.8 %. If they are not tolerable, Section 6.4.5 highlights a different optimization that would reduce the background rates with little effect on the signal acceptance.

The CDCHF and the OCDCHF aim to control the background levels so that the COMET experiment can select the cuts that maximise its overall signal acceptance. These algorithms will factor into "Online Track Trigger Efficiency" and "Tracking Efficiency" elements respectively in Table 9.1. Unfortunately, the final tracking fitting algorithm and track triggering firmware must be implemented before these two efficiencies can be calculated. This means the impact of the CDCHF and OCDCHF cannot be yet be quantified in the context of the final single event sensitivity. Similarly, the full analysis of the impact on these algorithms on the overall systematic error in the final momentum measurement must also wait for the final elements of the track fitting algorithm to be implemented. With that said, both

of these algorithms out-perform the ADC cut-based classifier that was part of the initial designs for the tracking and triggering systems, as discussed in Chapter 7 and Chapter 8.

9.3. Comments on Generalisation

The COMET tracking systems are required to identify a single, mono-energetic electron. The CDC hit filter is implemented for COMET with this goal in mind, but the concepts behind the algorithm can be generalised to other use cases. The key elements of the algorithm are:

- **Local Filtering**: Use a classifier to separate "signal hits" from "background hits", preferably using as many features as possible to describe each hit.
- **Shape Recognition**: Score each hit by its agreement with the shapes made by "signal hits" identified in the previous stage. Any track-finding or track-fitting algorithm can be used here, although ones that return a continuous score are favoured.
- **Track Filtering**: Design a new classifier that uses both the shape recognition score and the local filtering features to further separate the "signal" and "background" hits.

This method requires that some hit-level features are available, but beyond that is easily generalisable by adjusting the definition of "signal" and "background" hits. For example, if an experiment was interested in muon tracks that travel in straight lines, all muon hits are identified as signal, while the rest become background. A linear Hough transform can be used to detect the needed straight-line patterns. If an experiment is interested in multiple particle types, then multiple hit filters can constructed and applied in parallel.

9.4. Applications to Real Data

The main question that remains is: how will this algorithm work on real data? This method uses supervised learning, which requires a dataset in which the hits are already labelled by signal and background. The real data will not be labelled, and may be significantly different from the simulated data. This can be alleviated by:

- using the real CDC data to tune the simulated data,
- producing a data sample which is rich with electrons that make signal-like tracks in CDC, or
- creating machine learning algorithm that can learn from unlabelled data.

The real solution to the problem will likely involve all three of these steps. A common approach in LHC physics is to train machine learning methods using simulated signal events and background events from data[1]. This would require an accurate detector simulation of signal tracks such that the hit timing and ADC waveforms agree with data.

With that in mind, the second option can be achieved by lowering the strength of the magnetic field in the Detector Solenoid so that the DIO electrons can reach the CDC. This would provide some insight to electron tracks that curve through the CDC, with the caveat that the dynamics of the drifted particles changes with the magnetic field strength. Even so, the DIO electrons could provide a valuable reference for the algorithm's performance on real data.

The third option describes an algorithm that can deal directly with unlabelled data. Algorithms that cluster completely unlabelled data are referred to as *unsupervised* learning, which is reviewed in [71]. These algorithms aim to define distinct groups in the data set based on their features. They are significantly more challenging than supervised methods, but assuming that one could group signal-like hit and background like hits, an unsupervised algorithm could effectively replace the local filtering stage. The shape recognition stage would act overall distinct groups in the event to find the group that best matched they hypothesised track shape.

Alternatively, a *semi-supervised* algorithm could be used, which are able to combine labelled and unlabelled data when creating a classifier. These methods are reviewed in [72]. Such methods could provide a promising framework for using both real and simulated data together when training the algorithm.

9.5. Further Work

Even before real data is considered, there are several areas that should be addressed. The first is that CDC hits should be described as ADC waveforms that result from the charge pulses in the detector. The current treatment integrates this ADC waveform, which means coincident hits are summed into one hit. By studying the waveform, the algorithm would get closer to the lowest level data available and could possibly resolve two distinct hits on the same channel. Overall, this should improve the algorithm.

At the same time, the crosstalk effect between channels is currently ignored. This refers to the induced charge on neighbouring wires as a charge-pulse travels down a sense wire

[1] As an example, this was done by the author in [70] for the LCHb collaboration.

towards the readout board. Once it arrives at the readout board, there is some board-level crosstalk, where some voltage leaks on to other channels on the board. This correlates the ADC values and hit times of neighbouring channels, which would hurt the algorithms ability to use neighbouring features to separate signal from background. With that said, this size of the ADC pulses and their coincident timing follow a very regular pattern. Optimistically, a classifier reads the whole ADC waveform would be able to identify this pattern and distinguish it from the correlations that are described by the neighbouring features.

Appendix A.

Continuous Integration and Deployment

The Continuous Integration and Deployment elements of ICEDUST were added by the author to support this tracking work and ease the development overhead for future collaborators. These changes include adding a test suite and ensuring that these tests run automatically as a part of the existing development workflow. This workflow is based on a GitLab instance run by CCIN2P3. GitLab is a web-based front end for git where developers access and update the latest official copy of ICEDUST [38].

A.1. The Testing Suite

These tests aim to cover all functionality utilized in the mass production simulations. They fall into two categories:

- Build tests: ensure the code can compile. This is handled by the compiler.
- Unit tests: each function is provided with a test input. The output is checked against an expected result.
- Validation tests: higher level routines, which rely on many functions, are provided test inputs. The output is checked against an expected result.

These tests groups are designed to complement one another such that the cause and effect of any bugs can be determined quickly and accurately. The advantage of the testing suite is that developers can be confident that their new feature does not introduce new bugs into the software.

A.2. Continuous Integration and Deployment using Docker

Continuous Integration (CI) with automated testing includes these tests in the existing workflow in ICEDUST, whereby developers change their personal copy of the software when adding features, then open a "Merge Request" so that their changes are added to the official copy of the software. With Continuous Integration, the software tests are run automatically and remotely every time a Merge Request is opened or updated. The results are then reported to both the developer and the code reviewers. A diagram of this workflow can be seen in Figure A.1. This enables developers to fix low-level bugs as they occur and allows the code reviewer to focus on the higher level code-quality considerations.

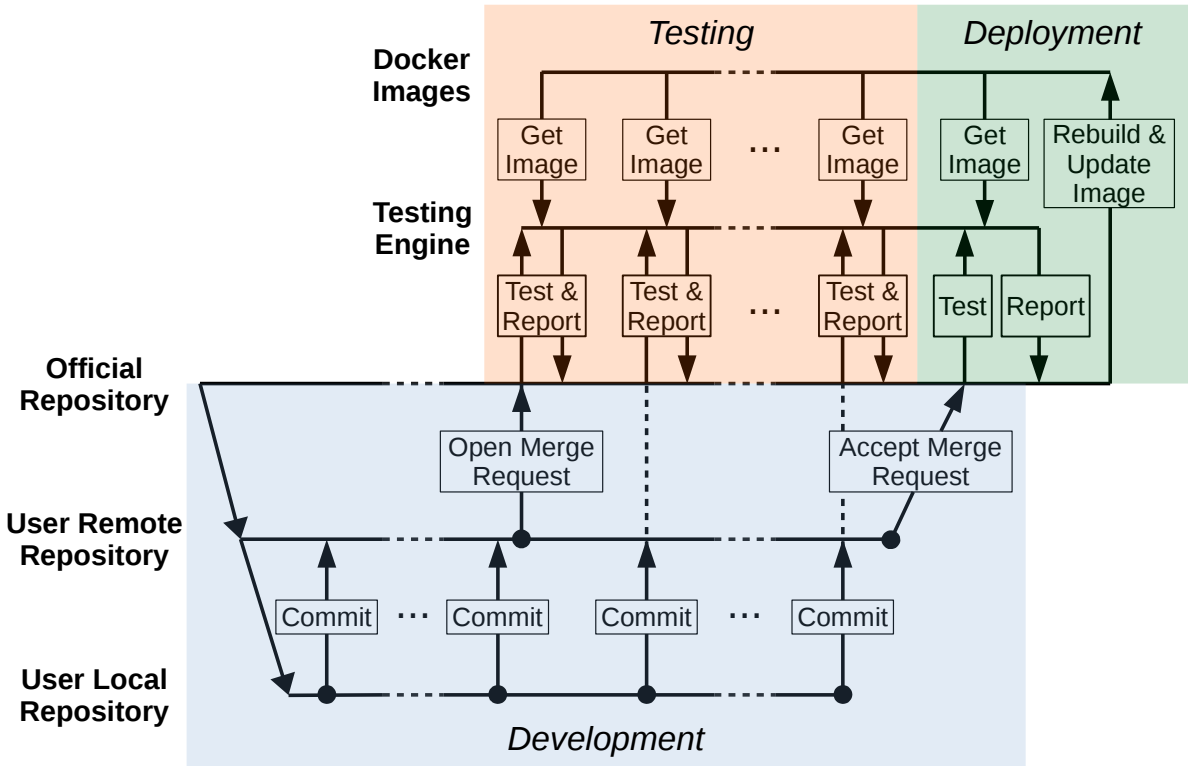


Figure A.1.: A diagram of the ICEDUST development and testing workflow using Continuous Integration and Continuous Deployment. The existing workflow is shown in blue, which is the part that requires user input. The orange element shows the testing, which is done remotely and automatically on GitLab. The green element shows the deployment, which provides all users with a virtual image of the latest working copy of ICEDUST and is also done automatically.

This functionality was implemented on GitLab using Docker. Docker is a relatively new but widely used virtualisation platform used for environment management. Docker provides

an interface to build and run virtual machines such that faithful copies of production environments can be easily run on any computer. A base image that contains all the needed packages for ICEDUST is created and hosted by GitLab. Each time the continuous integration is triggered by a merge request, this image is retrieved, ICEDUST is compiled inside the image, and the tests are run. Any error messages are reported via the GitLab web interface. This is done automatically, without any input from the developer or code reviewers.

GitLab and Docker can also be used for Continuous Deployment, which automatically builds and distributes the latest version of ICEDUST. This process is triggered whenever a merge request is accepted and the official copy is updated. This process starts with one more run of automated testing. If the tests pass, the base image is updated, rebuilt, and pushed to GitLab. This image will be used in future automated testing cycles. Similarly, the latest version is compiled on top of the base image in a new deployment image, which is also pushed to GitLab. This image can be used to run ICEDUST anywhere that uses Docker.

Appendix B.

The Geant4 Physics Lists

The physics models selected for this work prioritize accuracy over computational performance. This is to ensure that the results of the tracking studies are as accurate as possible, and to ensure that these first samples serve as a benchmark for accuracy for later samples.

B.1. Geant4 Physics Lists

In Geant4, particle trajectories are determined by breaking the track into steps, which are terminated by a physics process. To decide which physics process to use, Geant4 asks all the relevant processes for a proposed step length. The processes then each calculate their step length based on a random sampling of their probability distribution. This distribution often involves some characteristic interaction length or decay time. The process that returns the shortest length is selected to occur. If the particle is stopped, then the process that returns the shortest time is chosen.

When using Geant4, the user is responsible for registering all the physics processes he or she would like simulate [48]. These processes are all added to the physics list, which is then consulted whenever a particle is stepped through the simulation. ICEDUST uses a combination of pre-built physics processes that come with Geant4 and custom physics models based on recent experimental data.

B.2. Selecting a Low-Energy Physics List

Extra care was taken to ensure that the samples generated described all relevant potentially relevant physics processes. To this end, physics models that are accurate at low energies and that model hadronic scattering well were selected for ICEDUST. The QGSP_BERT_HP

was taken as a starting point. QGSP_BERT_HP is intended to provide the best accuracy for low-energy physics involving neutrons. This list was adapted slightly to improve the accuracy of low-energy electromagnetic interactions as well. The full list is made from these component lists:

1. Electromagnetic Physics
    - G4EmStandardPhysics_option4
        - Handles all electromagnetic processes, ionization, multiple-scattering, bremsstrahlung, etc.
        - Utilizes the most accurate standard and low energy models.
        - Recommended for applications that requires a high accuracy for electrons, hadrons and ion tracking.
    - G4ExtraEmPhysics
        - Adds several electromagnetic processes not included in the “standard” electromagnetic lists.
        - These include electron, positron, and gamma nuclear inelastic scattering in flight, all muon and anti-muon nuclear interactions in flight, and synchrotron radiation for electrons and positrons.
2. Hadronic Physics
    - G4HadronPhysicsQGSP_BERT_HP
        - High-precision modelling of inelastic scattering of hadrons, including protons, neutrons, pions, kaons, hyperons, and all anti-baryons.
        - Uses Bertini cascade to model particles below 9.9 GeV, i.e. in COMET’s initial proton energy range.
        - Takes special care of low-energy neutrons, where a data-driven model is used to incorporate both elastic and inelastic scattering below 19.5 MeV.
    - G4HadronElasticPhysicsHP
        - Designed as the elastic scattering complement G4HadronPhysicsQGSP_BERT_HP.

  - Modelling for all hadron elastic scattering *except* neutrons below 19.5 MeV, which are modelled in G4HadronPhysicsQGSP_BERT_HP.

3. Decay Physics
   - G4DecayPhysics
     - Provides decay physics for all unstable particles.
     - Present in most standard physics lists.
4. Ion Physics
   - G4IonPhysics
     - Provides elastic and inelastic scattering models for ions.
     - This includes particle and interaction definitions for alpha particles, helium-3, deuteron, triton. All other ions are modelled as "generic ions", which are parameterized by their atomic number and atomic mass.
5. Stopped Particle Physics
   - G4StoppingPhysics
     - Models stopped processes, such as nuclear capture, for hadrons. These include anti-protons, anti-nuclei, negative pions, and some negatively charged strange particles.
     - Processes involving stopped muons were disabled here and added back in using custom physics models, as discussed in the next section.

B.3. Custom Muon Physics for COMET in Geant4

In addition to the models listed in the previous section, custom physics processes have been developed and added to ICEDUST, as described in full in [38]. These models improve the description of stopped muon processes in aluminium. The two main processes of note are muon decay-in-orbit and nuclear muon capture.

As discussed in Section 2.3.3, the bound muon decay end-point energy for the resulting electron reaches the μ-e conversion limit of 105 MeV. The models used in Geant4 do not reproduce this tail by default. These models use the free muon decay and apply a small random boost factor set by the muon binding energy. To incorporate this tail, previous

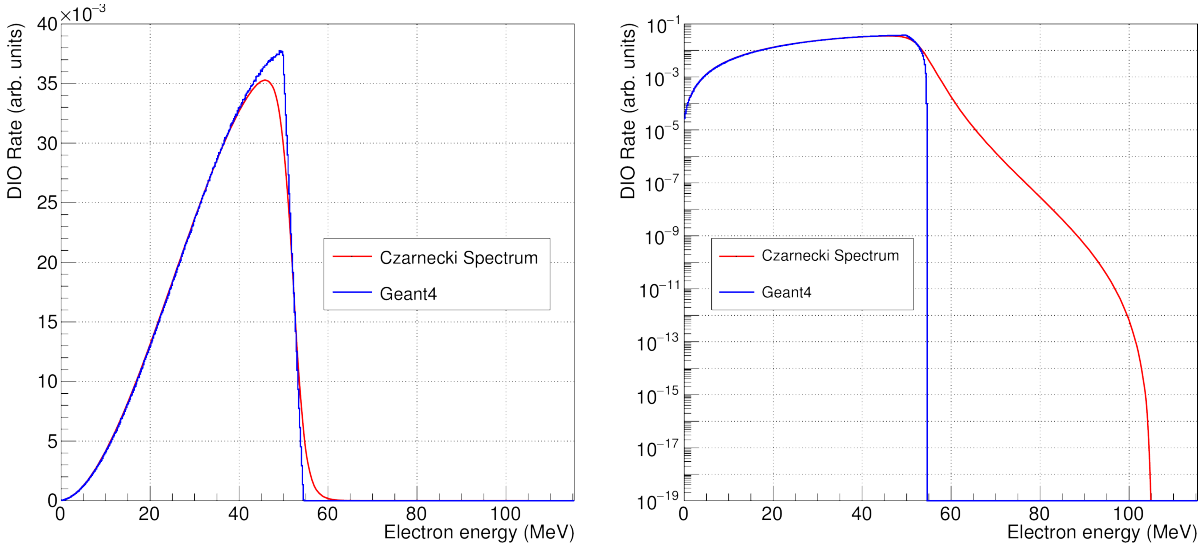


Figure B.1.: A comparison of the electron energy in the default muon decay-in-orbit modelling in Geant4 and the model used in ICEDUST as proposed [46]. The plots are normalized such that the agreement at lower momenta is clear. The right plot is the same as the left on a log scale. Note that the default model has a sharp cutoff, similar to free muon decay, while the model in ICEDUST describes the high-momentum decay-in-orbit background. These figures are from [38], where the models were first implemented in ICEDUST.

sensitivity studies have implemented the phenomenological model proposed in [46]. The differences in the emission spectrum between these two models can be seen Figure B.1.

Muon nuclear capture in aluminium is also a significant source of background in COMET Phase-I. As discussed in [47], the AlCap experiment has measured the resulting spectra from this process. The proton emission frequency and spectrum measured from AlCap has been fed back into the custom stopped muon physics model. By default, Geant4 uses a Bertini cascade to model this process. This model predicts that roughly 20% of all captured muons result in protons. AlCap data predicts a much lower rate of around 3%. The spectrum measured by AlCap is also less energetic than that predicted by the Bertini cascade, as seen in Figure B.2.

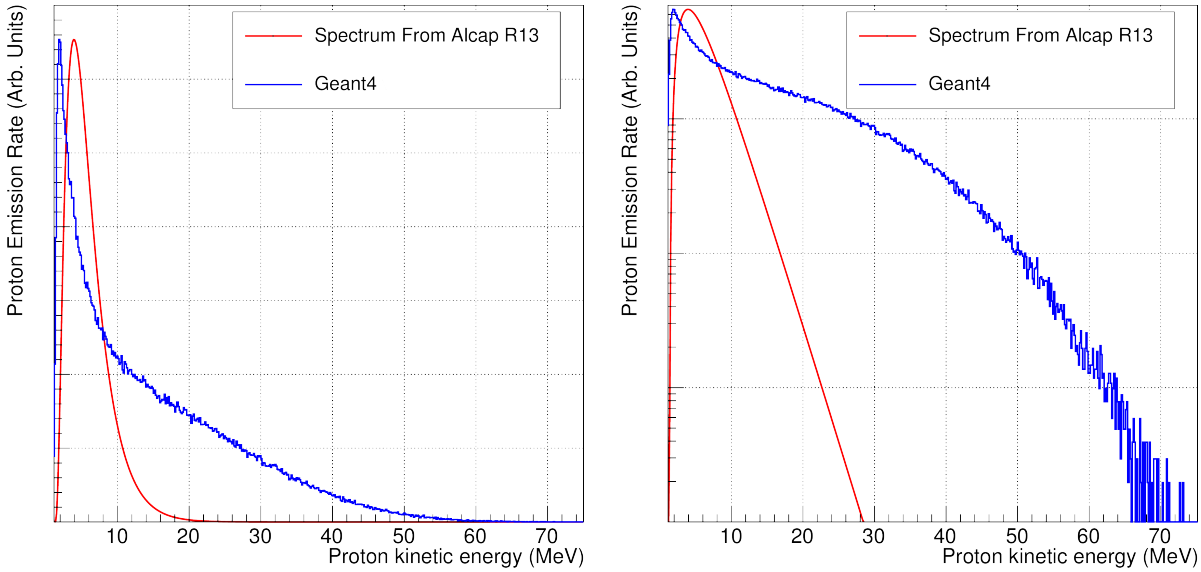


Figure B.2.: A comparison of the proton emission energy spectrum from nuclear muon capture between the Bertini cascade default in Geant4 versus the AlCap measured spectrum as implemented in ICEDUST. Note that the default Geant4 model also predicts 7 times more protons from nuclear muon capture than was measured by AlCap. The right plot is the same as the left on a log scale. These figures are from [38], where the models were first implemented in ICEDUST.

Appendix C.

Sample Summaries

This appendix identifies the key samples referenced in the remainder of this work. Each sample will be discussed briefly and have a summarizing entry in Table C.1. The entries in this table provide statistics about the run time and event size of the SimG4 portion of the simulation.

Table C.1.: Simulation Sample Summaries

Sample Name	Total Number of Events	Run Time/Event	Disk Size/Event
MC4n	8×10^6	3.61 s	2.665 MB
MC4o	9.8×10^9	2.77 s	8.77 kB
MC4p	9.7×10^{10}	0.015 s	490 B
MC4q	8.8×10^{11}	0.015 s	8.8 B

MC4n: This sample persists all tracks created across 8*e*6 POT events. As such, it is useful for describing phenomenology in the Pion Production Target section. This sample is limited by the available disk space.

MC4o: This sample is run with the same settings as MC4n but only saves particles that enter the Muon Transport or Detector Solenoid sections to the disk. The particle flux from the Pion Production Target into the Muon Transport or Detector Solenoid sections is recorded so that it can be can be resampled later, as discussed in Section 4.2.3.

MC4p: The resulting flux from MC4o is resampled to create to create 10 times more events. All tracks that enter the Muon Transport or Detector Solenoid sections are saved.

MC4q: Like MC4p, the resulting flux from MC4o is resampled to create to create 100 times more events. Only tracks that leave a hit in the CDC or CTH are saved.

Appendix D.

Training Machine Learning Algorithms

Training form the crucial first step in building machine learning algorithms. This step is analogous to fitting a generalized model to describe a given dataset. See [58, 59, 60] for a more detailed approach than provided in this summary.

D.1. Neural Networks And Back Propagation

Training a neural network means calculating the optimal values for all of the $\Theta^{(l)}$ matrices, as introduced in Section 5.2.1 To do so, the error is calculated at the output layer and then distributed back through the preceding layers. This technique is called the "backpropagation of errors" or "back propagation". This is essentially an implementation of the chain rule.

To demonstrate this mathematically, consider a regression problem that utilizes the mean squared error as a loss function with a regularisation term. This problem will generalize to consider K output values, denoted by the subscript:

$$J(Y,X,\Theta) = \frac{1}{2N}\sum_{t=1}^{N}\sum_{k=1}^{K}\left[f\left(\mathbf{x}^{(t)},\Theta\right)_k - y_k^{(t)}\right]^2 + \sum_{l=0}^{L}\sum_{i=1}^{n_l}\sum_{j=1}^{m_l}\Theta_{ij}^{(l)} \tag{D.1}$$

where layer l has m_l elements and layer $l+1$ has n_l elements. Now consider the update rule for an element of $\Theta_{ij}^{(L)}$, i.e. the matrix connecting the last hidden layer to the output layer. The form of this can be read off from equation (5.7):

$$\Theta_{ij}^{(L)} := \Theta_{ij}^{(L)}(1+\lambda) - \frac{\alpha}{N}\sum_{t=1}^{N}\sum_{k=1}^{K}\left[f\left(\mathbf{x}^{(t)},\Theta\right)_k - y_k^{(t)}\right]\frac{\partial}{\partial\Theta_{ij}^{(L)}} f\left(\mathbf{x}^{(t)},\Theta\right)_k \tag{D.2}$$

Now use equation (5.15) and equation (5.13) to provide a more explicit form of f. Calculate its differential:

$$\Theta_{ij}^{(L)} := \Theta_{ij}^{(L)}(1+\lambda) - \frac{\alpha}{N}\sum_{t=1}^{N}\sum_{k=1}^{K}\left[z_k^{(t)(L+1)} - y_k^{(t)}\right]\frac{\partial}{\partial\Theta_{ij}^{(L)}}\sum_{v=1}^{s}\Theta_{kv}^{(L)}a_v^{(t)(L)}$$

$$\Theta_{ij}^{(L)} := \Theta_{ij}^{(L)}(1+\lambda) - \frac{\alpha}{N}\sum_{t=1}^{N}\left[z_i^{(t)(L+1)} - y_i^{(t)}\right]a_j^{(t)(L)} \tag{D.3}$$

$$\Theta_{ij}^{(L)} := \Theta_{ij}^{(L)}(1+\lambda) - \frac{\alpha}{N}\sum_{t=1}^{N}\delta_i^{(t)(L+1)}a_j^{(t)(L)} \tag{D.4}$$

Two superscripts are now used, with the first indexing each datum in the sample, and the second indexing the layer value. Note that summations over k and v are reduced since only the $(k=i, v=j)$ term contributes from the differential. A new variable $\delta_i^{(t)(L+1)}$ defines the error on the i-th node in the output layer for the t-th datum. This gives the update rule the free parameters connecting the last two layers.

Now consider the updating the $(L-1)$ layer. The calculation starts the same, only the expansion for f goes two layers deep, which in turn incurs the activation function g:

$$\Theta_{ij}^{(L-1)} := \Theta_{ij}^{(L-1)}(1+\lambda) - \frac{\alpha}{N}\sum_{t=1}^{N}\sum_{k=1}^{K}\delta_k^{(t)(L+1)}\frac{\partial}{\partial\Theta_{ij}^{(L-1)}}\sum_{v=1}^{s}\Theta_{kv}^{(L)}a_v^{(t)(L)}$$

$$\Theta_{ij}^{(L-1)} := \Theta_{ij}^{(L-1)}(1+\lambda) - \frac{\alpha}{N}\sum_{t=1}^{N}\sum_{k=1}^{K}\delta_k^{(t)(L+1)}\frac{\partial}{\partial\Theta_{ij}^{(L-1)}}\sum_{v=1}^{s}\Theta_{kv}^{(L)}g\left(\sum_{w=1}^{r}\Theta_{vw}^{(L-1)}a_w^{(t)(L-1)}\right)$$

$$\Theta_{ij}^{(L-1)} := \Theta_{ij}^{(L-1)}(1+\lambda) - \frac{\alpha}{N}\sum_{t=1}^{N}\sum_{k=1}^{K}\delta_k^{(t)(L+1)}\Theta_{ki}^{(L)}\left(1-a_i^{(t)(L)}\right)a_i^{(t)(L)}a_j^{(t)(L-1)}$$

$$\Theta_{ij}^{(L-1)} := \Theta_{ij}^{(L-1)}(1+\lambda) - \frac{\alpha}{N}\sum_{t=1}^{N}\left[\left[\left(\Theta^{(L)}\right)^T\delta^{(t)(L+1)}\right]\circ\left(\mathbf{1}-\mathbf{a}^{(t)(L)}\right)\circ\mathbf{a}^{(t)(L)}\right]_i a_j^{(t)(L-1)} \tag{D.5}$$

The first two lines introduce the expansion to the second to last hidden layer. In calculating the differential in the second line, the following property of g was used:

$$g'(z) = [1-g(z)]\,g(z) \tag{D.6}$$

used. This differential reduces the summations over v and w since only the $v=i$ and $w=j$ terms survive. In the last line, the Hadamard product, $(A\circ B)_{ij} = A_{ij}B_{ij}$, has been introduced to vectorize part of the equation. This result can be extended to recursively

define $\delta^{(t)(L)}$:

$$\delta^{(t)(l)} = \begin{cases} z_i^{(t)(l+1)} - y_i^{(t)} & l = L+1 \\ \left[\left(\Theta^{(l)}\right)^T \delta^{(t)(l+1)}\right] \circ \left(\mathbf{1} - \mathbf{a}^{(t)(l)}\right) \circ \mathbf{a}^{(t)(l)} & l \leq 0 \leq L \end{cases} \tag{D.7}$$

From this definition, the update rule for all parameters in Θ can be deduced. First, note that determining the Θ values in the $L-2$ layer would proceed exactly as equation (D.5) had, but have one additional iteration of the chain rule on the $a_j^{(t)(L-1)}$ term. This additional iteration would mirror the chain rule used to get equation (D.5) from the previous result from the L layer. This motivates the recursive definition of $\delta^{(t)(L)}$ and along with it the generalized form of the update rule:

$$\Theta_{ij}^{(l)} := \Theta_{ij}^{(l)}(1+\lambda) - \frac{\alpha}{N}\sum_{t=1}^{N} \delta_i^{(t)(l+1)} a_j^{(t)(l)} \tag{D.8}$$

Finally, it is worth noting that exactly the same steps can be taken for a regression problem that uses the logarithmic loss function. Starting instead from equation (5.10), a slightly different definition is used for the base case for $\delta^{(t)(L)}$:

$$\delta^{(t)(l)} = \begin{cases} g\left(z_i^{(t)(l+1)}\right) - y_i^{(t)} & l = L+1 \\ \left[\left(\Theta^{(l)}\right)^T \delta^{(t)(l+1)}\right] \circ \left(\mathbf{1} - \mathbf{a}^{(t)(l)}\right) \circ \mathbf{a}^{(t)(l)} & 0 \leq l \leq L \end{cases} \tag{D.9}$$

The calculations of the update rule proceed in the same matter, such that equation (D.8) holds true for this case as well.

D.2. Growing a CART Tree

The most famous decision tree structure is called a Classification And Regression Tree (CART). As their name suggests, they can be used for both classification and regression. To grow a CART, the algorithm must determine the optimal split at each node of the data. This split is defined by three values:

- The feature to split on, x_j
- The threshold value of the selected feature, c
- The value of the prediction in both of the daughter node, (p_1, p_2)

These four values are determined by ensuring that the split minimises some loss function. Classification and regression trees only differ in the loss function choice. For regression, the mean squared error is normally used. Classification trees often use the Gini Impurity or the Cross Entropy. This review focuses on regression trees, which are used in GBDT algorithms.

To formalise this mathematically, consider a data set where each datum has a feature vector $\mathbf{x}^{(t)}$ and corresponding target $y^{(t)}$, where t indexes elements in the data set. Now consider a cut on x_j at value c. The resulting subsets of the data are defined as:

$$R_1(j,c) = \left\{\mathbf{x}^{(t)} \mid x_j^{(t)} \geq c\right\} \quad \text{and} \quad R_2(j,c) = \left\{\mathbf{x}^{(t)} \mid x_j^{(t)} > c\right\} \tag{D.10}$$

From here, the minimisation problem can be formalised in terms of the loss function, i.e. the mean squared error:

$$\{j,c,p_1,p_2\} = \min_{j,c}\left[\min_{p_1}\left(\sum_{\mathbf{x}^{(t)}\in R_1(j,c)}\left(y^{(t)}-p_1\right)^2\right) + \min_{p_2}\left(\sum_{\mathbf{x}^{(t)}\in R_2(j,c)}\left(y^{(t)}-p_2\right)^2\right)\right] \tag{D.11}$$

In this formalism, the solutions for p_1 and p_2 are the average value of $y^{(t)}$ of all elements in R_1 and R_2, respectively:

$$p_i = \frac{1}{\left|Y_i\right|}\sum_{y^{(t)}\in Y_i} y^{(t)} \quad \text{where} \quad Y_i = \left\{y^{(t)} \mid \mathbf{x}^{(t)} \in R_i(j,c)\right\}, \quad i \in \{1,2\} \tag{D.12}$$

This equation introduces the cardinality of a set $|S|$, which denotes the number of elements in the set. Due to the low computational complexity of these equations, the values of j and s are determined by scanning input feature for its best possible cut. From here, (p_1, p_2) are assigned to the output nodes. Note that the best split is calculated without considerations of how this will affect future nodes. This is called a "greedy" optimization method.

This training process starts by considering all data in the training sample in the first node, or "root" node. The first cut splits the data into two daughter nodes. The data splitting continues recursively on the daughters until some stopping criteria are reached. The stopping criteria used for the trees in ensemble models is often a limit of the total tree depth.

Decision trees are simple to construct and interpret, but often struggle to build complex predictors that generalise to the testing sample. They are far more powerful when used as a part of an ensemble classifier, such as a GBDT.

D.3. Gradient Boosting a Tree Ensemble

As mentioned in the main body of this work, Gradient Boosted classifiers admit the following form:

$$F_M(\mathbf{x}) = \sum_{i=1}^{M} \gamma_i h_i(\mathbf{x}) \tag{D.13}$$

This defines the output of an additive ensemble of M weak predictors as F_M. Each predictor is indexed by i and has a corresponding output of h_i. The weight of each predictor in the sum is denoted γ_i.

GBDTs build up F_M in a forward, stage-wise fashion. That is to say, each predictor in the ensemble is trained to correct for the error on the existing model. This can be formalised as:

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \gamma_m h_m(\mathbf{x}) = y \tag{D.14}$$

This is the ideal case, where F_m perfectly predicts y. This equation illustrates that when building the tree h_m, the model fixes what it has learned in the previous $m-1$ steps. To build some more intuition, the equation above can be rearranged as:

$$\gamma_m h_m(\mathbf{x}) = y - F_{m-1}(\mathbf{x}) = r_{m-1} \tag{D.15}$$

This equation demonstrates that the output of $\gamma_m h_m(\mathbf{x})$ should be the residual of the true value y to the predicted value from $F_{m-1}(\mathbf{x})$. Equivalently, the predictor trained at stage m should correct the disagreement from the previous $m-1$ predictors. This means that tree h_m should predict the target value r_{m-1} given vector $\mathbf{x}$.

If we consider a regression GBDT, the residuals r_m can be thought of as the negative gradient of the mean squared error loss, as in:

$$L(y, \hat{y}) = \frac{1}{2}[y - \hat{y}]^2 \quad \text{such that} \quad r_m = -\left[\frac{\partial L(y, \hat{y})}{\partial \hat{y}}\right]_{\hat{y}=F_{m-1}(\mathbf{x})} \tag{D.16}$$

This result is often extended to other loss functions, where r_m is calculated as the negative gradient of the loss function, as above, instead of the residual. In these cases, r_m is referred to as the "pseudo-residual". The new tree trained at stage m is then fit to predict the

pseudo-residuals from the previous step r_{m-1}. As a final step, γ_m is chosen to minimise the re-evaluated loss function, $L(y, F_m(\mathbf{x}))$. This is normally done by scanning values of γ_m.

This can all be summarised into an algorithm, which proceeds as follows:

1. Create a base-case predictor, which is normally just the average of all $y^{(t)}$ values in the sample:

$$F_0\left(\mathbf{x}^{(t)}\right) = \frac{1}{N}\sum_{i=1}^{N} y^{(i)}$$

2. For each $m = 1$ to M:

   - Calculate the pseudo-residuals:

$$r_m^{(t)} = -\left[\frac{\partial L\left(y^{(t)}, \hat{y}^{(t)}\right)}{\partial \hat{y}}\right]_{\hat{y}^{(t)}=F_{m-1}\left(\mathbf{x}^{(t)}\right)} \quad \text{for} \quad t \in [1, N]$$

   - Fit the h_m tree to predict $r_m^{(t)}$ given $\mathbf{x}^{(t)}$ for all elements, i.e. $t \in [1, N]$.

   - Determine γ_m by scanning over potential values to solve:

$$\gamma_m = \min_{\gamma} \sum_{t=1}^{N} L\left(y^{(t)}, F_{m-1} + \gamma h_m\left(\mathbf{x}^{(t)}\right)\right)$$

   - Update the model:

$$F_m\left(\mathbf{x}^{(t)}\right) = F_{m-1}\left(\mathbf{x}^{(t)}\right) + \gamma_m h_m\left(\mathbf{x}^{(t)}\right)$$

3. Output the final model, $F_M(\mathbf{x})$.

Bibliography

[1] Y. Nishina, M. Takeuchi, and T. Ichimiya. “On the Nature of Cosmic-Ray Particles”. In: *Phys. Rev.* 52 (1937), pp. 1198–1199. DOI: 10.1103/PhysRev.52.1198 (cit. on p. 1).

[2] E. J. Williams and G. E. Roberts. “Evidence for Transformation of Mesotrons into Electrons”. In: *Nature* 145 (1940), 102 EP –. URL: http://dx.doi.org/10.1038/145102a0 (cit. on pp. 1–2).

[3] Hideki Yukawa. “On the Interaction of Elementary Particles I”. In: *Proc. Phys. Math. Soc. Jap.* 17 (1935). [Prog. Theor. Phys. Suppl.1,1(1935)], pp. 48–57. DOI: 10.1143/PTPS.1.1 (cit. on p. 1).

[4] L. W. Nordheim. “On the Nature of the Meson Decay”. In: *Phys. Rev.* 59 (7 1941), pp. 554–555. DOI: 10.1103/PhysRev.59.554. URL: https://link.aps.org/doi/10.1103/PhysRev.59.554 (cit. on p. 2).

[5] C. M. G. LATTES et al. “PROCESSES INVOLVING CHARGED MESONS”. In: *Nature* 159 (1947). Article, 694 EP –. URL: http://dx.doi.org/10.1038/159694a0 (cit. on p. 2).

[6] E. P. Hincks and B. Pontecorvo. “Search for gamma-radiation in the 2.2-microsecond meson decay process”. In: *Phys. Rev.* 73 (1948), pp. 257–258. DOI: 10.1103/PhysRev.73.257 (cit. on pp. 3, 6).

[7] G. Danby et al. “Observation of High-Energy Neutrino Reactions and the Existence of Two Kinds of Neutrinos”. In: *Phys. Rev. Lett.* 9 (1962), pp. 36–44. DOI: 10.1103/PhysRevLett.9.36 (cit. on p. 3).

[8] Emmy Noether. “Invariant Variation Problems”. In: *Gott. Nachr.* 1918 (1918). [Transp. Theory Statist. Phys.1,186(1971)], pp. 235–257. DOI: 10.1080/00411457108231446. arXiv: physics/0503066 [physics] (cit. on p. 4).

[9] Raymond Davis Jr., Don S. Harmer, and Kenneth C. Hoffman. “Search for neutrinos from the sun”. In: *Phys. Rev. Lett.* 20 (1968), pp. 1205–1209. DOI: 10.1103/PhysRevLett.20.1205 (cit. on p. 4).

[10] B. Pontecorvo. "Neutrino Experiments and the Problem of Conservation of Leptonic Charge". In: *Sov. Phys. JETP* 26 (1968). [Zh. Eksp. Teor. Fiz.53,1717(1967)], pp. 984–988 (cit. on p. 4).

[11] Y. Fukuda et al. "Evidence for oscillation of atmospheric neutrinos". In: *Phys. Rev. Lett.* 81 (1998), pp. 1562–1567. DOI: 10.1103/PhysRevLett.81.1562. arXiv: hep-ex/9807003 [hep-ex] (cit. on p. 4).

[12] Ziro Maki, Masami Nakagawa, and Shoichi Sakata. "Remarks on the unified model of elementary particles". In: *Prog. Theor. Phys.* 28 (1962). [,34(1962)], pp. 870–880. DOI: 10.1143/PTP.28.870 (cit. on p. 4).

[13] William J. Marciano, Toshinori Mori, and J. Michael Roney. "Charged Lepton Flavor Violation Experiments". In: *Ann. Rev. Nucl. Part. Sci.* 58 (2008), pp. 315–341. DOI: 10.1146/annurev.nucl.58.110707.171126 (cit. on p. 5).

[14] Robert H. Bernstein and Peter S. Cooper. "Charged Lepton Flavor Violation: An Experimenter's Guide". In: *Phys. Rept.* 532 (2013), pp. 27–64. DOI: 10.1016/j.physrep.2013.07.002. arXiv: 1307.5787 [hep-ex] (cit. on pp. 5–6).

[15] H. N. Brown et al. "Precise measurement of the positive muon anomalous magnetic moment". In: *Phys. Rev. Lett.* 86 (2001), pp. 2227–2231. DOI: 10.1103/PhysRevLett.86.2227. arXiv: hep-ex/0102017 [hep-ex] (cit. on p. 5).

[16] Thomas Blum et al. "The Muon (g-2) Theory Value: Present and Future". In: (2013). arXiv: 1311.2198 [hep-ph] (cit. on p. 5).

[17] Aldo Antognini et al. "Proton Structure from the Measurement of $2S-2P$ Transition Frequencies of Muonic Hydrogen". In: *Science* 339 (2013), pp. 417–420. DOI: 10.1126/science.1230016 (cit. on p. 5).

[18] Roel Aaij et al. "Test of lepton universality using $B^+ \to K^+\ell^+\ell^-$ decays". In: *Phys. Rev. Lett.* 113 (2014), p. 151601. DOI: 10.1103/PhysRevLett.113.151601. arXiv: 1406.6482 [hep-ex] (cit. on p. 5).

[19] Roel Aaij et al. "Measurement of the ratio of branching fractions $\mathcal{B}(\bar{B}^0 \to D^{*+}\tau^-\bar{\nu}_\tau)/\mathcal{B}(\bar{B}^0 \to D^{*+}\mu^-\bar{\nu}_\mu$ In: *Phys. Rev. Lett.* 115.11 (2015). [Erratum: Phys. Rev. Lett.115,no.15,159901(2015)], p. 111803. DOI: 10.1103/PhysRevLett.115.159901,10.1103/PhysRevLett.115.111803. arXiv: 1506.08614 [hep-ex] (cit. on p. 5).

[20] A. Abdesselam et al. "Measurement of the branching ratio of $\bar{B}^0 \to D^{*+}\tau^-\bar{\nu}_\tau$ relative to $\bar{B}^0 \to D^{*+}\ell^-\bar{\nu}_\ell$ decays with a semileptonic tagging method". In: *Proceedings, 51st Rencontres de Moriond on Electroweak Interactions and Unified Theories: La Thuile, Italy, March 12-19, 2016*. 2016. arXiv: 1603.06711 [hep-ex] (cit. on p. 5).

[21] J. P. Lees et al. "Evidence for an excess of $\bar{B} \to D^{(*)}\tau^- \bar{\nu}_\tau$ decays". In: *Phys. Rev. Lett.* 109 (2012), p. 101802. DOI: 10.1103/PhysRevLett.109.101802. arXiv: 1205.5442 [hep-ex] (cit. on p. 5).

[22] G. Lafferty. "Pentaquarks and possible anomalies at LHCb". In: *Int. J. Mod. Phys.* A31.20n21 (2016), p. 1630032. DOI: 10.1142/S0217751X16300325 (cit. on p. 5).

[23] Albert M Sirunyan et al. "Search for lepton flavour violating decays of the Higgs boson to $\mu\tau$ and $e\tau$ in proton-proton collisions at $\sqrt{s} = 13$ TeV". In: *JHEP* 06 (2018), p. 001. DOI: 10.1007/JHEP06(2018)001. arXiv: 1712.07173 [hep-ex] (cit. on p. 5).

[24] Sheldon L. Glashow, Diego Guadagnoli, and Kenneth Lane. "Lepton Flavor Violation in B Decays?" In: *Phys. Rev. Lett.* 114 (2015), p. 091801. DOI: 10.1103/PhysRevLett.114.091801. arXiv: 1411.0565 [hep-ph] (cit. on p. 5).

[25] A. M. Baldini et al. "Search for the lepton flavour violating decay $\mu^+ \to e^+\gamma$ with the full dataset of the MEG experiment". In: *Eur. Phys. J.* C76.8 (2016), p. 434. DOI: 10.1140/epjc/s10052-016-4271-x. arXiv: 1605.05081 [hep-ex] (cit. on pp. 6–7).

[26] Wilhelm H. Bertl et al. "A Search for muon to electron conversion in muonic gold". In: *Eur. Phys. J.* C47 (2006), pp. 337–346. DOI: 10.1140/epjc/s2006-02582-x (cit. on pp. 8, 16, 19).

[27] U. Bellgardt et al. "Search for the Decay mu+ —> e+ e+ e-". In: *Nucl. Phys.* B299 (1988), pp. 1–6. DOI: 10.1016/0550-3213(88)90462-2 (cit. on p. 7).

[28] Ann-Kathrin Perrevoort. "Status of the Mu3e Experiment at PSI". In: *EPJ Web Conf.* 118 (2016), p. 01028. DOI: 10.1051/epjconf/201611801028. arXiv: 1605.02906 [physics.ins-det] (cit. on p. 7).

[29] R. M. Dzhilkibaev and V. M. Lobashev. "On the Search for $\mu \to e$ Conversion on Nuclei. (In Russian)". In: *Sov. J. Nucl. Phys.* 49 (1989). [Yad. Fiz.49,622(1989)], pp. 384–385 (cit. on p. 8).

[30] R. M. Dzhilkibaev and V. M. Lobashev. "The solenoid muon capture system for the MELC experiment". In: *AIP Conf. Proc.* 372.1 (1996), pp. 53–60. DOI: 10.1063/1.50918 (cit. on p. 8).

[31] James L. Popp. "The MECO experiment: A Search for lepton flavor violation in muonic atoms". In: *Nucl. Instrum. Meth.* A472 (2000), pp. 354–358. DOI: 10.1016/S0168-9002(01)01270-0. arXiv: hep-ex/0101017 [hep-ex] (cit. on p. 8).

[32] Y. K. Semertzidis. "The MECO experiment at BNL". In: *Nucl. Phys. Proc. Suppl.* 149 (2005). [,372(2005)], pp. 372–374. DOI: 10.1016/j.nuclphysbps.2005.05.072 (cit. on p. 8).

[33] Jeffrey Mervis and Adrian Cho. "Costs force NSF to cancel Brookhaven project". In: *Science* 309.5738 (2005), pp. 1163–1164 (cit. on p. 8).

[34] Luca Morescalchi. "The Mu2e Experiment at Fermilab". In: *PoS* DIS2016 (2016), p. 259. DOI: 10.22323/1.265.0259. arXiv: 1609.02021 [hep-ex] (cit. on pp. 8–9).

[35] Cesar Bonilla et al. "Perspectives for Detecting Lepton Flavour Violation in Left-Right Symmetric Models". In: *JHEP* 03 (2017), p. 027. DOI: 10.1007/JHEP03(2017)027. arXiv: 1611.07025 [hep-ph] (cit. on p. 11).

[36] Lorenzo Calibbi and Giovanni Signorelli. "Charged Lepton Flavour Violation: An Experimental and Theoretical Introduction". In: *Riv. Nuovo Cim.* 41.2 (2018), p. 1. DOI: 10.1393/ncr/i2018-10144-0. arXiv: 1709.00294 [hep-ph] (cit. on p. 11).

[37] E. Arganda, M. J. Herrero, and A. M. Teixeira. "mu-e conversion in nuclei within the CMSSM seesaw: Universality versus non-universality". In: *JHEP* 10 (2007), p. 104. DOI: 10.1088/1126-6708/2007/10/104. arXiv: 0707.2955 [hep-ph] (cit. on p. 11).

[38] Benjamin Krikler. "Sensitivity and Background Estimates for Phase-II of the COMET Experiment". PhD thesis. Imperial Coll., London, 2016. URL: http://hdl.handle.net/10044/1/45365 (cit. on pp. 12, 16–18, 20–21, 36, 104, 109, 115–117).

[39] Andre de Gouvea and Petr Vogel. "Lepton Flavor and Number Conservation, and Physics Beyond the Standard Model". In: *Prog. Part. Nucl. Phys.* 71 (2013), pp. 75–92. DOI: 10.1016/j.ppnp.2013.03.006. arXiv: 1303.4097 [hep-ph] (cit. on p. 12).

[40] Giovanni Marco Pruna. "Charged lepton flavour violation: precise background calculation and effective field theoretical interpretation". In: *Proceedings, Prospects in Neutrino Physics (NuPhys2016): London, UK, December 12-14, 2016*. 2017. arXiv: 1705.04826 [hep-ph] (cit. on p. 12).

[41] F. Gursey, Pierre Ramond, and P. Sikivie. "A Universal Gauge Theory Model Based on E6". In: *Phys. Lett.* 60B (1976), pp. 177–180. DOI: 10.1016/0370-2693(76)90417-2 (cit. on p. 13).

[42] Edward Witten. "Quest for unification". In: *Supersymmetry and unification of fundamental interactions. Proceedings, 10th International Conference, SUSY'02, Hamburg, Germany, June 17-23, 2002*. 2002, pp. 604–610. arXiv: hep-ph/0207124 [hep-ph]. URL: http://www-library.desy.de/preparch/desy/proc/proc02-02/Proceedings/special/hertz_pr.ps (cit. on p. 13).

[43] J. Bernabeu, E. Nardi, and D. Tommasini. "μ - e conversion in nuclei and Z' physics". In: *Nucl. Phys.* B409 (1993), pp. 69–86. DOI: 10.1016/0550-3213(93)90446-V. arXiv: hep-ph/9306251 [hep-ph] (cit. on p. 13).

[44] R. Phillip Litchfield. "Status of the AlCap experiment". In: *PoS* NUFACT2014 (2015), p. 095. DOI: 10.22323/1.226.0095. arXiv: 1501.04880 [physics.ins-det] (cit. on p. 14).

[45] M. Tanabashi et al. "Review of Particle Physics". In: *Phys. Rev.* D98.3 (2018), p. 030001. DOI: 10.1103/PhysRevD.98.030001 (cit. on p. 14).

[46] Andrzej Czarnecki, Xavier Garcia i Tormo, and William J. Marciano. "Muon decay in orbit: spectrum of high-energy electrons". In: *Phys. Rev.* D84 (2011), p. 013006. DOI: 10.1103/PhysRevD.84.013006. arXiv: 1106.4756 [hep-ph] (cit. on pp. 15, 116).

[47] Andrew Edmonds. "Latest Updates from the AlCap Experiment". In: *13th Conference on the Intersections of Particle and Nuclear Physics (CIPANP 2018) Palm Springs, California, USA, May 29-June 3, 2018*. 2018. arXiv: 1809.10122 [physics.ins-det] (cit. on pp. 16, 116).

[48] S. Agostinelli et al. "GEANT4: A Simulation toolkit". In: *Nucl. Instrum. Meth.* A506 (2003), pp. 250–303. DOI: 10.1016/S0168-9002(03)01368-8 (cit. on pp. 17, 35, 38, 113).

[49] M. Friend. "J-PARC accelerator and neutrino beamline upgrade programme". In: *J. Phys. Conf. Ser.* 888.1 (2017), p. 012042. DOI: 10.1088/1742-6596/888/1/012042 (cit. on p. 21).

[50] W. R. Leo. *Techniques for Nuclear and Particle Physics Experiments: A How to Approach*. 1987. ISBN: 9783540572800 (cit. on p. 31).

[51] Erik Dietz-Laursonn. "Peculiarities in the Simulation of Optical Physics with Geant4". In: (2016). arXiv: 1612.05162 [physics.ins-det] (cit. on p. 32).

[52] Thomas Lindner. "Evolution of the T2K-ND280 Computing Model". In: *J. Phys. Conf. Ser.* 664.3 (2015), p. 032021. DOI: 10.1088/1742-6596/664/3/032021 (cit. on p. 35).

[53] Koji Niita et al. “PHITS: A particle and heavy ion transport code system”. In: *Radiat. Meas.* 41 (2006), pp. 1080–1090. DOI: 10.1016/j.radmeas.2006.07.013 (cit. on p. 35).

[54] T. T. BÃűhlen et al. “The FLUKA Code: Developments and Challenges for High Energy and Medical Applications”. In: *Nucl. Data Sheets* 120 (2014), pp. 211–214. DOI: 10.1016/j.nds.2014.07.049 (cit. on p. 35).

[55] David C. Carey, Karl L. Brown, and F. C. Iselin. “Decay TURTLE (Trace Unlimited Rays Through Lumped Elements): A Computer Program for Simulating Charged Particle Beam Transport Systems, Including Decay Calculations”. In: (1982) (cit. on p. 38).

[56] Alexander Radovic et al. “Machine learning at the energy and intensity frontiers of particle physics”. In: *Nature* 560.7716 (2018), pp. 41–48. DOI: 10.1038/s41586-018-0361-2 (cit. on p. 43).

[57] Kim Albertsson et al. “Machine Learning in High Energy Physics Community White Paper”. In: (2018). arXiv: 1807.02876 [physics.comp-ph] (cit. on p. 43).

[58] T. Hastie, R. Tibshirani, and J.H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics. Springer, 2009. ISBN: 9780387848846. URL: https://books.google.co.uk/books?id=eBSgoAEACAAJ (cit. on pp. 43, 47, 121).

[59] L. Breiman et al. *Classification and Regression Trees*. The Wadsworth and Brooks-Cole statistics-probability series. Taylor & Francis, 1984. ISBN: 9780412048418. URL: https://books.google.co.uk/books?id=JwQx-WOmSyQC (cit. on pp. 49, 121).

[60] Jerome H. Friedman. “Greedy Function Approximation: A Gradient Boosting Machine”. In: *Annals of Statistics* 29 (2000), pp. 1189–1232 (cit. on pp. 50, 121).

[61] P. E. Hart. “How the Hough transform was invented [DSP History]”. In: *IEEE Signal Processing Magazine* 26 (2009) (cit. on p. 51).

[62] Richard O. Duda and Peter E. Hart. “Use of the Hough Transformation to Detect Lines and Curves in Pictures”. In: *Commun. ACM* 15.1 (1972), pp. 11–15. ISSN: 0001-0782. DOI: 10.1145/361237.361242. URL: http://doi.acm.org/10.1145/361237.361242 (cit. on p. 51).

[63] Eric Jones, Travis Oliphant, Pearu Peterson, et al. *SciPy: Open source scientific tools for Python*. [Online; accessed <today>]. 2001–. URL: http://www.scipy.org/ (cit. on p. 76).

[64] Wes McKinney. "Data Structures for Statistical Computing in Python". In: *Proceedings of the 9th Python in Science Conference*. Ed. by Stéfan van der Walt and Jarrod Millman. 2010, pp. 51 –56 (cit. on p. 76).

[65] Fabian Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *J. Machine Learning Res.* 12 (2011), pp. 2825–2830. arXiv: 1201.0490 [cs.LG] (cit. on p. 77).

[66] Tianqi Chen and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System". In: (2016). DOI: 10.1145/2939672.2939785. arXiv: 1603.02754 [cs.LG] (cit. on p. 77).

[67] Tomohisa Uchida et al. "Readout electronics for the central drift chamber of the Belle II detector". In: *Proceedings, 2011 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC 2011): Valencia, Spain, October 23-29, 2011*. 2011, pp. 694–698. DOI: 10.1109/NSSMIC.2011.6154084 (cit. on p. 95).

[68] V. V. Gligorov and Mike Williams. "Efficient, reliable and fast high-level triggering using a bonsai boosted decision tree". In: *JINST* 8 (2013), P02013. DOI: 10.1088/1748-0221/8/02/P02013. arXiv: 1210.6861 [physics.ins-det] (cit. on p. 96).

[69] Burkhard Schmidt. "The High-Luminosity upgrade of the LHC: Physics and Technology Challenges for the Accelerator and the Experiments". In: *J. Phys. Conf. Ser.* 706.2 (2016), p. 022002. DOI: 10.1088/1742-6596/706/2/022002 (cit. on p. 103).

[70] Roel Aaij et al. "Measurement of the $B_s^0 \to J/\psi\eta$ lifetime". In: *Phys. Lett.* B762 (2016), pp. 484–492. DOI: 10.1016/j.physletb.2016.10.006. arXiv: 1607.06314 [hep-ex] (cit. on p. 107).

[71] *Advanced Lectures on Machine Learning*. Vol. LNAI 3176. Max-Planck-Gesellschaft. Berlin, Germany: Springer, 2004, p. 240 (cit. on p. 107).

[72] Xiaojin Zhu et al. *Introduction to Semi-Supervised Learning*. Morgan and Claypool Publishers, 2009. ISBN: 1598295470, 9781598295474 (cit. on p. 107).

List of figures

List of tables