Study of Sudden Beam Losses of SuperKEKB and Development of 3D Track Hardware Trigger using Machine Learning at Belle II

LIU Yuxin

Master of Science (M.Sc.)

Department of Particle and Nuclear Physics

School of High Energy Accelerator Science

The Graduate University for Advanced Studies, SOKENDAI

September 2023

Acknowledgments

I would like to express my deepest gratitude and appreciation to the following individuals and institutions who have played a significant role in the completion of my master's thesis.

First and foremost, I am incredibly grateful to my supervisors, Taichiro Koga-san and Akimasa Ishikawa-san, for their unwavering support, invaluable guidance, and scholarly expertise throughout this research journey. Their insightful feedback, patience, and mentorship have been instrumental in shaping the direction of this thesis. I am truly fortunate to have had the opportunity to work under their supervision.

I am indebted to SOKENDAI, for providing me with the necessary resources and a stimulating academic environment. The excellent faculty and staff at SOKENDAI have been instrumental in my growth as a researcher.

I am grateful to my colleagues in KEK, who provided valuable insights, engaging discussions, and a supportive community throughout of my two years study and research process. Their camaraderie and intellectual contributions have enriched my research and learning experience.

Finally, I would like to express my heartfelt thanks to my friends and family for their unconditional love, encouragement, and understanding. Their unwavering support and belief in my abilities have been a constant source of motivation, especially during challenging times.

In conclusion, the successful completion of this thesis would not have been possible without the support and contributions of the aforementioned individuals and institutions. I am truly grateful for their involvement and trust in my capabilities.

Yuxin Liu

Abstract

The Belle II Experiment, located at the SuperKEKB asymmetric electron-positron collider in Japan, is at the next generation of B-factories, aiming to explore new physics (NP) in the flavor sector and enhance the precision of Standard Model (SM) measurements. SuperKEKB is expected to achieve the luminosity of 6×10^{35}, $\mathrm{cm}^{-2}\mathrm{s}^{-1}$, enabling unprecedented NP searches and measurements of the CKM matrix. However, the increase of higher luminosity faces challenges of sudden beam loss events and increasing level-1 trigger rates.

Sudden beam loss events, characterized by rapid beam loss within a few turns, pose risks to the SuperKEKB and Belle II components, with the underlying causes still unknown. To address this, beam loss monitor with fine timing resolution have been installed to pinpoint the location of initial beam loss. Timing analyses have identified the LER D06 section as the region where the earliest loss occurs, suggesting the occurrence of initial beam instability. Based on the analysis results, countermeasures, such as fast beam abort and additional sensors at the D06 section, are planned to protect detectors and collimators from sudden beam loss.

The 3D Track Hardware Trigger, responsible for triggering physics events, faces limitations of trigger rate. With increasing luminosity, the large increased background trigger rate nears the maximum limit. Based on the newly deployed fourth generation universal trigger board (UT4, which have general 4 times logic gates than previous board, new neural-network 3D track trigger architectures have been developed with software simulations. These architectures achieved an 50 % reeducation of total CDC trigger background. Further work will focus on simplifying the architecture and implementing it in UT4 modules.

Contents

List of Figures

List of Tables

Introduction

The Belle II Experiment, located at the SuperKEKB asymmetric electron-positron collider in KEK, Japan, is the next generation B-factory. Its primary physics goals are to explore new physics (NP) in the flavor sector at the intensity frontier and enhance the precision of measurements for Standard Model (SM) parameters. The SuperKEKB facility is specifically designed to collide electrons and positrons at center-of-mass energies near the Υ resonances. Most of the data will be collected at the $\Upsilon(4S)$ resonance, which is just above the threshold for B-meson pair production, thereby avoiding the production of fragmentation particles. To enable measurements of time-dependent charge-parity (CP) symmetry violation, the accelerator is designed with asymmetric beam energies to boost the center-of-mass system.

Building upon the achievements of its predecessor, the Belle experiment, the Belle II detector has undergone upgrades, including the addition of a new Pixel Detector (PXD) employing DEPFET technology and a larger Central Drift Chamber (CDC). The KEKB facility has been upgraded to SuperKEKB to achieve a peak luminosity of $6 \times 10^{35}, \mathrm{cm}^{-2}\mathrm{s}^{-1}$ and aims to accumulate a integrated luminosity of 50 ab^{-1}. As of June 2022, SuperKEKB has already achieved a luminosity of $4.7 \times 10^{34}, \mathrm{cm}^{-2}\mathrm{s}^{-1}$.

During recent operations of the SuperKEKB, there has been a noticeable increase in the occurrence of "sudden beam loss" (SBL) events. These SBL events result in a rapid loss of the stored beam within just a few turns, and the exact reasons behind them remain unknown. Notably, the occurrence of large SBL events has also caused significant damage to the vertical collimators, posing challenges in effectively controlling the beam background. In some cases, these events have even led to substantial radiation doses in the vicinity of the interaction point (IP), posing a serious risk of damaging the sensors of the Belle II detector. Consequently, there is an urgent

need to investigate the underlying causes of these sudden beam loss events. To identify the precise location in the accelerator ring where the earliest beam loss is observed, we have installed fast loss sensors and are conducting timing analyses that involve fast loss monitors and other beam monitors.

Despite that, as the luminosity increases, each sub-detector of Belle II faces a limitation of a finite bandwidth for data transfer. Therefore, in order to manage this, a Level-1 trigger is employed to select interesting events for recording with a maximum limit for the Level-1 trigger rate, set at 30 kHz. And it is worth noting that this rate has already reached approximately 11 kHz at a luminosity of $4.7 \times 10^{34}, \mathrm{cm}^{-2}, \mathrm{s}^{-1}$. To facilitate future data acquisition, it is imperative to update the Level-1 trigger to effectively reduce the background trigger rate. Through our investigations, we have identified the CDC trigger with tracks off the IP as the primary source of background triggers. Thus, our plan entails augmenting the input information and upgrading the existing Neural-Network 3D track trigger architectures by incorporating the advancements provided by the universal trigger board 4 hardware upgrade.

This thesis is organized as follows. Chapter 1 of this thesis will delve into the physics motivations driving the Belle II experiment. In Chapter 2, an extensive overview of the SuperKEKB collider and the Belle II detector will be presented. The significance of SBLs and the fast loss monitor system will be outlined in Chapter 3, while Chapter 4 will illustrate the methodology employed for timing analysis using fast loss monitors. The preliminary findings and analysis of SBLs will be shown in Chapter 5. Furthermore, Chapter 6 will elucidate the current pipeline of the level-1 CDC trigger, followed by Chapter 7 which explores several optimized approaches for the level-1 CDC trigger. Finally, Chapter 8 will show the performance evaluation of each proposed method.

1 Physics motivation of Belle II

The chapter will give an overview of the physics motivations for the Belle II.

1.1 Measurements of Standard Model parameters

At its current level of experimental precision and with the energies achieved thus far, the Standard Model (SM) stands as the most extensively tested theory of elementary particle physics. The Belle II experiment aims to contribute to this endeavor by precisely measuring the parameters of the SM, particularly focusing on the elements of the CKM matrix.

1.1.1 CKM matrix

The Cabibbo-Kobayashi-Maskawa (CKM) matrix [1, 2] is a unitary matrix that describes the mixing of quark flavors in weak interactions and appears in the couplings of the W boson to quarks:

$$\mathcal{L}_W^q = \frac{g}{\sqrt{2}}[V_{jk}\bar{u}_{Lj}\gamma^\mu d_{Lk}W_\mu^+ + V_{jk}^*\bar{d}_{Lk}\gamma^\mu u_{Lj}W_\mu^-]$$

In this equation, g represents the gauge coupling constant, u_{Lj} and d_{Lk} are left-handed up-type and down-type quark fields, respectively, W_μ^+ and W_μ^- are the charged gauge bosons of the weak force, and V_{jk} represents the elements of the CKM matrix.

It relates the mass eigenstates of quarks (d', s', b') to their weak interaction eigenstates (d, s, b). as:

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \begin{pmatrix} d \\ s \\ b \end{pmatrix}$$

where V_{ij} represents the CKM matrix elements corresponding to the transition from the ith-type quark to the jth-type quark.

A arbitrary 3×3 complex matrix have 18 free parameters. Considering the unitary condition $V_{CKM}V_{CKM}^\dagger = 1$, we can reduce the free parameter by 9, and 5 phase parameter can be absorbed in the quark field redefinition, thus the CKM matrix can be characterized with four parameters: three Euler angles (θ_{12}, θ_{23}, θ_{13}) and one complex phase (δ). The angles represent the mixing between different generations of quarks, while the phase accounts for CP violation. The mixing angles determine the probabilities of flavor-changing transitions in weak interactions.

The CKM matrix can be parameterized as follows:

$$\begin{pmatrix} c_{12}c_{13} & s_{12}c_{13} & s_{13}e^{-i\delta} \\ -s_{12}c_{23} - c_{12}s_{23}s_{13}e^{i\delta} & c_{12}c_{23} - s_{12}s_{23}s_{13}e^{i\delta} & s_{23}c_{13} \\ s_{12}s_{23} - c_{12}c_{23}s_{13}e^{i\delta} & -c_{12}s_{23} - s_{12}c_{23}s_{13}e^{i\delta} & c_{23}c_{13} \end{pmatrix}$$

where $s_{ij} = \sin\theta_{ij}$ and $c_{ij} = \cos\theta_{ij}$.

Another common parametrization is the Wolfenstein Parametrization, which expands the CKM matrix in terms of the parameter $\lambda = \sin\theta_{12} \sim 0.22$. Wolfenstein then defined four parameters (λ, A, ρ, η):

$$\lambda = sin\theta_{12};\ A\lambda^2 = sin\theta_{23};\ A\lambda^3(\rho - i\eta) = sin\theta_{12}e^{-i\delta} \tag{1.1}$$

so that up to $O(\lambda^3)$, the CKM matrix can be written as:

$$\begin{pmatrix} 1-\frac{\lambda^2}{2} & \lambda & A\lambda^3(\rho-i\eta) \\ -\lambda & 1-\frac{\lambda^2}{2} & A\lambda^2 \\ A\lambda^3(1-\rho-i\eta) & -A\lambda^2 & 1 \end{pmatrix} + O(\lambda^3)$$

We can also expand the unitary condition $V_{CKM}V_{CKM}^{\dagger} = 1$ to $O(\lambda^3)$. If we take a product of the down and bottom columns:

$$V_{ud}V_{ub}^* + V_{cd}V_{cb}^* + V_{td}V_{tb}^* = 0 = A\lambda^3(\rho+i\eta) - A\lambda^3(1-i\eta-\rho) - A\lambda^3 + O(\lambda^5)$$

which contains three complex terms of O(λ 3) that must form a closed triangle in the complex plane, as shown in Fig. 1.1. It is convenient to normalize these terms so that one side is purely real with length 1.

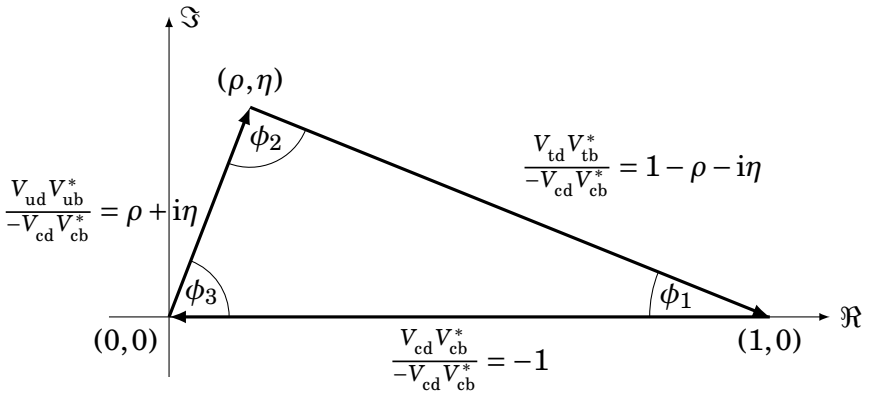


Figure 1.1: The CKM unitary triangle[3]

We can define three angles of the unitary triangle:

$$\phi_1 = \arg\left(-\frac{V_{cd}^*V_{cb}}{V_{td}^*V_{tb}}\right);\ \phi_2 = \arg\left(-\frac{V_{td}^*V_{tb}}{V_{ud}^*V_{ub}}\right);\ \phi_3 = \arg\left(-\frac{V_{ud}^*V_{ub}}{V_{cd}^*V_{cb}}\right)$$

The accurate measurement of these three angles can test the consistency of the

CKM matrix and search for possible deviations that could indicate the presence of new physics beyond the Standard Model.

The angle ϕ_1 is measured from the interference between B_d oscillation with $b \to cs\bar{s}$ decays. Most of the hadronic uncertainties cancel out in this CP-violating observable, and it therefore provides a very clean and precise determination of ϕ_1 already with Belle, BaBar and LHcb.

The angle ϕ_2 is measured from interference between the tree level $b \to ud\bar{u}$ and the B meson mixing, with decays such as $B \to \pi\pi, \pi\rho, \rho\rho$. The penguin contribution pollutes this ϕ_2 measurement. The experimental error on ϕ_2 is still very large, and more precise measurements by Belle II have the potential to reveal a deviation from the other unitary triangle fit inputs.

The third angle ϕ_3 is measured via the CP asymmetry, which occurs due to the interference between $b \to c\bar{u}s$ and $b \to u\bar{c}s$, and both decay to the same final state. The Decay mode of the type $B \to D^{(*)}K^{(*)}$, where the D meson decays to a flavor non-specific hadronic decays, can be used to obtain a very precise determination of ϕ_3. The measurement of ϕ_3 is highly statistics-limited and will be greatly improved in the era of Belle II.

1.2 New physics searching in flavor sector

Another target of Belle II experiments is to search for NP beyond SM that includes more specific flavor couplings, for which indirect searches can push the new physics scale much higher. Many flavor physics questions may be addressed by Belle II experiments. Here we list some examples from [3].

- New CP violating phases in the quark sector and flavour-changing neutral currents (FCNC) beyond the SM: The amount of CP violation in the SM quark sector is orders of magnitude too small to explain the baryon-antibaryon asymmetry. Measurements of time-dependent CP violation in penguin transitions of $b \to s$ and $b \to d$ quarks, such as $B \to \phi K^0$ and $B \to \eta^0 K^0$ decay at Belle II, may provide new insights. Additionally, Belle II can improve the FCNC measurements of $b \to d$, $b \to s$, and $c \to u$ transitions, shedding light on the presence of new physics.

- New CP violating phases in the quark sector: The amount of CP violation in the SM quark sector is orders of magnitude too small to explain the baryon-antibaryon asymmetry. Measurements of time-dependent CP violation in penguin transitions of $b \to s$ and $b \to d$ quarks, such as $B \to \phi K^0$ and $B \to \eta^0 K^0$ decay at Belle II may provide new insights.

- Extended Higgs sectors: Many extensions to the SM, such as two-Higgs doublet models, predict charged Higgs bosons in addition to a neutral SM-like Higgs. It can be searched in flavor transitions to τ leptons $B \to \tau\nu$ and $b \to s\gamma$. The extended Higgs sector can also introduce additional sources of CP violation in the $B \to X_s\gamma$ process.

- lepton flavor violation (LFV) : LFV in charged lepton decay at such rates are key predictions in many neutrino mass generation mechanisms and other models of physics beyond the SM. Belle II are expected to achieve unrivalled sensitivities of τ decays and can analysis the LFV.

2

SuperKEKB and Belle II experiments

This chapter presents an overview of SuperKEKB and the sub-detectors of Belle II. We focus on the beam monitors, Central Drift Chamber and level-1 trigger system.

2.1 SuperKEKB

SuperKEKB is an asymmetric-energy electron-positron double-ring collider, which consists of a 7 GeV electron ring (high energy ring, HER), 4 GeV positron ring (low energy ring, LER), and an injector linear accelerator (linac) with a 1.1-GeV positron damping ring (DR), as shown in Fig. 2.1.

SuperKEKB is designed to reach peak luminosity of 6×10^{35}, $\mathrm{cm}^{-2}\mathrm{s}^{-1}$ which is up to 30 times higher than its predecessor KEKB and aims to collect 50 ab−1 of data as showed in Fig. 2.2. The luminosity L is given as:

$$L = \frac{\gamma_{\pm}}{2er_e}\left(1 + \frac{\sigma_y^*}{\sigma_x^*}\right)\left(\frac{I_{\pm}\epsilon_{y\pm}}{\beta_y^*}\right)\left(\frac{R_L}{R_{\epsilon_y}}\right) \tag{2.1}$$

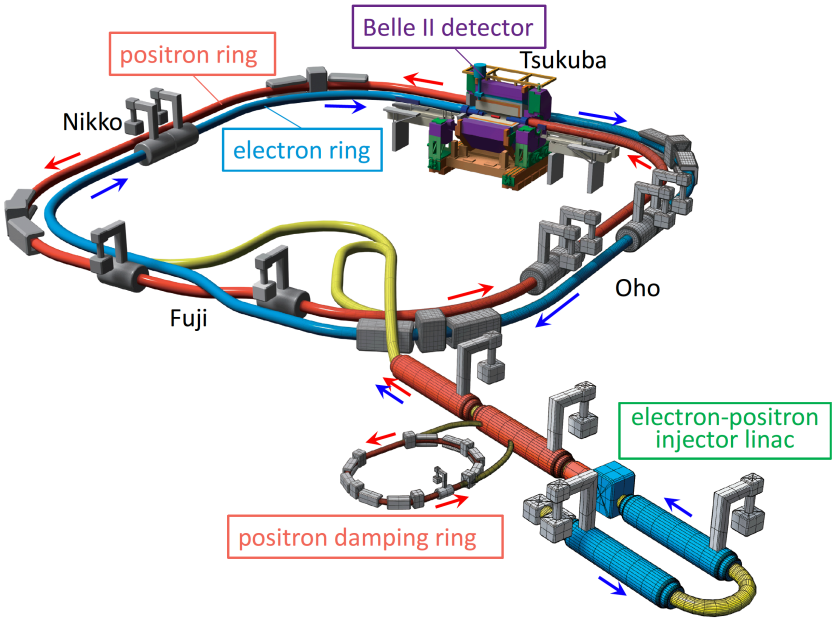


Figure 2.1: Schematic view of SuperKEKB[4].

where $\gamma_{\pm}$ are the Lorentz factor, e is the elementary charge, r_e is the classical electron radius, $\sigma^*_{x,y}$ are the beam sizes at the IP, $I_{\pm}$ are the beam current, β^*_y is the vertical β function at IP, $\epsilon_{y\pm}$ are vertical beam-beam tune-shift parameters and R_L, R_{ϵ_y} are correction factors for the geometrical loss due to the hourglass effect and the crossing angle at the IP. To achieve higher luminosity, higher beam currents, larger vertical beam-beam tune-shift parameters, and smaller vertical β functions and beam size are required at IP. In practical, SuperKEKB pursed much smaller β^*_y to increasing luminosity with the nanobeam collision scheme [6]. In the nanobeam collision scheme, beam bunches with sufficiently smallσ^*_x collide at a large horizontal crossing angle, as shown in Fig. 2.3. Table. 2.1 show the main Machine parameters proposed at target luminosity for SuperKEKB comparing with KEKB.

2.1.1 Final-focus superconducting magnet system

Final-focus superconducting magnet (QCS) system [7] is a very precise and complex system for realizing extremely small β^*_y, consists of eight main super conducting quadrupole magnets for focusing or defocusing beams, 43 super conducting corrector

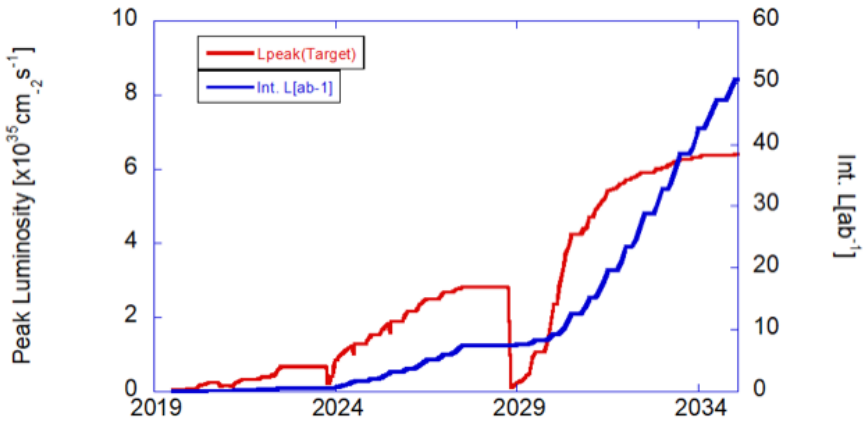


Figure 2.2: Luminosity plan of SuperKEKB [5].

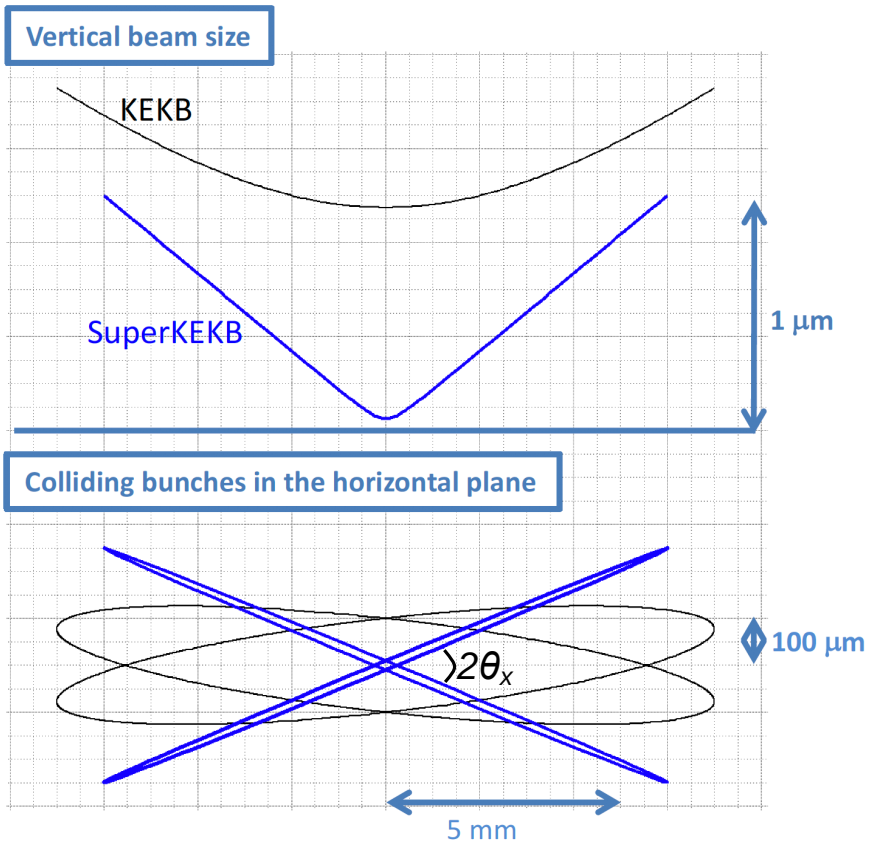


Figure 2.3: Schematic view of the nanobeam collision scheme. A large Piwinski angle($\phi_{\mathrm{Piw}} = \theta_x \sigma_z / \sigma_x^*$, where θ_x is the half of horizontal cross angle) was adapted to reduce β_y^*. [4]

	SuperKEKB		KEKB	
	LER	HER	LER	HER
Beam energy(GeV)	4.0	7.0	3.5	8.0
β_x^*/β_y^*(mm)	32/0.27	25/0.30	1200/5.7	1200/5.9
σ_x(μm)	10.1	10.7	147	170
σ_y(nm)	48	62	940	940
Beam current (A)	3.6	2.6	1.64	1.19
Number of bunches	2500		1584	
Number of e^-/e^+ in bunch(10^{10})	9.04	6.53	6.47	4.72
Luminosity ($cm^{-2}\ s^{-1}$)	6×10^{35}		2.108×10^{34}	

Table 2.1: Main Machine Parameters of KEKB and SuperKEKB (Designed)

magnets for tuning beams canceling the leak field from the main quadrupole magnets, and four compensation solenoid coils for canceling the detector solenoid field, as shown in Fig. 2.4. All magnets are installed in liquid helium vessels and accommodated in cryostats.

The superconduction magnets in QCS had quench events, which is a sudden transition to the normal state of the superconductor in the magnet, mainly induced by beam. In recent times, a large sudden beam loss at IP can also lead to QSC quench. After the QCS quench, it takes more than a few hours to resume the beam operation and re-optimize of beam optics. Thus, it is crucial to reduce the QCS quenches.

2.1.2 Collimators

The collimator [8] is one of the vacuum components used to shield the non-Gaussian tail in bunches by bringing heavy metal blocks in proximity to the circulating beam. They also function as machine protection systems by limiting physical apertures locally in the rings, and also prevent quenches in the QCS. Currently, two type of collimators are used, KEKB type which tapered chamber itself approaching the beam and SuperKEKB type which is a chamber has two movable jaws to approach the beam as Fig. 2.5. The SuperKEKB collimator is operated with a distance of 0.4 to 25 mm between the center of the beam channel and the tip of the jaw (that is, half aperture) for the vertical collimators and 2 to 30 mm for the horizontal collimators. In high-current operations, the jaws were occasionally damaged by hitting abnormal beams, and beam

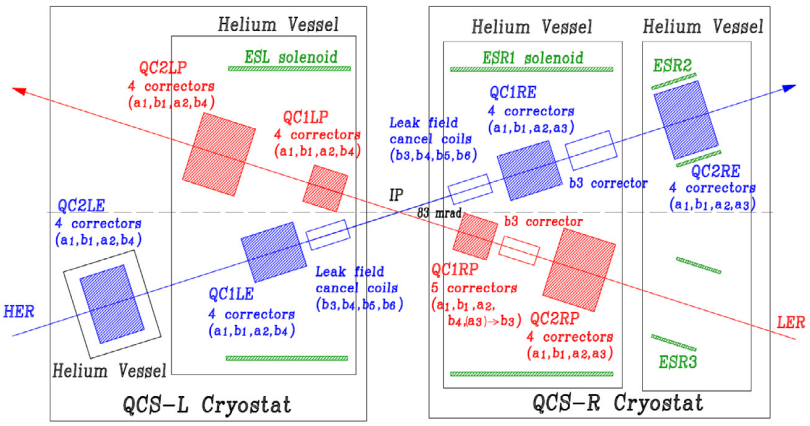


Figure 2.4: Schematic view of the QCS [7]. QCS consist of eight super conducting quadrupole magnets (QC1RP,QC1LP,QC2RP,QC2LP,QC1RE,QC1LE,QC2RE,QC2LE), 43 super conducting corrector magnets and four compensation solenoid coils (ESL,ESR1,2,3)

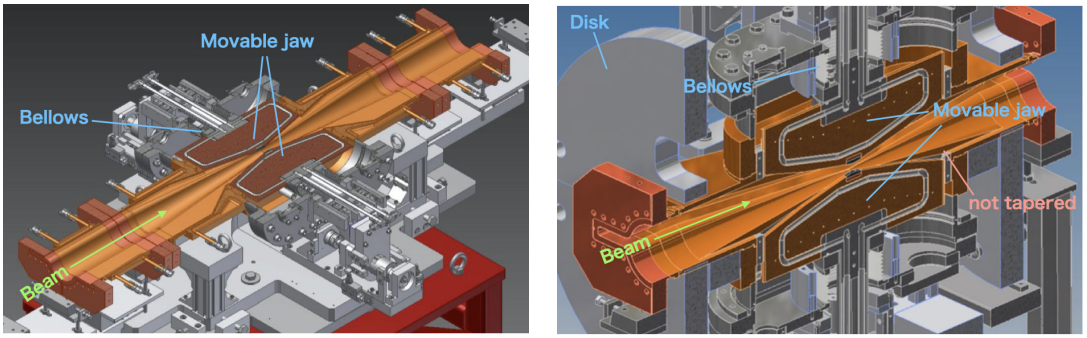
loss can also lead to its damage.The location of all collimators on SuperKEKB showed in Fig. 2.6.

2.1.3 Beam monitors

Beam monitors play an important role in SuperKEKB operations, which can measure of beam characteristics and stabilize of beam. Beam monitors can provide beam information including beam position, beam current, synchrotron radiation, beam loss etc. Here we focus on the monitors related to beam losses.

PIN photo-diodes

PIN photo-diodes (PDs) are semiconductor with high speed and high radiant sensitivity. Model BPW34[9] PDs are used as beam loss monitors at SuperKEKB [10] to protect hardware of SuperKEKB against beam loss and trigger beam abort. PDs are installed at downstream of all collimators and part of upstream to protect the collimators. Fig .5.3 (a) show the concept diagram of the PDs, it consists of P-region, N-region and intrinsic region in between. When the Beam losses occurred, electrons/positrons within a



(a) (b)

Figure 2.5: Structure of SuperKEKB type (a)horizontal collimator and (b) vertical collimator[8].

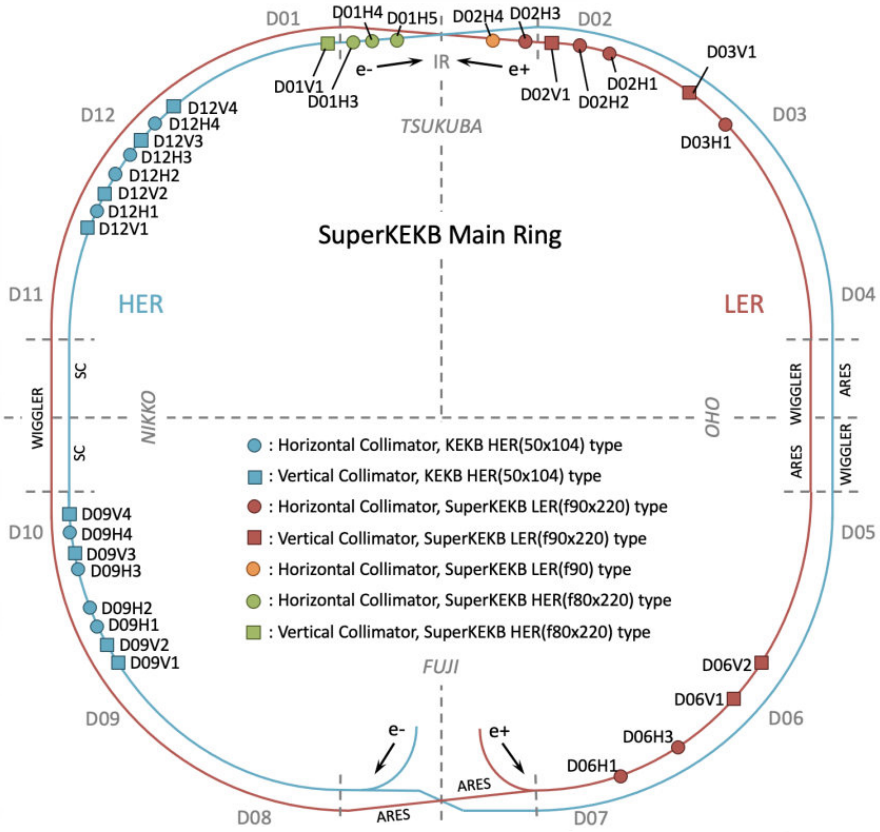


Figure 2.6: Location of all collimators at SuperKEKB[8]

beam collide with the inner walls of the beam chamber/collimators and create large electromagnetic showers. When a particle from such showers of sufficient energy enters the depletion region - intrinsic region or N-region, an electron - hole pair is generated and swept out by the reverse-bias field, creating current finally. Fig .5.3 (b) is the circuits applied followed the PDs, an integral circuit and amplifier are applied to readout signal. This integral circuit cause a delay for the signal readout and trigger abort. Special PDs directly pass raw signal to oscilloscope are installed at upstream and downstream of D06V1 and D06V2 collimators for sudden beam loss detection.

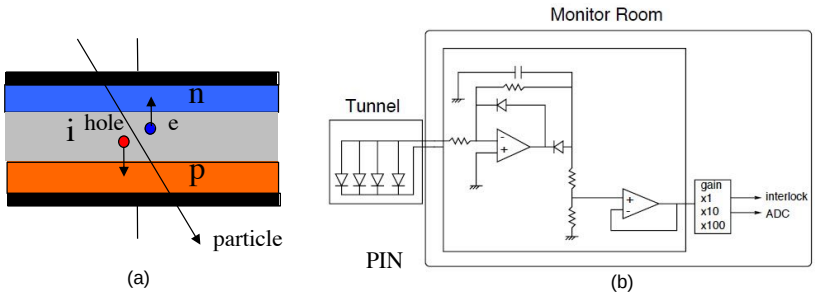


Figure 2.7: (a) Concept diagram of the PDs, which consist of P-region, N-region and intrinsic region in between. (b) Readout circuit for PDs [11]

Fiber detectors

Optical fiber sensors are also used as beam loss monitors in SuperKEKB. The schematic diagram of optical fiber sensors are demonstrated as Fig. 2.8. Once a charged high-speed particle passes through the optical fiber, it generates the Cherenkov light inside the optical fiber. And the Cherenkov light transport through the fiber can be detected with the PMT attached to the fiber. One optical fiber sensor was installed at downstream of D06V2 collimator as beam loss monitor. A 6-bundle multimode optical fiber with a diameter of 62.5 µm was laid from the D6 power supply building to the D06V2 collimator. Its length is 180 m, and from there it is connected to a single-mode optical fiber of 30 m and extended upstream and downstream of the collimator. The fibers are input to the PMT module and converted to electrical signals. Electrical signals are transported for abort trigger and waveform recorder.

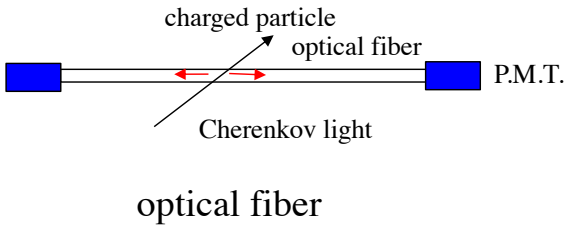


Figure 2.8: Schematic diagram of optical fiber sensors. Once a charged high-speed particle passes through the optical fiber, it generates the Cherenkov light inside the optical fiber. And the Cherenkov light transport through the fiber can be detected with the PMT attached to the fiber.

Bunch current monitor and beam oscillation recorder

The bunch current monitor (BCM) and the beam oscillation recorder (BOR) of the bunch-by-bunch (BxB) feedback system [12] measures charge and horizontal/vertical position of each bunch for each turn. The beam abort triggers BCM and BOR to record 4096 turns of data for all 5120 RF buckets with a sample rate as 2 ns. Once the beam loss occurs, the certain bunch current should be decrease comparing with previous turn, which can be measured by BCM. The BCM can indicate a certain bunch in turn has a beam loss. However, since only one BCM is installed in each ring, it is not possible to obtain information on where in the ring beam loss occurred. Besides, BOR provide the beam position information, which can infer the beam instability and used to analysis beam loss reason.

Diamond sensor

Diamond sensors [13], as solid-state ionization chambers, measure the current of electrons and holes produced by particles due to beam loss. Diamond sensors installed at the beam pipe, SVD detectors and QCS (Fig. 2.9) measure the radiation level inside the Belle II detector. If the measured radiation level is higher than 4 mrad in 10 µs or higher than 40 mrad in 1 ms, a beam abort request signal is issued and delivered to SuperKEKB abort system. Diamond sensors have a sample rate of 50 MHz and record ADC by integral 125 samples, which corresponding to 2.5 µs for each recorded ADC value.

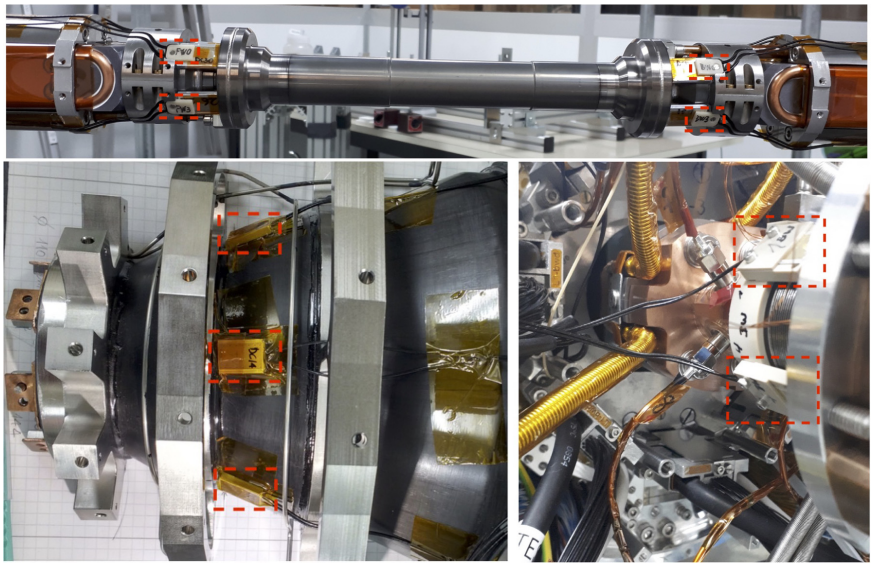


Figure 2.9: Photographs of diamond sensors (red dashed boxes) mounted on the beam pipe (top), on the backward SVD support cone (bottom left), and on the bellows close to the backward QCS (bottom right) [13]

2.1.4 Beam Abort system

In order to protect the hardware components of the detector and the accelerator against the high beam currents, beam abort system [14] was installed at SuperKEKB. Beam abort system consist of kicker magnets, pulsed quadrupole magnets, a Lambertson septum magnet and a beam dump. If received beam abort signal, the abort kicker will extract the circulating beam through the extraction window and the Lambertson DC septum magnet can lead the beam to the beam dump. Pulsed quadrupole magnets can enlarge the horizontal beam size at the extraction window to prevent the heating damage to the window. The dumped beam has a length of one revolution time as 10 µs.

Beam abort can be triggered by many sources, including magnets, loss monitors, RF cavity, Vacuum Pressure and so on. The loss monitors, including PDs, Diamonds sensors, optical fiber sensors can all issue beam abort with beam loss. And the issued beam abort signal will be sent to central control room through optical cables. After synchronization with beam revolution, beam abort signal will be delivered to beam kickers and every loss monitors for data recording. The process will take 17 µs to 30 µs varied from the source of beam abort. Given that the typical time for one turn of the

beam in SuperKEKB is approximately 10 microseconds (μs), the abort process typically takes 2 to 3 turns.

2.2 Belle II Detectors

The Belle II detector is the centerpiece of the experiment. It is a full–solid–angle detector with many sub-detector layers surrounding the interaction point of SuperKEKB. The detector is based on the design of the predecessor Belle detector, with the goal of maintaining the performance of the Belle detector in the presence of considerably higher background levels. A sketch of the Belle II detector and coordinate system is shown in Fig. 2.10. The detector consists of the following sub-detector: Pixel Detector (PXD), Silicon Strip Detector (SVD), Central drift chamber (CDC), Particle identification (PID) detector, Electromagnetic calorimeter (ECL), K–Long and Muon detector (KLM). A superconducting solenoid magnet is placed between ECL and KLM, generating a 1.5 T magnetic field along the beam axis. For the coordinate system, the x-axis is horizontal and toward the outside of the accelerator tunnel, which is roughly northeast. y is vertical upward. z is the Belle solenoid axis, which is the bisector of two beams; roughly toward the direction of the electron beam. ϕ is azimuthal angle around z-axis. $\phi = 0$ is defined for $(x, y, z) = (1, 0, 0)$. θ is zenith angle with respect to z-axis. $\theta = 0$ is defined for $(x, y, z) = (0, 0, 1)$

Here we give a brief introduction for every part, the CDC which is related to our work will be detailed explained in subsection 2.2.1.

PXD: The PXD is now the innermost sub-detector and directly surrounds the Beam pipe. The two layers of the PXD are at radii 14 mm and 22 mm from the beam line. PXD based on DEPleted Field Effect Transistor technology (DEPFET) has a small pixel as 50×55 μm^2 for inner layer and 50×75 μm^2 for outer layer. The primary purpose of the PXD is to measure the decay vertices under high hit rates coming from beam–related backgrounds.

SVD: The SVD comprises the outer four layers of the vertex detection sub-detector at radii 38, 80, 115, 140 mm. Three sizes of double–sided silicon microstrip detectors (DSSDs) are used for the outer, inner, and forward sections. SVD work for decay vertices measurement and low–momentum particle track reconstruction.

PID: The PID sub-detector contains two components: a Time Of Propagation(TOP)

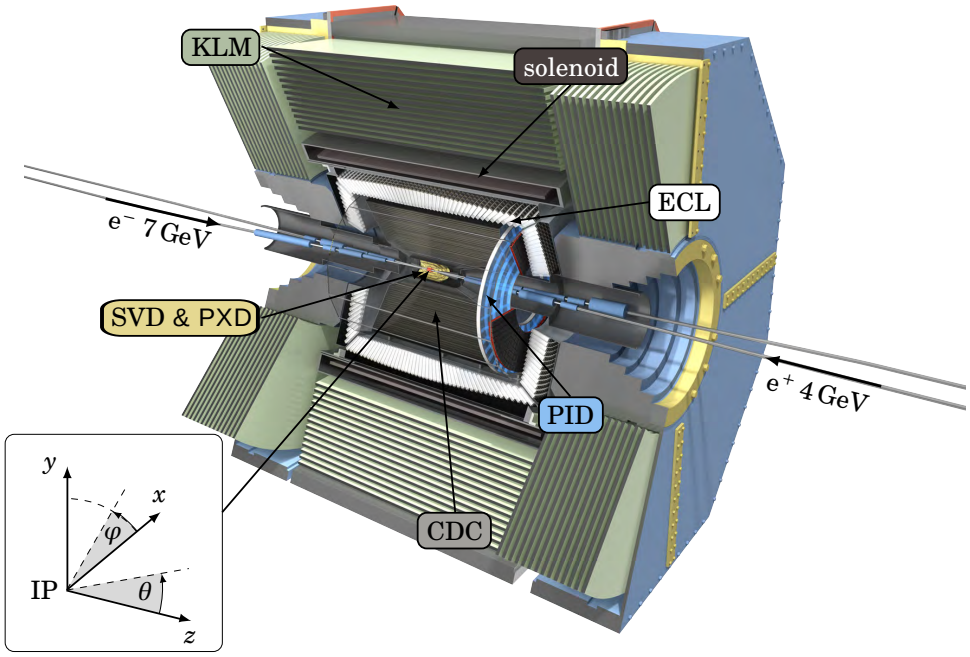


Figure 2.10: The Belle II detector and coordinate system [15]. For the coordinate system, the x-axis is horizontal and toward the outside of the accelerator tunnel, which is roughly northeast. y is vertical upward. z is the Belle solenoid axis, which is the bisector of two beams; roughly toward the direction of the electron beam. ϕ is azimuthal angle around z-axis. $\phi = 0$ is defined for $(x, y, z) = (1, 0, 0)$. θ is zenith angle with respect to z-axis. $\theta = 0$ is defined for $(x, y, z) = (0, 0, 1)$

detector and an Aerogel Ring Imaging Cerenkov (ARICH) detector at barrel and forward endcap region. The TOP detector is used for particle identification in the barrel region of Belle II, while the ARICH detector performs particle identification in the forward endcaps region. The main task for PID is to separate Kaon and Pion by separating angles of Cherenkov photon.

ECL: ECL consists of total 6624 thallium doped caesium iodide CsI(TI) crystals in the barrel, and 2112 CsI(TI) crystals at end-caps. Photodiodes are glued to every crystal to detect the scintillation light. The key roles of the ECL is to detect photons, identify electrons.

KLM: The KLM is made of alternating layers of 470 mm thick iron plates and detector components. Scintillators are used in the entire endcaps and first two layers of the barrel section, with RPCs used for the remaining barrel layers. In the barrel there are 15 detector components and 14 iron plates. In the forward (backward) end–cap there are 14 (12) detector layers and 14 (12) iron plates. The KLM is used for K_L^0 and $\mu^{\pm}$ identification.

2.2.1 Central Drift Chamber

The Central Drift Chamber (CDC) is the main tracking detector in the Belle II Experiment. The main task of the CDC is to measure the momenta of charged particles precisely by reconstructing charged tracks which curve in the 1.5 T magnetic field along the z axis. The CDC can also contribute to the trigger system and particle identification.

The CDC is a wire chamber consisting of 42240 field wires and 14366 sense wires, filled with the gas mixture of 50% He and 50 % C_2H_6. The wires are arranged radially in rectangular cells. Every 8 field wires surrounding each sense wire as illustrated in Fig 2.11. A positive high voltage is applied on the sense wire while field wires are connected to ground. Charged particles passing through the chamber will ionize the gas and generate electrons. The electrons accelerate and drift towards the sense wires, ionizing more gas atoms in the high electrical field surrounding the wire and finally resulting in an electron avalanche. When the induced signal exceeds the discriminator threshold on CDC front-end electronics (FE)[16], it is judged as "CDC hit", and its TDC and ADC are measured.

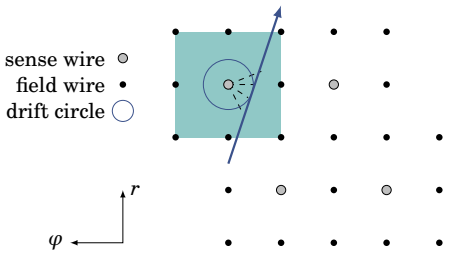


Figure 2.11: Sense wire and field wire distribution[17]. Each sense wire is surrounded by 8 field wires

Fig. 2.12 shows the concept diagram of cross-section for the CDC wires. The total 56 layers of wires in the chamber are arranged into 9 super layers (SL) totaling a cylindrical volume with an outer radius of 113 cm and an inner radius of 16 cm. The innermost SL consists of 8 layers of wires with small cell configuration to cope with the higher background near the IP. The remaining SL have 6 layers. Two types of orientation of wires are used in CDC, Axial and Stereo as Fig. 2.13. Axial wires are parallel to the z-axis and used for track 2-dimension reconstruction in $r - \Phi$ plane. And stereo wires are inclined/skewed with respect to the beamline, allowing for a 3D reconstruction of tracks. Stereo SLs are skewed between ±45 mradto ± 74 mrad, where sign corresponding to the two orientations for stereo wires as Fig. 2.13 (b).

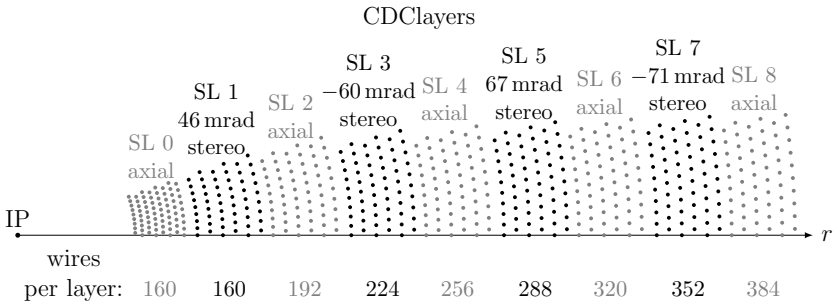
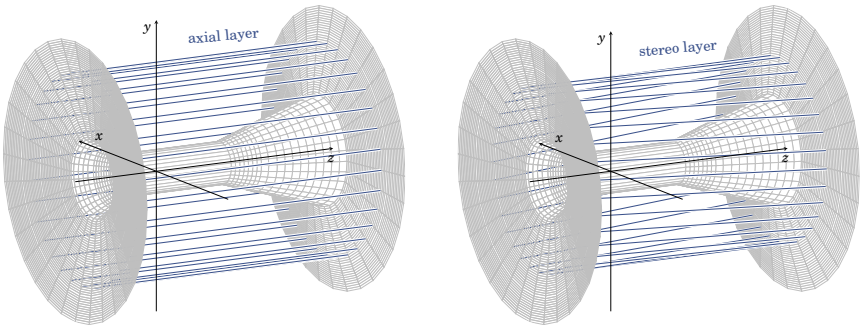


Figure 2.12: Layer configuration of the CDC with 9 SLs [17].



(a) (b)

Figure 2.13: Wire orientations of CDC: (a) Axial wires are parallel to beamline (z-axis); (b) Stereo wires are skewed with respect to beamline [18].

2.2.2 Trigger System

The Belle II detector is expected to receive a total 20 kHz trigger rate for physics process at designed luminosity, as illustrated in Table 2.2. Samples of Bhabha and $\gamma\gamma$ events will be used to measure the luminosity and to calibrate the detector responses. In order to deal with the copious amount of beam-induced background and select events for physics analysis with high efficiency, the Belle II trigger system is partitioned into two consecutive levels. The level-1 trigger, implemented in deadtime-free pipelined hardware, performs a partial online event reconstruction and sends a signal to the High level trigger(HLT) whenever certain criteria are satisfied. The HLT is implemented in software and performs a more detailed selection on the events triggered by the L1 trigger in order to further reduce the background among the events. Here we give a brief introduction for level-1 trigger.

level-1 trigger

Level-1 trigger which implemented in deadtime-free pipelined hardware, should fulfill the following requirements:

1. High efficiency for physics events from $\Upsilon(4S) \rightarrow B\bar{B}$ and from continuum;
2. Maximum average trigger rate of 30 kHz;
3. Fixed latency of about 4.2 µs;

Physics Process	Cross Section (nb)	Rate (Hz)
$\Upsilon(4S) \longrightarrow B\bar{B}$	1.2	960
Hadron production from continuum	2.8	2200
$\mu^+\mu^-$	0.8	640
$\tau^+\tau^-$	0.8	640
Bhabha ($\theta_{\text{lab}} \geq 17^\circ$)	44	$350^{(*)}$
$\gamma\gamma$ ($\theta_{\text{lab}} \geq 17^\circ$)	2.4	$19^{(*)}$
2γ processes ($\theta_{\text{lab}} \geq 17^\circ$, $p_t \geq 0.1, \text{GeV}/c$)	~80	~15000
Total	~130	~20000

* The $\gamma\gamma$ and Bhabha rate is pre-scaled by a factor of 1/100 due to the large cross section.

Table 2.2: Total cross-section and trigger rates at designed luminosity from various physics processes at $\Upsilon(4S)$ [19]. The θ_{lab} is the difference θ of two particles in the laboratory coordinate system.

4. Timing precision of less than 10 ns;

Considering expected physics trigger rate, we should reduce the total level-1 background trigger rate to 10 kHz at luminosity of $6 \times 10^{35}, \text{cm}^{-2}\text{s}^{-1}$. The schematic overview of the Belle II trigger system is shown in Fig. 2.14. Full level-1 trigger consist of four sub-trigger collect information from each corresponding Belle II sub-detectors, Global Reconstruction Logic (GRL) and Global Decision Logic (GDL). The CDC sub-trigger provides the charged track information (momentum, position, charge, multiplicity). The ECL sub-trigger gives energy deposit information, energy cluster information, Bhabha identification, and cosmic-ray identification. The TOP sub-trigger gives precise event timing. The KLM sub-trigger gives muon track information. GRL will collect information from four sub-trigger and apply algorithms for the combination of sub-trigger information. Finally, all the information deliver to GDL to make a final decision of whether the event meet certain condition and should be kept or not. The Level-1 trigger signal is designed to be output to the determination in 4.2 µs after the beam collision.

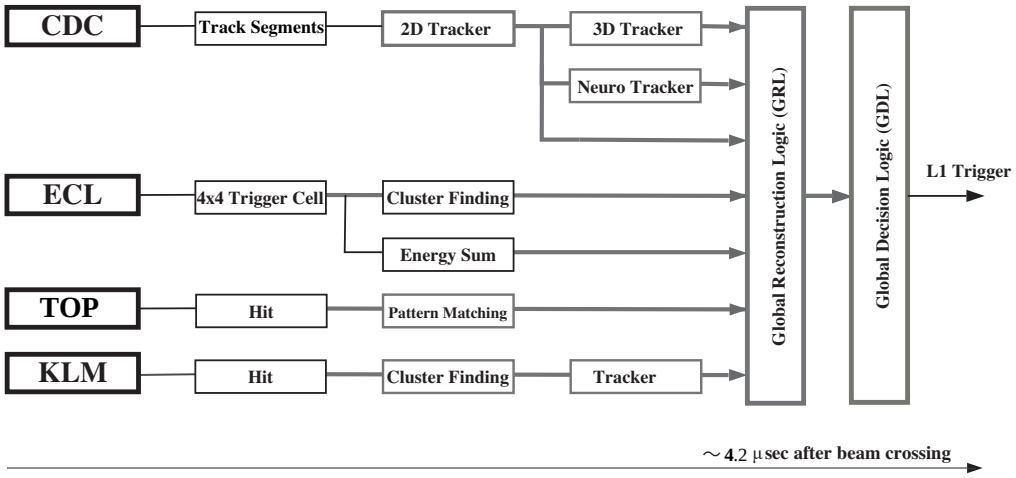


Figure 2.14: Schematic diagram of the full Level-1 trigger system[20]

3

Sudden beam loss and Fast beam loss monitors

This study focuses on the observation of an increasing occurrence of sudden beam loss (SBL) events during recent SuperKEKB operations. These events generated from the rapid loss of a portion of the stored beam, and their underlying causes remain unknown. The SBL-induced background can be detrimental to the Belle II detectors, and in some cases, cause damage. Our research seeks to identify the specific locations where SBL events occur and to gain a deeper understanding of their root causes. This knowledge will enable us to develop strategies to mitigate the adverse effects of SBL and prevent damage to the detectors.

3.1 Beam loss

Beam losses occur when electrons/positrons within a beam collide with the inner walls of the beam chamber, leading to a reduction in the overall beam intensity.

Electromagnetic showers can be generated as a result of these collisions, with some of the shower particles detectable outside the beam chamber. Generally speaking, beam losses are typically unavoidable and are localized at the collimator system or other aperture limits. These beam losses can occur continuously during accelerator operation and are associated with the beam lifetime and transport efficiency within the accelerator. The minimum possible loss rate is determined by the theoretical limit on the beam lifetime, which is mainly influenced by collective effects, including Touschek effect, beam-beam interactions, collisions, transverse and longitudinal diffusion, residual gas effect, halo scraping and beam instabilities [21]. Typically, these beam losses manifest over extended time scales, surpassing 10 turns, and certain instances can be mitigated through the utilization of collimators.

3.1.1 Sudden Beam loss

Sudden beam losses (SBLs) differ from general beam losses in that they are rapid, harmful, and their causes remain unknown. These events occur less than three or four turns, as illustrated in Fig. 3.1. At SuperKEKB SBLs more likely occurred in the Low Energy Ring (LER) which is roundly two times than in the High Energy Ring (HER). Certain SBLs that occurred in the LER led to severe damage to the vertical collimators, as depicted in Fig. 3.2. This damage made collimators difficult to effectively control the beam background, and some SBLs resulted in large radiation doses around the interaction region (IR), which in the worst case caused quenches of QCS. These "catastrophic" events seem to be more frequent at higher beam currents, limiting the maximum beam currents during machine operation. Thus, our primary objective and immediate concern is to identify the root causes of these sudden beam loss events and devise solutions to attain high luminosity.

3.2 Fast loss monitor system

To pinpoint the locations of SBLs, a beam loss monitor system that covers the most critical regions and possesses fast response times is essential. At present, the beam monitoring system primarily used a combination of PIN diodes and ionization chambers that typically have a response time of 20 µs [10] determined by the readout electronics,

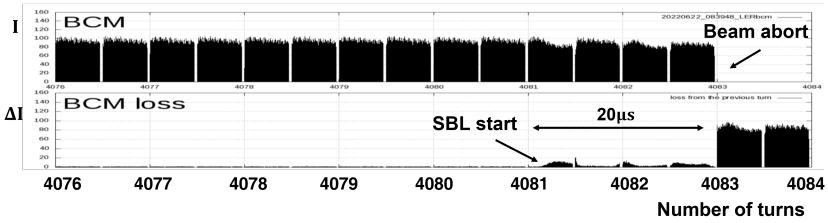


Figure 3.1: Bunch Current monitor record in June 2021. I is the current for single bunch and $\Delta I = I(b, n) - I(b, n-1)$, where b is the bucket number and n is the number of turns. This sudden beam loss happens in only 20 μs

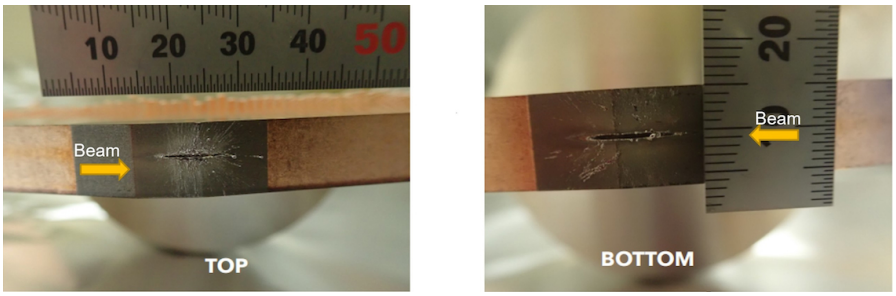


Figure 3.2: LER D2V1 collimator heads severely damaged by the sudden beam loss in June 2021.

which is insufficient for the precise location determination for SBLs, because for previous SBLs we found that most beam loss monitors fired at same time and can not determine a certain first fired monitors. Besides, in order to perform measurements in proximity to the main rings, it is imperative that the monitors should meet the requirements of minimal efficiency deduction at a radiation level of approximately few hundreds to one thousand μGy/h, which is the typical radiation level at collimators[22]. To meet this requirement, we have developed a fast loss monitor system consisting of two types of fast loss monitors: Electron Multiplier Tubes (EMTs) and Photon Multiplier Tubes (PMTs) with scintillators. Both are directly connect to the new readout system utilizing oscilloscope and TDC module, recording timing and waveform with a precision of ~ 8 ns.

3.2.1 Fast loss monitor detectors

Photon multiplier tube with scintillator

Excepted EMT, we use pure CsI attached to PMT using optical glue as an option for our loss monitors, as showed in Fig. 3.3.

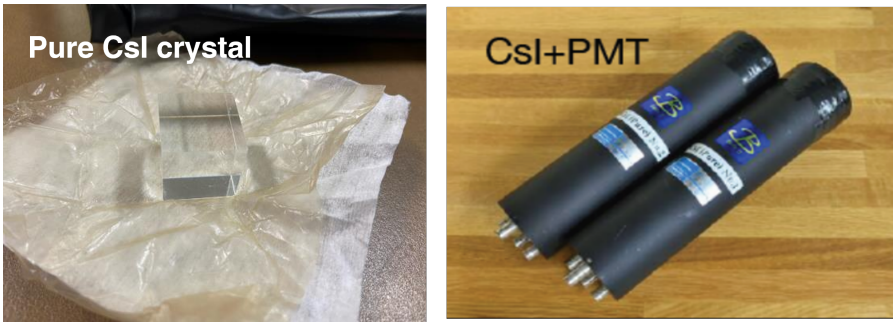


Figure 3.3: Pure CsI crystal and PMT attached with CsI

This setup works as showed in Fig. 3.4. An Incident particle from beam loss can excite electrons in CsI crystal and then it rapidly de-excite by emitting scintillation photons. Emitted photons are converted into photo-electrons by the photo-cathode and then focused and multiplied by an arrangement of dynodes in multiple stages connected in series to the externally applied High Voltage. The amplified signal is directly readout from oscilloscope. Pure CsI crystal have a very short decay time ~

16 ns, which can satisfied the fast response requirement.

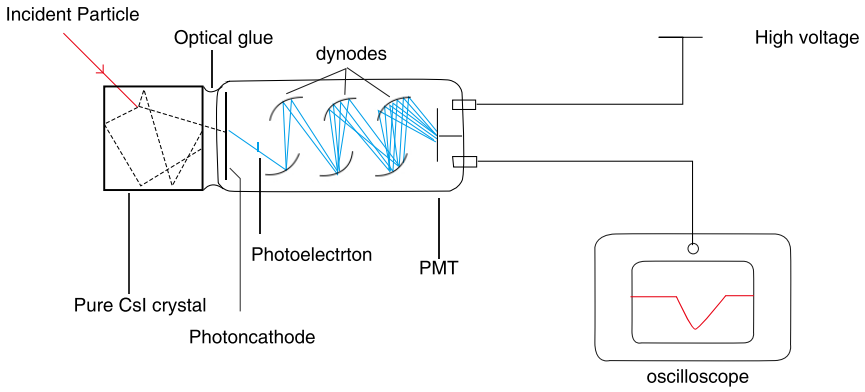


Figure 3.4: Schematic drawing of the PMT setup. Incident particle generate photons in pure CsI and photons are converted into photo-electrons by the photo-cathode and then focused and multiplied by an arrangement of dynodes in multiple stages connected in series to the externally applied High Voltage. Signal are directly readout from oscilloscope.

Electron multiplier tube

As shown in Fig. 3.5, Electron Multiplier Tubes (EMTs) are essentially Photon Multiplier Tubes (PMTs) without a photocathode; instead, aluminum is deposited on the cathode to achieve high radiation tolerance. EMTs originally have been developed as muon beam monitor in the T2K experiment [23]. EMTs were examined to have great radiation tolerances that have less than 5% degradation under $\sim 2\,\mathrm{MGy/h}$[24].

When a charged particle passes through the EMT, it produces secondary electrons either at the surrounding aluminum cathode or at the dynodes. The emitted electrons are then accelerated, bombard the downstream dynodes, and produce additional electrons.

For our fast beam loss monitors, we used the same prototype EMTs as T2K experiments but change the divider circuits to E10679-Y003 [26], which provides a maximum operation voltage at 1100 V. These EMTs are manufactured from R9880U PMT and provided with time response$\sim$ 2ns [25]. Table 3.1 show the Specifications of EMTs and PMTs setups.

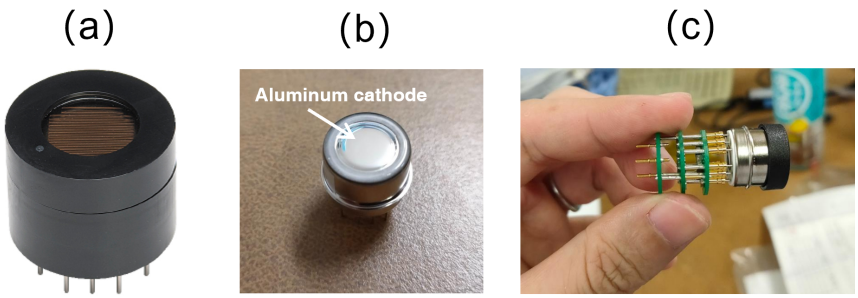


Figure 3.5: (a)Origin R9880U PMT [25] (b) Prototype EMT with replaced aluminum cathode (c) Prototype EMT with circuit

	Size(mm)	Gain	Decay time(ns)	Rise Time (ns)	Transit Time (ns)	Max HV(V)
EMT Setup	Dia.16	2.0×10^6	NAN	0.57	2.7	1100
PMT Setup	Dia.60	2.5×10^6	16	0.8	16	3500

Table 3.1: Specifications of EMT and PMT setup. Here, taking the same time response as R9880U for EMTs. PMTs setup show H2431 PMT here.

Detectors location

To detect the beam loss effectively with limited number of fast loss monitors and cable, we installed at possible location for sudden beam loss at LER, included:

- D06H3: A horizontal collimator. It is an important collimator to protect the accelerator components in the main ring from the accidental firing of the LER kicker magnet.
- D06V1: A primary vertical collimator used in the LER. Its position is phase-matched with the D02V1 collimator located closest to the interaction point (IP) or Belle II detector. The D06V1 is primarily employed to control the beam background resulting from beam injection.
- D06V2: A vertical collimator. It is located downstream of the D06V1 and serves as a complement to D06V1 for beam background control by having a phased difference from D06V1.
- D02V1: A primary vertical collimator for LER. It is located closest to the IP (or Belle II). If the collimator head gets damaged, it could significantly increase the

beam background and may require stopping the machine operation.

All fast loss monitors are installed around collimator because that the collimators are the narrowest parts in the mainly where we most likely to detect SBLs first. We also installed 3 fast loss monitor at HER rings. But with the limited number of monitors, we can hardly perform detail analysis for HER sudden beam loss, thus in this thesis we will focus on the analysis for LER. These collimators and fast loss monitors location in the main ring show in Fig. 3.6.

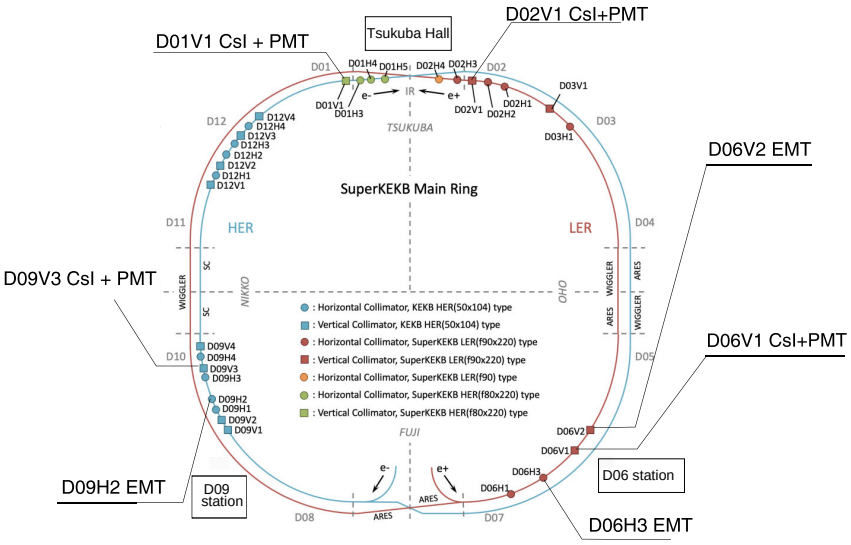


Figure 3.6: Location of each collimator in the main ring and the installed fast loss monitors.

The photo of install EMT and PMT+ CsI scintillator are showed in Fig. 3.7. Considering the good radiation hardness of EMTs, they are set closed to collimators ($\sim$ 10 cm). While for PMTs, they are set $>$ 100 cm from the collimators.

3.2.2 Readout system, Time synchronization and White rabbit module

NH-5D-2E (BNC) cables are used for signal transition and NH-TVECX (SHV) cables are used for HV providing, both of them uses polyethylene sheath material for flame retardant. In order to minimize electronics noise and cable costs, each fast loss monitor

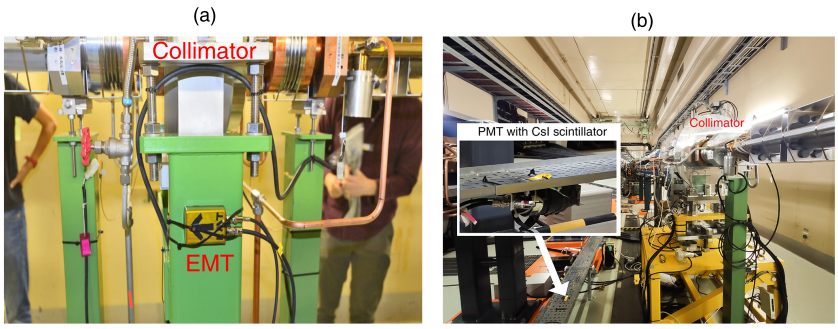


Figure 3.7: (a) Installed EMT at D06H3 collimator (b) Installed PMT+ CsI scintillator at D02V1 collimator

is connected to the nearest local control room (LCR) using coaxial cable for data acquisition. We employ a dual system setup, as illustrated in Fig. 3.8, to facilitate timing and signal recording from the fast loss monitors. The monitor signals are directed to the 3403D MSO oscilloscopes for waveform storage and precise timing measurements; the oscilloscopes are triggered by the beam abort signal and would record 5 ms waveform before and after beam abort. Simultaneously, the signals are passed through a discriminator, which applies a specific threshold, and a level adapter to convert the NIM signal to TTL. The resulting signal is then fed into the TDC module, enabling the recording of timing information for every signal exceeding the threshold.

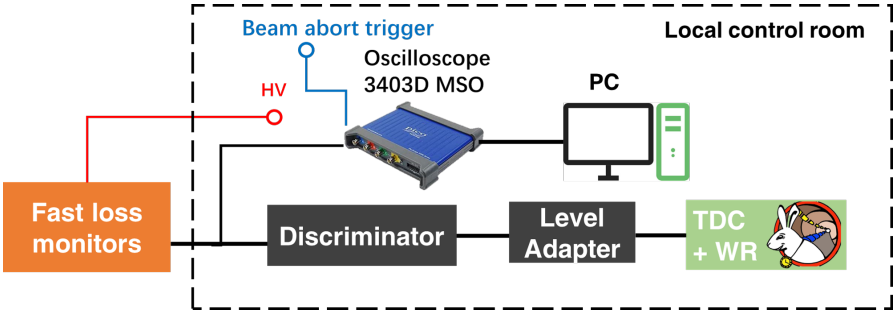


Figure 3.8: Concept diagram for fast loss monitor readout system.

To achieve synchronization among the various fast loss monitors dispersed over a few kilometers, we employ two primary strategies. Firstly, the oscilloscopes are equipped to capture the beam abort trigger signal sent from the Central Control Building (CCB) using optical cables of known length. By comparing the signal from

the fast loss monitor with the beam abort trigger, we can calculate a relative time difference, enabling effective comparisons across different local systems.

Secondly, we use the White Rabbit system [27](WR). WR is a fully deterministic Ethernet-based network for general purpose data transfer and synchronization. It can synchronize over 1000 nodes with sub-ns accuracy over fiber lengths of up to 10 km.

The slave nodes in our system comprise the Simple PCIe FMC Carrier (SPEC)[28] and the FMC-DIO card[29], which are connected to the grand-master module and synchronize their internal timestamp with that of the grand-master. These slave nodes function as TDCs in our system and offer an accuracy of 8 ns in providing the GPS timestamp for all input entries. The pictures of installed WR system shown in Fig 3.9

Figure 3.9: White Rabbit modules: the upper white module in the left figure is the GPS receiver. The black 1U-height module under it is the grand-master module of White Rabbit timing system. The right two pictures show the slave node. The PCI Express type slave module is inserted into the commercial PC. The individual slave nodes are connected with the grand-master module via the single mode optical cable.

In addition, injection signal, revolution signal, beam abort trigger, and beam gate timing are also recorded and synchronized with White Rabbit to enable a comparative study between fast loss monitors, beam monitor, and beam status.

The dual system setups can serve as complementary to each other, allowing for comprehensive analysis. One aspect involves recording the waveform of each fast loss monitor in conjunction with beam aborts, providing detailed information about the monitor's response. The other aspect involves precise timing measurements of signals that exceed a specified threshold using White Rabbit (WR) technology. By combining

these two approaches, we can obtain a comprehensive understanding of the fast loss monitor behavior and ensure accurate timing measurements for relevant signals.

Table. 3.2 summarized the installed Fast loss monitors specification.

Location	Type	HV (V)	LCB	Cable length (m)
LER				
D02V1	PMT	500	Tsukuba B4	158
D06V1	PMT	600	D06 station	151
D06V2	PMT	1200	D06 station	114
D06V2*	EMT	650	D06 station	114
D06H3	EMT	650	D06 station	250
HER				
D01V1	PMT	500	Tsukuba B4	157
D09V3	PMT	950	D10 station	223
D09H2	EMT	650	D10 station	200

* At D06V2, we switch from PMT to EMT at April 21th 2022 because of possible PMT damage.

Table 3.2: Summary of installed PMTs and EMTs setup.

4

Timing analysis of sudden beam loss

In order to pinpoint the locations of SBLs, we conducted a meticulous timing analysis of both fast loss monitors and other beam monitors. The subsequent sections will show the details of the timing analysis, covering topics including methodology employed, calibration procedures, and the synchronization.

4.1 Beam loss timing detection

We employed distinct timing methods based on the signal discrepancies observed across various monitors. These methods are categorized into three types: 1. Fast loss monitors, 2. Bunch current monitors and Beam Oscillation Recorder, and 3. Interaction region monitors and PIN diode.

4.1.1 Fast loss monitor

The timing analysis is mainly focused on Fast loss monitors. The typical signal from fast loss monitors is showed in Fig. 4.1. Both EMT and PMT have fast response for beam loss with rising edges ~ 20 ns for PMT and ~ 10 ns for EMT.

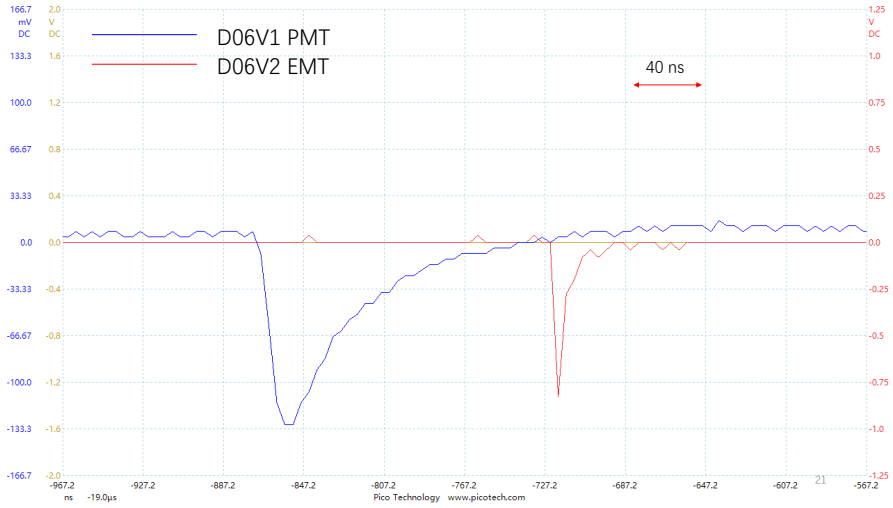


Figure 4.1: Waveform for D06V1 PMT (blue line) and D06V2 EMT (Red line) when injection cased beam loss happen at 5th June 2022

As outlined in Section 3.2.2 of Chapter 3, we have implemented a dual system for the readout of fast loss monitors. In the case of the oscilloscope configuration, we employed a "double threshold method" to measure the precise timing of the rising edges of the beam loss signal, as depicted in Fig. 4.2. This approach involved setting a high threshold to capture large beam loss events while minimizing incorrect timing due to pedestal noise. Then, a low threshold was utilized to searching for every first point crossing it, in the range of $(-100\,\mu s, 100\,\mu s)$ centered on the point crossing the high threshold. The timing cross low threshold is regarded as timing for this event. When faced with beam loss events resulting in multiple peaks, we discerned individual peaks by detecting every two instance of the signal crossing the low threshold. The specific values for the high threshold and low threshold were adjusted accordingly, taking into account the gain and pedestal noise inherent in the system. In detailed, We summarized the amplitude for pedestal noise into a histogram and calculated its standard deviation σ; then we set the low threshold as the 5 times of the σ and round

up to a multiple of 10 for simplification. For high threshold setting, we manually pick one hundred events with injection beam loss, check the minimum of signal amplitude, and set the threshold as half of the minimum of signal and round down to a multiple of 10. If it less than 10σ, we forced it to be 10σ for pedestal noise rejection. And for TDC & WR setup, we directly used the timing from TDC module which recorded the timing from discriminator crossing threshold with precision ~ 8 ns. The threshold setting shows in Table. 4.1.

Fast loss monitor	D02V1	D06V1	D06V2	D06H3
High Threshold (mV)	-50	-50	-100	-300
Low Threshold (mV)	-10	-30	-40	-40
Discriminator (mV)	-75	-130	-80	-90

Table 4.1: Threshold setting for fast loss monitor timing

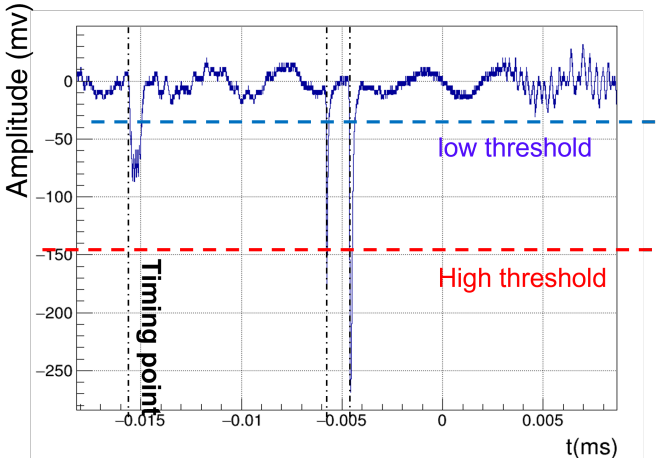


Figure 4.2: Double threshold timing method. Find a point crossing the high threshold (red line), then search for every first points crossing the low threshold (blue line) as precise timing point. Multi-peaks events are searched in the range (-100 μ s, +100 μ s) and distinguished when every peaks cross the low threshold twice (black dash line

4.1.2 Bunch current monitors and Beam Oscillation Recorder

In the case of the BCM and BOR, our objective is to identify specific bunches that exhibit changes in bunch current or bunch oscillation. To achieve this, we analyze the difference in BCM/BOR data between every single bunch and the same bunch previous turn. As showed in Fig. 4.3, We determined the timing from $\Delta I = I(b, n-1) - I(b, n)$ for BCM, and $\Delta x = x(b, n-1) - x(b, n)$ for BOR, where b is the bucket number, n is number of turns, I for bunch current and x is the beam vertical/horizontal position. The start point of ΔI and Δx are identified by locating the first instance of two consecutive bunches with above a certain threshold, or a single bunch above twice the threshold. The thresholds are set as 3 times of standard deviation for pedestal noise amplitude distribution.

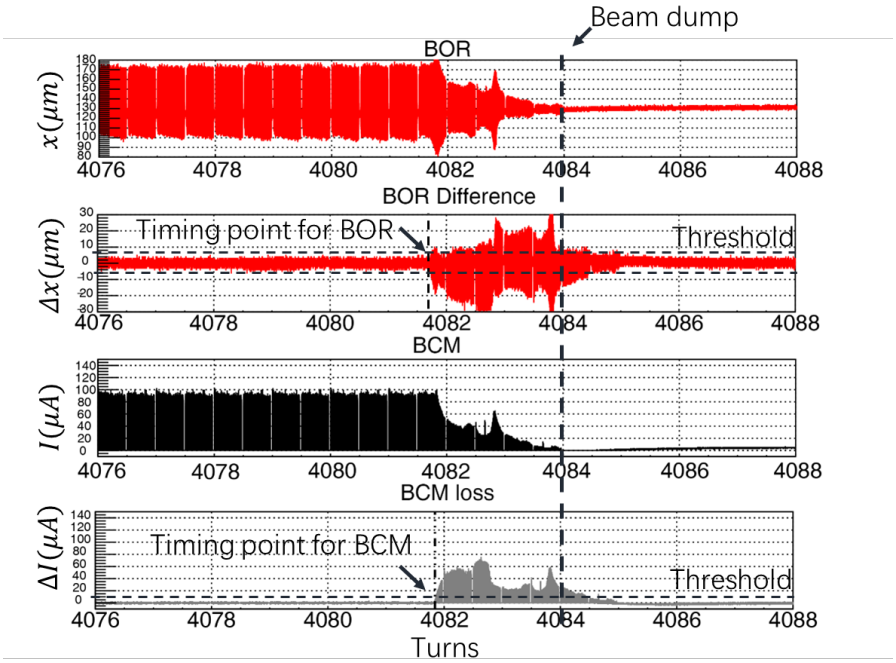


Figure 4.3: BCM and BOR timing method. We determined the timing from $\Delta I = I(b, n-1) - I(b, n)$, and $\Delta x = x(b, n-1) - x(b, n)$ where b is the number of bunches, n is number of turns, I for bunch current and x is the beam vertical/horizontal position. The start point of ΔI and Δx are identified by locating the first instance of two consecutive bunches with above a certain threshold, or a single bunch above twice the threshold.

4.1.3 PIN photo-diodes

In the case of PDs, which are readout from an integral circuit and exhibit a time responding time of approximately 20 µs, our primary objective is to accurately determine the starting point of each signal. The waveform and timing analysis method are illustrated in Fig. 4.4. Initially, we identify the continuous rising segments of the waveform that exceed the threshold, which typically defined as three consecutive sample points with an increasing trend. This step helps mitigate errors introduced by pedestal jitters. Subsequently, we simultaneously examine both the forward and backward portions of the waveform to identify sections with a 10-point positive average slope. We locate the last point with a positive slope in the opposite direction. Finally, we impose a minimum edge width requirement to discriminate against background noise.

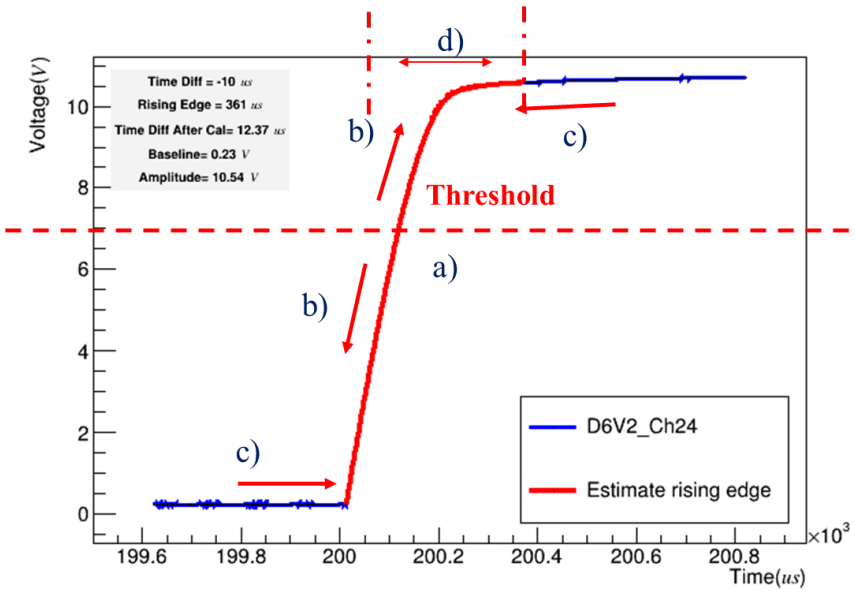


Figure 4.4: PIN diodes timing method. Blue line is the waveform of PIN diode, red line show the estimated rising edge from following 4 steps: (a) Find the continuous rise parts above threshold. (b) Simultaneously look forward and backward for waveform with a positive average slope. (c) Find the last point with a positive slope in the opposite direction. (d) Reject background with a cut on width of edge

4.1.4 Diamonds sensors and Optical Fiber sensors and others

For Diamonds sensor's case, due to the limited sample rate as 2.5 µs/s, we only time the first point crossing the threshold. As for optical fiber sensors and special PDs which deliver the raw signal to recorder, due to their fast time response around few nanoseconds, we also timing it by finding the first point crossing threshold.

4.2 Calibration and synchronization

Calibration and synchronization for all timing from different monitors and local readout systems are necessary for comparison between them. As showed in Fig. 4.5, a beam loss causing beam abort will send the abort signal to every local system, which is the best reference for comparisons. For BCM/BOR case, the timing of beam dumped is recorded and can deduce the beam abort timing also (see Fig. 4.3). To calibrate out the cable length from fast loss monitors to local system and from central control building (CCB) to local system, we directly measured delay of analogue signal and optical cables by using a test pulse and showed as Table. 4.2.

analogue cable	Delay time (µs)
D02V1 PMT to Tsukuba B4	0.799
D06V1 PMT to D06 station	0.765
D06V2 PMT/EMT to D06 station	0.579
D06H3 EMT to D06 station	1.250
Optical cable	Delay time (µs)
CCB to D06 station	3.77
CCB to D02 station	10.01

Table 4.2: Delay of the optical and analogue cable length for LER BLMs

We use the $\Delta T = t_{monitors} - t_{abortsignal}$ to compare different monitors. And including the calibration, ΔT can be writ en as:

$$\begin{aligned} \Delta T &= t_{\text{monitor}} - t_{\text{abort}} \\ t_{\text{monitor}} &= t^{\text{local}}_{\text{monitor}} - c^{\text{local}}_{\text{monitor}} \\ t_{\text{abort}} &= t^{\text{local}}_{\text{abort}} - c^{\text{local}}_{\text{CCB}} \end{aligned} \tag{4.1}$$

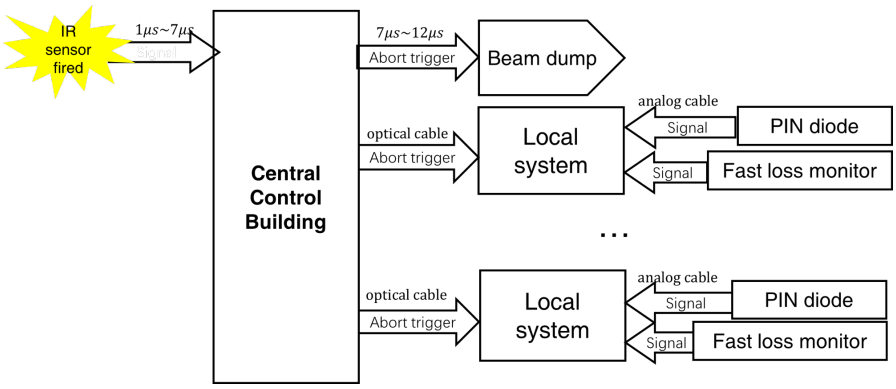


Figure 4.5: Timeline for a general beam loss events

Where t_{monitor} and $t_{\text{abort signal}}$ are the actual monitor timing and abort timing; $t^{\text{local}}_{\text{monitor}}$ and $t^{\text{local}}_{\text{abort}}$ are monitor timing and abort timing recorded in local system; $c^{\text{local}}_{\text{abort}}$ and $c^{\text{local}}_{\text{monitor}}$ are the transition time for abort signal and monitors signal due to analogue/optical cable length. By calculating the ΔT for different monitors, we can pinpoint which monitors detected beam loss first. All the following discussion is based on ΔT.

4.3 Multi-sensor comparison

Upon completing the calibration and synchronization process, we acquired the ΔT values for each sensor. To examine the timing of beam loss events and determine if they originated from the same bunches, we compiled the data into a location versus timing plot. A representative example of a sudden beam loss event in June 2022 is illustrated in Fig. 4.6. The x-axis represents the distance from various detectors to the interaction region, with each vertical dashed line indicating the position of a specific detector. The y-axis represents the measured ΔT. The upward dashed line denotes the inferred time position relationship based on the first bunch with observed current loss at the BCM, and the interval between adjacent slashes corresponds to one turn (approximately 10.061 us). From the plot, we can discern the following information:

a) The EMT at D06V2 detected the initial beam loss.

b) D06V2 and D02V1 observed the beam loss in the first turn, while the BCM and D06V1 detected it in the subsequent turn. These losses originated from nearly

identical bunches.

c) The beam loss was initially detected two turns prior to the beam abort.

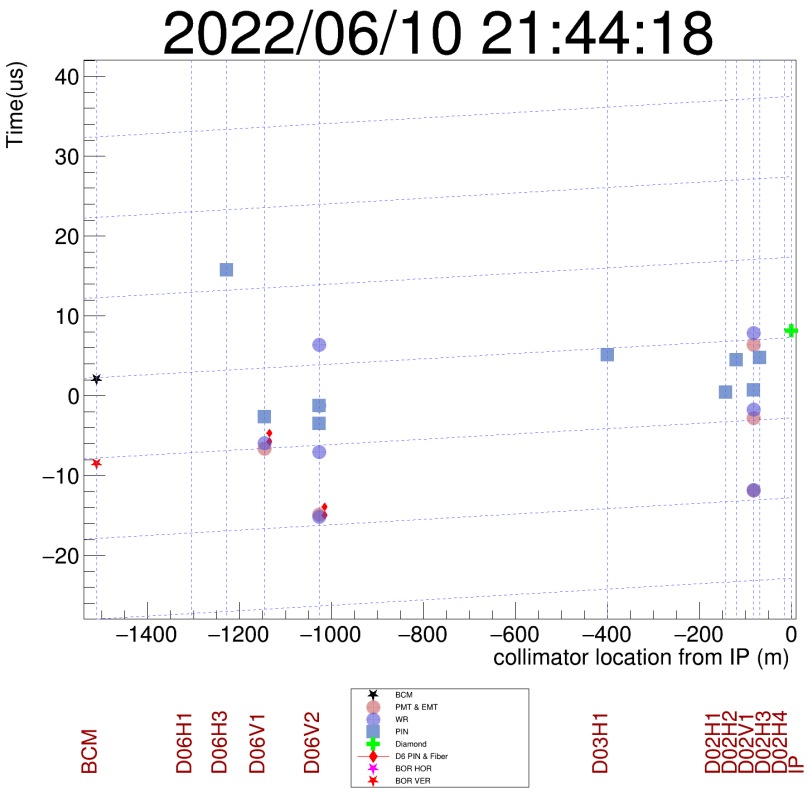


Figure 4.6: Measured beam loss timing of the all sensors with the sudden beam loss at 10th June. The X axis shows the distance from different detectors to the interaction region, and each vertical dashed line represents the position of a particular detector. Y-axis show the ΔT. The upward dashed line represents the time position relationship inferred from the first bunch with current loss observed at BCM. Every point with corresponding to one timing point of various sensors.

5

Analysis result for sudden beam loss

We have analyzed all beam loss events in LER within 2022 physics run period. Sudden beam loss events were selected out from all beam abort events by requiring a visible current loss in BCM and not related with beam injection. Between February 2022 and July 2022, a total of 57 beam loss events were selected out. We performed a comprehensive timing analysis for each of these events using various monitoring systems, including fast loss monitors, PIN diodes, BCM/BOR, IR monitors, and additional PIN diodes/fiber detectors located upstream and downstream of the D06V1 collimators, which were installed during this period. In this chapter, we present a detailed analysis of all the sudden beam loss events that occurred in 2022, followed by a comprehensive discussion of the results.

5.1 overview of all sudden beam loss events in 2022

We begin by assessing the selected SBLs and the resulting occurrence of QCS in relation to the beam current, bunch current, and collimator conditions, as depicted in Fig. 5.1.

We can manually divide the timeline into two periods: before and after May 17th, when significant beam loss led to damage in the D06V1 collimator. Prior to May 17th, it is evident that large SBLs occurred whenever the bunch current reached 0.7 mA, causing QCS quenches and occasional collimator damage. Following the widening of the damaged D06V1 collimator on May 17th, QCS quenches became more frequent even with lower bunch currents. In response, we tightened the D06V2 collimator twice on June 3rd and again on June 14th, aiming to safeguard the D02V1 collimator and prevent further QCS quenches. These observations highlight a pronounced correlation between the condition of the collimators in the D06 section and the occurrence of QCS quenches. As for the overall frequency of SBLs, no significant correlation can be deduced from it and the bunch current.

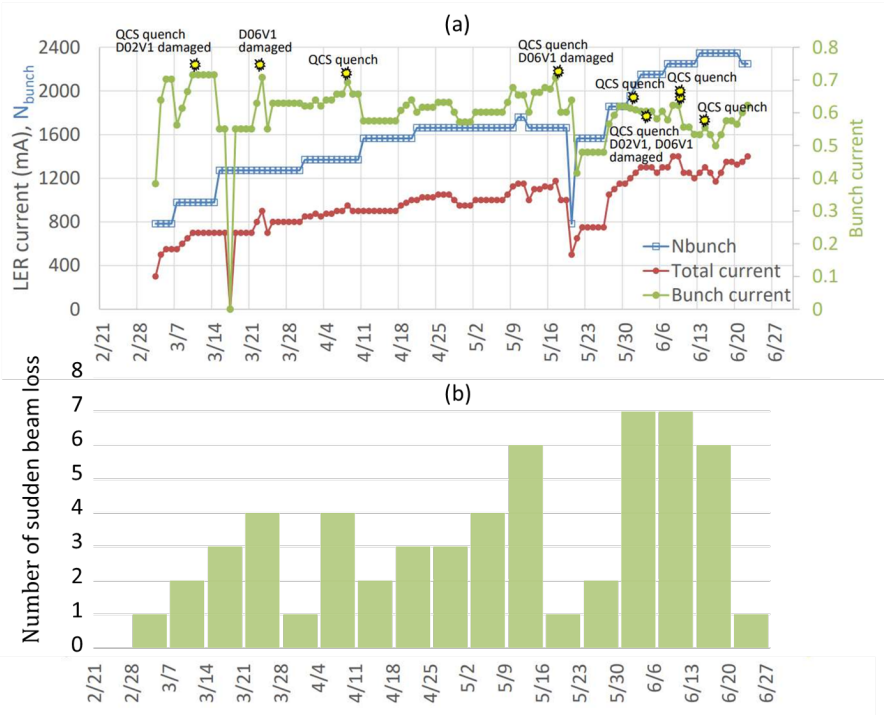


Figure 5.1: (a) QSC quench events comparing with beam current and bunch current [30] (b) Total number of sudden beam losses per weeks

5.2 Timing analysis result for sudden beam loss

The 57 events that occurred between March 2022 and July 2022 were analyzed using the methodology outlined in Chapter 4. To determine which sensor detected the beam loss first in each event, we calculated the ΔT between the sensors, as shown in Fig. 5.2. In this figure, we excluded the timing data from PDs since our analysis revealed that PIN diodes with integral circuits exhibited slower response compared to fast loss monitors. Details of PDs timing comparing with fast loss monitors (FLMs) are showed in Fig. 5.3. If a sensor detected multiple peaks within an event, we selected the fastest ΔT. A detailed summary of the timing results is presented in Tab. A.1 in the appendix. In 56 out of the 57 cases, the fastest ΔT observed between the fast loss monitors and BCM fell within the range of $-10\,\mu s$ to $-40\,\mu s$, indicating that these beam losses occurred within 4 turns. The rest one event at 22nd June was detected at D06V2 WR 7 turns before and continued to see the signal every turn before beam abort. We summarize the fastest sensor location into a histogram as illustrated in Fig. 5.4, the sensors at D06V1 or D06V2 detected the initial beam loss in 43 out of 49 beam loss events that did not result in QCS quenching. On the other hand, 7 out of 8 beam loss events that caused QCS quenching were first detected at D02V1. By combining the information presented in Fig.5.4 and Fig.5.1, we observe a correlation between the condition of the collimators and the initial location of detected SBLs. Following the damage and widening of the D06V1 collimator, it became more frequent for SBLs to be initially detected at D02V1, and in all such instances, a subsequent QCS quenching occurred. Only after tightening the D06V2 collimator did the D02V1 fast loss monitor cease to detect SBLs as the first indication, with the D06V2 monitor assuming that role instead. This suggests that in cases where D02V1 is the first to detect SBLs, the onset of sudden beam loss may also be before the D02V1 section. Despite that, 5 of 57 events we saw the beam position deviation at BOR before our beam loss monitors detected SBLs, this may indicate a beam instability before the beam loss.

We have selected a summary figure illustrating beam loss events with significant radiation doses at the interaction region (IR) (> 300 mrad), as shown in Fig.5.5. From these figures, it is evident that in cases where there was no QSC quench, the fast loss monitors at the D06 collimators detected beam loss events one or two turns before the current loss was detected by the BCM. Moreover, in 5 out of 8 events, the fast loss

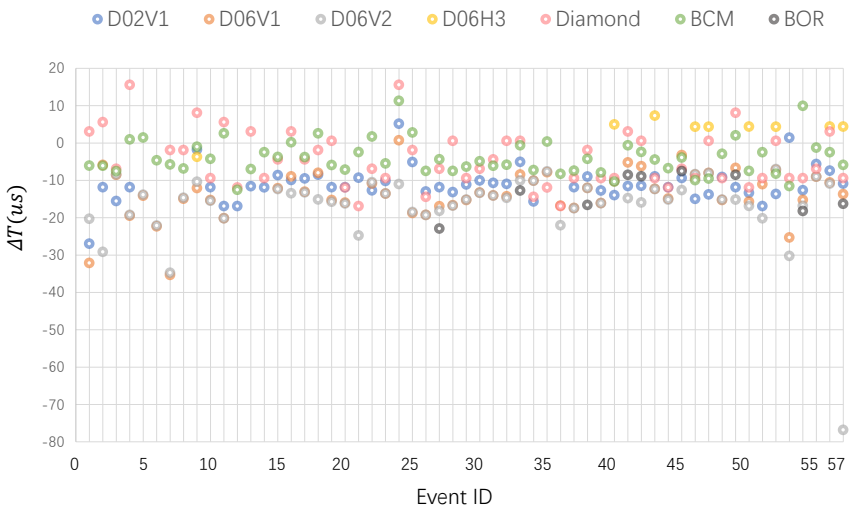


Figure 5.2: ΔT for fast loss monitors, IP sensors and BCM/BOR for all beam loss events in LER.

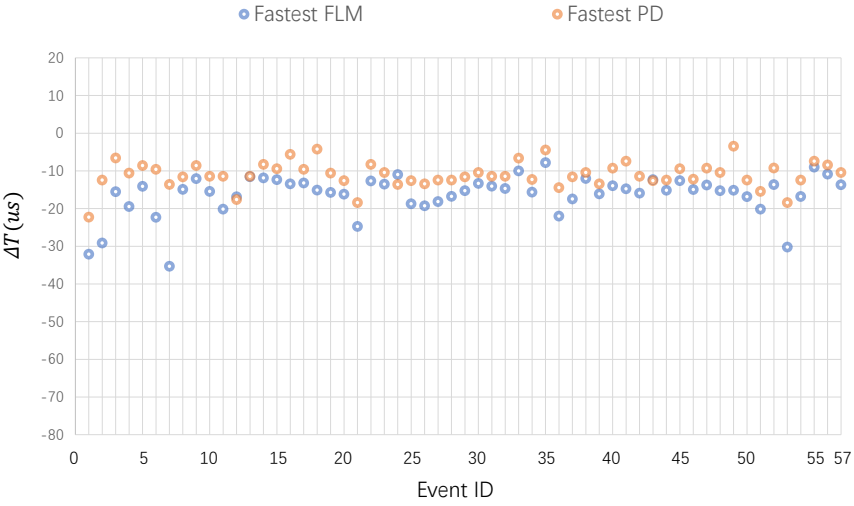


Figure 5.3: ΔT for fastest fast loss monitors and fastest PDs in LER. Only 1 events PDs had similar timing as FLMs, others PDs were slower than FLM.

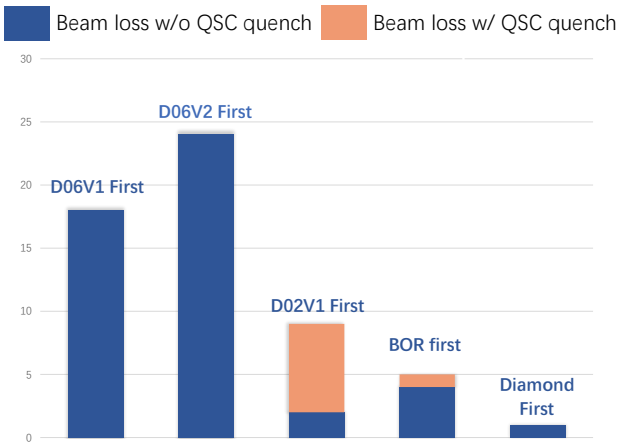


Figure 5.4: Categorization for all beam loss events

monitors detected losses in nearly the same bunches that experienced beam current loss. In the remaining three events, the BCM detected a slight loss in bunch current beforehand, while the fast loss monitor signals corresponded to subsequent bunches with significant current loss.

Regarding QCS quench events presented in Fig.5.6, the fast loss monitor at the D02V1 collimator primarily detected the initial beam loss. In 5 out of 8 events, the fast loss monitor detected losses in almost the same bunches as the BCM. In two of the remaining three events, we observed a single bunch current loss beforehand, and in the remaining event, one of the BCMs detected a small bunch current loss prior to the quench. Notably, in all these events, the bunches where the fast loss monitors first detected beam loss were consistent with the bunches that exhibited a large current loss detected by the BCM.

SBLs are more likely to be first detected in the D06 section first. To investigate this further, PIN diodes without integral circuits and fiber detectors were installed upstream and downstream of the D06V1 and D06V2 collimators starting from June 8th. The cable lengths for these newly installed sensors were assumed to be the same as those for the D06V2 and D06V1 fast loss monitors, which may introduce a deviation of a few microseconds. A comparison between these upstream/downstream monitors and our fast loss monitors is presented in Tab. 5.1. After June 8th, in 10 beam loss events, the sensors downstream of D06V1 (D06V2) detected the beam loss at the same turn in 7 (10) of these events, while the sensors upstream of D06V1 and D06V2 did not detect

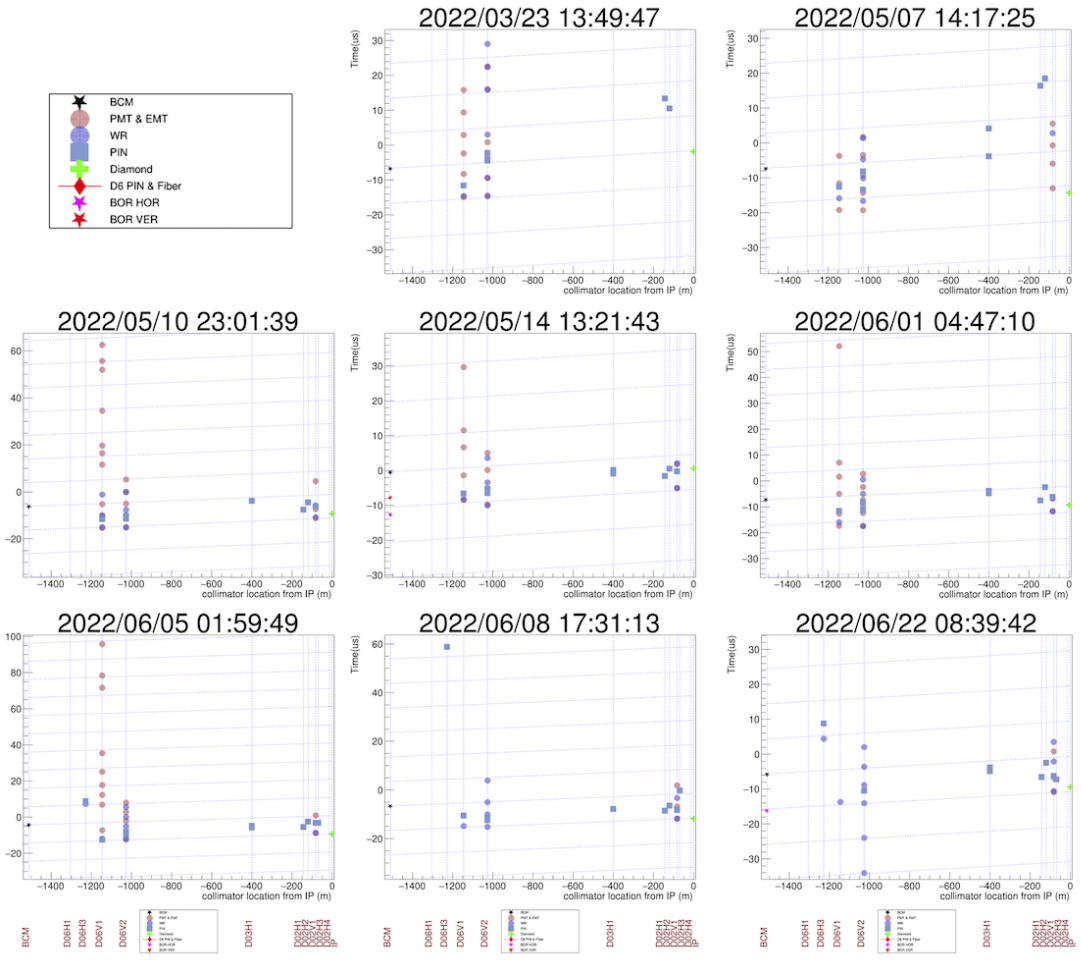


Figure 5.5: Measured beam loss timing of the all sensors with the large diamond does of > 300 mrad or BCM loss >15%.

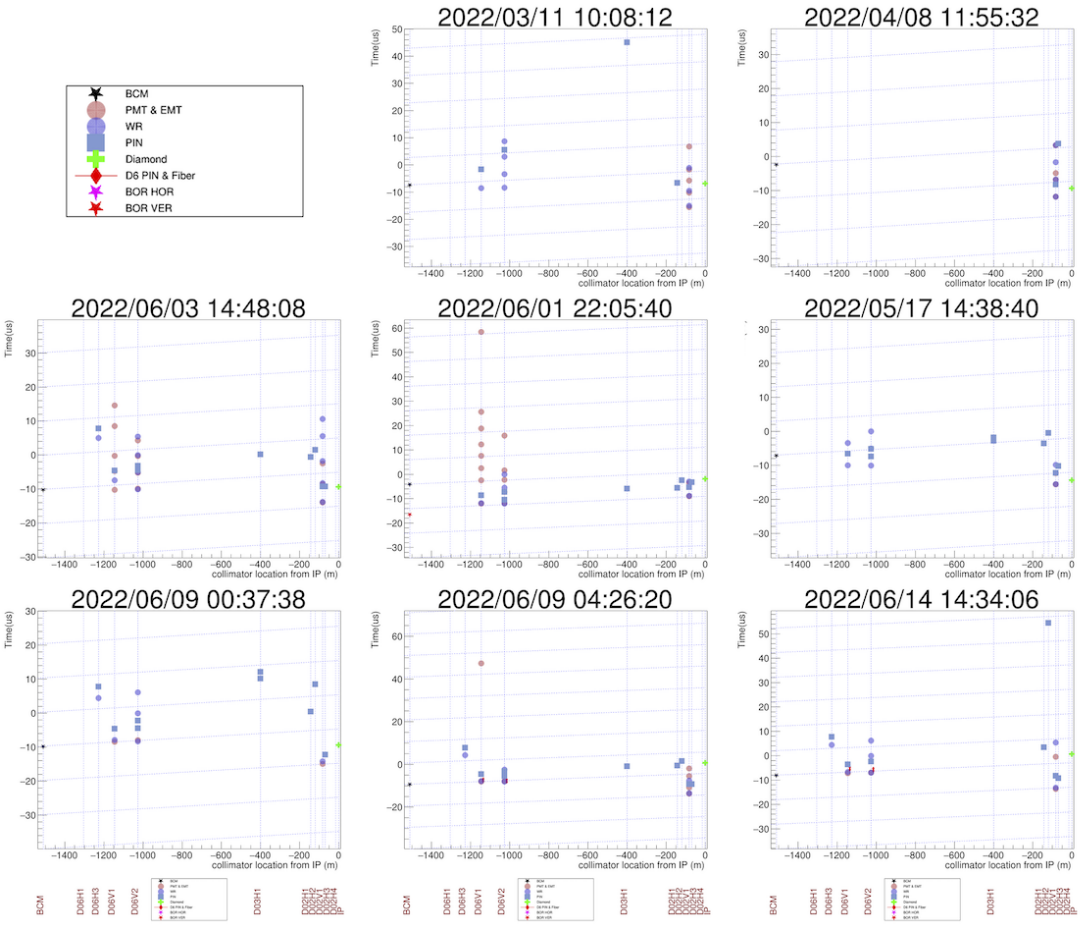


Figure 5.6: Measured beam loss timing of the all sensors with the sudden beam loss which causes QCS quench.

any signal at all in these 10 events.

Events Date	Upstream	D06V1	Downstream	Upstream	D06V2	Downstream
2022/6/9 4:26	NAN	-8.0	-7	NAN	-7.8	-8
2022/6/10 15:15	NAN	-15.3	-10	NAN	-14.9	-13
2022/6/10 21:44	NAN	-6.7	-5	NAN	-14.9	-14
2022/6/13 8:47	NAN	-15.7	NAN	NAN	-16.4	-15
2022/6/14 12:44	NAN	-11.0	-8	NAN	-19.9	-17
2022/6/14 14:34	NAN	-7.2	-5	NAN	-6.8	-6
2022/6/16 2:01	NAN	-25.3	-4	NAN	-25.4	-22
2022/6/16 22:18	NAN	-15.2	2	NAN	-16.5	-15
2022/6/18 20:32	NAN	-9.0	-9	NAN	-8.8	-9
2022/6/20 11:28	NAN	-10.5	-10	NAN	-10.78	-10

Table 5.1: $\Delta T(\mu s)$ for D06V1 and D06V2 upstream and downstream timing comparing with fast loss monitor for beam loss events after installing

5.3 Hypothesis for sudden beam loss

Based on the aforementioned observations, we have explored various potential causes for SBL events. Our analysis revealed that the SBLs occur rapidly, without any preceding small or gradual increase in beam losses, which is characteristic of conventional beam instabilities. Furthermore, the BOR data did not exhibit significant dipole oscillations prior to the SBLs, as would typically be observed in parallel with conventional beam instabilities. These findings strongly suggest that the SBLs are not attributable to conventional beam instabilities and SBLs may be located between BCM/BOR and D6. And the SBLs can occur at different bunch current and beam current without certain threshold, while the conventional cause of beam loss including transverse and longitudinal diffusion, intra-beam scattering and Touschek scattering have a well-defined threshold of bunch current or total beam current. And all these SBLs should not be related with injection events since we reject events which related with beam injection in events selection.

Considering the influence of the dust effect, [31] conducted simulations assuming the SBLs to be extreme cases of vacuum burst (dust) events. Their simulations assumed

aluminum dust particles with a radius of 500 μm, and the scattered particles were expected to impact the D06V2 collimators. One of their simulation results is shown in Fig. 5.7, indicating that vacuum burst events of this nature should be initially detected by the horizontal collimators D06H1 and D06H3. However, in our observations, it is consistently the vertical collimators D06V1, D06V2, or D02V1 that detect the SBLs first, contrary to the simulation results.

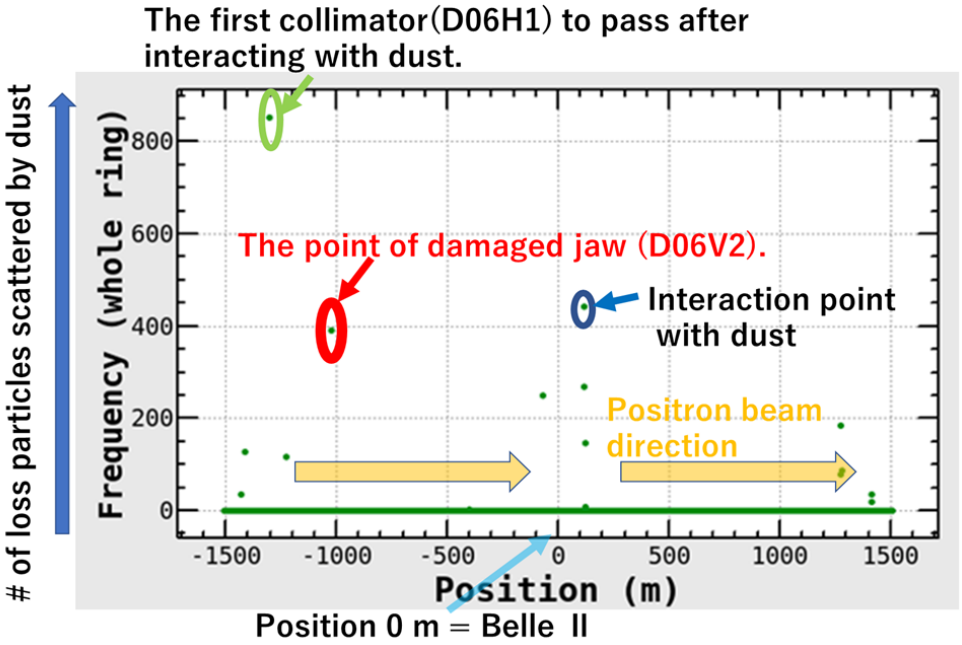


Figure 5.7: Result of tracking the scattered beam that interacted with dust using PHITS and SAD.[31]

And [32] proposed a ” fireball” hypothesis. It illustrated the physical process as:

1. Micro particles in a vacuum with high sublimation points such as carbon and molybdenum are heated by a strong RF field, turning into a fireball with a temperature reaching 1000°C or higher.

2. The fireball lands on a metal surface with a low sublimation point;

3. The plasma is generated around the landing point due to the rapid and substantial temperature increase;

4. The plasma evolves with timely eating the RF-field energy, growing up into a macroscopic vacuum arc.

5. The vacuum arc interacts with the circulating beam and causes beam losses.

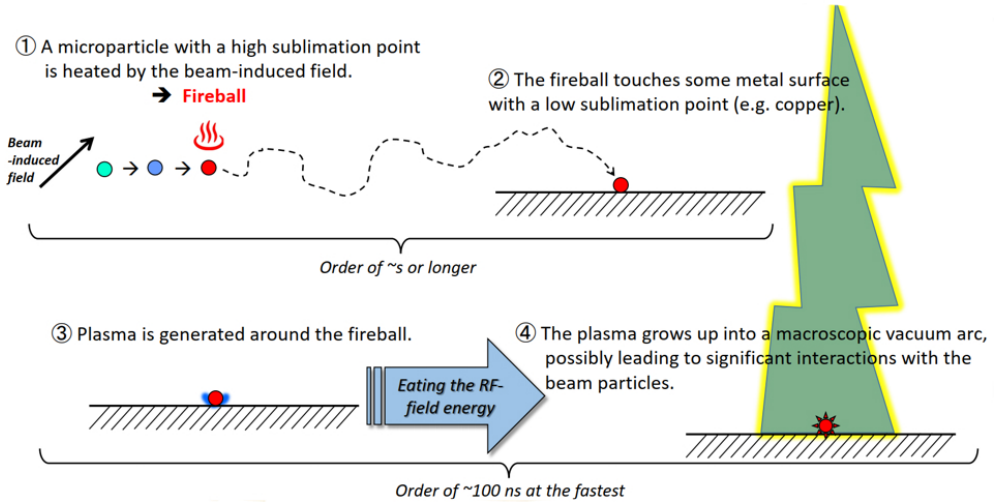


Figure 5.8: Physical process proposed by ” fireball” hypothesis. Micro particles in a vacuum with high sublimation points can be heated by a strong RF field, turning into a fireball, landing on a metal surface with a low sublimation point and generating plasma. Plasma growing up with RF-field energy and interacted with circulating beam, inducing beam loss.

The collimators within the system consist of two distinct materials with significantly different sublimation points. The collimator head is composed of tantalum, tungsten, or carbon, while the vacuum chamber is made of copper. Consequently, in principle, fireball breakdowns can potentially manifest in the proximity of the collimators. To address this possibility, the use of acoustic emission detection has been proposed, leading to the installation of acoustic sensors at the D06 section at the end of the 2022 physics data collection period. As of now, there is a lack of direct evidence supporting the fireball hypothesis, and ongoing investigations are being conducted to shed further light on this matter.

In conclusion, the cause of sudden beam loss events is not fully understood yet. Most of collective effects seem not to be the cause of sudden beam loss events. Beam-dust interaction cannot explain the vertical beam loss in sudden beam loss events. The

fireball hypothesis looks interesting, but there is still no observable evidence that an electric discharge is occurring around the collimator.

5.4 Further investigation and Countermeasure for sudden beam loss

Based on this study, to further investigate the reason of SBLs and protect detectors from SBLs, we are working on following issues:

1. Implement the additional fast loss monitors at the LER/HER injection points and the extra collimators. This expansion in monitoring capabilities allows us to delineate a more precise and stringent region for identifying potential causes of sudden beam loss.

2. Send abort requests using laser transmission through air (instead of optical fiber). As showed in Fig. 5.9. This can speed up the transmission time of beam abort by 30 percent in theoretical to mitigate damage by SBLs. [33]

3. Add additional BOR at D06 section and additional scintillator detector at new installed D05V1 collimator. This can provide more comprehensive monitors for beam profiles. And later one can also be used to directly send abort signal and fast the abort process. [34]

4. All collimator heads will coat with copper. One possible cause of the "fireball" is the particle from damaged collimator heads. This measure is aimed to prevent such issue.

5. Utilize the fast loss monitors for collimator tuning in order to prevent QCS quench and Belle II detector damage. At the end of the 2022 physics run, we have adjusted the collimator based on the fast loss monitor finds, which finally reduce the occurrence of QSC quenches. We prefer to continue to refer to fast loss monitors and adjust collimators.

6. Exam the fireball hypnosis with the installed acoustic sensors. [32]

The location of all sensors are showed in Fig. 5.10. Based on our study, these implemented measures aim to enhance our comprehension of SBLs and mitigate their associated damages.

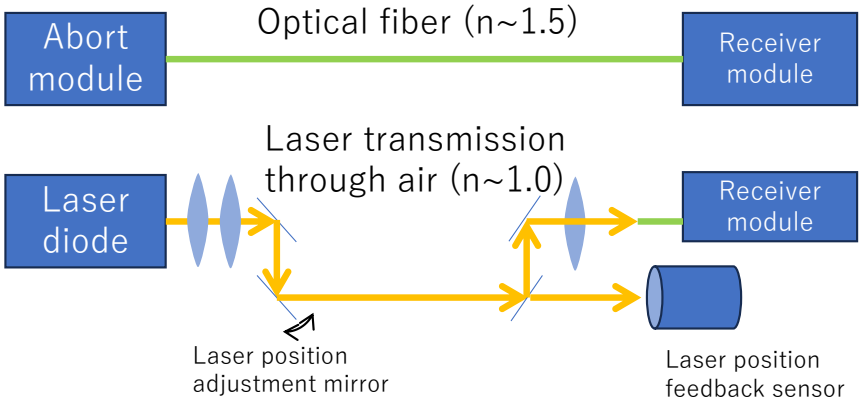


Figure 5.9: Concept diagram of sending abort requests using laser transmission[33]. The underdeveloped Laser position adjustment mirror Laser position feedback sensor are used to stabilize the laser orbit after long distance transmission.

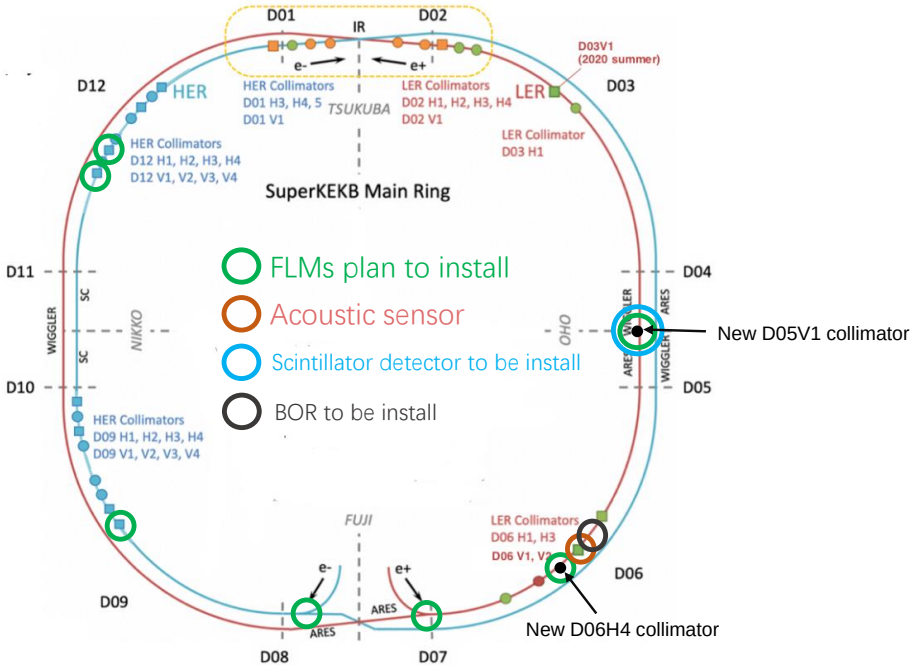


Figure 5.10: Location for monitors that planned to install. Extra fast loss monitors will be installed at LER/HER injection points and D06H4, D05V1, D09V1, D12V1 and D12V2 collimators. BOR will be installed at D06 section. Acoustic sensors are already installed at D06V1. The scintillator detector which can trigger beam abort will be installed at D05V1. Besides, a new beam abort line will be installed at D06 section and proposed to utilize the laser transmission for abort request to CCR.

6 level 1 CDC trigger system

This chapter presents the current status of Level 1 trigger rate and an overview of the CDC trigger system, which is a real-time trigger system specifically designed to identify charged tracks detected by the CDC. The workflow and challenges encountered in the current implementation of the level 1 CDC trigger system are discussed in detail in this chapter.

6.1 Level-1 trigger rate and limitation

The Level-1 trigger rate is required to be less than 30 kHz, as a limitation from Belle II DAQ and sub detector frontend. However, in June 2022, SuperKEKB reaching a luminosity of $4.7 \times 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$, which is about 13 times smaller than the target luminosity of $6.0 \times 10^{34}, \mathrm{cm}^{-2}\mathrm{s}^{-1}$, when the level 1 trigger rate exceeded 10 kHz. It becomes crucial to reduce the level 1 trigger rate in order to achieve higher luminosity.

The level 1 trigger rate is influenced by various trigger conditions (trigger bit). Once any of these trigger conditions are met, the corresponding data will be sent to the

HLT for further processing. The typical trigger condition can be categorized into two pars: $B\bar{B}$ trigger and low-multi trigger, where the later one corresponding to the events with a low particle multiplicity including $e^-e^+ \to \tau^+\tau^-$, $e^-e^+ \to \mu^+\mu^-$, and possible dark sector. The ECL trigger (ECLTRG) and CDC trigger (CDCTRG) are the two primary components of $B\bar{B}$ trigger and low-multi trigger. Table 6.1 provides an overview of the main $B\bar{B}$ trigger bits and low-multi CDCTRG bits with their associated trigger conditions. A Bhabha veto is applied for some trigger bits to reject Bhabha scattering $e^-e^+ \to e^-e^+$ and detail condition in the same table.

Trigger bit	Condition
CDCTRG $B\bar{B}$ bits	(# CDC 2D track >=3 AND #CDC 3D track >=1) OR (#CDC 2D track >=2 AND #CDC 3D track >=1 AND ϕ between 2 2D track $>\pi/2$ And Bhabha veto)
ECLTRG $B\bar{B}$ bits	(#ECL cluster 4 AND Bhabha veto) OR (ECL energy Sum 1 GeV AND Bhabha veto)
CDC low-multi bits*	#CDC 2D track >=1 AND #CDC 3D track >=1 AND p >0.7GeV
BhaBha veto	$165^\circ < \sum \theta_{\mathrm{CM}} < 190^\circ$ AND $160^\circ < \Delta\phi_{\mathrm{CM}} < 200^\circ$ $E^0_{\mathrm{CM}} > 3$ GeV AND $E^1_{\mathrm{CM}} > 3$ GeV AND ($E^0_{\mathrm{CM}} > 4.5$ GeV OR $E^1_{\mathrm{CM}} > 4.5$ GeV)

*Here we only consider single track trigger case, which dominate the CDC low-multi bits

Table 6.1: Typical trigger bits and their corresponding condition. $\sum \theta_{\mathrm{CM}}$ is the sum of polar angles for two ECL clusters and $\Delta\phi_{\mathrm{CM}}$ is the difference of azimuthal angles for two ECL clusters. And $E^{0,1}_{\mathrm{CM}}$ are the deposit energy of two ECL clusters. Injection veto is applied for all bits.

The trigger rates for these bits in a physics run conducted at a luminosity of $4.5 \times 10^{34}, \mathrm{cm}^{-2}\mathrm{s}^{-1}$ are presented in Table 6.3, in which a total level-1 trigger rate is 11.50 kHz. It is evident that the CDC trigger-related bits contribute significantly to the overall level 1 trigger rate. In order to mitigate this issue, we have categorized the components of CDCTRG bits within the same events, as outlined in Table 6.4. In this table, we categorize events into three distinct types based on the presence of charged tracks: signal events, off IP background events, and fake track background events, which are defined in Table 6.2. Off-IP background events primarily originate from beam background outside the interaction point (IP). On the other hand, fake track events can arise from incorrect track reconstructions with electronics noise, CDC

Signal Events	#offline Tracks from IP($\lvert z_0^{\text{offline}} \rvert$ <= 1cm)>=1
Fake Track Events	#offline Tracks == 0
Off-IP Events	#offline Tracks ≠ 0 AND #offline Tracks from IP == 0

Table 6.2: Definition of signal events Off-IP background events, and fake track background events

cross-talk or beam-induced background. It is important to note that both fake track and off IP background events contribute significantly to the overall background trigger rate. Our primary objective is to optimize the level 1 CDC trigger by improve the location resolution and reject more "Off IP Background events" while maintaining the same level of efficiency. And if possible, with better z_0 resolution, we can restrict the z_0^{NN} selection condition and reject more fake track background events.

Trigger bit	CDCTRG $B\bar{B}$ bits	ECLTRG $B\bar{B}$ bits	CDC low-multi bits
Raw rate (kHz)	2.91	2.49	2.93
Exclusive rate (kHz)	2.91	1.80	1.37

Table 6.3: Trigger rate for each bit at luminosity of $4.5 \times 10^{34} cm^{-2} s^{-1}$. Each of the events may fulfill more than one condition, thus we use "Exclusive rate" to show the rate excluded events which already included in left trigger bits.

Trigger bit	CDCTRG $B\bar{B}$ bits	CDC low-multi bits
Signal Events rate (kHz)	0.76	1.02
Off IP Background rate (kHz)	1.36	1.04
Fake track Background rate (kHz)	0.79	0.87

Table 6.4: Raw trigger rate of different trigger components. Events with at least one offline track from IP are categorized as "Signal event"; events with at least one track, but none from IP are categorized as "Background events off IP" and events with no offline track are categorized as "Fake track events"

6.2 level 1 CDC trigger workflow

level 1 CDC trigger workflow is illustrated in Fig. 6.1. The CDC Front-End Electronics (FE) are responsible for providing the raw CDC hits with the precise drift time in

resolution of 2 ns, which are then collected by the merger boards [35]. Each merger board aggregates information from four FEs. Subsequently, the merger boards transmit this information to a four-module pipeline. The first module, known as the Track Segment Finder (TSF), utilizes the raw CDC hits from the merger boards to build characteristic wire patterns which we called Track Segment in every SLs. This step aims to reduce the data volume transmitted to the subsequent module. The second module, referred to as the 2D Track Finder, constructs tracks in the transverse plane by combining Track Segments from the axial SLs using a Hough transformation. Simultaneously, the Event Time Finder module determines the event time in parallel with the 2D Finder, allowing for the calculation of drift times to obtain precise spatial information from the hits. Finally, the Neurotrigger module integrates the 2D track, Track Segments in stereo SLs, and event timing to estimate the 3D track parameters. This section provides an overview of the pipelined level 1 track trigger modules, which take CDC hits as input and generate low-level tracking information for the GDL.

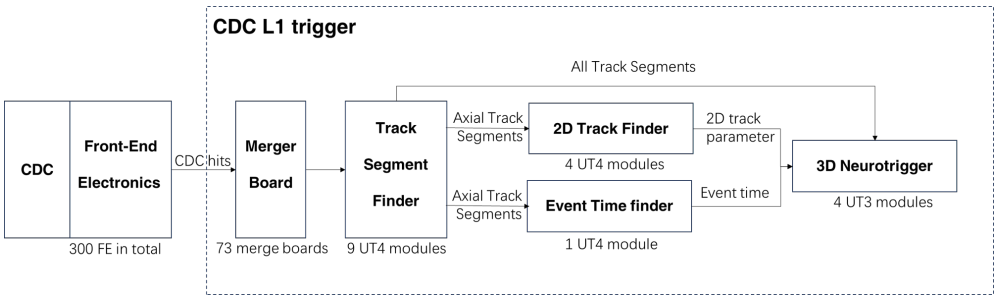


Figure 6.1: The first level track trigger modular pipeline. The implemented hardware modules are showed below every entity.

6.2.1 Track Segments Finder

The TSF serves as the initial module in the level 1 track trigger pipeline. Its primary function is to process the raw CDC hits patterns and timing information obtained from the merger board and generate the Track Segments. The visual representation of the Track Segment's shape can be seen in Fig. 6.2. To construct a Track Segment, a minimum requirement of four out of the five layers must contain hits. The selection of the priority wire follows a predetermined order, starting with the first priority location. If no hit is present, the selection proceeds to the second priority location. And

because of the CDC wire position, as showed in Fig 2.12, a ϕ angle shift exists between every two layers, we can determine the relative positioning of tracks and priority wires by analyzing the hit pattern within each Track Segment, exemplified by Fig. 6.3. Each Track Segment encompasses a pattern that corresponds to the existence of CDC wire hits, TDC of the first/second priority wire with a resolution of 2 ns, left/right directional data encoded with two bits, TDC for the fastest hit in a Track Segment with a resolution of 2 ns, and timing data for other wires with a precision of 32 ns. The Track Segments are used as a basic element for following procedure.

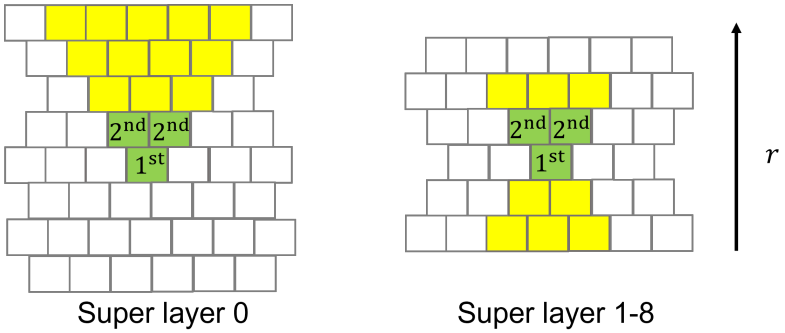


Figure 6.2: The shape of Track Segments. The yellow part is wires in the Track Segment and the green part is the first/second prior wires of the Track Segment. Left for the innermost SLs, where use 15 wires from outer 5 of 8 layers to form Track Segments. Right for all the outer SLs, where using 11 wires from inner 5 of 6 layers to form Track Segments.

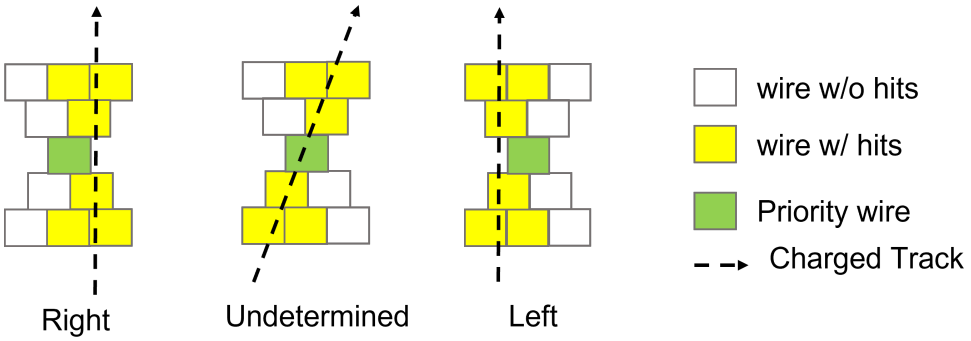


Figure 6.3: The example of right, undetermined and left state of Track Segments. Charged tracks are assuming from Track Segments pattern.

6.2.2 2-dimensional Track Finder

After the TSF module, the axial SLs provide Track Segments which are then combined to form circular tracks in the transverse plane using a technique called Hough transformation. The Hough transformation is a widely used method in image processing and pattern recognition, originally developed for detecting lines in an image. However, it can also be extended to detect other shapes such as circles and curves.

In 2D track finder, the Hough transformation is applied to identify circular tracks from the Track Segments. The transformation works by mapping points in the geometric space to a parameter space with the equation:

$$\rho(\phi) = \frac{2}{r_{\mathrm{TS}}} \sin(\phi - \phi_{\mathrm{TS}}) \tag{6.1}$$

After the Track Segment Finder (TSF) module, the axial Super Layers (SLs) provide Track Segments which are then combined to form circular tracks in the transverse plane using a technique called Hough transformation. The Hough transformation is expressed mathematically as:

$$\rho(\phi) = \frac{2}{r_{\mathrm{TS}}} \sin(\phi - \phi_{\mathrm{TS}}) \tag{6.2}$$

In this equation, $(r_{\mathrm{TS}}, \phi_{\mathrm{TS}})$ represents the polar coordinates corresponding to the priority wires of the Track Segments in the geometric space, while (ρ, ϕ) denote the polar coordinates within the Hough parameter space. By applying this transformation, a circle in the geometric space is mapped to two points in the Hough parameter space. Similarly, a point in the geometric space appears as a sine curve in the Hough parameter space. This can be visualized in Fig. 6.4, which illustrates the Hough transformation of a circular track. The crossing points in the parameter space correspond to positive and negative curvatures, indicating clockwise and counterclockwise tracks, respectively. Each point is associated with a sine curve of the same color in the parameter space.

The process of identifying a specific circular track then converts into locating a cross point within the Hough parameter space. This is achieved by implementing a grid separation on the Hough parameter space. Each curve present in the parameter space contributes a count to the corresponding grid cell, and curves originating from each SL are counted only once per cell. Subsequently, cells exceeding a specified count

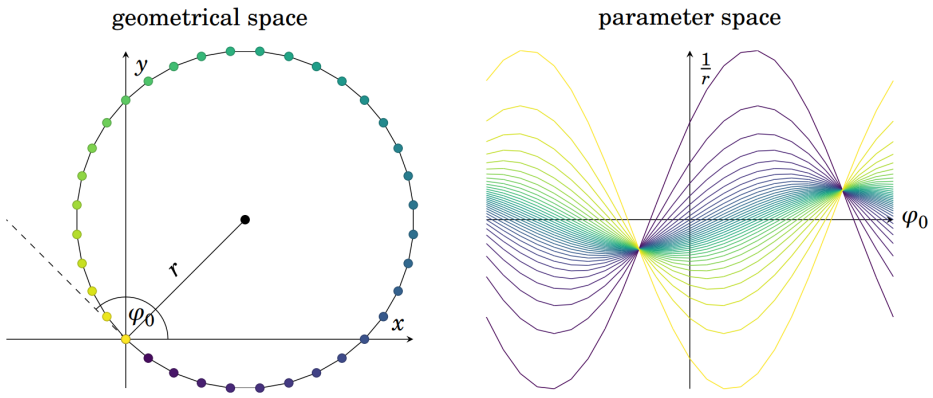


Figure 6.4: Hough transformation of a circular track. There are two crossing points, one for positive and one for negative curvature, where positive for clockwise track and negative for counterclockwise. Each point is corresponding to the Sine curve with same color in parameter space [18].

threshold are selected as the cross point which corresponding to 2D tracks in the geometric space (Fig. 6.5 provides an illustrative example). The threshold value and the number of cells are adjusted to optimize performance. The recent investigation [36] has further developed an algorithm that utilizes all wires in the Track Segments, as opposed to solely the priority wires, aiming to enhance efficiency and mitigate background track rates. The 2D track finder module provides 2D tracks for subsequent 3D fitting procedures, while the transverse momentum (p_T) and charge information of the tracks can be derived from the 2D track finder.

6.2.3 Event Timing Finder

The Event Timing Finder (ETF), working parallel with 2D track finder, provides the start time T_0 of an event. The T_0 is necessary for calculated drift time in CDC: $T_{drift} = T_{wire} - T_0$. The main idea of ETF is to find out the fastest signal timing in all Track Segment. But the problem is that Track Segments contaminates a large amount of background which is not derived from events such as beam background and crosstalk, whose hit timing is independent of event timing (see Fig. 6.6). So Current ETF in [37] first apply a 2D track finder inside, to only select track segments related with a 2D track. Then in order to further suppress the influence of the residual background

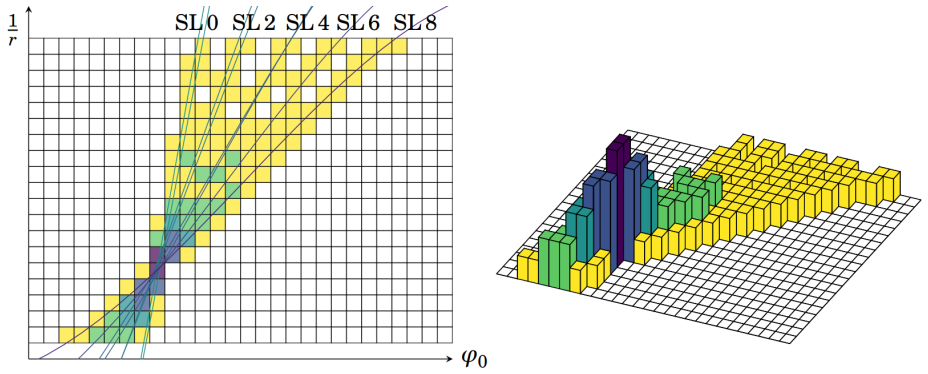


Figure 6.5: Left: Constructed curve and grid in the parameter space. Right: Histogram for each grid cells counts [18].

hits, the ETF calculate a median value from the fastest timing of all the fastest timing from Track Segments. ETF provide the T_0 to follow 3D fitting.

6.2.4 3-dimensional track reconstruction modules

3D track reconstruction module get all the information include track segments, 2D track and event time to reconstruct the 3D track and fit the z_0 and θ of each track.

For the 3D reconstruction, the track of a charged particle in a uniform magnetic field follows a helical shape, aligned with the axis of the magnetic field, as expressed by

$$\begin{pmatrix} x(\mu) \\ y(\mu) \\ z(\mu) \end{pmatrix} = \begin{pmatrix} r \cdot (\sin(\mu/r - \phi_0) + \sin\phi_0 + x_0) \\ r \cdot (\cos(\mu/r - \phi_0) - \cos\phi_0 + y_0) \\ \cot\theta_0 \cdot \mu + z_0 \end{pmatrix} \tag{6.3}$$

where $\mu \equiv 2\alpha r$ is the arc length of the transverse track projection from the reference point to a general point on the helix and α is the crossing angle of tracks with each SL. (x_0, y_0, z_0) is the reference point on helix called pivot, ϕ_0 and $\theta_0$are the azimuth and polar angle of the momentum at the reference point. This helix is showed in Fig. 6.7.

From 2D track reconstruction, the ϕ_0 and r can be determined. It is worth noting that with the Hough transformation employed in the 2D finder, only tracks originating from the origin ($(x_0, y_0) = (0, 0)$) are selected. And the *alpha* can also be calculated with 2D tracks and each SL parameters. The only left unknown parameters are the θ

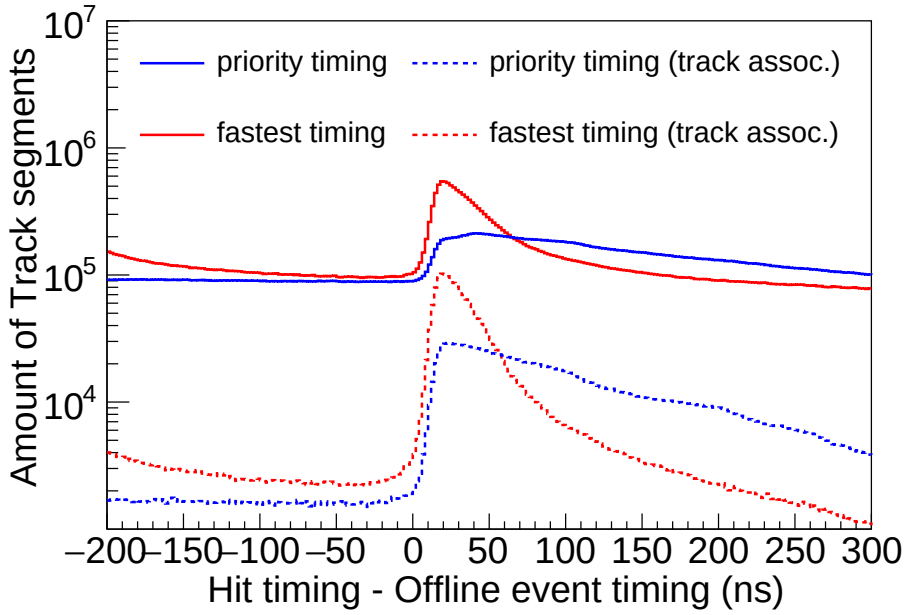


Figure 6.6: The hit timing distribution relative to offline reconstructed event timing. The red lines are priority timing and the blue lines are the fastest timing. The solid and dashed lines show before and after background reduction by association with 2D track [37]

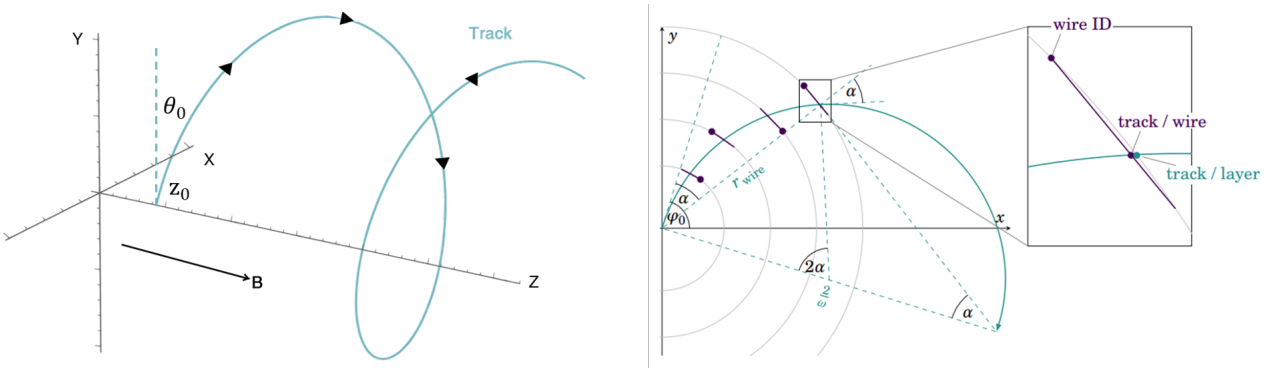


Figure 6.7: Left:The track in helix shape. Right: Track projection on x-y plane and the related stereo wires [18]

and z_0, both of which are related to the stereo wires which are not parallel to z-axis and need 3D reconstruction.

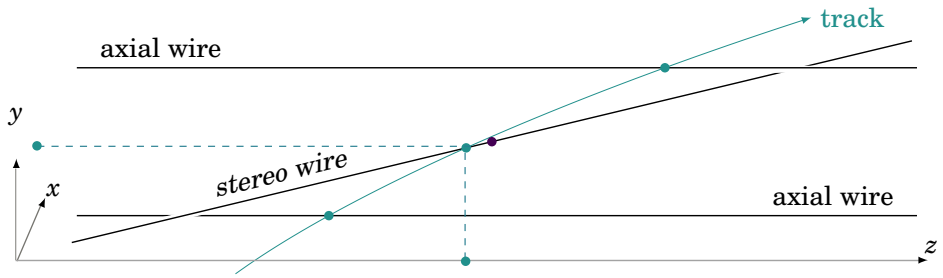


Figure 6.8: The track and CDC wires in y-z plane. The hit points on the stereo wires can be estimated from the ϕ_{cross} and r_{cross} on x-y plane.[18]

As showed in Fig 6.8, assuming charged particle directly cross the stereo wires, with determined stereo wires crossing point $(r_{cross}, \phi_{cross})$, we could calculate the z_{cross}. Consider stereo wires' radius r_{cross} are known, only ϕ of crossing point is enough for determination. And it follows this equation:

$$\frac{z_{cross} - z_B}{z_F - z_B} = \frac{\phi_{cross} - \phi_B}{\phi_F - \phi_B} \tag{6.4}$$

where the index F/B denotes the forward/backward endplate and z_B, z_F, ϕ_B, ϕ_F are constant specific to each stereo wire.

Taking into account the drift time, which represents the distance between the track and the sensing wire (Fig. 6.9), we can calculate the hit position by incorporating the drift time as follows:

$$\phi_{hit} = \phi_{cross} \pm \arcsin\left(\frac{v_{drift} \cdot t_{drift}}{r_{wire}}\right) \sim \phi_{cross} \pm \frac{v_{drift} \cdot t_{drift}}{r_{wire}} \tag{6.5}$$

t_{drift} and v_{drift} are drift time and drift speed. We approximately take the $r_{hit} == r_{wire}$ and $v_{drift} \cdot t_{drift} << r_{wire}$.

By utilizing more than two crossing points from different stereo wires, we can perform a fit to determine z_0 and θ_0 in Equation 6.3. Furthermore, from Fig. 6.7, it is

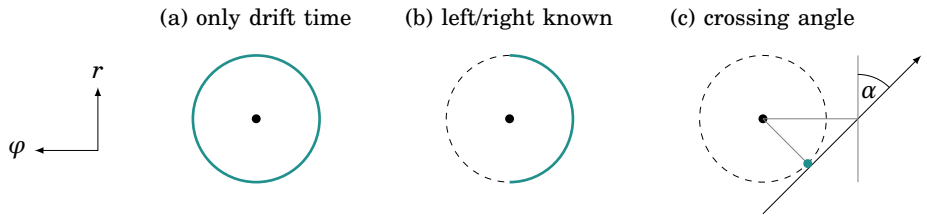


Figure 6.9: Determined position of track hit point (in cyan line) related with stereo wires. Left: only know drift time. Middle: know drift time and left/right state. Right: know drift time, left/right state and crossing angle [18]

obviously that ϕ_{cross} can be calculated using the 2D parameters:

$$\phi_{cross} = \phi_0 - \alpha \tag{6.6}$$

The 3D reconstruction process involves two parallel modules: the 3D neural network and the 3D fitter. Our work primarily focuses on the 3D neural network approach. After 3D reconstruction, the z_0 and θ_0 will be provided to GRL and GDL.

6.2.5 Software simulation

As previously mentioned, the level 1 trigger system is implemented on dedicated hardware. However, in order to facilitate investigations into the trigger performance, software simulations of the trigger have also been incorporated within the Belle II analysis software framework, known as basf2 [38]. The present study is based on the software simulation of the trigger.

6.3 3D Neural-Network trigger

6.3.1 Current Hardware implemented

For the 3D track reconstruction in the level 1 trigger, a neural-network (NN) methodology called NeuroTrigger was employed. As indicated in section 6.1.4, the variables required for fitting z_0 include ϕ_{cross}, t_{drift}, and the left/right (L/R) state obtained from the stereo wires, as well as α from the 2D track. The constants such as r_{wire}, z_B, z_F, ϕ_B,

ϕ_F, and v_{drift} are not directly inputted, as the neural network is designed to learn and incorporate them internally. Pytorch lib [39] is used for neural network training.

NeuroTrigger input

In the actual implementation, the variable $\phi_{rel} = \phi_{cross} - \phi_B$ is utilized instead of ϕ_{cross} to facilitate the learning process. Additionally, for t_0, the fastest timing among all priority wires is employed instead of relying on the ETF output. And t_{drift} are combined with L/R state where Left for positive, Right for negative and undecided for zero drift time (assuming track is close to wire). For the selection of input Hits, following criteria are applied:

1. ϕ_{rel} in the range of (ϕ_{min}, ϕ_{max}), which is pre-trained with the offline data to limit the maximum ϕ range between the 2D track and stereo wire.

2. $t_{drift} \in (0, 503\text{ns})$, priority wire with negative drift time are rejected.

Since the architecture of NN is determined, and we may have multi Track Segment in each SL, only one per every SLs are selected out. First it finds the Track Segment with L/R state decided. If found, then pick up fastest one with smallest t_{drift}. If no, pick up the fastest one in all Track Segment in this SL. All input are scaled to $(-1, 1)$ for convenient training.

NeuroTrigger architecture

The parameters of the selected wires are input to the NeuroTrigger. Because of possible missing hits in each SL, a combination of 5 different NN are used to handle a different missing SL case (One for all SLs valid, and 4 for each SL missing case). Meanwhile, a 3D track will only build with at least 3 out of 4 stereo SLs have hits. Such single NN are called “expert”. The current architecture of every expert of NeuroTrigger is depicted in Fig. 6.10, which consist of an input layer with 27 nodes, a hidden layer with 81 nodes, and an output layer with 2 nodes. Two output for z_0 and θ_0. All layers are fully connected.

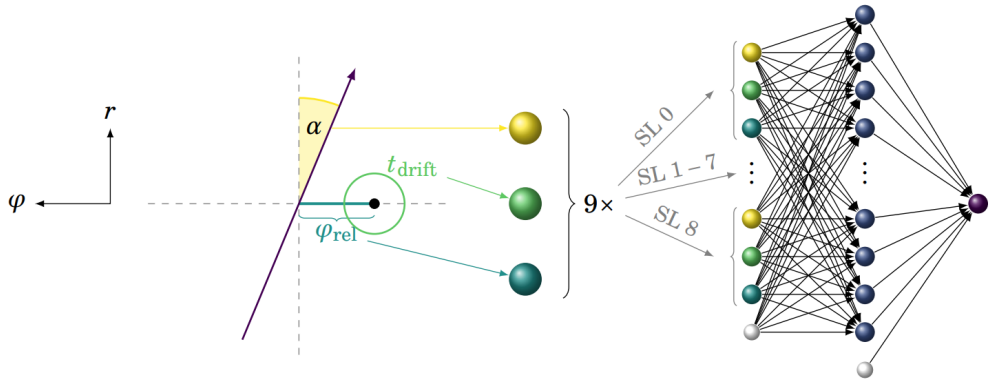


Figure 6.10: Architecture for implemented neural-network. It has one input layer of 27 nodes, consists of ϕ_{rel}, t_{drift} (include L/R) and α per every SL, One hidden layer with 81 nodes and one output layer of two nodes for z_0 and θ [18].

Neural network training

The NN undergoes a training process to adjust its internal weights in order to capture the relationship between the input and output. The training process generally involves a sequence of forward and backward propagation steps. Let's consider a complete propagation process for an input vector with n samples consisting of m_1 input features and corresponding m_2 target features:

$$X^{in} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1m_1} \\ x_{21} & x_{22} & \dots & x_{2m_1} \\ \dots & & & \\ x_{n1} & x_{n2} & \dots & x_{nm_1} \end{pmatrix}, Y^{in} = \begin{pmatrix} y_{11} & y_{12} & \dots & y_{1m_2} \\ y_{21} & y_{22} & \dots & y_{2m_2} \\ \dots & & & \\ y_{n1} & y_{n2} & \dots & y_{nm_2} \end{pmatrix} \tag{6.7}$$

In the forward step, we process each sample of X into NN and propagate to each node in NN as:

$$N_{ik} = A(\sum^{j} w_{jk} \cdot x_{ij} + b_k) \tag{6.8}$$

where i represent sample number, k represent the node number in next layers and j represent the node number in this layer; w_jk and b_{jk} are the elements of weights matrix and bias matrix, respectively, which are stored in the NN and updated

iteratively. A nonlinear activation function $A()$ applied to introduces nonlinearity into the propagation. The forward propagation is performed layer by layer until reaching the output layer. Once the output Y^{out} is obtained, it is compared to the target value to calculate the loss matrix:

$$L_i = \sum^{j} E(y_{ij}^{in}, y_{ij}^{out}) \tag{6.9}$$

L_i is the elements of Loss matrix; $E()$ is the loss function which can be adjusted based on the desired target. In common case we could use mean-square error (MSE) as $E(y_{ij}^{in}, y_{ij}^{out}) = (y_{ij}^{in} - y_{ij}^{out})^2$.

Subsequently, the loss is backpropagated to adjust the weights and bias matrices in each layer:

$$\Delta w_{ijk} = lr \cdot O(\frac{\partial L_i}{\partial w_{jk}}) \tag{6.10}$$

where lr is the learning rate determined the speed they learn from every sample and O is the optimization algorithms which can be adjusted. Notably, the weights w_{jk} are either updated by accumulating and summing the Δw_{ijk} over i from a few samples (referred to as a "batch"), updating them with every sample, or updating them after processing all the samples, based on the chosen training strategy. Once all input samples have been traversed and the w_{jk} have been updated, one epoch of training is completed. Through the forward and backward propagation steps, the NN's output gradually approaches the target. The number of epochs required for convergence depends on the learning rate and can range from a few tens to a few thousands.

It should be noticed that with a multi-layer NN, $\frac{\partial L_i}{\partial w_{jk}}$ is calculated from chain derivation method as $\frac{\partial L_i}{\partial w_{jk}} = \frac{\partial L_i}{\partial f_0}\frac{\partial f_0}{\partial f_1}\cdots\frac{\partial f_n}{\partial w_{jk}}$. Then a problem called vanishing gradient may occur. If $\frac{\partial f_{n-1}}{\partial f_n}$ got a small or zero value, the following gradient will be close to zero and can hardly update. It is important to choose suitable activate function to reduce this effect.

To prevent overfitting to the training samples, a validation sample is essential. By comparing the loss obtained from the validation sample, one can determine if overfitting is occurring and if the results worsen for the validation sample. In such cases, the training process can be appropriately halted.

In the current implementation of the Neurotrigger, the MSE loss function is

employed. $O\left(\frac{\partial L_i}{\partial w_{jk}}\right) = \Delta_{ijk}\frac{\partial L_i}{\partial w_{jk}}$, where Δ_{ijk} depends on the sign of $\frac{\partial L_i}{\partial w_{ijk}}$ and $\frac{\partial L_{i-1}}{\partial w_{(i-1)jk}}$. And the weights and bias matrices are updated after processing every 2024 input samples. Offline reconstructed track from real physics data was utilized for NN training.

6.3.2 Firmware logic

The Universal Trigger board serves as a versatile FPGA board for the trigger system, providing a platform for implementing advanced and high-performance logic. Current trigger logics mainly deployed on the three generation Universal Trigger board (UT3) and part on fourth generation UT (UT4). As part of an upgrade to enhance the capabilities and performance of the L1 trigger system, the hardware transition will be made from three generation UT (UT3) to fourth generation UT (UT4). A detailed comparison between UT3 and UT4 is presented in Table 6.5. The UT4 board offers more than three times the number of logic gates and a communication bandwidth that is twice as large as that of UT3. These improvements enable the utilization of larger Neural-Network architectures and facilitate the transfer of additional inputs, contributing to enhanced functionality and performance.

	UT3	UT4
FPGA	Virtex 6 XC6VHX380/565T	Virtex UltraScale 7 XCVU080/160
Logic gate	382k/580k	975k/2026k
Optical IO bandwidth (total)	530Gbps	1300Gbps
Internal independent RAM	No	DDR432GiB

Table 6.5: UT4 and UT3 comparison

The CDC trigger logics discussed above have been implemented on the UT3 FPGA already. Here we focus on the FPGA implementation of Neurotrigger[40]. The architecture of the Neurotrigger FPGA implementation is illustrated in Figure 6.11. This implementation comprises three stages. The first stage is responsible for input handling, where data from the ETF, TSF, and 2D Track Finder within the CDC trigger system is received and processed. The second stage, known as preprocessing, incorporates various processing modules that perform different tasks. These tasks include hit selection to identify stereo and axial Track Segments, calculation of α and ϕ_{rel}, and scaling all parameters. These additionally show the data flow related dependencies in

	α	ϕ_{rel}	t_{drift}	Scaled Input	MLP Weights
assigned bits	14	24	8	13	13

Table 6.6: Bandwidth for α,ϕ_{rel} t_{drift} and Scaled Input[40]

processing. Modules without a data dependence are operating in parallel and data is synchronized to compensate for different delays at the respective stages. And in the final stage, processing, the scaled parameters are fed them into the trained Multi-Layer Perceptron (MLP) network. This processing step yields the desired outputs of z_0 and θ_0. The length of assigned bits for every parameter are showed in Tab. 6.6.

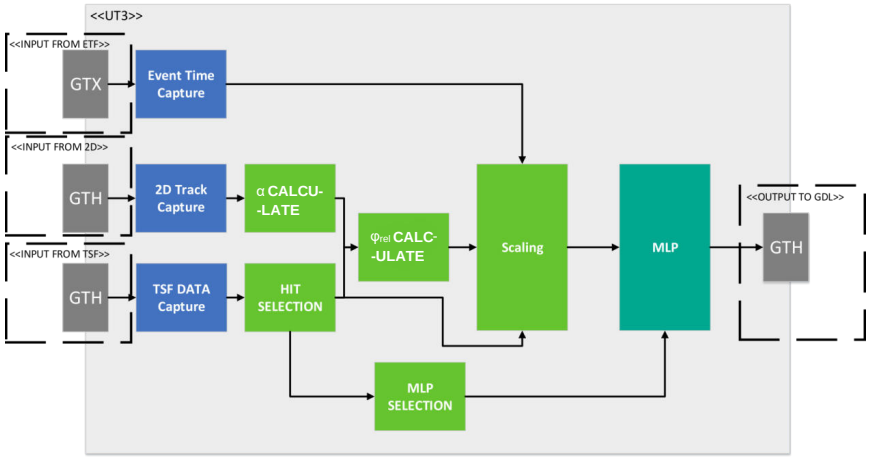


Figure 6.11: The architecture of the FPGA implementation of Neurotrigger [40]. It is divided into three stages. The input handling that receives the data from the ETF, TSF and 2D Track Finder within the CDC trigger system. The preprocessing is represented by the different processing modules used within the design, including hit selection to select stereo and axial Track Segment, α and ϕ_{rel} calculate and input scaling. And in the final stage, processing, the scaled parameters are fed them into the trained Multi-Layer Perceptron (MLP) network. This processing step yields the desired outputs of z_0 and θ_0.

6.3.3 Performance of Neurotrigger

The Neurotrigger produces reconstructed 3D tracks with the variables z_0 and θ_0, which are subsequently processed by the GRL and GDL to make the final decision. To discriminate against events originating from sources other than the IP, a z_0 selection condition as $|z_0| < 15$cmis implemented to select 3D tracks after the Neurotrigger. In this regard, we present the comparison between z_0^{NN}, the output from the NN, and z_0^{offline}, which represents the precise z_0 obtained through offline analysis with data taking during the 2022 physics run. The current Neurotrigger z_0 resolution for signal tracks originating from the IP ($|z_0^{\mathrm{offline}}| < 1, \mathrm{cm}$) [41] is illustrated in Fig.6.12. A double Gaussian fitting is applied. Since we want to keep 95% signal efficiency, we also focus on the σ_{95} which represent the standard deviation of central 95% events and directly related to the selection condition we could set. For current Neurotrigger, $\sigma_{95} = 3.05$. However, the current Neurotrigger still faces a significant challenge. As depicted in Fig.6.13, a considerable number of off-IP events fall within the selection region defined by $|z_0^{\mathrm{NN}}| < 15$cm, which is what we use in actual data taking, resulting in a substantial increase in the level 1 CDC trigger rate. It is of utmost importance to mitigate this "Off IP background" originating from the Neurotrigger. And we list the target for our developed new NN trigger as Tab. 6.7, in which we plan to reduce the σ_{95} of both IP track and Off-IP tracks by 1 cm and reject further 70% background events which were triggered by current Neurotrigger, while keep same efficiency. Considering a same total trigger rate at 11.5 kHz, we aim to reduce the background CDCTRG $B\bar{B}$ raw trigger rate by 1.1 kHz and background CDCTRG low-multi trigger rate by0.9 kHz.

Parameters	Target
z_0 resolution at IP (σ_{95}^{IP}	<2 cm
Trigger efficiency	>95%
Extra background rejection rate	>50%
Reduction for CDCTRG $B\bar{B}$ raw trigger rate*	>1.1 kHz
Reduction for CDCTRG low-multi raw trigger rate*	>0.9 kHz

*Assuming same total trigger rate and ratio as we showed in Table 6.3,6.4

Table 6.7: Target of new developed NN trigger. Extra background rejection rate is the rejection rate for the background events that pass Neurotrigger selection.

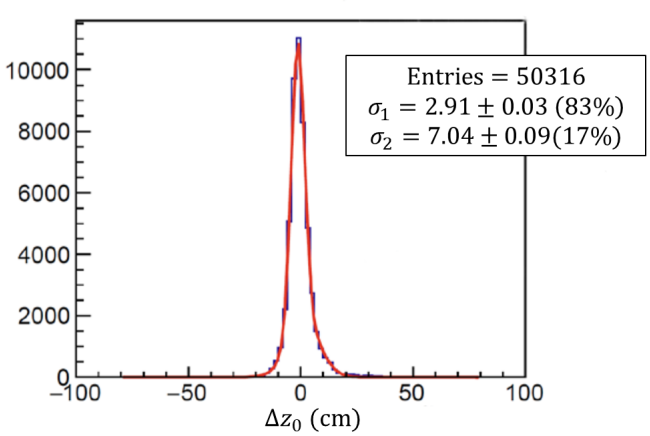


Figure 6.12: $\Delta z_0 \equiv z_0^{\mathrm{NN}} - z_0^{\mathrm{offline}}$ distribution at $|z_0^{\mathrm{offline}}| < 1$ region (IP). Using double Gaussian fit to evaluate the resolution. Data taking from 2022 physics run [41].

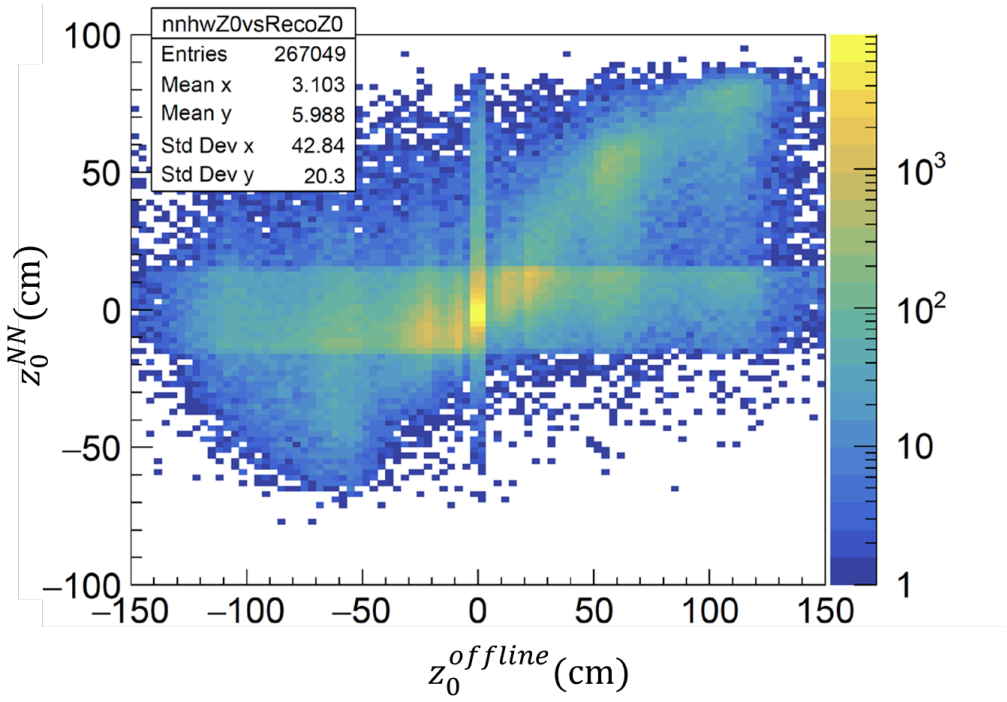


Figure 6.13: z_0^{NN} comparing with z_0^{offline}. Selection condition of Neurotrigger track are set at 15/20 cm for multi-track and single track events. Large amount of events with $|z_0^{\mathrm{offline}}| > 15cm$ drop into the Selection region. Data taking from 2022 physics run. [41].

7 Development of Neural-Network 3D track trigger

This chapter presents the strategy employed to develop the novel Neurotrigger with the primary objective to reduce resolution by 1 cm and to improve the total background rejection rate further by 50%. This chapter encompasses two aspects: additional inputs, and optimization of the Neural-Network architecture.

7.1 Extra input information

This section presents the supplementary data utilized to enhance the performance of the Neurotrigger in conjunction with the UT4 board. The additional information is categorized into three distinct components: Extra wires, ADC information, and ETF input.

7.1.1 Extra wires information

A general approach to enhance the fitting performance involves incorporating additional data points for the fitting process. With the introduction of the UT4 board, it becomes feasible to accurately transfer timing information from extra wires to the Neurotrigger, achieving a precision of 2 ns. As depicted in Fig. 7.1, the current implementation only utilizes the priority wire within each Track Segment. In this, we include hits other than the priority one, in the same Track Segment (extra wires) as extra input feature of the fitting process. One challenge we encounter is the need for deterministic NN inputs, which is crucial for hardware implementation. The number of hits in a Track Segment varies from event to event, posing a hindrance to achieving a fixed number of inputs. Additionally, the presence of potential invalid inputs corresponding to wires with no hits can significantly impair the performance of the existing fully connected NN [18]. To address these challenges, we propose two strategies for incorporating the extra wires: "Selected extra wires input" and "Full wire input with partial connect NN".

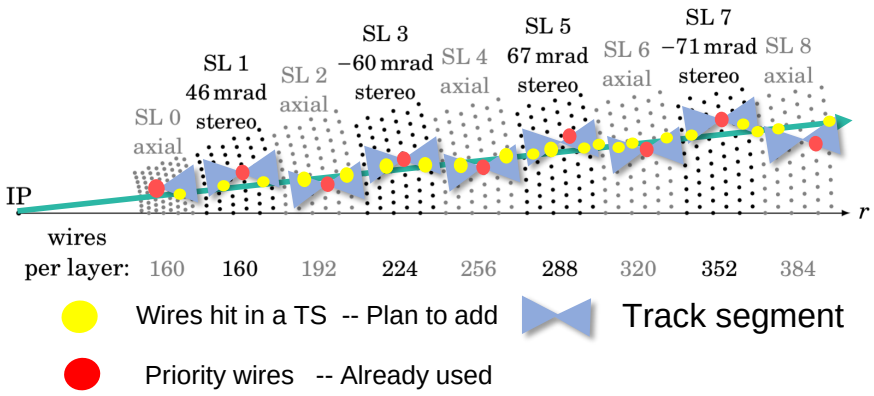


Figure 7.1: The input CDC hits for Neurotrigger, red dots for already used and yellow dots for we plan to added

Selected extra wires input

It is natural to select only the hit wires as input to avoid the issue of missing input features. To ensure a valid input, considering that a Track Segment should have at

least 4 out of 5 layers hit, we can guarantee a minimum of 3 valid extra wires for the Neurotrigger. Since the relative location of each input wire is not determined in relation to the priority wire, it is necessary to include ϕ_{rel} and α as input along with t_{drift} for the extra wires.

The remaining challenge lies in determining the L/R state. The Track Segment can only provide the L/R state for the priority wires. To address this, we construct a full L/R Look-Up Table (LUT) that maps the L/R state for every wire in the Track Segment based on the Track Segment hit pattern. The mapping from hit pattern to L/R state is determined through Monte Carlo (MC) simulated tracks. For each hit in the track segment, the hit pattern and the true L/R state for every wire are determined in the simulation. Subsequently, for each pattern, the number of hits with the true left (right) passage, denoted as n_L (n_R), are counted for each wire. To determine the left/right state for the pattern, the following condition is checked:

$$\text{Left/Right state} = \begin{cases} \text{Left} & if\ n_L > p \cdot (n_L + n_R) + 3\sigma \\ \text{Right} & if\ n_R > p \cdot (n_L + n_R) + 3\sigma \\ \text{Undecided} & \text{otherwise} \end{cases} \tag{7.1}$$

Here, $\sigma = \sqrt{(n_L + n_R) \cdot p \cdot (1-p)}$ represents the width of a binomial distribution, and p is the probability of the binomial distribution, with the assumption that the L/R state follow the binomial distribution. p is adjustable for different here. Fig. 7.2 illustrates an example of the Track Segment hit pattern and the corresponding L/R state. However, it should be noted that the shape of the Track Segment may not be suitable for determining the L/R state of certain wires. Fig. 7.3 demonstrates the rate of undetermined L/R state for different wires. The Wires with number 0,2,8, and 9 which are at edge of a Track Segment, can hardly determine L/R only from the Track Segment pattern with more than 80% undecided rate.

Considering in real physics events, a certain hit may not be related to signal but from background, which is neither left pass nor right pass. Thus, the n_B are used to mark hits from background. And an extra equation was introduced to separate

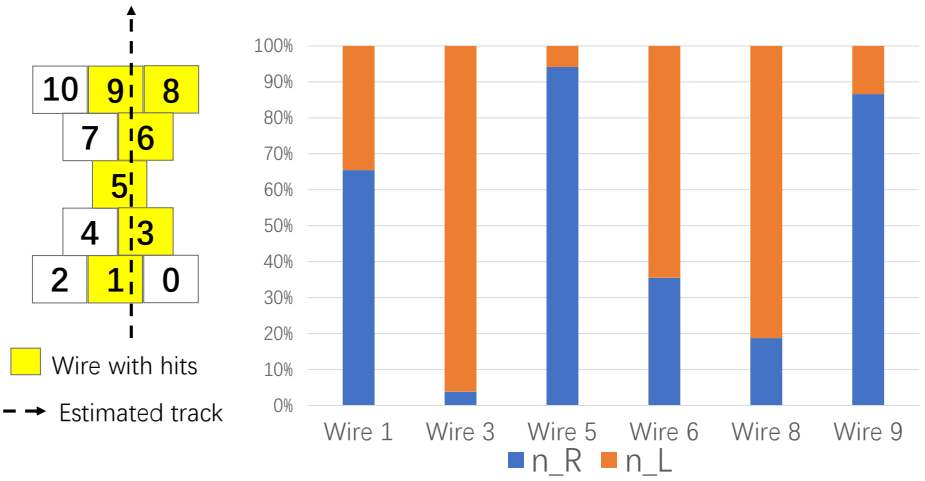


Figure 7.2: Example of Track Segment with specific hit pattern (left) and corresponding n_L and n_R ratio for each wire. For this pattern, Wire 3 and wire 8 are determined left state, while wire 5 and wire 9 as right state. Others are set as undecided state.

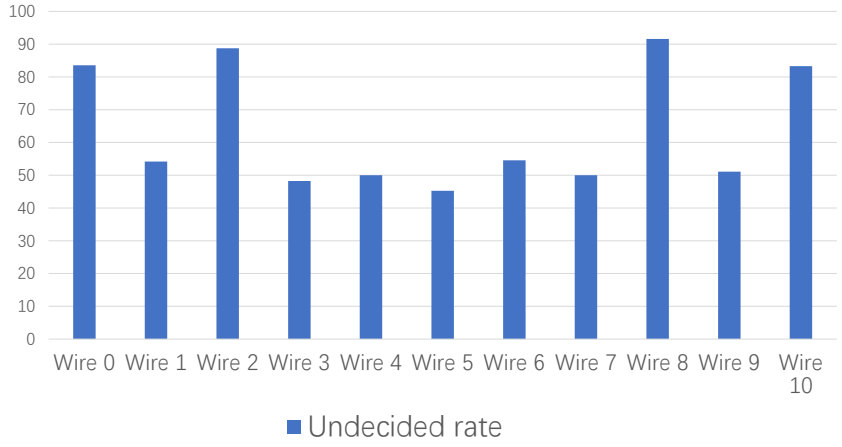


Figure 7.3: The undecided rate of all wires in Track Segment with $p = 0.7$. Wire 0,2,8,9, with undecided rate $> 80\%$ can hardly provide correct L/R state for input.

background like pattern:

$$\text{Left/Right state} = \begin{cases} \text{Signal} & if\ n_L + n_R > (1-b)\cdot(n_L + n_R + n_B) + 3\sigma \\ \text{Undecided} & \text{otherwise} \end{cases} \tag{7.2}$$

Here the b can be adjusted to reach a balance of signal efficiency and background reject rate.

We use the Monte Carlo (MC) simulation to generate 100k single charge muon tracks events for LUT training. Each event was mixed with Belle II early phase 3 background. Additional 20k events are generated for test.

Full L/R LUT has been trained and evaluated with b cut ranged from 0.4 to 0.99 and p from 0.5 to 0.99. Every hit have three real state in MC: background hit, right pass and left pass, the later two are signal hits. And every hit also have three predict state, undecided, left pass and right pass. We assign "undecided" for background like hits and give it low priority in hits selections. For the signal hits evaluation, we focus on the $Correct\ L/R\ Rate = \frac{\#Correct\ L/R\ hits}{\#Total\ Signal\ hits}$ and the $Undecided\ Rate = \frac{\#Undecided\ Signal\ Hits}{\#Total\ Signal\ hits}$. High correct rate and low undecided rate are preferred. For Background hit, we focus on the $Correct\ Background\ rate = \frac{\#Undecided\ background\ hits}{\#Background\ hits}$. Fig. 7.4 show these three parameters comparing with p and b. We could tell from this figure that $Correct\ L/R\ Rate$ does not depend on the b but depends on p. $Undecided\ Rate$ and $Correct\ Background\ Rate$ are influenced by both p and b. To get high $Correct\ L/R\ Rate$ with relative low $Undecided\ Rate$, we use $p = 0.7$ and $b = 0.9$ for following train sample generation. A $Correct\ L/R\ Rate = 91.8\%$ and $Undecided\ Rate = 59.2\%$ are expected. Considering at least 4 wires in one Track Segments, we have 80.5% probability to obtain at least one extra wires with correct decided L/R state. Considering the $Correct\ L/R\ Rate = 91.8\%$, we prefer to say it can be approximated as binomial distribution.

Another approach is to treat the 3D track reconstruction as a fitting process that aims to find a helix track minimizing the distance between the track and the circles which center of the wires and with a radius of $t_{drift} \times v_{drift}$ in the plane perpendicular to the wire (see Fig. 7.5). By providing the complete set of ϕ_{rel}, α, and t_{drift} as inputs for multiple wires in the Track Segment, it becomes feasible to reconstruct the track without explicitly determining the L/R state. This can be achieved by employing a deep neural network structure.

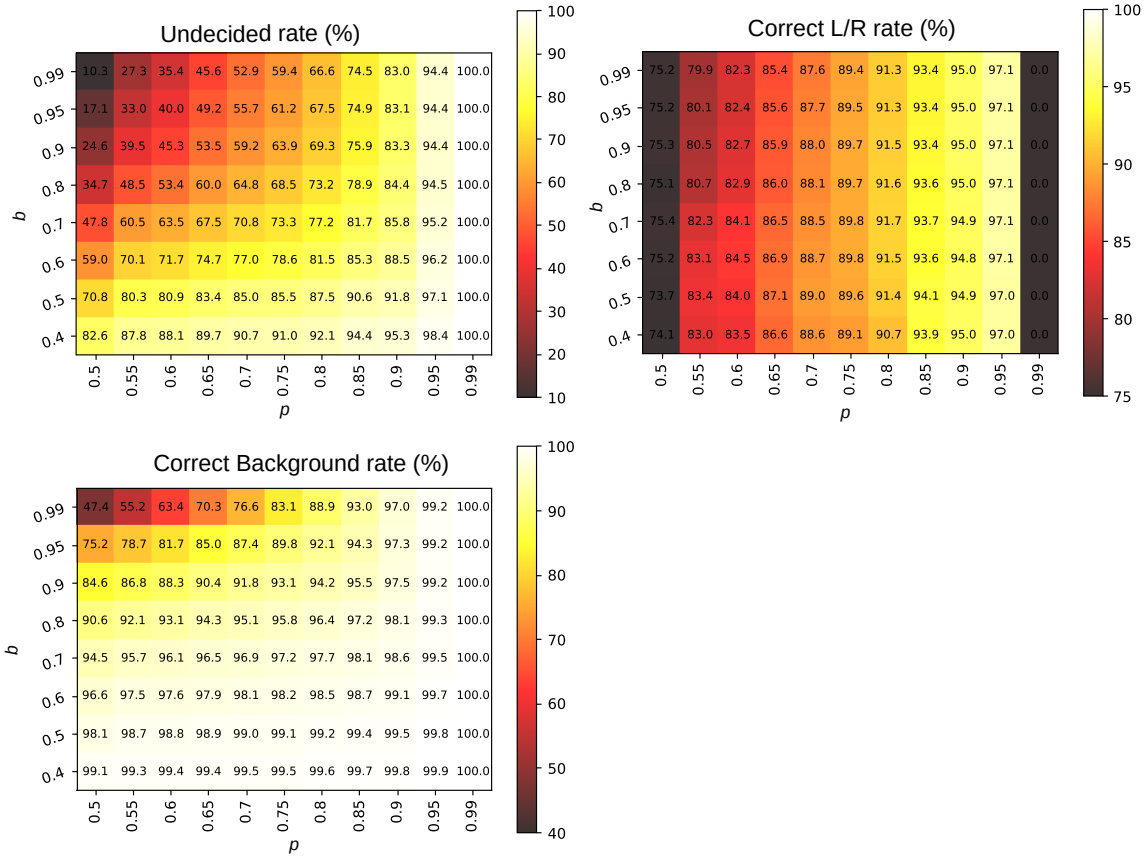


Figure 7.4: Upper Left: *Undecided Rate* compares with b and p. Upper Right: *Correct L/R Rate* compares with b and p. Lower Left: *Correct Background Rate* compares with b and p

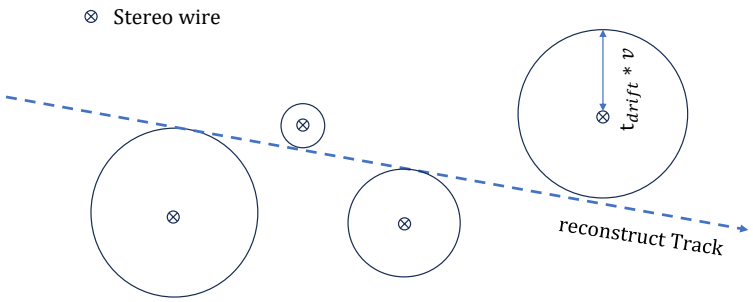


Figure 7.5: schematic diagram for a 3D track reconstruction with full wire information in a Track Segment. A linear approximation of the track is make in this figure.

Full wire input

As previously mentioned, the main challenge with the full wire input approach is the mismatch between the deterministic nature of the neural network (NN) and the variability of the signal wire. However, it is possible to address this issue by leaving input space for every wires no matter if it has signal which converts it as a missing data problem. In the case of our problem, the missing data can be classified as Missing Not At Random (MNAR) [42], which means that the missing values are related to the reasons for their absence and can be well modeled. We propose using a neural network approach to model the missing data, allowing us to directly input all the t_{drift} values into the NN. In this case, since the relative location is constant for each input node, additional location inputs such as ϕ_{rel} and α, as well as the L/R state, are not necessary. Therefore, the positive t_{drift} values are scaled to the range $(0, 1)$, while the no signal case we input -1. The specific model we employed is described in Section 7.3.

7.1.2 ADC information

It is feasible to include an additional bit for the ADC of each wire within the Track Segment in order to reject background noise from electronics and low energy gamma with low charge. Considering the CDCEF bandwidth only one bit for every wire is available currently. Thus, our preference is to transmit a boolean value indicating whether the ADC value crosses a certain threshold.

Fig. 7.6 illustrates the significant discrepancy in ADC values between signal and background events in the $ADC < 20$ region. In the 2022 physics run, events characterized by a substantial background composition resulted in approximately 10% of false track reconstructions [41], thereby affecting the hit selection process of the Neurotrigger. By incorporating the ADC input, it becomes possible to enhance the optimization of the Neurotrigger by refining the selection process.

Given that all inputs in the CDC Trigger pipeline operate at the Track Segment level, our intention is to extend the optimization efforts to this level as well. By utilizing the one-bit ADC information, it becomes possible to generate a hit pattern at the Track Segment level by applying an ADC threshold, as depicted in Fig. 7.7. To facilitate the rejection of background-like Track Segments, we have developed a LUT based on the hit ADC pattern, which has been trained using real physics events. In this process,

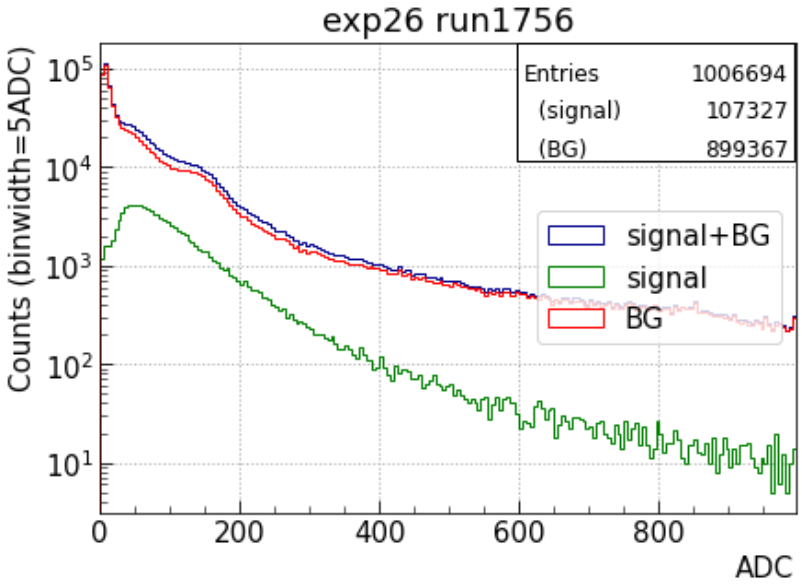


Figure 7.6: ADC Distribution for Signal and Background events. A significant deviation occurred at $ADC < 20$ region [43]

each Track Segment is labeled as either "signal" if it is used in any offline track reconstruction or "background" if it has not been utilized. The Track Segments are then categorized according to their hit ADC patterns. Subsequently, the number of background Track Segments, denoted as n_b, and the number of signal Track Segments, denoted as n_s, are tallied for each hit ADC pattern. To determine the background/signal classification for a particular pattern, the following condition is examined:

$$\text{Background/Signal state} = \begin{cases} \text{Background} & if\ n_b > p \cdot (n_b + n_s) + 3\sigma \\ \text{Signal} & \text{otherwise} \end{cases} \tag{7.3}$$

where $\sigma = \sqrt{(n_b + n_s) \cdot p \cdot (1 - p)}$ represents the width of a binomial distribution. The parameter p is optimized to achieve the optimal balance between efficiency and rejection rate. Figure 7.8 illustrates a typical example of this classification process.

However, it should be noted that even for background state patterns, there might still be some signal events present. In order to avoid any potential loss in

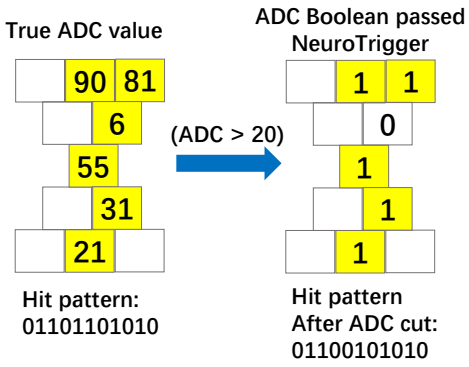


Figure 7.7: Example of hit pattern (yellow block at left) and hit pattern after ADC Cut (yellow block at right)

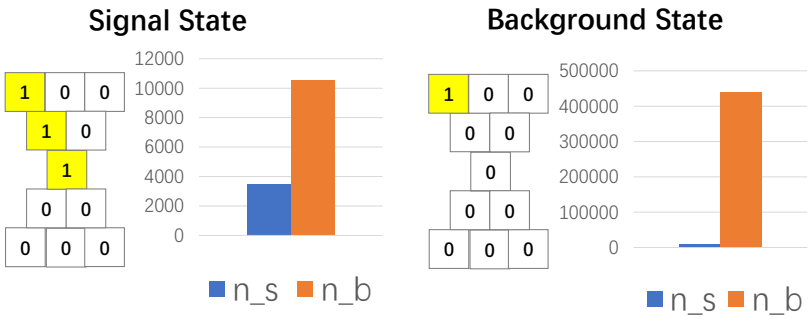


Figure 7.8: Example of typical signal state (Left) and background state (Right), with ADC cut at 20 and $p = 0.9$

efficiency, we prioritize the signal state patterns during the hit selection stage rather than directly rejecting the background state patterns. This allows us to focus on capturing the desired signal events while minimizing the risk of excluding any potential signal candidates . To train the Hit ADC pattern, 100,000 tracks from the 2022 Belle II physics run data are used, and the validation is performed with an extra 30,000 tracks. We use the Hit ADC pattern LUT has been trained and evaluated with ADC threshold ranged from 5 mV to 40 mV and p ranged from 0.5 to 0.95. We focus on the $background\ reject\ rate = \frac{\#Correct\ Background\ Track\ Segments}{\#Total\ Background\ Track\ Segments}$ and $signal\ efficiency = \frac{\#Correct\ Signal\ Track\ segments}{\#Total\ Signal\ Track\ Segments}$. Fig. 7.9 shows the evaluation result. It clearly that with increasing p, we could obtain better signal track segment efficiency while reduce the background track segment reject rate. As for $ADC\ CUt$, there is the best value at 15 mV or 20 mV, which is consistent with the ADC distribution in Fig. 7.6.To keep best signal efficiency, we choose $ADC\ Cut = 15$ and $p = 0.95$, where we could obtain 97.7% signal track segment while reject 83.7% background track segments. These values are applied for DNN training sample generalization. It should be noticed that $ADC\ Cut = 20$ is also preferred with same level efficiency.

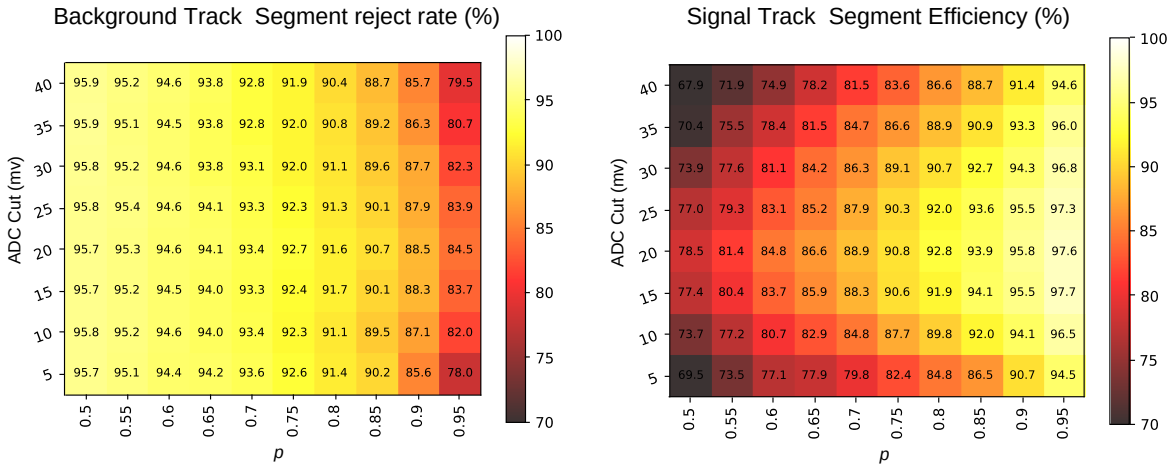


Figure 7.9: Left: The background track segments reject rate with different ADC cut and p value. Right: The signal track segments efficiency with different ADC cut and p value

7.1.3 Event timing finder input

In the current Neurotrigger implementation, a "Fastest Priority" t_0 is utilized instead of the Event Timing Finder (ETF) t_0 to determine the t_{drift}. However, with the availability of the newly implemented ETF module [37], we plan to transition to using the ETF t_0 output instead. The comparison between the ETF t_0 and Fastest Priority t_0 is illustrated in Fig. 7.10. The ETF module offers a t_0 resolution of 10 ns and helps reduce the possibility of obtaining t_0 values from background hits, which is the main negative part in Figure 7.10.

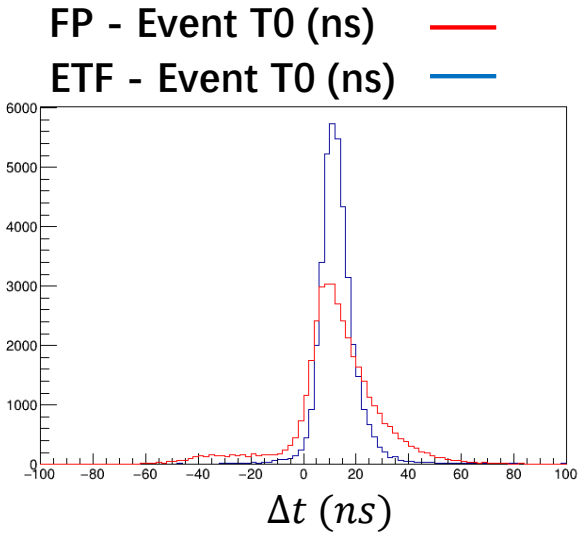


Figure 7.10: $\Delta t \equiv t_0^{\text{ETF(FastestPriority)}} - t_0^{\text{Events}}$ distribution. t_0^{Events} was got from offline reconstruction. ETF module has t_0 resolution of 10 ns, which is a two factor improvement of Fastest Priority.

Incorporating the ETF timing introduces a potential issue where the ETF t_0 may not always be smaller than the t_{hit} values from the CDC wires, resulting in negative t_{drift} values. To address this, we implement the following approach. First, taking into account the resolution of 10 ns, we assign low priority to wires with $t_{drift} < -10$ ns. Then, we enforce that all negative drift times are set to zero. This ensures that the t_{drift} values remain non-negative and helps mitigate the problem arising from negative drift times.

Parameter	Number of input feature	Assigned bits in FPGA
Original Input feature		
ϕ_{rel} for priority wire	9	9×13
α for priority wire	9	9×13
t_{drift} for priority wire	9	9×13
New Input feature		
extra ϕ_{rel} for selected wires	9/18/27	$9/18/27 \times 13$
extra α for selected wires	9/18/27	$9/18/27 \times 13$
extra t_{drift} for selected wires	9/18/27	$9/18/27 \times 13$
Total Input features	54/81/108	$9/18/27 \times 13$

Table 7.1: Input features for Selected extra wires input. The number of new input features depends on the number of extra wire(s) we use, 9/18/27 for 1/2/3 extra wire(s) case.

7.1.4 Summary of extra input features

As described above, the inclusion of extra wires from Track Segments is implemented using two strategies: "Selected extra wires input" and "Full wires input". ADC boolean are included to help hit selection, and ETF t_0 are used to improve t_{drift} resolution. The complete set of new input features is outlined as follows.

Selected extra wires input

The inputs are defined as Table 7.1:

A total of nine SLs are considered, consisting of five axial SLs and four stereo SLs. The axial SLs are directly obtained from the 2D track found by the 2D track finder. The selection of stereo Track Segments is restricted to the relevant range mentioned in Section 6.3.1. In the case where no Track Segment falls within the relevant range, no hit is used, and a value of 0 is inputted to the NN. Subsequently, the Track Segments are selected following a priority order, as follows:

1. ADC state as a signal (if ADC cut is applied)
2. $t_{drift} > -10$ ns
3. Determination of the L/R state
4. Fastest positive t_{drift}

Parameter	Number of input feature	Assigned bits in FPGA
Original Input feature		
ϕ_{rel} for priority wire	9	9×13
α for priority wire	9	9×13
t_{drift} for priority wire	9	9×13
New Input feature		
extra t_{drift} for all wires	99	99×13
Total Input features	126	126×13

Table 7.2: Input features for Full wires input input.

The Track Segment with the highest priority is selected, and the ϕ_{rel}, α, and t_{drift} values of its priority wire are used as input for the NN.

For the selection of extra wires, the priority order is as follows:

1. $t_{drift} > -10$ ns
2. Determination of the L/R state (if L/R is used)
3. Fastest non-negative t_{drift}

The top three extra wires with the highest priority are selected. Depending on the input nodes of the NN, either one, two, or three extra wires' ϕ_{rel}, α, and t_{drift} values are inputted to the NN. In this mode, the NN has a total of 54/81/108 input nodes for the cases of 1/2/3 extra wires, respectively.

Full wires input

the inputs are defined as Tab. 7.2:

The Track Segment selection follows the same rules as described earlier. No selection condition is applied for $extra - t_{drift}$ since all values are inputted. The timing of priority wires are included twice since we want to keep the origin input. In the case of SL 0, which has 15 wires, only the first 11 wires are included as inputs.

A total of 126 input nodes are used for the NN in this mode.

7.2 Neural-Network optimization

With the introduction of UT4, the architecture of the NN can be expanded and optimized. This section presents the optimization methods from three aspects: architecture modification and training optimization algorithm tuning and parameter tuning.

7.2.1 Architecture modification

In the previous architecture, which consisted of a single hidden layer NN or multi-layer perceptron (MLP) with fully connected nodes, the model can be viewed as a mathematical function that maps input values to output values. This type of architecture is suitable for well-defined fitting problems. However, it requires careful feature engineering prior to training [44], as we did previously by selecting the best signal-like inputs to minimize the impact of background noise. The single hidden layer structure has limitations in performing complex feature engineering on its own. Consequently, as the number of background hits increases, the performance of this architecture tends to deteriorate. Furthermore, considering our large dataset, which comprises more than 1 million events, a simple MLP structure with only approximately 2000 free parameters may not fully capture the intricate structures within the dataset. This limitation prompted us to explore the application of deep learning (DL) methods.

Multi hidden layers and deep learning

In contrast to MLPs, recent research [44, 45] has demonstrated that deep neural networks (DNNs) have the capability to extract features internally within the network itself, as depicted in Fig. 7.11. This ability presents a promising opportunity to process enriched inputs using DNN architectures. However, a significant challenge arises from hardware limitations. The current Neurotrigger implementation reveals that even a single MLP in the UT3 module incurs a latency of approximately 300 ns and utilizes around 44% of the DSP resources[40], which dominated the resource's usage of UT3 FPGA. Therefore, considering the new neural network architecture on UT4, 1 hidden layer with 300 nodes or 3 hidden layers with 81 nodes can certainly be implemented, thus we set a restriction of three hidden layers and 300 nodes per layer for best NN searching.

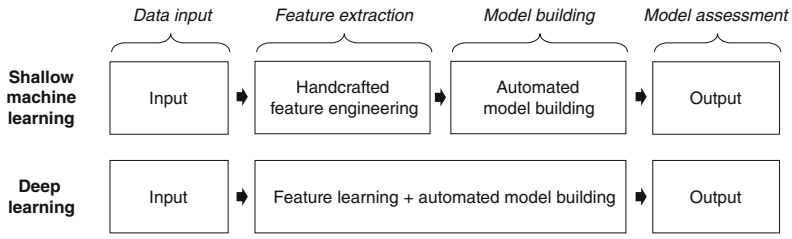


Figure 7.11: Process of analytical model building[44]

We begin with a simple case of adding more hidden layers and adjusting the number of nodes in each layer. The optimization ranges for the number of nodes are set between 20 and 300, while the number of hidden layers can vary from 1 to 3. For simplicity, we initially assume an equal number of nodes in each hidden layer. The concept diagram of this DNN fitter is illustrated in Fig. 7.12.

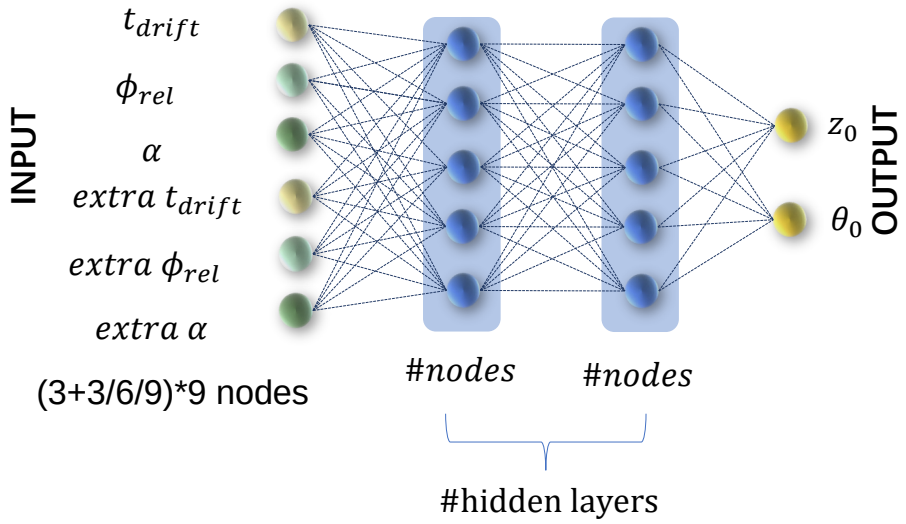


Figure 7.12: concept diagram of DNN fitter. The number of input features change based on the number of extra wires we use, every one extra wire will increase input features in one SL by 3.

Neural-Network classifier for vertex-z distribution

In order to improve the rejection rate of tracks not originating from the IP ($|z_0| > 1$cm), we can utilize the capabilities of a DNN to directly predict whether a track is from the IP or not. Instead of applying a wide cut on the output z_0 as a manual selection criterion, we can train a new DNN classifier with a binary target: 1 for tracks Off-IP and 0 for tracks from the IP.

The structure of the DNN classifier follows the same architecture as depicted in Fig. 7.12, but the output and target are modified to accommodate the classification task. This DNN classifier only provided a output p as the probability of background track. The limitation on the number of hidden layers and nodes remains the same as before. By training this new DNN classifier, we aim to achieve better rejection rates for tracks not originating from the IP, surpassing the performance of a simple manual cut based on z_0 alone. And with the same input feature as DNN fitter, it is possible to perform them parallel.

Attention based architecture for missing data modelling

As discussed in Section7.2, the incorporation of the second input strategy necessitates the use of a specialized model capable of effectively handling missing data. In this regard, the transformer model [46] has emerged as a promising candidate due to its ability to evaluate the relationships between long-distance and short-distance features and assigning attention to specific extracted features. This model may tackle the missing data problem as evidenced by similar approaches employed in [47].

The main components of the transformer model include the attention structure, as depicted in Fig. 7.13. In this structure, the attention weights and attention values are calculated independently from the input vector and then multiplied together to obtain the output. A softmax function is applied to attention weights to scale the sum of them to one. The softmax can be described as:

$$softmax(z_i) = \frac{e^{z_i - max(z)}}{\sum e^{z_i - max(z)}} \tag{7.4}$$

This process enables the model to emphasize certain attention values based on their corresponding attention weights. A typical transformer model consists of a

combination of multiple attention structures.

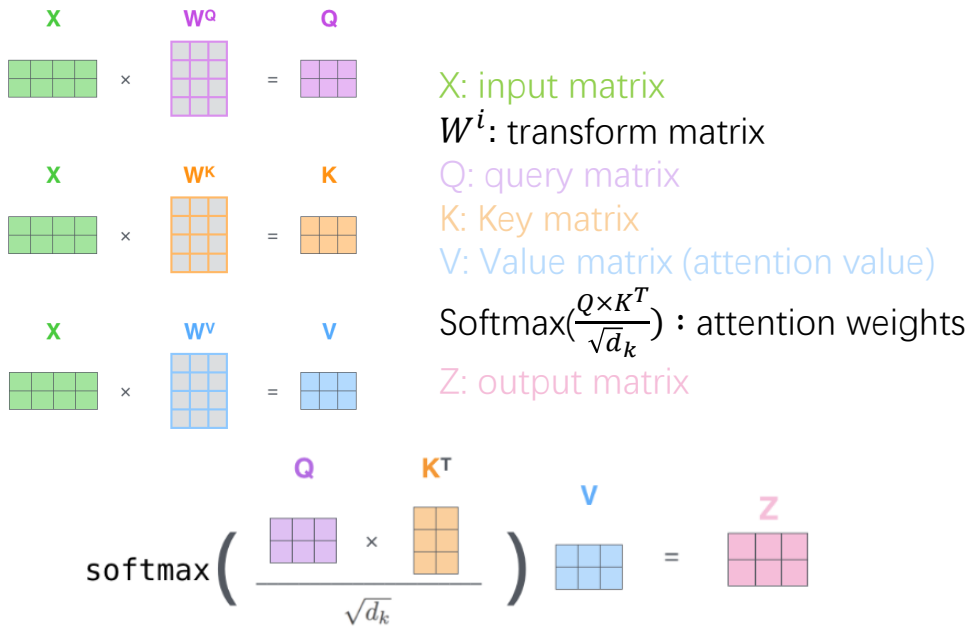


Figure 7.13: Single head attention structure. Q, K, Vmatrix are generated from input matrix X with transform weights matrices W^i. Weights matrix update during training. Attention weights calculated from $softmax(\frac{Q\times K^T}{\sqrt{d_K}})$, where softmax function scale the sum matrix it to 1 and d_k is the dimension of matrix K, also used to scale it. The attention weights are then multiply with attention value V to get the output Z.

In our specific case, where we deal with two-dimensional input data and the subsequent fitting/classification process can be effectively handled by a fully connected NN, we adopt the architecture depicted in Fig. 7.14. Initially, the full wires input is utilized to compute attention values and attention weights. The resulting values obtained by multiplying each attention value with its corresponding attention weight are then fed into a fully connected NN for the purpose of fitting z_0 and θ_0, or for IP track classification. To serve as a control group, we also train a fully connected NN with an equivalent number of parameters and depth, employing the same input.

This architecture allows us to leverage the benefits of the attention mechanism in extracting relevant features from the full wires input. The subsequent processing with the fully connected NN enhances the modeling and prediction capabilities, thereby

enabling accurate determination of z_0 and θ_0, or reliable classification of IP tracks.

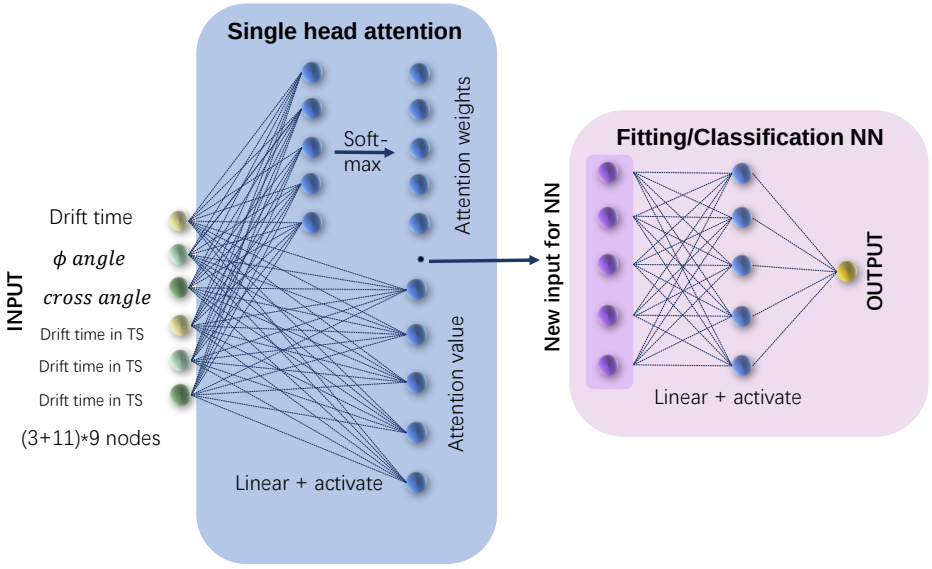


Figure 7.14: concept diagram of Attention Based NN

Activate function

In the context of the aforementioned architectures, an important aspect to consider is the choice of activation functions. Activation functions introduce non-linear relationships within neural networks, and recent studies [48] have demonstrated that different activation functions can have a significant impact on the performance of neural networks. The currently used activation function, tanh(x/2), can also be optimized.

Taking into account their widespread usage, we have compiled a list of potential activation functions for optimization, including:

1. Hyperbolic Tangent Function (Tanh): The current Neurotrigger utilizes Tanh(x/2) as its activation function. Tanh is a smooth, zero-centered function that maps input values to the range [-1, 1]. It is advantageous because it can handle negative, zero, and positive values effectively. However, it does not solve the vanishing gradient problem. The output of Tanh(x/2) is given by Equation 7.5. We keep the

activate function for z_0 fitter output as Tanh(x/2) to scale it to (-1,1).

$$\text{Tanh(x/2)} = \frac{exp(x/2) - exp(-x/2)}{exp(x/2) + exp(-x/2)} \tag{7.5}$$

2. Sigmoid Function: The sigmoid function transforms input values to the range [0, 1]. It is a smooth and continuously differentiable function, and its derivative is always positive. However, like Tanh, it suffers from the vanishing gradient problem. Sigmoid activation is suitable for the classifier output as it scales the output to (0, 1) thus we keep it as activate function for DNN classifier output. The sigmoid function is defined in Equation 7.6.

$$\text{Sigmoid(x)} = \frac{1}{1 + exp(-x)} \tag{7.6}$$

3. Rectified Linear Unit Function (ReLU). Relu has been the most widely used activation function for deep learning applications. The ReLU activation function performs a threshold operation on each input element where values less than zero are set to zero thus the ReLU is given by Equ. 7.7. ReLU can solve the vanishing gradient problem since the gradient at positive part is always 1. But it may lead to some dead neurons because of set negative part to 0.

$$\text{Relu(x)} = \begin{cases} x, & if\ x > 0 \\ 0, & otherwise \end{cases} \tag{7.7}$$

4. Leaky ReLU (LReLU): LReLU addresses the "dead" neuron issue by introducing a small negative slope for negative input values. The parameter α determines the slope and is typically set to a small value, such as 0.01. The LReLU function is defined in Equation 7.8.

$$\text{leaky Relu(x)} = \begin{cases} x, & if\ x > 0 \\ \alpha \times x, & otherwise \end{cases} \tag{7.8}$$

5. Exponential Linear Units (Elu). Elu expressed in Equ. 7.9 has negative values which allows for pushing of mean unit activation closer to zero thereby reducing

computational complexity thereby improving learning speed. It has the same feature as ReLU in positive to solve the vanishing gradient.

$$\text{Elu(x)} = \begin{cases} x, & if\ x > 0 \\ \alpha \times exp(x) - 1, & otherwise \end{cases} \tag{7.9}$$

The figures of every activate functions are showed in Fig. 7.15.

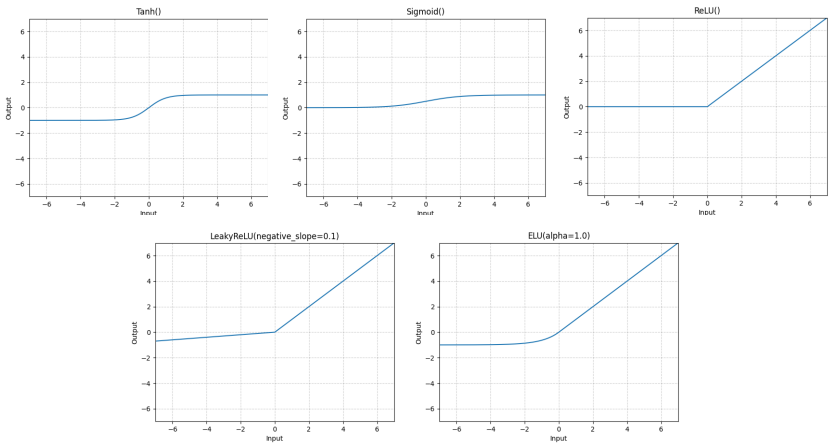


Figure 7.15: Activate functions. From left to right, up to down are Tanh, Sigmoid, Relu, LRelu and Elu.

7.2.2 Training optimization algorithms tuning

In addition to the neural network architecture, the optimization algorithms for the training process can also be fine-tuned. In this regard, we present three potential candidates for gradient descent optimization algorithms to train our DNN.

Resilient backpropagation algorithm

Resilient backpropagation algorithm (RProp) [49] is a popular gradient descent algorithm that only uses the signs of gradients to compute updates. This algorithm was used in Neurotrigger training process. In (RProp), the parameter update can be written

as:

$$w_{(i+1)jk} = w_{ijk} - lr \times \Delta_{ijk} \frac{\partial L_i}{\partial w_{ijk}} \tag{7.10}$$

where Δ_{ijk} transforms follow:

$$\Delta_{(i+1)jk} = \begin{cases} Max(\Delta_{max}, \eta^+ \Delta_{ijk}), & \text{if } \frac{\partial L_i}{\partial w_{ijk}} \frac{\partial L_{i-1}}{\partial w_{i-1jk}} > 0 \\ Min(\Delta_{min}, \eta^- \Delta_{ijk}), & \text{if } \frac{\partial L_i}{\partial w_{ijk}} \frac{\partial L_{i-1}}{\partial w_{i-1jk}} < 0 \end{cases} \tag{7.11}$$

where $(\Delta_{min}, \Delta_{max})$ are preset range for Δ_{ijk} and (η^-, eta^+) follow $0 < \eta^- < 1 < \eta^+$. A common value for (η^-, eta^+) are set as $(0.5, 1.2)$.

As showed in [49], RProp can speed up training process. Meanwhile, RProp provide a different step size for each weight which can update each weights in independent speed. But RProp generally requires large batch for update which may have not good performance with too much randomness in small batches training.

Stochastic gradient descent algorithm with momentum

Stochastic gradient descent (SGD) calculated gradient descent from a stochastic selected subset from the full sample. And the momentum was introduced to help accelerate SGD in the relevant direction and dampens oscillations [50]. The full expression can be written as:

$$\begin{aligned} g_i &= \frac{\partial L_i}{\partial w_{ijk}} \\ b_i &= \mu b_{i-1} + (1 - \tau) g_i \\ w_{(i+1)jk} &= w_{ijk} - lr \times b_i \end{aligned} \tag{7.12}$$

In the above equations, b_i represents the gradient with momentum from the previous iteration, initialized as $b_0 = 0$. The parameter μ denotes the momentum factor, and τ represents the dampening factor. By incorporating the momentum from the previous gradient, the training process can accelerate towards an optimal point while mitigating oscillations when approaching it. In typical cases, μ is set to 0.9, and τ is set to 0, indicating no dampening.

Adaptive Moment Estimation algorithm

Adaptive Moment Estimation (Adam) [51] is an optimization algorithm that enables different learning rates for different parameters in the neural network training process. It follows a set of equations outlined as follows:

$$
\begin{aligned}
g_i &= \frac{\partial L_i}{\partial w_{ijk}} \\
m_i &= \beta_1 m_{i-1} + \Phi 1 - \beta_1) g_i \\
v_i &= \beta_2 v_{i-1} + \Phi 1 - \beta_2) g_i^2 \\
\hat{m}_i &= m_i / (1 - \beta_1^i) \\
\hat{v}_i &= v_i / (1 - \beta_2^i) \\
w_{(i+1)jk} &= w_{ijk} - lr \hat{m}_i / (\sqrt{\hat{v}_i} + \epsilon)
\end{aligned}
\tag{7.13}
$$

Here, (β_1, β_2) are coefficients used to calculate the sums of the gradients and their squares, typically set to $(0.9, 0.999)$ by default. The variables m_i and v_i represent the momentum terms obtained from the gradient and its square, initialized with a value of 0. The scaled versions $\hat{m}_i$ and $\hat{v}_i$ remove the bias from the initial 0 values. The parameter ϵ is a small constant added to ensure that $(\sqrt{\hat{v}_i} + \epsilon)$ is not equal to zero.

According to the findings presented in [51], Adam has exhibited superior performance compared to other optimization algorithms. Therefore, we consider it as the primary algorithm and perform a simple comparison with the other two algorithms mentioned.

Batch size and learning rate

All three algorithms mentioned above can be applied at the batch level, which involves using subsets of the total sample. Consequently, the selection of batch size and learning rate becomes crucial, particularly for SGD and RProp. It is widely recognized that a larger batch size can expedite model training through the parallelism of GPUs. However, an excessively large batch size can result in poor generalization. The choice of batch size is also closely related to the learning rate, as parameters are only updated after processing a batch. Therefore, a large batch size is typically associated with a small learning rate. Additionally, adjusting the global learning rate during the training process using a technique known as a learning rate scheduler can be beneficial.

This approach is also applicable to Adam though it could adjust the learning rate adaptive[52]. Hence, we fine-tune the batch size, learning rate, and learning rate scheduler prior to commencing the full training process.

Pytorch Lib

Our training and model building process rely on the PyTorch library [39], which offers fast and flexible experimentation capabilities, efficient production workflows, and a user-friendly interface.

7.2.3 Parameter tunning

We have a set of parameters that need to be tuned for our models, and they are as follows:

- Batch size: Ranging from 256 to 4096.
- Learning rate: Ranging from 10^{-4} to 10^{-2}.
- Optimization algorithm: Adam, SGD, or RPROP.
- Number of hidden layers: Ranging from 1 to 3.
- Number of hidden nodes: Ranging from 20 to 300.
- Activation function: Relu, Tanh(x/2), LRelu, or ELU.

For the DNN fitter, our tuning target is to minimize the σ_{95}, which is defined as the standard deviation of the $z_0^{NN} - z_0^{offline}$ distribution for the central 95% of events. This 95% cut is applied to reduce the contribution of outliers. For the DNN classifier, the target is to maximize precision, which is calculated as the ratio of the number of correctly classified tracks to the total number of tracks.

To conduct the multi-variable optimization, we utilized the nondominated sorting genetic algorithm II (NSGA-II) [53] implemented in Optuna [54]. Optuna provides a powerful framework for efficient and effective parameter optimization. We have not performed any optimization for the attention-based fitter/classifier yet, as it requires more than 10 times training time. This optimization process including few hundreds

times of NN training and validation, and each full process of training and validation is call a “trial”.

8

Performance evaluation of DNN 3D track trigger

This section illustrates the performance of DNN 3D track trigger performance, including the parameters tuning results, and the finalized DNN fitter, classifier and attention based NN performance.

8.1 Training samples

For the DNN training, a portion of the 2022 physics run data, which is specially taken without HLT filtering is used. We randomly separated total offline events and generate training sample, validation sample and test sample with certain fraction as 70%, 20% and 10%. Only CDC trigger tracks related with a real offline reconstructed tracks are used for training and validation. Fake tracks which only reconstructed in CDC trigger but not related to real offline reconstructed tracks are saved only for testing. A total of 1.1 million tracks are used for each expert training, with 0.34 million tracks for

validation and 0.2 million tracks for testing. Besides about 80k fakes tracks saved for testing. To compare different feature engineering strategy, six different types of samples are generated from the same events, as shown in Table 8.1. The sample #1 follows the same condition as previous Neurotrigger training and is used as control group.

Number	Input type	t_0	ADC LUT	Full L/R LUT
#1	Selected extra wires	Fastest Priority	×	×
#2	Selected extra wires	ETF	×	×
#3	Selected extra wires	Fastest Priority	✓	×
#4	Selected extra wires	Fastest Priority	×	✓
#5	Selected extra wires	ETF	✓	×
#6	Full wires	ETF	✓	×

Table 8.1: Type of training samples

As for parameter tuning process, 100,000 tracks from the above training samples are used for training and another 50,000 tracks from the training samples are used for validation.

8.2 Parameters tuning results

We tune the parameters mentioned in 7.3.3 before the finalize the DNN training. Since the learning rate (lr), batch size and optimization algorithm highly influence the training speed, we first tune these three. First we fixed other parameters and now the model is as Table 8.2

We just enlarge other parameters to maximum to make sure this learning rate is suitable even with the largest model. Only 54 input feature which corresponding to one extra wire was used to simplify and speed up the tuning process. A total a 100 trial tuning was performed and lr and batch size are suggested with exponential order. Optimization algorithm are chosen from Adam, SGD and RPROP. The result of σ_{95} distribution is showed in Fig. 8.1. All trials stopped before reach the maximum epoch which suggests it converged. We could tell that the learning rate and batch size have a little influence ~ 0.1 cm on the σ_{95}, and best batch size/learning rate is at $512/2.95 \times 10^{-3}$. Optimization algorithm contribute a large difference. Adam has

Parameter	Value
Fixed parameter	
Data type	samples #5
Model type	DNN fitter
Input features	54 (1 extra wire)
Number of hidden layers	3
Number of hidden nodes	300
Activation function	Relu
Maximum Epoch	2000
Tunning parameter	
Batch size	Ranging from 256 to 4096
Learning rate	Ranging from 10^{-4} to 10^{-2}
Optimization algorithm	Adam, SGD, or RPROP

Table 8.2: Parameter setting for tuning of batch size, learning rate and Optimization algorithm

general 1 cm improvement on σ_{95} comparing with Rprop or SGD. We pick up the best trials for SGD, Adam and RProp and compare the loss decrease in the training process as showed in Fig. 8.2. It is clear that Adam can not only speed up the training to around only 100 epochs even with the largest model.

Thus, we fixed learning rate as 2.9×10^{-3}, batch size as 512 and optimization algorithm as Adam. This is applied for both DNN fitter and DNN classifier since they have similar structure with only different target.

8.2.1 DNN fitter

For DNN fitter tuning, the tuning model as Table 8.3:

It should be mentioned that a callback function used to stop training if no improvement after 50 epochs was applied to speed up tuning process. A total of 300 trials of tuning was performed, and the results are showed in Fig. 8.3. It is clearly that ReLU and LReLU got a better final $sigma_{95}$ in general. And 3 hidden layers trials have a more than 1 cm improvement for σ_{95} with 1 or 2 hidden layers trials, large number of hidden layers have not been tried yet due to the latency requirement. The best point of number of hidden nodes is at 207. Fig. 8.6 show the combination influence of #hidden layers and #hidden nodes while keep activate function as LRelu, at where we could see

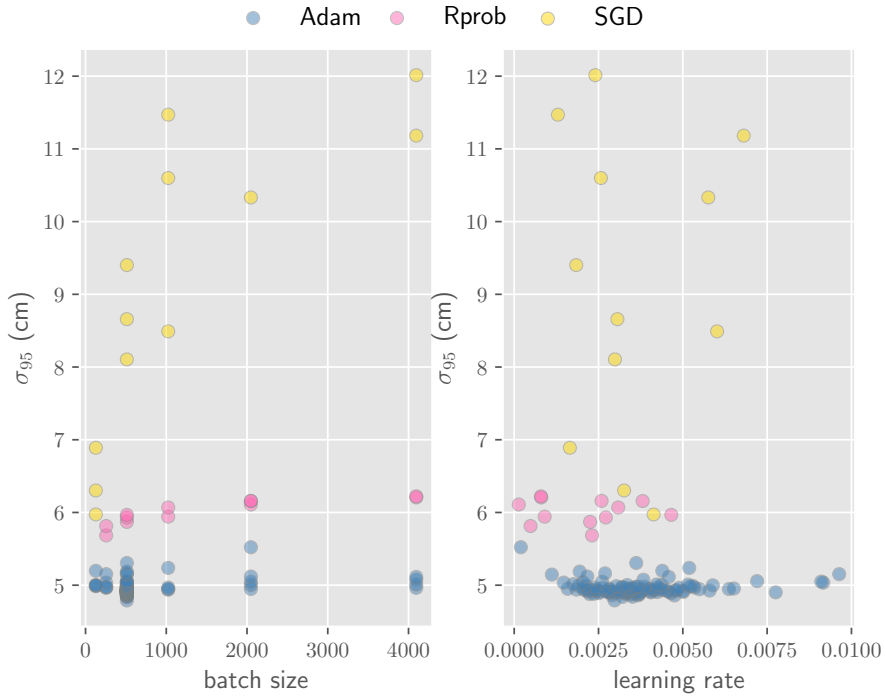


Figure 8.1: Batch size, learning rate and optimization algorithm tuning results. Left for batch size versus σ_{95} and right for learning rate.

Parameter	Value
Fixed parameter	
Data type	samples #5
Model type	DNN fitter
Input features	54 (1 extra wire)
Batch size	512
Learning rate	2.9×10^{-3}
Maximum Epoch	2000
Optimization algorithm	Adam
Tunning parameter	
Number of hidden layers	from 1 to 3
Number of hidden nodes	from 20 to 300
Activation function	Relu, Tanh(x/2), LRelu, or ELU.

Table 8.3: Parameter setting for tuning of DNN fitter

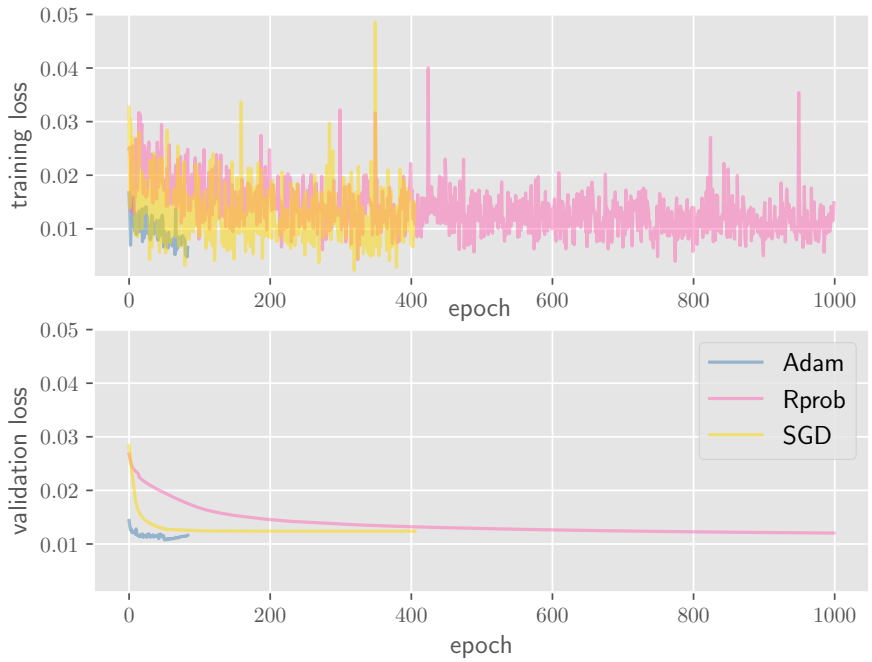


Figure 8.2: Error cure for the Best trial of Adam, SGD and RProp. Upper is the loss for training sample (in a single batch) and lower is the loss for validation sample (in full sample).

that, three hidden layers trials always have a 1 cm improvement comparing with one or two hidden layers trials. And for a minimize model, even with 69 hidden nodes we could keep σ_{95} ~5.2 cm, which can be easier to deployed on FPGA. We choose LRelu as activate function, with *#hidden nodes* = 207 and *#hidden layers* = 3 for DNN fitter full training to see the best performance of this structure.

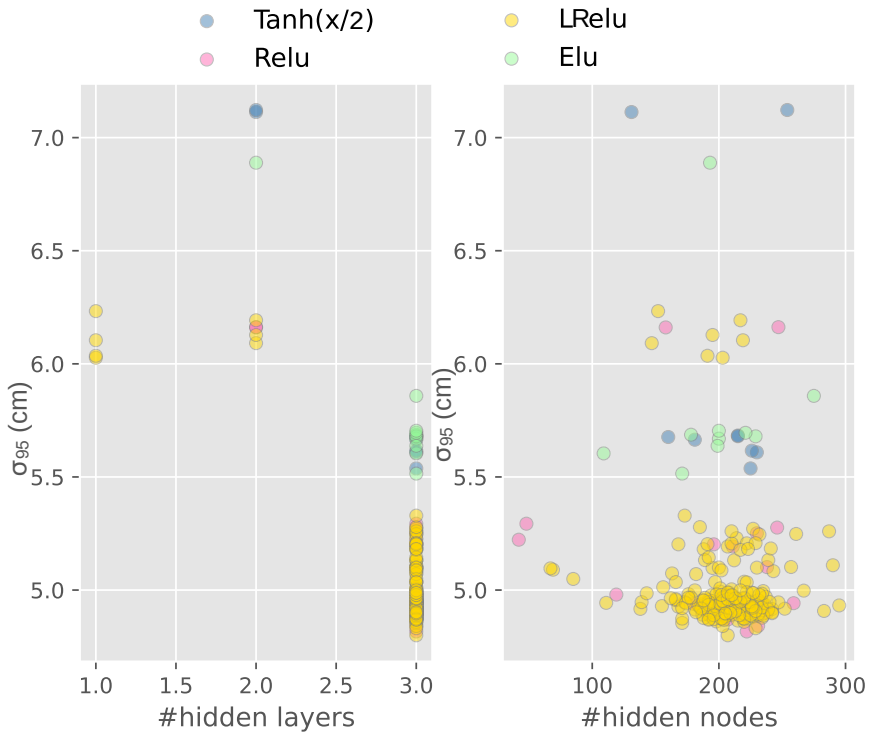


Figure 8.3: DNN fitter tuning results. Left: σ_{95} comparing with *#hidden layers*. Right: σ_{95} comparing with *#hidden nodes*

8.2.2 DNN classifier

For DNN classifier tuning, the tuning model is as Table 8.4. The others are the same as DNN fitter, only with a shift of NN output from z_0 and θ_0 to the boolean of if track originated from IP ($z_0 < 1$cm). And the tuning target is to maximize the $accuracy \equiv \frac{\#Correct\ classified\ tracks}{\#Total\ Tracks}$. A total of 300 trials are performed. The results are showed in Fig. 8.5. For classifier case, Tanh(x/2) gives a better performance with a 1 percent improvement in *accuracy* from ReLU. And similar tendency in *#hidden layers*

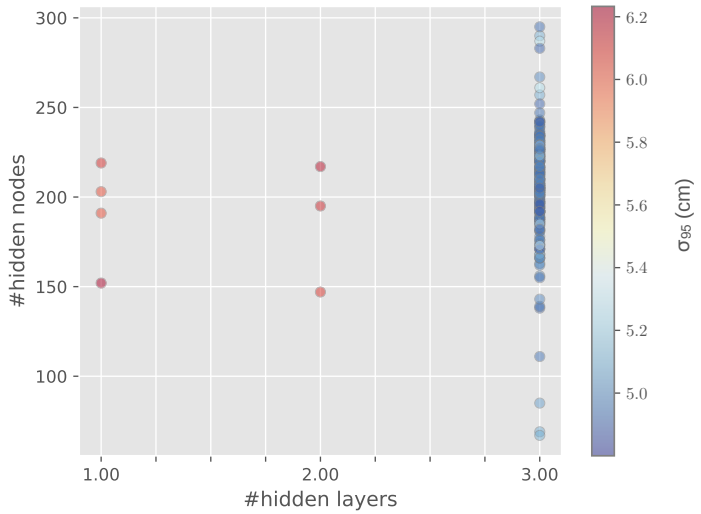


Figure 8.4: σ_{95} of different combination of #hidden nodes and #hidden layer, keeping activate function as LRelu

was seen, that 3 hidden layers trials have a 1 percentage improvement in *accuracy* while 1 hidden layer and 2 hidden layers got similar result. The best point of number of hidden nodes is at 160. Fig. 8.6 show the combination influence of #hidden layers and #hidden nodes while keep activate function as Tan(x/2). It is interesting that the with only one hidden layer and 72 hidden nodes, it can obtain a precision of 92%, which is only one percent decrease as the best one, while two hidden layers trials have a worse performance. We choose Tanh(x/2) as activate function, with $\#hidden\ nodes = 160$ and $\#hidden\ layers = 3$ for DNN classifier full training.

8.2.3 Attention based Architecture

We have not tuned the attention based NN yet since it take 10 times longer training time. Thus, We set the default model parameters as Table 8.5, which try to keep same depth as DNN cases and possible smallest free parameters.

Parameter	Value
Fixed parameter	
Data type	samples #5
Model type	DNN Classifier
Input features	54 (1 extra wire)
Batch size	512
Learning rate	2.9×10^{-3}
Maximum Epoch	2000
Optimization algorithm	Adam
Tuning parameter	
Number of hidden layers	from 1 to 3
Number of hidden nodes	from 20 to 300
Activation function	Relu, Tanh(x/2), LRelu, or ELU.

Table 8.4: Parameter setting for tuning of DNN Classifier

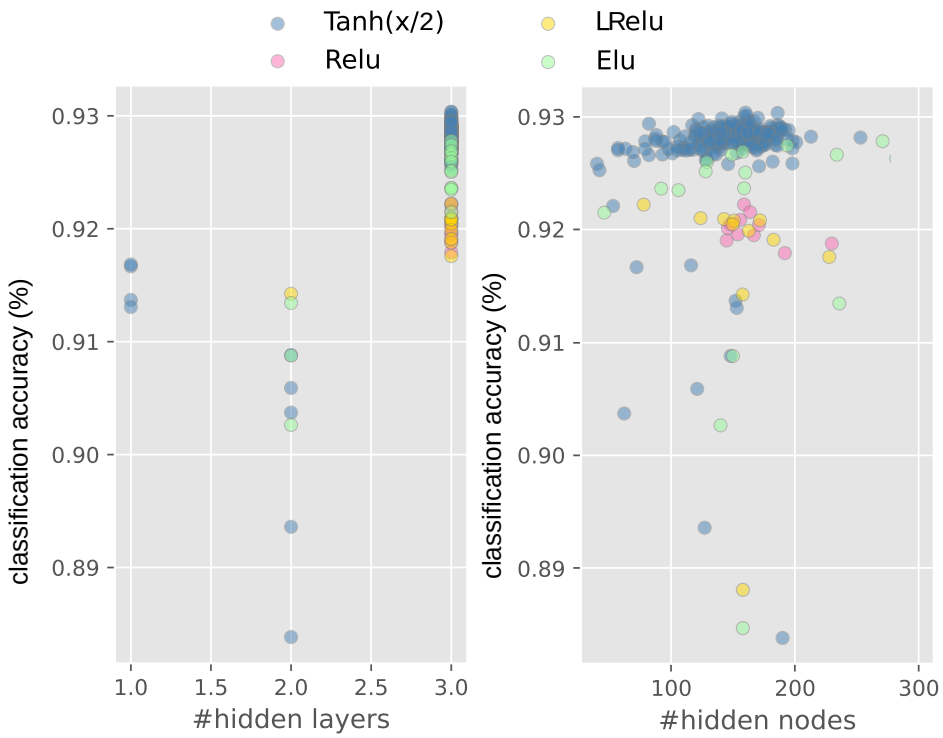


Figure 8.5: DNN Classifier tuning results. Left: *accuracy* comparing with *#hidden layers*. Right: *accuracy* comparing with *#hidden nodess*

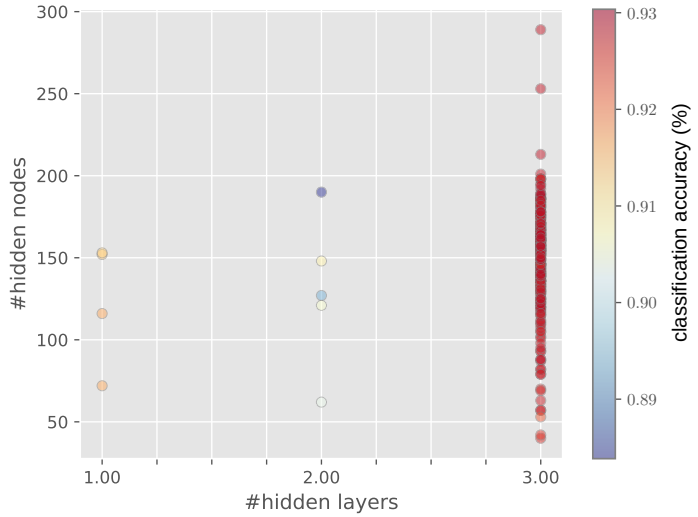


Figure 8.6: *accuracy* of different combination of #hidden nodes and #hidden layer, keeping activate function as Tanh(x/2)

Parameter	**Value**
Data type	samples #6
Model type	Attention Based architecture
Input features	126
Batch size	512
Learning rate	2×10^{-3}
Maximum Epoch	2000
Optimization algorithm	Adam
Number of attention values/weights	81
Number of hidden layers after attention structure	1
Number of hidden nodes after attention structure	81
Activation function	Relu

Table 8.5: Parameter setting for Attention Based architecture

Parameter	Value
Data type	samples #1
Model type	Neurotrigger
Input features	27 (no extra wire)
Batch size	2048
Learning rate	10^{-3}
Maximum Epoch	2000
Optimization algorithm	Adam
Number of hidden layers	1
Number of hidden nodes	81
Activation function	Tanh(x/2)

Table 8.6: Parameter setting for original Neurotrigger retraining

8.3 Performance evaluation

This is section shows the performance evaluation for retrained original Neurotrigger, DNN fitter, DNN classifier and attention based structure.

8.3.1 Control group: Retrained original Neurotrigger

To comparing the DNN 3D track trigger performance, first we retrained the origin Neurotrigger with new data taken from physics run as control group. The training parameters are showed in Table 8.6

We tested it with the test sample, and the Fig. 8.7 show its Δz_0 distribution at full range and at IP region with $|z_0^{\text{offline}}| < 1$. The z_0 resolution of track from IP region directly related to the trigger efficiency, thus we extract it from the origin distribution. σ_{95} is calculated for both distribution. Besides, we also care about the minimized range including 95% entries (range_{95}), which is directly represent the selection condition we could apply to keep 95% signal tracks. For the retrained Neurotrigger, $\text{range}_{95} = 13.6$ The 2D plot of z_0^{NN} and z_0^{offline} are showed in Fig. 8.8. Comparing with the hardware trigger performance as Fig. 6.12 and Fig. 6.13, retrained Neurotrigger with $\sigma_{95}^{\text{IP}} = 5.53$ even have worse IP resolution. This can be induced by different test samples, because we use a later phase data from 2022 run contaminated with high background level. Meanwhile, the linear relationship of z_0^{NN} and z_0^{offline} became better, which lead to a reeducation of background trigger rate.

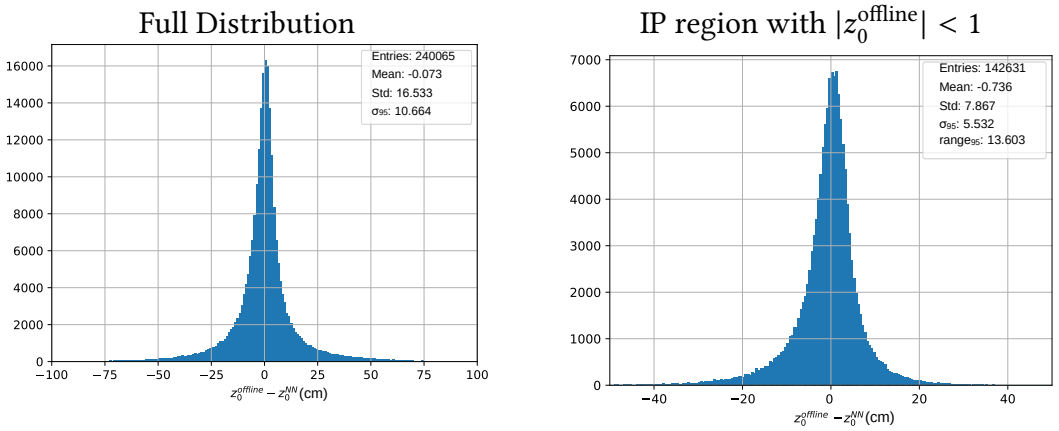


Figure 8.7: Δz_0 distribution for retrained Neurotrigger. Left: Full scale; Right: IP region with $|z_0^{\text{offline}}| < 1$.

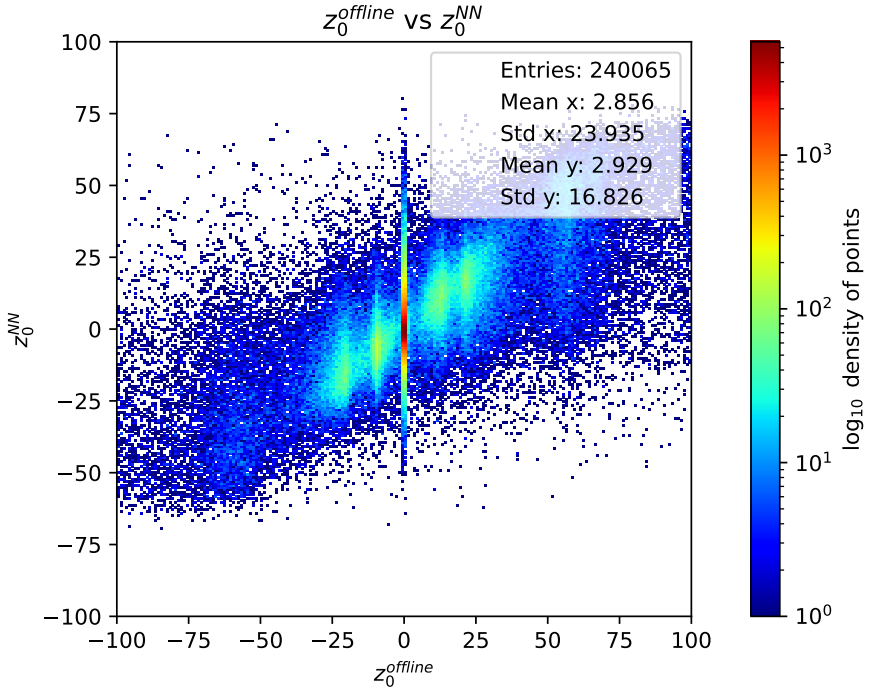


Figure 8.8: 2D plot of z_0^{NN} and z_0^{offline} from retrained Neurotrigger.

8.3.2 DNN fitter Performance

After well tuned DNN fitter, we trained it with full training samples. Fig.8.9 show the test result base on samples #5 and with 1 extra wire. The $\sigma_{95}^{\mathrm{FUll=6.21}}$ for full scale distribution improved by 3 cm and IP resolution $\sigma_{95}^{\mathrm{IP=2.34}}$improved by factor 2 comparing with Fig. 8.9. Besides, considering the range$_{95}$ = 7.19cm, we could restrict the selection condition by 5 cm while maintain the same efficiency. As for the 2D plot as Fig. 8.10, a well linear relationship can be seen even at $z_0^{\mathrm{offline}} > 50$cm, which is not premised in retrained Neurotrigger.

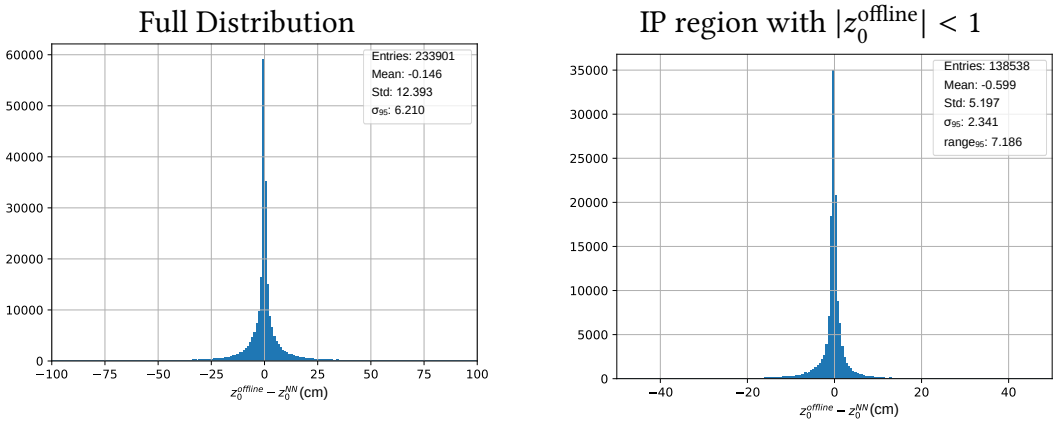


Figure 8.9: Δz_0 distribution for tuned DNN fitter #5 and with extra wire 1. Left: Full scale; Right: IP region with $|z_0^{\mathrm{offline}}| < 1$.

We also focus on the trigger efficiency and background reject rate, since our DNN fitter works on the track level, here we also use the signal track efficiency and background track reject rate for tracks here, which are defined as:

$$\begin{aligned}\text{Signal Track Effiency} &= \frac{\#\text{Signal Tracks Pass selection}}{\#\text{Total Signal Track}} \\ \text{Background Track Reject Rate} &= \frac{\#\text{Off} - \text{IP Background Tracks Not Pass selection}}{\#\text{Total Off} - \text{IP Background Track}}\end{aligned} \tag{8.1}$$

It should be noted that only tracks that related with a certain offline reconstructed track, which have certain z_0^{offline}, were used for training, validation, and testing here. Fake tracks which are built into CDC track trigger which not related with real track were not considered. Thus, the background track only contributes from Off-IP background

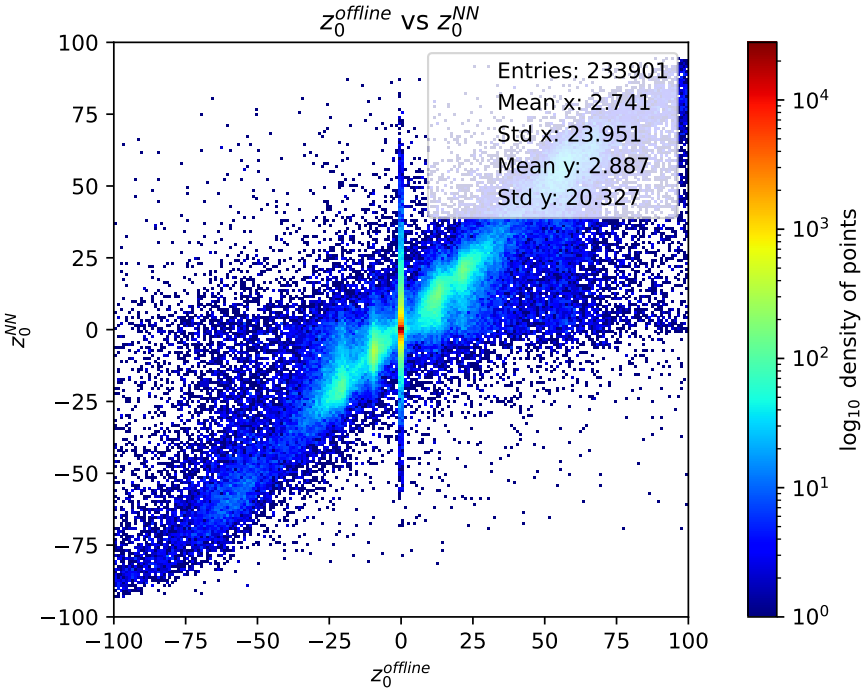


Figure 8.10: 2D plot of z_0^{NN} and z_0^{offline} from tuned DNN fitter.

tracks. The detail discussion of fake track events was showed in Section. 8.3.5. Signal tracks are defined as track with $|z_0^{\text{offline}}| < 1$. Here we use selection condition as $|z_0^{\text{NN}}| < 15$, which is same as CDCTRG low-multi bits z_0 condition. It should notice that here the efficiency did not take track selection into consideration since it is same for all NN trigger method, where we always require 3 of 4 stereo layers has hits related with 2D track.

To check the different feature engineering strategy influence on resolution, signal track efficiency, and background track reject rate, we compare the performance from sample #1~ #5 at different p_T with fixing input feature = 54, which corresponding to 1 extra wire used as showed in Fig. 8.11. Neurotrigger performance was included as control group. From upper left figure, it is clear that DNN fitter resolution improve by more than 2 factors at every point comparing with Neurotrigger. But the different feature engineering strategy can not show distinguishable difference. At the Upper right figure, we can also see the resolution is better at every point, especially for IP tracks. As for signal track efficiency, at $p_T < 0.5$GeV the samples #2 #3 #4 gain an improvement with 0.5% comparing with sample #1 and sample #5 which applied both ADC LUT and ETF timing gain a 1% more improvement. While for the background track reject rate, a distinguishable difference can be told at $p_T > 2$GeV region. Thus, we prefer to conclude that for DNN fitter z_0 resolution case, new introduced ADC LUT, Full L/R LUT and ETF timing provide an improvement at both efficiency and background track reject rate, while this do not work for the full resolution improvement.

We all checked impact of extra wires inputs samples #5. We trained and test 4 DNN with #input feature = (27,54,81,108) which corresponding to origin input, 1 extra wire, 2 extra wires and 3 extra wires. The hidden structures are kept as the same for these four. The results are showed in Fig. 8.12. At every figure, we could see that the extra wires 1/2/3 input have a improvement comparing with extra wires 0 input, even for resolution, a 1 cm improvement can be seen at p_T <0.5 GeV. But considering 1,2 or 3 extra wire(s) themselves, we did not see significant difference in between. Thus, to keep a minimal resource cost, we prefer to use the extra 1 wire case. Combining Fig. 8.12 with Fig. 8.3, we demonstrate that for DNN fitters architecture, optimization of the number of hidden layers and hidden nodes make main contribution, while the extra input features only provides a slight improvement, and feature engineering followed.

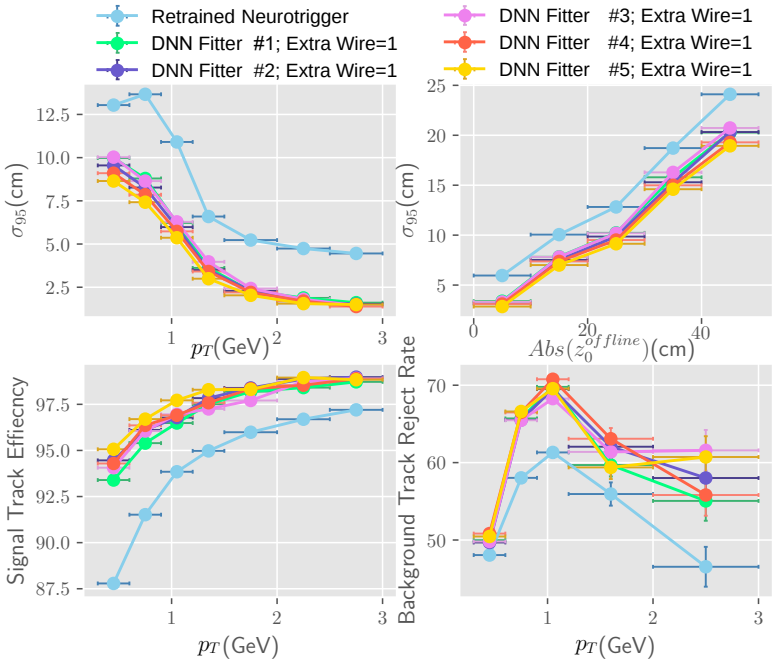


Figure 8.11: Performance comparison between different feature engineering strategies. Upper left: σ_{95} versus transverse momentum p_T; Upper right:σ_{95} versus $|z_0^{offline}|$; Lower right: Efficiency versus p_T; Lower right: Background reject rate versus p_T.

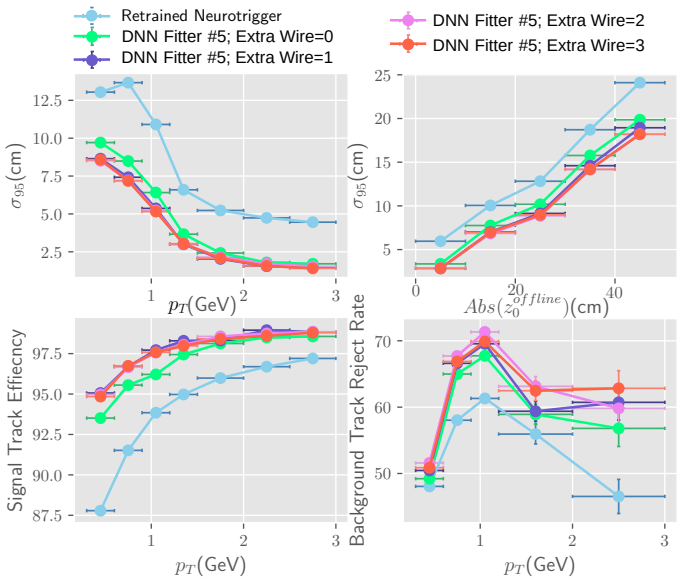


Figure 8.12: Performance comparison of different extra wire(s). Upper left: σ_{95} versus transverse momentum p_T; Upper right:σ_{95} versus $|z_0^{offline}|$; Lower right: Efficiency versus p_T; Lower right: Background reject rate versus p_T.

8.3.3 DNN classifier Performance

After well tuned DNN fitter, we also trained it with full training samples. Fig.8.9 show the test result base on samples #5 and with 54 input features. Regarding tracks with $p >= 50\%$ as background tracks, we can calculate the signal track efficiency = 92.7% and background track reject rate = 89.1%. By adjusting the p cut, we can adjust the signal track efficiency to meet the efficiency requirement.

To compare the DNN classifier performance with DNN fitters, we applied both of them on same test samples and combine its output. Fig.8.14 show the 2D plot of DNN fitter while we applied the $p < 50\%$ cut on each track. It is clear that the p cut can keep the tracks from IP even with predicted $z_0^{\mathrm{NN}} > 25$cm from DNN fitter, while reject most background tracks inside the $z_0^{\mathrm{NN}} < 15$cm cut region. However, we still could see about 3% of signal track with predicted $z_0^{\mathrm{NN}} < 1$ are rejected by p cut. To maintain the advantage from both DNN fitter and DNN classifier, we propose to use it parallel.

Same as DNN fitter, we check the impact of increasing the different feature engineering strategy, we trained and tested DNN classifier with sample #1 $\sim$ #5 and fixed #input feature = 54. Fig. 8.15 show the result. In this figure, we keep Cut of p as 65

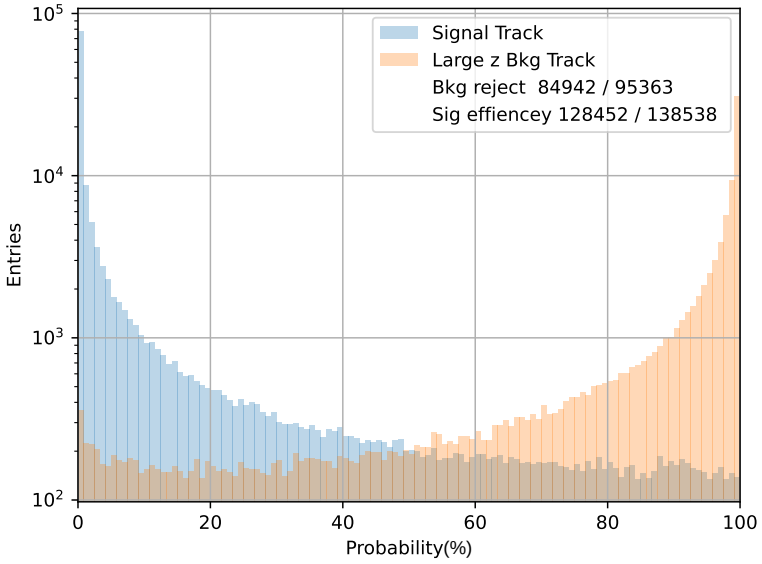


Figure 8.13: Background Probability p distribution for signal track (blue) and background track (orange), tested with sample #5 and DNN Classifier. Reject rate and signal efficiency are calculated regarding tracks with $p >= 50\%$ as background tracks.

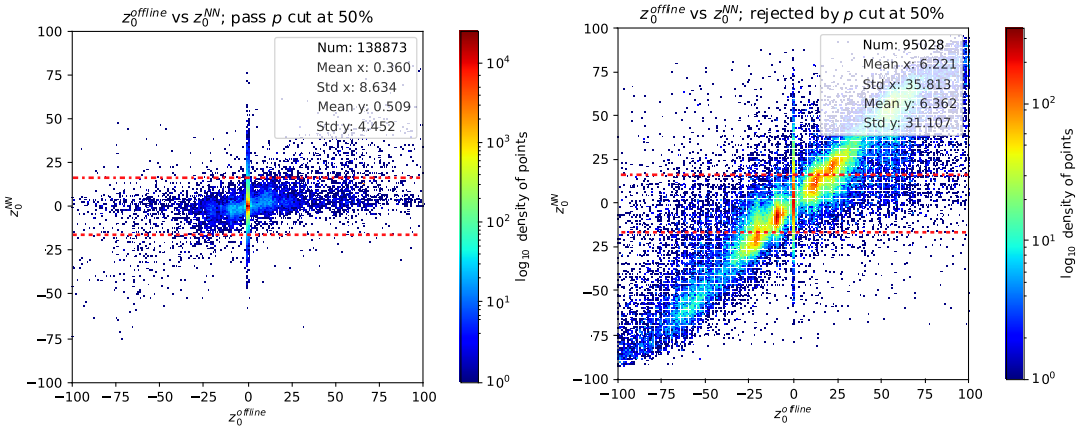


Figure 8.14: 2D plot of z_0^{NN} and z_0^{offline} from tuned DNN fitter passed the $p < 50\%$ cut (Left) and rejected by the cut (Right). The red dash line shows the cut of $|z_0^{\mathrm{NN}}| < 15$cm

.

for DNN classifier and cut of z_0^{NN} as 15 cm for retrained NeuroTrigger, where the later one is the default cut with CDCTRG low-multi bits. It is clear that all DNN classifier have $\sim$ 80% improvement of *Background Reject Rate* with p_T <1 GeV. Meanwhile, the *Signal Efficiey* is also better at full scale of p_T except p_T <0.5 GeV.For efficiency at low p_T, we can improve it by increasing p cut. Considering different samples case, the *Signal Efficiey* keeps almost same while we can see the improvement in *Background Reject Rate* with applying ADC CUT (sample #4), using ETF (sample #2) and using L/R LUT (sample#3). The sample #5 have the best performance with ETF and ADC LUT applied, which have a general 5% *Background Reject Rate* improvement with p_T >2 GeV.

We also check the impact from input features. Fig. 8.16 show the comparison for #input feature = (27,54,81,108) with sample #5. The same cut as above has been applied. We can see a more than 2 percent improvement for *Background Reject Rate* comparing no extra wire case (#input feature = 27) and with extra wire cases(#input feature = 54,81,108). A large deviation about 3 percent can also be seen for these two types as *Signal efficiency* especially with p_T <0.5 GeV. And considering 1, 2 or 3 extra wire(s) cases, no distinguishable difference can be told.

In conclusion, the DNN classifier architecture can improve *Background Reject Rate* by 80% at low p_T while keeping almost same *Signal Efficiency* in full scale. But this must use extra wires input otherwise we'll lose efficiency. A detail comparison of the *Background Reject Rate* and *Signal Efficiency* with different cuts for are showed in Section. 8.4.

8.3.4 Attention based NN trigger

For Attention Base Fitter, the Δz_0 distribution is showed in Fig. 8.17 and 2D plot in Fig. 8.19. The full resolution improved by 1 cm further comparing with DNN fitter #5. And IP resolution also improved by 0.5 cm. To check if this improvement is from architecture difference, we trained a special DNN fitter#6 with same depth, input feature and free parameters of Attention Base Fitter as a control group. This DNN fitter#6 have 3 hidden layers and 90 hidden nodes per layer, with total free parameters of 27,810. And Fig. 8.19 show the σ_{95} at different p_T for Attention Based Fitter and DNN fitter #5 and DNN fitter #6. Attention Based Fitter got a $(0.5 \sim 1.5)$cm resolution

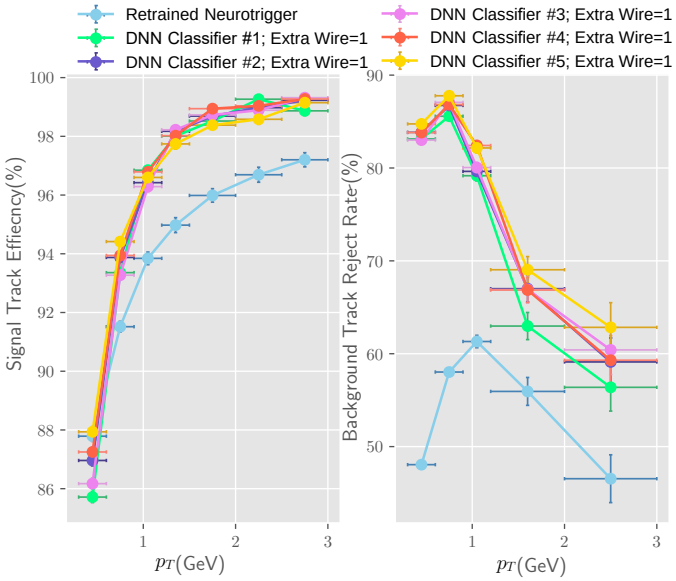


Figure 8.15: *Signal Efficiey* (Left) and *Background Reject Rate*(Right) versus transverse momentum p_T for sample #1 ~ #5 cases. Cut of p was set as 65 for DNN classifier and cut of z_0^{NN} was set as 15 cm for retrained NeuroTrigger.

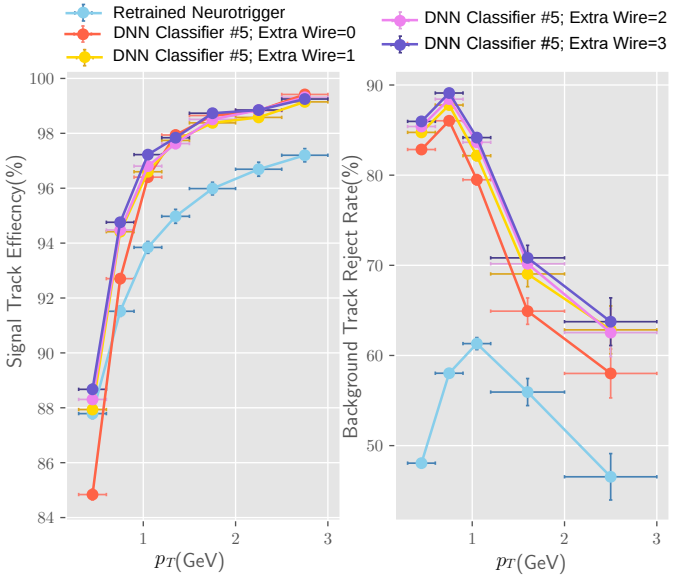


Figure 8.16: *Signal Efficiey* (Left) and *Background Reject Rate*(Right) versus transverse momentum p_T for # input feature = (27,54,81,108) cases. Cut of p was set as 65 for DNN classifier and cut of z_0^{NN} was set as 15 cm for retrained NeuroTrigger.

improvement at different p_T comparing with both DNN fitters. And we can see DNN fitter #6 have almost same performance as DNN fitter #5, even with enlarge input feature and free parameters. This implied that the attention based structure do help to make better use of the extra wires input. Another thing can be seen from DNN fitter #5 and DNN fitter #6 is that the missing data problem no longer reduce the resolution with deep learning structure, which is still a main problem for single-hidden-layer NN as illustrate in [18]. Thus, a possible attempts to merge the five experts into single one with DNN will be tested in further.

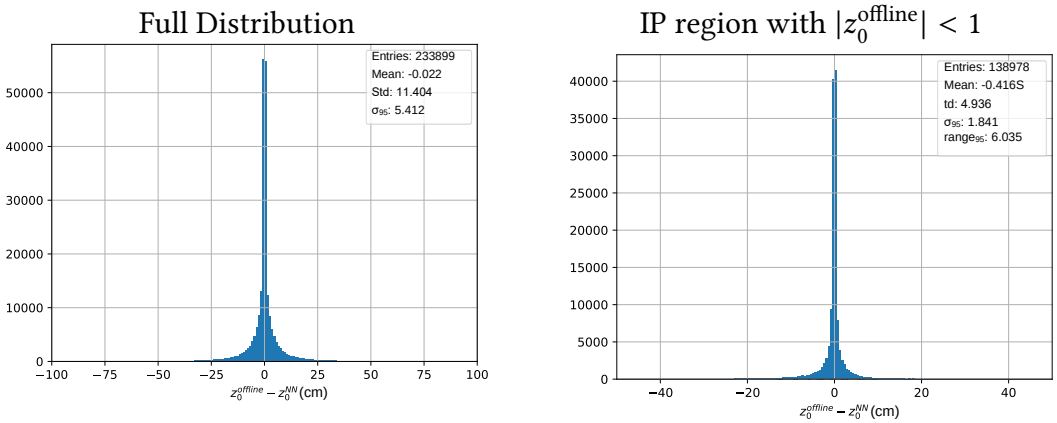


Figure 8.17: Δz_0 distribution for Attention Based Fitter. Left: Full scale; Right: IP region with $|z_0^{\text{offline}}| < 1$.

And we also check the performance of Attention Based Classifier as Fig. 8.20. The Attention Based Classifier has a 1% accuracy improvement further comparing with DNN classifier. It should be stressed that current Attention Based NN trigger have not been well tuned. So it may still remain space for improvement. And We will conduct a full tuning for it then.

8.3.5 Fake tracks background

As mentioned in 6.1, main background tracks are from two sources: Off-IP tracks and fake track, for training, validation and testing we did not include fake tracks since they do not have any true z_0 or θ_0. Since we want to check the performance of current NN with the fake tracks, we generated a sample with 87,287 fake tracks and applied every

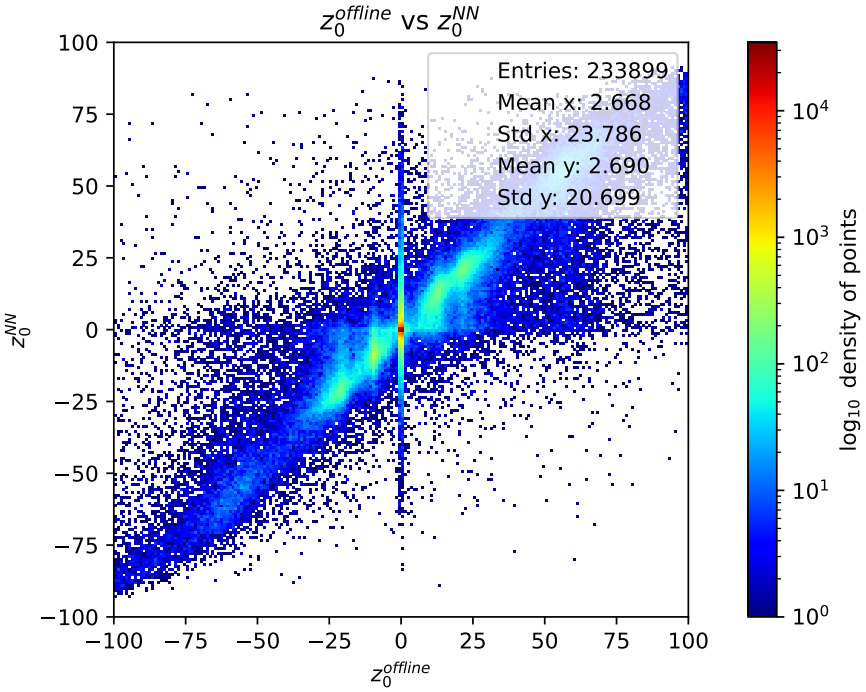


Figure 8.18: 2D plot of z_0^{NN} and z_0^{offline} from Attention based fitter.

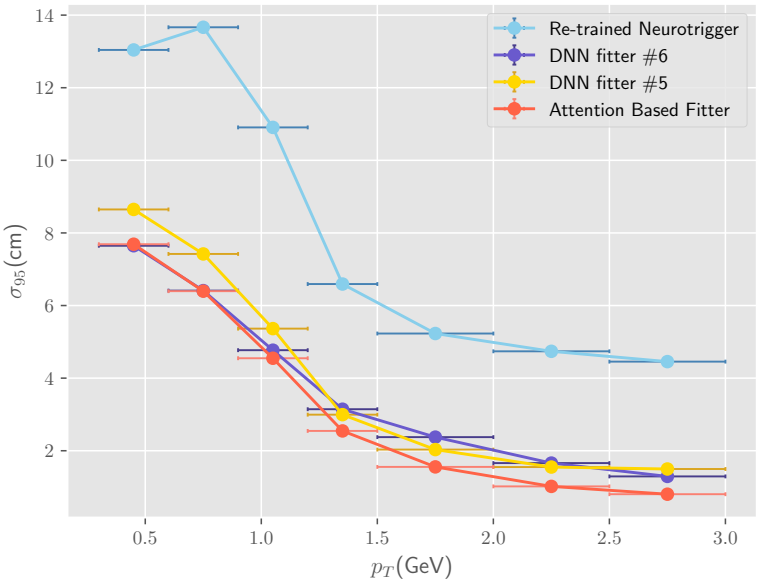


Figure 8.19: σ_{weighted} versus p_T for Attention Based Fitter, comparing with DNN fitter #5, #6 and retrained Neurotrigger. DNN fitter #6 has same depth, free parameters and input features as Attention Base Fitter, with only different in architecture—DNN fitter #6 is full connected while attention based fitter have a transformer layer.

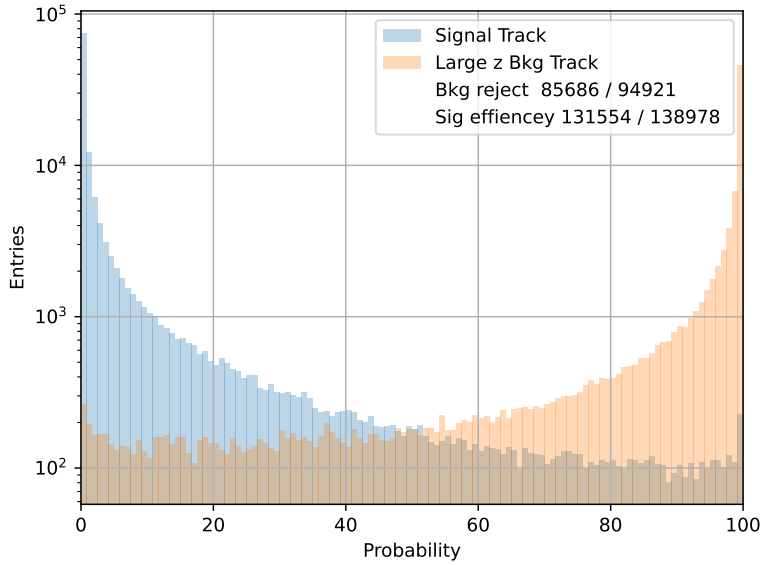


Figure 8.20: Attention Based Classifier's background Probability p distribution for signal track (blue) and background track (orange), tested with sample #6

NN on it to see the output. For z_0 fitter case, the results are showed in Fig. 8.21. It is clear that the distribution of z_0^{NN} for fake tracks has a certain structure. But with z_0^{NN} selection criteria as z_0^{NN} <15 cm, we could reject 59.5%(61.0% /59.4%) fake tracks with retrained Neurotrigger(DNN fitter #5 /Attention based fitter). There is no significant difference at the same selection criteria. However, with better IP resolution, we could apply more strict selection criteria, which provide better fake track events rejection rate. Considering the classifier cases, the output background probability distribution is

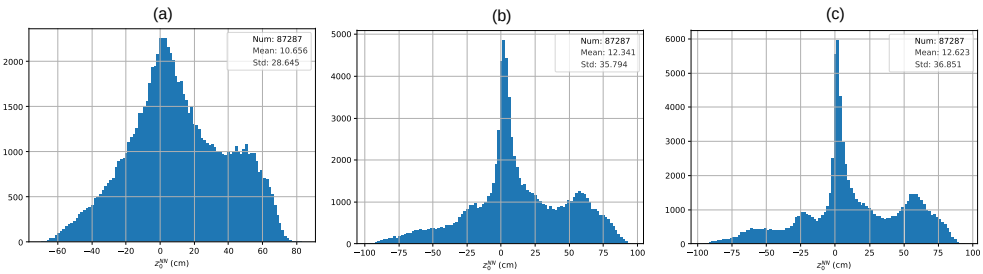


Figure 8.21: Output z_0^{NN} distribution for fake tracks from (a) retrained Neurotrigger (b) DNN fitter #5 with extra 1 wire (c) Attention based fitter

showed in Fig 8.22. It is interesting that even without inputting any fake tracks for

training, the classifier can distinguish the fake tracks somehow and output a large background probability for most case. Applying selection criteria as $p^{NN} < 50\%$ we can obtain 76.6%(77.2%) fake tracks events rejection rate with DNN classifier #5 (Attention based classifier). The detail comparison of efficiency and fake track events rejection rate are showed in next section.

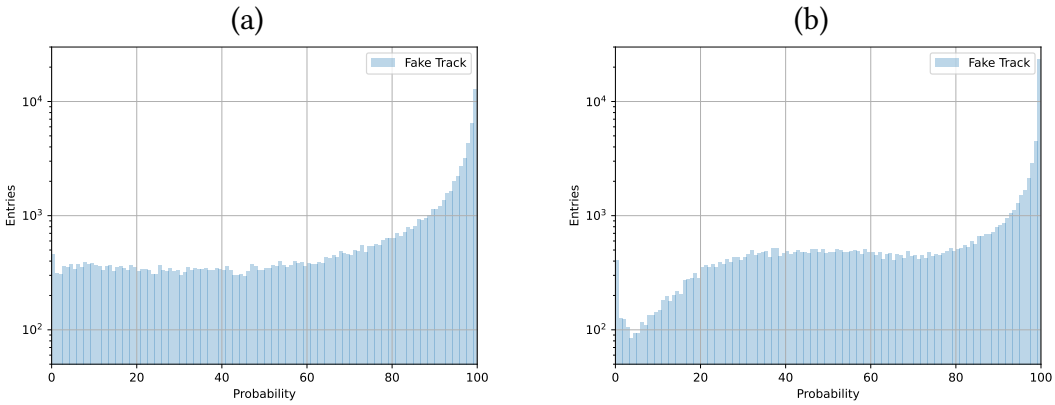


Figure 8.22: Output background probability distribution from (a) DNN classifier #5 with extra 1 wire (b) Attention based classifier

8.4 Summary

We summarized the Signal Events Efficiency and Total Background Events Rejection Rate for the above models with different selection condition to give a direct comparison as Fig. 8.23. In order to show the possible architecture for directly implemented on UT4, small DNN fitter (classifier) with 3(1) hidden layers, 69(72) hidden nodes and Other parameters keep same as DNN fitter (classifier) #5 are trained and included in Fig. 8.23. The signal events and backgrounds events follows the same definition as Table 6.2. To simplify the trigger conditions, we regard events with at least 1 CDC 3D trigger track pass selection condition as the triggered events, otherwise not triggered. To separate the effects of Off-IP Background Events and Fake Track Background Events, we show each of it rejection rate comparing with Signal Events Efficiency in Fig.8.24.

Considering we want to keep a 95% Signal Track Efficiency, we summarize the performance of these NN into Tab. 8.7 with selection to obtain the same efficiency. It is

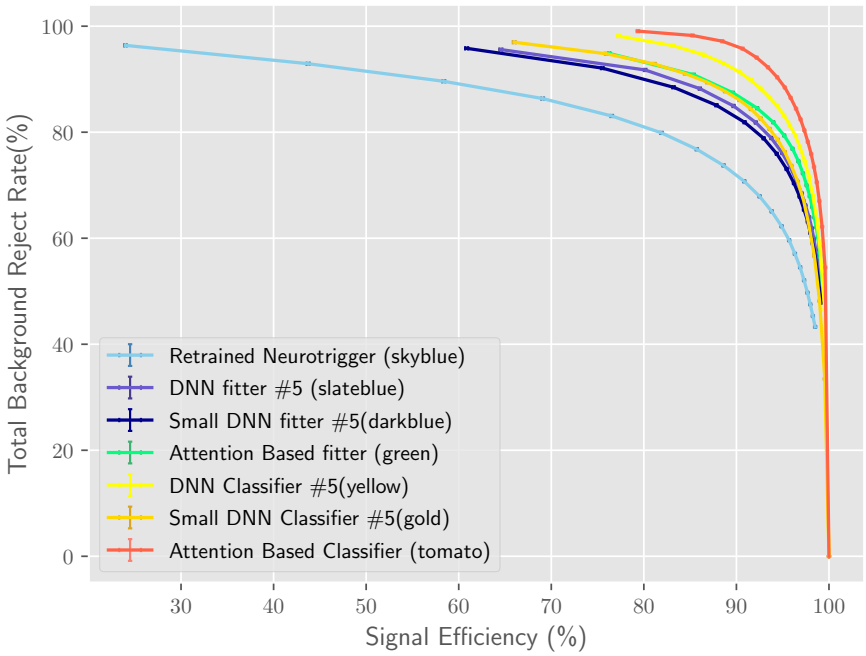


Figure 8.23: ROC curve of Signal Events Efficiency and Total Background Events Rejection Rate for the retrained Neurotrigger, DNN fitter (sample #5, extra wire = 1), DNN classifier (sample #5, extra wire = 1), small DNN fitter & classifier, Attention Based Fitter and Attention Based Classifier

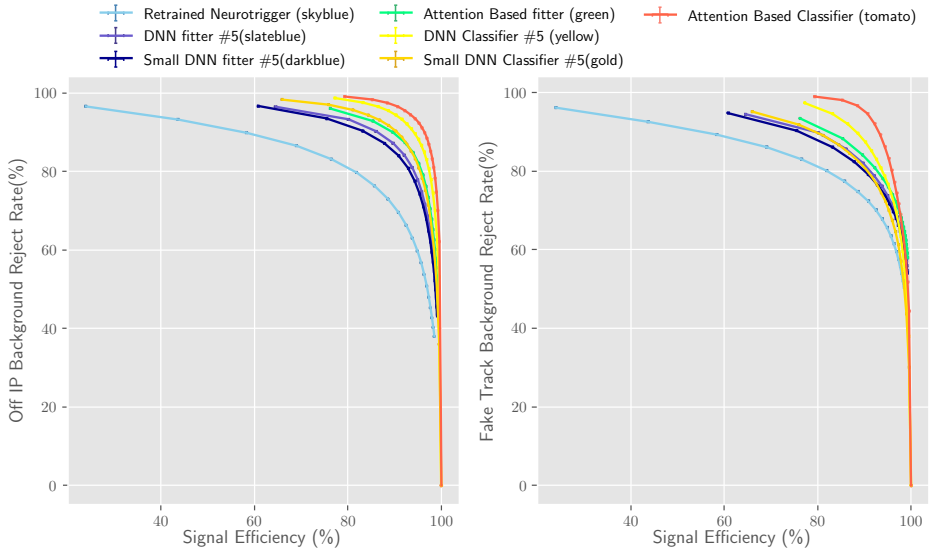


Figure 8.24: ROC curve of Signal Events Efficiency and Off-IP Background Events Rejection Rate (Left); Signal Events Efficiency and Fake Track Background Events (Right)

clear that the Attention based classifier got the best performance, which can improve the Background Reject Rate from 59.6% to 88.4% comparing with origin Neurotrigger. Assuming the same fraction and trigger rate as we showed in Tab. 6.3,6.4, we can reduce the CDCTRG $B\bar{B}$ raw trigger rate by 1.5 kHz and CDCTRG low-multi raw trigger by 1.4 kHz. And the classifier always got a better performance rather than the fitter even with the small one.

8.5 Discussion for firmware implementation

We plan to implement our CDC Neural Network (NN) triggers on the UT4 platform utilizing the Virtex UltraScale 7 XCVU160 FPGA, which offers a fourfold increase in logic gate capacity compared to the UT3 platform. When considering the dominant resource usage of the FPGA, namely the digital signal processing (DSP) capability, it is important to note that the previous Neurotrigger implementation accounted for 44% of the UT3 DSP resources and 30 % LUT resources [40], with a total of 2,349 free parameters.

Currently, the best-performing Deep Neural Network (DNN) fitter, DNN classifier,

Type	Selection	Signal Efficiency (%)	Background Efficiency (%)	Depth	Free Param.
Neurotrigger	$\lvert z_0^{\mathrm{NN}}\rvert < 13$cm	95.6	40.4	3	2,349
DNN fitter	$\lvert z_0^{\mathrm{NN}}\rvert < 8$cm	95.8	26.5	3	99,066
Small DNN fitter	$\lvert z_0^{\mathrm{NN}}\rvert < 8$cm	95.4	26.9	3	13,386
Attention Based Fitter	$\lvert z_0^{\mathrm{NN}}\rvert < 6$cm	95.2	20.6	3	27,621
DNN Classifier	$p < 51\%$	95.2	17.1	3	60,160
Small DNN Classifier	$p < 60\%$	95.2	23.8	1	4,093
Attention Based Classifier	$p < 40\%$	95.2	11.6	3	27,540

Table 8.7: Selection condition, Signal Efficiency and Background Efficiency (1 − background rejection rate) comparison of Neurotrigger, (small) DNN fitter & classifier and Attention Based fitter & Classifier. Manually set integer selection condition to keep efficiency above 95% for every module. Free param. includes all the weights in the NN which should be recorded in FPGA.

Attention-Based fitter, and Attention-Based classifier have a significantly larger number of free parameters, specifically 99,066, 60,160, 27,621, and 27,540, respectively. These structures can not be directly implementable on the UT4 platform. However, the small DNN fitter (Classifier) with only 13,386 (4,093) free parameters, is feasible for implementation on the current UT4 platform without any modification and also improve the background events rejection from 59.6% to 76.2%.

Moreover, the Attention-Based fitter/classifier, which is approximately 10 times larger than the current structure, may also be optimized and pruned to reduce its overall size, and subsequently implemented on the UT4 platform. Looking towards the future upgrade of UT5, which offers a further fourfold improvement in capacity, all of these logic implementations become feasible and can be accommodated.

Conclusion

The Belle II Experiment, situated at the SuperKEKB asymmetric electron-positron collider in Japan, is the next generation B-factory, aiming to explore new physics (NP) in the flavor sector at the intensity frontier and enhance the precision of measurements for Standard Model (SM) parameters. SuperKEKB is expected to achieve the highest luminosity in the world, reaching $6 \times 10^{35}, \mathrm{cm}^{-2}\mathrm{s}^{-1}$, enabling unprecedented precision in NP searches and measurements of the CKM matrix. However, two obstacles hinder the increase in luminosity at present. One is sudden beam loss events which prevent for reaching higher bunch current and cause severe damaged to the collimators and detectors. Another one is that the increasing level-1 trigger rate will soon reach designed limitation and background trigger contribute a large part of.

To address this sudden beam loss, we have installed fast loss monitors along the SuperKEKB main ring, supplemented by existing loss monitors and beam monitors. Through detailed timing analysis, we have identified the possible location for sudden beam loss. Under nominal collimator settings, the earliest loss was observed in the LER D06 section, indicating the occurrence of initial beam instability in or upstream of the D06 section. Notably, after opening the D06 collimator due to severe damage, the earliest loss was observed at the D02 collimator. Precursor phenomena such as beam size blowup or beam orbit deviation were rarely observed before the earliest beam loss. Although the cause of sudden beam loss events is not fully understood, we plan to implement countermeasures such as fast beam abort and additional sensors at the D06 section to issue abort requests and obtain multi-angle beam information. These measures aim to safeguard our detectors and collimators from sudden beam loss.

Regarding the level-1 CDC trigger, we have developed three different architectures for a new neural-network trigger. Through software simulations, we examined the

performance of each architecture, including the introduction of extra input features, pre-selection methods, and changes to the architecture itself. The results confirmed that the new architectures led to an 80% improvement in off IP background track reject rate while maintaining the same efficiency as the original Neurotrigger architecture. This improvement is expected to reduce the total raw CDC trigger background by more than 2 kHz. Moving forward, we will continue working on simplifying the architecture and implementing it in the UT4 modules to achieve practical functionality.

Bibliography

[1] Nicola Cabibbo. Unitary symmetry and leptonic decays. *Phys. Rev. Lett.*, 10:531–533, Jun 1963.

[2] Makoto Kobayashi and Toshihide Maskawa. CP-Violation in the Renormalizable Theory of Weak Interaction. *Progress of Theoretical Physics*, 49(2):652–657, 02 1973.

[3] E Kou et al. The belle II physics book. *Progress of Theoretical and Experimental Physics*, 2019(12), dec 2019.

[4] Kazunori Akai, Kazuro Furukawa, and Haruyo Koiso. SuperKEKB collider. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 907:188–199, nov 2018.

[5] Luminosity projection. https://www-superkekb.kek.jp/Luminosity_projection.html.

[6] M. Bona et al. SuperB: A High-Luminosity Asymmetric e+ e- Super Flavor Factory. Conceptual Design Report. 5 2007.

[7] N. Ohuchi, Y. Arimoto, et al. Superkekb beam final focus superconducting magnet system. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 1021:165930, 2022.

[8] T. Ishibashi, S. Terui, Y. Suetsugu, K. Watanabe, and M. Shirai. Movable collimator system for SuperKEKB. *Phys. Rev. Accel. Beams*, 23(5):053501, 2020.

[9] Vishay. Silicon pin photodiode bpw34.

[10] Hitomi Ikeda, Mitsuhiro Arinaga, John Walter Flanagan, Hitoshi Fukuma, and M. Tobiyama. Beam loss monitor at superkekb. In *IBIC2014*, 2015.

[11] H. Fukuma and H.Ikeda. Lecture notes in introduction to accelerators ii, January 2022.

[12] Makoto Tobiyama, John W. Flanagan, and Alessandro Drago. Bunch by bunch feedback systems for superkekb rings. Proceedings of the 13th annual meeting of Particle Accelerator Society of Japan, page 1420, Japan, 2016. PARTICLE ACCELERATORS.

[13] S. Bacher et al. Performance of the diamond-based beam-loss monitor system of Belle II. *Nucl. Instrum. Meth. A*, 997:165157, 2021.

[14] Toshihiro Mimashi, Naoko Iida, Mitsuo Kikuchi, Takashi Mori, Kazuhiko Abe, Atsushi Sasagawa, and Akira Tokuchi. SuperKEKB Beam abort System. In *5th International Particle Accelerator Conference*, page MOPRO023, 7 2014.

[15] Belle ii experiment main page. https://belle2.jp/.

[16] Tomohisa Uchida, Masahiro Ikeno, Yoshihito Iwasaki, Masatoshi Saito, Shoichi Shimazaki, Manobu Tanaka, Nanae Taniguchi, and Shoji Uno. Readout electronics for the central drift chamber of the belle ii detector. In *2011 IEEE Nuclear Science Symposium Conference Record*, pages 694–698, 2011.

[17] Henrikas Svidras. The Central Drift Chamber of Belle 2. In *5th Belle II Starterkit Workshop*, January 2020.

[18] Sara Pohl. *Track Reconstruction at the First Level Trigger of the Belle II Experiment.* PhD thesis, Munich U., 2017.

[19] T. Abe, I. Adachi, et al. Belle ii technical design report, 2010.

[20] Yoshihito Iwasaki, ByungGu Cheon, Eunil Won, Xin Gao, Luca Macchiarulo, Kurtis Nishimura, and Gary Varner. Level 1 trigger system for the belle ii experiment. *IEEE Transactions on Nuclear Science*, 58(4):1807–1815, 2011.

[21] Kay Wittenburg. Beam loss monitoring and control. 06 2002.

[22] Takuya ISHIBASHI. Collimator plans during ls1. In *MDI taskforce meeting*, 7 2022.

[23] Y Ashida, M Friend, A K Ichikawa, T Ishida, H Kubo, K G Nakamura, K Sakashita, and W Uno. A new electron-multiplier-tube-based beam monitor for muon monitoring at the T2K experiment. *Progress of Theoretical and Experimental Physics*, 2018(10), 10 2018. 103H01.

[24] Takashi Honjou. T2k実験ミューオンモニターに用いる電子増倍管の放射線耐性試験(1)：試験の概要. In 日本物理学会*2021*年春季大会, 3 2021.

[25] Hamamatsu. Metal package photomultiplier tube r9880u series. 8 2022.

[26] Hamamatsu. D-type socket assembly e10679-51. 2019.

[27] White rabbit. http://www.ohwr.org/projects/white-rabbit.

[28] Simple pcie fmc carrier. https://www.ohwr.org/project/spec/wikis/home.

[29] Fmc-dio. https://www.ohwr.org/project/fmc-dio-5chttla/wikis/home.

[30] Y. Funakoshi. Personal view on large beam loss events. In *Beam abort meeting*, 5 2022.

[31] S. Terui, Yoshihiro Funakoshi, Hiromi Hisamatsu, Takuya Ishibashi, Ken ichi Kanazawa, Y. Ohnishi, K. Shibata, Mitsuru Shirai, Yusuke Suetsugu, and M. Tobiyama. Report on collimator damaged event in superkekb. 2021.

[32] Tetsuo ABE. A fireball hypothesis to explain the trigger of the catastrophic beam-loss events. In *Beam abort meeting*, 5 2022.

[33] Kazuki Kitamura. Superkekb 加速器におけるビームアボート高速化に向けた基礎研究. In *JPS*, 3 2023.

[34] Hiroshi Kaji. New ler abort system by mdi. In *MDI taskforce meeting*, 2 2023.

[35] Y.-S. Teng, C.-H. Wang, S.-M. Liu, J.-G. Shiu, Y.-T. Lai, and C.-S. Lin. The status of high-speed trigger multiplexer module with aurora protocol implemented on arria ii fpga for the belle ii cylindrical drift chamber detector. In *2013 IEEE Nuclear Science Symposium and Medical Imaging Conference (2013 NSS/MIC)*, pages 1–3, 2013.

[36] Ping Ni. *Upgrade of Two Dimensional Track Trigger on Central Drift Chamber aimed for Belle II Targeted Luminosity*. PhD thesis, Tokyo, University of Tokyo, Tokyo, 2022. Presented on 09 08 2022.

[37] Yuki Sue, Bae Hanwook, Toru Iijima, Yoshihito Iwasaki, Taichiro Koga, Yun-Tsung Lai, Hideyuki Nakazawa, and Kai Lukas Unger. The Event Timing Finder for the Central Drift Chamber Level-1 Trigger at the Belle II experiment. *J. Phys. Conf. Ser.*, 2374(1):012103, 2022.

[38] T. Kuhr, C. Pulvermacher, M. Ritter, T. Hauth, and N. Braun. The Belle II Core Software. *Comput. Softw. Big Sci.*, 3(1):1, 2019.

[39] Adam Paszke et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[40] Steffen Baehr, Sara McCarney, Felix Meggendorfer, Julian Poehler, Sebastian Skambraks, Kai Unger, Juergen Becker, and Christian Kiesling. Low latency neural networks using heterogenous resources on fpga for the belle ii trigger, 2019.

[41] C. Kiesling. Neural track trigger in 2022. In *B2GM*, 10 2022.

[42] Søren Nielsen. 1. statistical analysis with missing data (2nd edn). roderick j. little and donald b. rubin, john wiley sons, new york, 2002. no. of pages: xv+381. isbn: 0-471-18386-5. *Statistics in Medicine - STAT MED*, 23:1181–1181, 04 2004.

[43] Hiroto Sudo. Update for 3d fitter in cdctrg. In *B2GM*, 2 2023.

[44] Christian Janiesch, Patrick Zschech, and Kai Heinrich. Machine learning and deep learning. *Electronic Markets*, 31(3):685–695, Sep 2021.

[45] Pramila P. Shinde and Seema Shah. A review of machine learning and deep learning applications. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pages 1–6, 2018.

[46] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[47] Yen-Pin Chen, Chien-Hua Huang, Yuan-Hsun Lo, Yi-Ying Chen, and Feipei Lai. Combining attention with spectrum to handle missing values on time series data without imputation. *Information Sciences*, 609:1271–1287, 2022.

[48] Sagar Sharma, Simone Sharma, and Anidhya Athaiya. Activation functions in neural networks. *Towards Data Sci*, 6(12):310–316, 2017.

[49] M. Riedmiller and H. Braun. A direct adaptive method for faster backpropagation learning: the rprop algorithm. In *IEEE International Conference on Neural Networks*, pages 586–591 vol.1, 1993.

[50] Ning Qian. On the momentum term in gradient descent learning algorithms. *Neural Networks*, 12(1):145–151, 1999.

[51] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[52] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019.

[53] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[54] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

[55] Akane Maeda. Superkekbビームロスモニター開発のためのテストビームラインを用いたemtの検出効率の評価. In *JPS*, 3 2023.

A

Beam loss timing table

Table A.1: Summary of the sudden beam loss events from February to July 2022 with the measured beam loss timing on each sensor. The sensor with the fastest timing is written in the rightest column. Radiation does at the diamond, amount of beam loss at BCM, and if QCS is quenched are shown to explain size of the beam loss.

Date	Diamond does(mrad)	BCM loss(%)	QCS quench	Sensor timing (ΔT, μs)						Fastest Sensor
				BCM	D06H3	D6V1	D06V2	D02V1	Diamond	
2022/2/28-3/13										
2022/3/06 18:35	15	12.68%		-6.07	-	-32.1	-20.29	-26.94	3.1	D06V1 PMT
2022/3/11 7:09	-	14.79%		-6.09	-	-5.86	-29.16	-11.85	5.6	D06V2 PMT
2022/3/11 10:08	544	17.88%	○	-7.5	-	-8.51	-8.40	-15.54	-6.9	D02V1 PMT
2022/3/14-3/27										
2022/3/15 3:18	-	5.90%		0.96	-	-19.46	-19.22	-11.85	15.6	D06V1 PMT
2022/3/20 7:39	-	0.66%		1.46	-	-14.12	-13.63	-	-	D06V1 PMT
2022/3/21 10:36	-	0.56%		-4.63	-	-22.33	-21.89	-	57.1	D06V1 PMT
2022/3/22 8:15	-	6.89%		-5.76	-	-35.3	-34.75	-	-1.9	D06V1 PMT
2022/3/23 13:49	9	12.09%		-6.84	-	-14.91	-14.48	-	-1.9	D06V1 PMT
2022/3/23 22:20	12	59.36%		-1	-3.71	-12.08	-9.2	-1.78	8.1	D06V1 PMT
2022/3/27 10:34	106	8.03%		-4.24	-	-15.26	-15.25	-11.85	-9.4	D06v2 pMT
2022/3/28-4/10										
2022/4/03 17:14	37	1.52%		2.58	-	-20.12	-20.15	-16.88	5.6	D06V2 PMT
2022/4/07 0:16	40	4.10%		-12.51	-	-	-	-16.87	-11.9	D02V1 PMT
2022/4/08 8:52	17	1.86%		-6.96	-	-	-	-11.57	3.1	BOR HOR
2022/4/08 11:55	91	2.17%	○	-2.47	-	-	-	-11.85	-9.4	D02V1 PMT
2022/4/11-4/24										
2022/4/11 12:09	139	7.95%		-3.72	-	-12.11	-12.2	-8.61	-4.4	D06V2 PMT
2022/4/15 23:30	51	6.65%		0.22	-	-8.94	-13.43	-9.91	3.1	D06V2 PMT
2022/4/16 5:02	-	0.46%		-3.76	-	-13.01	-13.21	-9.47	-4.4	D06V2 PMT
2022/4/19 0:58	42	3.20%		2.57	-	-7.96	-14.89	-8.49	-1.9	D06V2 PMT
2022/4/20 23:36	41	4.81%		-5.87	-	-15.26	-15.69	-11.84	0.6	D06V2 PMT
2022/4/21 11:08	14	0.94%		-7.11	-	-15.94	-16.14	-11.84	-11.9	D06V2 PMT
2022/4/25-5/8										
2022/4/28 17:25	95	13.42%		-2.4	-	-	-24.76	-9.3	-16.9	D06V2 EMT
2022/4/29 5:12	111	13.05%		1.75	-	-10.82	-10.49	-12.66	-6.9	D02V1 PMT

Date	Diamond	BCM	QCS	Sensor timing ($\Delta T, \mu s$)						Fastest
	does(mrad)	loss(%)	quench	BCM	D06H3	D6V1	D06V2	D02V1	Diamond	Sensor
2022/5/02 1:19	57	4.84%		-5.48	-	-13.54	-13.32	-10.19	-9.4	D06V1 PMT
2022/5/03 3:02	12	1.14%		11.29	-	0.73	-10.99	5.18	15.6	D06V2 EMT
2022/5/07 8:40	55	19.18%		2.81	-	-18.73	-18.47	-5.08	-1.9	D06V1 PMT
2022/5/07 14:17	107	5.47%		-7.49	-	-19.24	-19.27	-13	-14.4	D06V2 EMT
2022/5/08 2:44	51	3.75%		-4.33	-	-16.9	-18.16	-11.84	-6.9	BOR HOR
2022/5/9-5/22										
2022/5/10 15:56	-	0.54%		-7.48	-	-16.66	-16.68	-13.15	0.6	D06V2 EMT
2022/5/10 23:01	175	44.88%		-6.33	-	-15.26	-14.92	-11.1	-9.4	D06V1 PMT
2022/5/11 1:58	19	0.54%		-4.89	-	-13.26	-13.22	-10.01	-6.9	D06V1 PMT
2022/5/11 8:00	53	4.30%		-6.09	-	-14.03	-13.91	-10.68	-4.4	D06V1 PMT
2022/5/13 13:48	27	1.18%		-5.83	-	-14.25	-14.47	-10.95	0.6	D06V2 EMT
2022/5/14 13:21	618	17.24%		-0.61	-	-8.46	-9.8	-5.06	0.6	BOR HOR
2022/5/17 14:38	471	54.39%	○	-7.24	-	-10.05	-10.10	-15.63	-14.4	D02V1 PMT
2022/5/23-6/5										
2022/5/28 5:11	48	5.63%		0.39	-	-7.82	-7.58	-	-11.9	D06V1 PMT
2022/5/28 10:13	48	3.87%		-8.25	-	-16.78	-22.01	-	-16.9	D06V2 EMT
2022/6/01 4:47	96	18.42%		-7.39	-	-17.37	-17.47	-11.84	-9.4	D06V2 EMT
2022/6/01 22:05	739	28.50%	○	-4.25	-	-12.04	11.83	-8.97	-1.9	BOR VER
2022/6/02 22:55	-	0.95%		-7.9	-	-16.06	-16	-12.71	-9.4	D06V1 PMT
2022/6/3 14:48	1018	53.16%	○	-10.31	4.98	-10.23	-9.9	-13.95	-9.4	D02V1 PMT
2022/6/04 16:35	158	7.15%		-0.6	-	-5.19	-14.56	-11.51	3.1	D06V2 EMT
2022/6/05 5:31	229	37.40%		-2.4	-	-6.22	-15.92	-11.44	0.6	D06V2 EMT
2022/6/05 1:59	436	7.88%		-4.41	7.34	-12.26	-12.15	-8.94	-9.4	D06V1 PMT
2022/6/6-6/19										
2022/6/08 17:31	341	28.71%		-6.74	-	-14.86	-15.14	-11.85	-11.9	D06V2 EMT
2022/6/08 23:49	161	5.54%		-3.9	-	-3.22	-12.33	-9.38	-6.9	D06V2 EMT
2022/6/09 0:37	644	6.70%	○	-9.93	4.38	-8.47	-7.96	-14.96	-9.4	D02V1 PMT
2022/6/09 4:26	1249	24.57%	○	-9.53	4.37	-8.02	-7.83	-13.76	0.6	D02V1 PMT
2022/6/10 15:15	61	3.30%		-2.88	-	-15.25	-14.91	-9.11	-9.4	D06V1 PMT
2022/6/10 21:44	42	1.18%		2.04	-	-6.65	-14.91	-11.85	8.1	D06V2 EMT
2022/6/13 8:47	69	1.25%		-7.48	4.39	-15.7	-16.41	-13.35	-11.9	D06V2 EMT
2022/6/14 12:44	130	4.12%		-2.46	-	-11.02	-19.93	-16.87	-9.4	D06V2 EMT
2022/6/14 14:34	1056	9.03%	○	-8.2	4.37	-7.2	-6.88	-13.64	0.6	D02V1 PMT
2022/6/16 2:01	108	2.61%		-11.54	-	-25.3	-25.46	1.43	-9.4	D06V2 EMT
2022/6/16 22:18	70	3.08%		9.95	-	-15.22	-16.53	-12.65	-9.4	BOR HOR

Date	Diamond does(mrad)	BCM loss(%)	QCS quench	Sensor timing ($\Delta T, \mu s$)						Fastest Sensor
				BCM	D06H3	D6V1	D06V2	D02V1	Diamond	
2022/6/18 20:32	198	19.76%		-1.24	-	-9.02	-8.76	-5.62	-6.9	D06V1 PMT
2022/6/20-6/22										
2022/6/20 11:28	195	6.15%		-2.46	4.4	-10.55	-10.66	-7.38	3.1	D06V2 EMT
2022/6/22 8:39	306	7.97%		-5.87	4.39	-13.68	-74.32	-10.86	-9.4	D06V2 EMT

B

EMT amplitude and Efficiency

Define secondary emission efficiency for each dynodes as δ_i, where i is the number of layers. δ_i should be different for different layers since it depends on the particle energy. And Δ for the second secondary emission efficiency for the aluminum cathode. We could write the gain of EMT as:

$$G = \Delta \cdot \delta_0 \cdot \delta_1 \cdot ... \cdot \delta_n = \Delta \prod_{i=1}^{n} \delta_i; \tag{B.1}$$

And the amplitude of signal can be write as:

$$Q = e \cdot \phi_e \{A_{sur} \cdot \Delta \cdot \prod_{i=1}^{n} \delta_i + \sum_{i=1}^{n-1}; (A_i \cdot \delta_i \cdot \prod_{j=i+1}^{n} \delta_i)\} \tag{B.2}$$

where A_{sur} and $A_i$is the area of surface for aluminum cathode and each dynodes. ϕ_e for is the electron flux [cm^2]. First term is from secondary emission at aluminum cathode and second from each dynodes.

We have made a beam test with KEK Accelerate Ring Test Beamline to evaluated the detect efficiency of the EMTs and the result as[55] : $P = 0.3\%$ Where P define as :

$$P = \frac{\#signal}{\#Input\ Electron \times \Delta \times Acceptance} \tag{B.3}$$