Low-Latency On-Chip τ Event Classification with Machine Learning for the Belle II Level-1 Trigger

(Belle II 実験用レベル 1 トリガーのための機械学習を用いた低遅延な FPGA 上のタウ事象選別アルゴリズムの開発)

Deven Misra

A Master's Thesis
修士論文

Submitted to
The University of Tokyo
Department of Physics, Graduate School of Science
August 4th, 2026

Thesis Advisor: Takeo Higuchi 樋口岳雄

Abstract

Belle II is a luminosity frontier experiment located at the SuperKEKB e^+e^- collider, focusing on the study of B, D, and τ physics under the broad umbrella of "flavor physics". The Belle II experiment is expected to continue operation through 2039, over which time the luminosity at SuperKEKB will increase from 5.2×10^{34} cm^{-2}s^{-1} to 6.0×10^{35} cm^{-2}s^{-1}. Belle II relies on a hardware-based "Level-1" trigger system for real-time event selection during operation, which filters events in order to limit the overall data rate. While the existing event-selection algorithms are sufficient to achieve the requisite signal efficiency on standard model τ decays under the current accelerator conditions, the the expected increase in luminosity will require the development of new trigger algorithms to maintain a total trigger rate below the < 30 kHz limit of Belle II's Data Acquisition System. In this thesis, we present a low-latency machine learning-based τ event selection algorithm implemented on an AMD Xilinx UltraScale FGPA as part of the Belle II Level-1 trigger. We show that this new algorithm consistently outperforms existing trigger bits on low-multiplicity standard model τ decays, with signal efficiencies reaching 97% across all modes at a luminosity of 6.0×10^{35} cm^{-2}s^{-1} while remaining below the 30 kHz limit. This new trigger bit has been fully enabled for Belle II physics runs from Summer 2026 onward. Performance of the new firmware logic is consistent with results from simulations, while further direct evaluation of the trigger behavior during physics runs also suggests several directions for ongoing study and refinement of the neural network algorithm.

Contents

1 Introduction

The Standard Model of particle physics is one of the crowning achievements of 20th century physics, unifying the fundamental forces (with the notable exception of gravity) and providing an exhaustive account of the quantum particles and fields which constitute the natural world at the smallest scales. The fundamental particles identified by the Standard Model are broadly classified into quarks, leptons, and gauge bosons. Leptons are those particles with half-integer spins and do not couple to the strong interaction: the electron ($e^{\pm}$), muon ($\mu^{\pm}$), and tau ($\tau^{\pm}$), along with their associated neutrinos. The τ lepton is the heaviest among these, and consequently has the shortest lifetime and the greatest number of different decay modes. As a result, τ decays provide a powerful probe of physics beyond the Standard Model due to the comparatively large region of the kinematic phase space they occupy (compared to e or μ decays). The most common τ decay modes predicted by the Standard Model are given in Table 1 along with their associated (inclusive) branching ratios [1].

Table 1: Common decay modes for the Standard Model τ lepton

Decay Mode	**Branching Ratio**
$\tau^- \to \pi^-(\geq 0\pi^0)\nu_\tau$	$\sim 46.6\%$
$\tau^- \to e^-\bar{\nu}_e\nu_\tau$	$\sim 17.8\%$
$\tau^- \to \mu^-\bar{\nu}_\mu\nu_\tau$	$\sim 17.4\%$
$\tau^- \to \pi^-\pi^+\pi^-(\geq 0\pi^0)\nu_\tau$	$\sim 13.6\%$
$\tau^- \to K^-(\geq 0\pi^0)\nu_\tau$	$\sim 1.2\%$

Belle II is an experiment located at the SuperKEKB collider which focuses on B, D, and τ physics. As a luminosity frontier experiment, Belle II relies on high-efficiency hardware triggers to limit systematic uncertainties and maximize statistical power. The first collisions occurred in April 2018 and operation is expected to continue until at least ∼2040, collecting a total of 50 ab^{-1} of data. In this time, the accelerator luminosity will increase by over an order of magnitude. As a result, the trigger algorithms utilized for real-time event selection will require substantial improvements to their background rejection rates while maintaining or improving on the current efficiencies for $B\bar{B}$ and $\tau^+\tau^-$ decays.

In this thesis, we develop a new machine learning-based algorithm for low-latency τ decay mode classification and event selection at Belle II, focusing on low-multiplicity (1- and 3-prong) Standard Model τ decays. In Section 2, we discuss τ physics in the context of the Standard Model, highlighting key measurements and probes of new physics involving these decays. Section 3 introduces the Belle II experiment and the SuperKEKB collider, including detailed descriptions of the hardware-based Level-1 trigger system. In Section 4, we provide a broad overview of machine learning and its applications in particle physics, focusing on feed-forward neural networks and hardware acceleration.

Section 5 explicitly defines the signal we will attempt to reconstruct and introduces the new machine learning-based algorithm for τ event classification. Section 5 also includes an overview of the methods used for compression and training of the neural network, along with performance evaluation and validation of the trained model. In section 6, the on-chip deployment of the neural network is discussed, with an emphasis on the high-level synthesis algorithms used to generate firmware for the hardware target. Section 7 evaluates the performance of the newly implemented trigger algorithm during a Belle II physics run. Finally, in Section 8, we summarize the primary achievements of the thesis and present several directions for future study and development.

2 Tau Physics

2.1 Lepton Flavor Universality

In the Standard Model, the electroweak gauge couplings are universal across all generations of leptons, giving rise to an accidental global flavor symmetry. Under this symmetry, interchanging two lepton fields results in identical gauge dynamics, meaning that any differences in the observable physics arises solely from phase space and kinematic constraints imposed by their differing masses. More generally, any unitary combination of two particles which couple to the same gauge fields should share identical gauge couplings. In practice, confirmation of lepton flavor-universality (LFU) requires precision measurements of the branching ratios associated with the decays of leptons in order to determine the gauge couplings of each generation. If this symmetry is observed to be broken in the charged-lepton sector, it would be an unambiguous indication of the existence of new physics beyond the standard model. Recent measurements of the branching ratios $\mathcal{B}(\tau^- \to \mu^- \bar{\nu}_\mu \nu_\tau)$ and $\mathcal{B}(\tau^- \to e^- \bar{\nu}_e \nu_\tau)$ at Belle II have provided a stringent test of LFU in the 1×1 topology, finding that

$$R_\mu = \frac{\mathcal{B}(\tau^- \to \mu^- \bar{\nu}_\mu \nu_\tau)}{\mathcal{B}(\tau^- \to e^- \bar{\nu}_e \nu_\tau)} = 0.9675 \pm 0.0007 \pm 0.0036, \tag{1}$$

and thus that

$$|g_\mu / g_e|_\tau = 0.9974 \pm 0.0019, \tag{2}$$

where g_μ and g_e are the respective coupling strengths of the μ^- and e^- to the W boson [2].

2.2 Lepton Flavor Violation

The Standard Model of particle physics also requires that the *total number* of leptons of each flavor is conserved in all interactions. While this symmetry is assumed to hold in the Standard Model, many theoretical extensions of the Standard Model include the potential for decays which violate this symmetry. The existence of so-called lepton flavor-violation (LFV) in τ decays is one of the key experimental observations which would establish the existence of physics beyond the standard model. There are a wide variety of modes which have been hypothesized depending on the specific theoretical model in question, however some LFV decays appear across a wide range of theories. Some of the most commonly studied among these include $\tau \to e\gamma$, $\tau \to \mu\gamma$, and $\tau \to 3\mu$. The existing statistical limits on several of the most studied LFV decays are shown in Fig. 1.

2.3 CP Violation

In the context of the Standard Model, both charge-conjugation (C) and parity (P) symmetries are maximally violated by the weak interaction due to the chiral coupling of the W boson to *only* left-handed fermions (and right-handed anti-fermions). Consequently, neither C nor P are good quantum numbers for weak interactions. CP symmetry was originally introduced by Landau in 1957 [4], restoring a modified form of spatial reflection invariance by mapping left-handed fermions to right-handed anti-fermions:

$$CP : \psi_L \to (\psi^c)_R \tag{3}$$

From the perspective of the Lagrangian, exact CP symmetry requires all coupling constants to be real. Therefore, any complex phase in the Lagrangian which can not be eliminated via field redefinition implies explicit CP violation. In the Standard Model, these phases originate from the quark (CKM) and lepton (PMNS) mixing matrices, as well as the non-perturbative QCD θ-term.

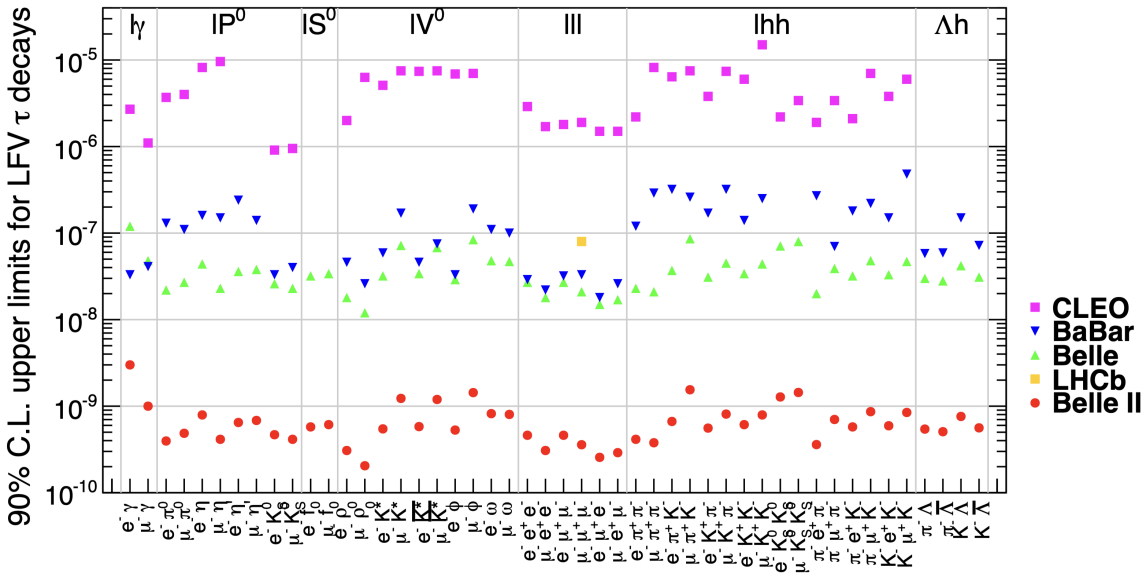


Figure 1: 90% confidence-interval upper bounds for τ LFV branching ratios [3]

2.3.1 CP Violation in QCD

The QCD Lagrangian includes a topologically non-trivial term,

$$\mathcal{L}_\theta = \theta \frac{g_s^2}{32\pi^2} G^a_{\mu\nu} \tilde{G}^{a\mu\nu}. \tag{4}$$

where $G^a_{\mu\nu}\tilde{G}^{a\mu\nu}$ is the product of the gluon field strength tensor and its dual. Under a time-reversal (T) transformation, this operator changes sign:

$$T : G^a_{\mu\nu}\tilde{G}^{a\mu\nu} \to -G^a_{\mu\nu}\tilde{G}^{a\mu\nu} \tag{5}$$

However, CPT is an exact symmetry in the Standard Model, therefore this term must violate CP symmetry. In practice, experimental measurements of the electric dipole moment of the neutron have placed stringent restrictions on the magnitude of the effective observable phase $\bar{\theta} \equiv \theta + \arg\det(M_q) < 10^{-10}$ where M_q is the quark mass matrix, strongly suggesting that QCD does *not* violate CP-symmetry [5][6]. This apparent fine-tuning is known as the "strong" CP problem.

2.3.2 CP Violation in the Weak Sector

The weak force operates directly on the weak flavor eigenstates. For the down-type quarks, these are represented as:

$$\psi_d^I = \begin{pmatrix} d' \\ s' \\ b' \end{pmatrix}. \tag{6}$$

The weak eigenstates are linear combinations of the physical mass eigenstates,

$$\psi_d^m = \begin{pmatrix} d \\ s \\ b \end{pmatrix}. \tag{7}$$

The relationship between the two is governed by the unitary *Cabibbo-Kobayashi-Maskawa* (CKM) matrix V_{CKM}:

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = V_{CKM} \begin{pmatrix} d \\ s \\ b \end{pmatrix}. \tag{8}$$

The CKM matrix has the general form

$$V_{\text{CKM}} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix}, \tag{9}$$

which is frequently approximated via the *Wolfenstein parameterization* to highlight its hierarchical structure. To leading order, this can be written as

$$V_{\text{CKM}} = \begin{pmatrix} 1 - \frac{\lambda^2}{2} & \lambda & A\lambda^3(\rho - i\eta) \\ -\lambda & 1 - \frac{\lambda^2}{2} & A\lambda^2 \\ A\lambda^3(1 - \rho - i\eta) & -A\lambda^2 & 1 \end{pmatrix} + \mathcal{O}(\lambda^4). \tag{10}$$

In this form, the dependence on the four parameters λ, A, ρ, and η is made explicit, with $\eta \neq 0$ permitting the existence of a CP-violating phase. For three generations of quarks, V_{CKM} necessarily possesses a single irreducible, CP-violating complex phase, and experimental measurements of η confirm the existence of CP-violation in the quark sector with $\bar{\eta} \approx 0.35$ [7][8].

Since the CKM matrix must be unitary,

$$V_{\text{CKM}} V_{\text{CKM}}^{\dagger} = \mathbb{I}, \tag{11}$$

it follows that its elements must satisfy the orthogonality condition

$$V_{ud}V_{ub}^{*} + V_{cd}V_{cb}^{*} + V_{td}V_{tb}^{*} = 0. \tag{12}$$

This condition can be represented geometrically in the complex plane as the so-called CKM *unitarity triangle* (Fig. 2). The lengths of the sides are determined by products of the magnitudes of the

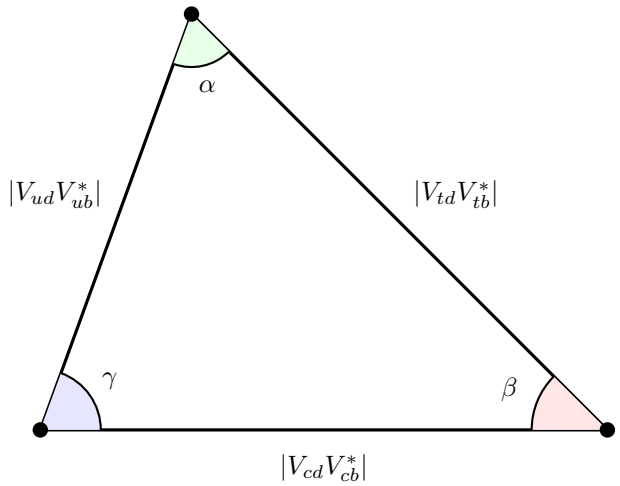


Figure 2: The CKM unitarity triangle representing the orthogonality condition (Eq. 12)

elements of the CKM matrix, while the angles α, β, and γ measure the CP-violating phase. Over the last several decades, a variety of experiments have made precise measurements of the magnitudes of the elements of the CKM matrix (i.e. the sides and angles of the CKM unitarity triangle) [9–12].

2.3.3 CP Violation in the Lepton Sector

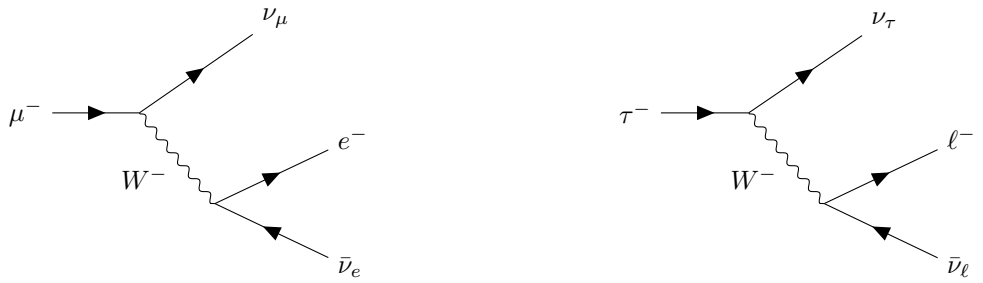
The Pontecorvo – Maki – Nakagawa – Sakata (PMNS) matrix is the analogue of the CKM matrix for the lepton sector, characterizing the mixing of the neutrino flavor and mass eigenstates,

$$\begin{pmatrix} \nu_e \\ \nu_\mu \\ \nu_\tau \end{pmatrix} = U_{PMNS} \begin{pmatrix} \nu_1 \\ \nu_2 \\ \nu_3 \end{pmatrix}. \tag{13}$$

Modern long-baseline neutrino experiments have also recently measured the mixing angles for the PMNS matrix, finding strong evidence that CP symmetry is violated in the neutrino sector *if* the neutrino mass hierarchy is inverted [13]. However, the observed magnitude of CP violation from both the weak sector and neutrino oscillation remains insufficient to explain the observed baryon asymmetry in the universe [14][15]. Consequently, significant experimental and theoretical efforts remain dedicated to uncovering novel sources of CP violation. In 2012, the BaBar experiment observed anomalies which suggested CP violation in the $\tau^- \to \pi^- K_S^0 (\geq 0\pi^0)\nu_\tau$ decay may exceed that attributable to quark mixing [16]. While further measurements at Belle and Belle II have not confirmed these anomalies, observation of direct CP violation in τ decays would provide concrete evidence of physics beyond the Standard Model, and thus remains a priority at modern collider experiments such as Belle II.

2.4 Michel Parameters

The *Michel parameters* are a set of parameters which characterize the phase space distribution of the three-body decay of a heavy charged lepton into a lighter charged lepton and two neutrinos, mediated by a charged vector boson $W^\pm$ (Fig. 3). In the Standard Model, the structure of the weak



(a) Muon decay ($\mu^- \to e^- \bar{\nu}_e \nu_\mu$)

(b) Tau decay ($\tau^- \to \ell^- \bar{\nu}_\ell \nu_\tau$)

Figure 3: The tree-level Feynman diagrams for lepton decays characterized by the Michel parameters

interaction is $V - A$ (vector minus axial-vector), hence the W boson couples only to left-handed particles and right-handed antiparticles. This structure is a specific feature of the Standard Model rather than a fundamental property of any quantum field theory; the most general, Lorentz-invariant, local, derivative-free matrix element for lepton decays also allows for scalar (S), vector (V), tensor (T), axial-vector (A), and pseudo-scalar (P) couplings. The Michel parameters therefore must take on specific values in the Standard Model in order for the weak interaction to be a purely $V - A$ interaction. The four "main" Michel parameters are ρ, η, ξ, and δ. For the weak interaction to be purely $V - A$, these parameters must be exactly

$$\rho = \frac{3}{4}, \quad \eta = 0, \quad \xi = 1, \quad \text{and} \quad \delta = \frac{3}{4}. \tag{14}$$

Consequently, high-precision experimental measurements of the phase space distribution of these decays provide a robust test of the Standard Model. If any of these parameters deviates from the values required for a pure $V - A$ weak interaction, it would indicate the existence of new physics.

2.5 τ Electric Dipole Moment

The Standard Model makes specific predictions for the electric dipole moment (EDM) of each of the charged leptons. While the specific theoretical calculations for these quantities remain subject to some uncertainty, experimental bounds on the τ EDM remain significantly less restrictive than those for the other charged leptons. The ACME II experiment established strong bounds on the electron EDM d_e in 2018 [17],

$$|d_e| < 1.1 \times 10^{-29}\text{e cm } (CL = 90\%), \tag{15}$$

which were then further improved upon by the JILA collaboration in 2023 [18],

$$|d_e| < 4.1 \times 10^{-30}\text{e cm}. \tag{16}$$

The current limits on the muon EDM d_μ are less restrictive than those for d_e, and were set by the Brookhaven National Laboratory (BNL) E821 experiment in 2009 [19],

$$|d_\mu| < 1.8 \times 10^{-19}\text{e cm } (CL = 95\%). \tag{17}$$

On the other hand, the most stringent kinematic bounds on the τ EDM d_τ come from the Belle experiment, using measurements of the 1-prong τ decays $\tau \to \pi\nu_\tau$, $\tau \to \rho\nu_\tau$, $\tau \to e\bar{\nu}_e\nu_\tau$, and $\tau \to \mu\bar{\nu}_\mu\nu_\tau$. The resulting limits on d_τ were computed as [20]

$$\begin{gathered} -0.22 \times 10^{-16}\text{e cm} < \text{Re}(d_\tau) < 0.45 \times 10^{-16}\text{e cm} \\ \text{and} \\ -0.25 \times 10^{-16}\text{e cm} < \text{Im}(d_\tau) < 0.08 \times 10^{-16}\text{e cm}. \end{gathered} \tag{18}$$

Thus, precision measurements of d_τ at Belle II remain a key probe of the Standard Model.

3 Belle II

3.1 SuperKEKB

SuperKEKB (Fig. 4) is an asymmetric-energy e^+e^- collider (4 GeV e^+, 7 GeV e^-), operating at the $\Upsilon(4S)$ resonance to efficiently produce $B\bar{B}$ pairs. Collisions at $\Upsilon(4S)$ also generate a large number of τ leptons, enabling a robust τ physics program. At the $\Upsilon(4S)$ resonance, the cross-section for the

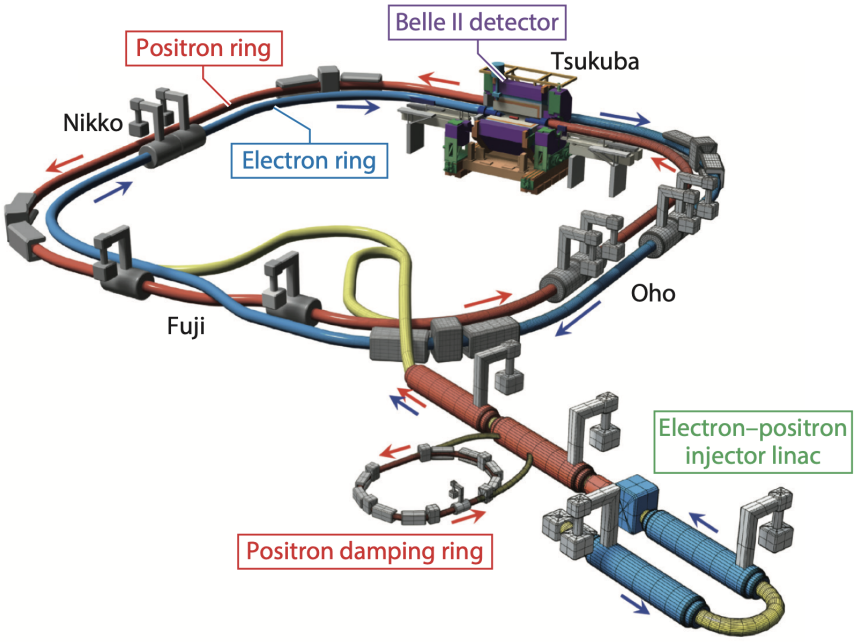


Figure 4: The SuperKEKB e^+e^- collider []

$e^+e^- \to \tau^+\tau^-$ process (Fig. 5) is $\sigma_{e^+e^-\to\tau\tau} = 0.919 \pm 0.03$ nb [21]. The previous Belle experiment collected a total of 711 fb^{-1} of data at $\Upsilon(4S)$, comprising approximately 6.5×10^8 τ pairs. In comparison, Belle II aims to collect at least 50 ab^{-1} of data at $\Upsilon(4S)$, or approximately 4.6×10^{10} τ pairs.

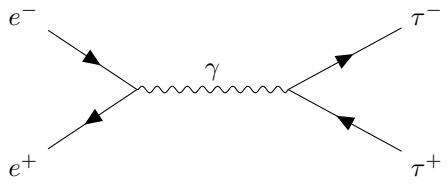


Figure 5: The dominant tree-level Feynman diagram for $e^+e^- \to \tau^+\tau^-$ at the $\Upsilon(4S)$ resonance

The SuperKEKB accelerator utilizes a unique nano-beam collision scheme [22] which has allowed it to achieve the world's highest instantaneous luminosity, measured at $\mathcal{L} = 5.2 \times 10^{34}$ cm^{-2}s^{-1} in March 2026. The *luminosity* of an accelerator relates the rate of collision events R to the interaction cross-section σ,

$$R = \mathcal{L}\sigma. \tag{19}$$

In a head-on collision between two bunches of particles with Gaussian spatial distributions,

$$\mathcal{L} = \frac{f_{cross} N_1 N_2}{4\pi\sigma_x\sigma_y} F, \tag{20}$$

where f_{cross} is the bunch crossing rate, N_1 and N_2 are the number of particles in each bunch, and σ_x and σ_y are the horizontal and vertical root-mean-square (RMS) beam sizes at the interaction point (IP).
The interaction cross-section σ can be decomposed into the *emittance* ϵ and the *beta function* β as

$$\sigma = \sqrt{\epsilon\beta}. \tag{21}$$

We denote quantities *at the IP* with a superscript *, hence the horizontal and vertical beam sizes σ_x and σ_y at the IP can be written as

$$\sigma_x^* = \sqrt{\epsilon^*\beta_x^*} \quad \text{and} \quad \sigma_y^* = \sqrt{\epsilon^*\beta_y^*}. \tag{22}$$

It immediately follows that the luminosity is related to the beta functions at the IP by

$$\mathcal{L} \propto \frac{1}{\sqrt{\beta_x^*}} \quad \text{and} \quad \mathcal{L} \propto \frac{1}{\sqrt{\beta_y^*}}. \tag{23}$$

The high luminosity at SuperKEKB is achieved by "squeezing" the vertical beta function β_y^* as the beams approach the interaction point, as illustrated in Fig. 6. In a traditional head-on collision scheme, the overlap length at the interaction point is approximately equal to the bunch length σ_z. This requires ultra-short bunches to avoid the "hourglass" effect, in which the heads and tails of each bunch (where the transverse size of the beam envelope is largest) pass through each other *before* or *after* passing through the IP. On the other hand, the nano-beam scheme employed by SuperKEKB utilizes a large crossing angle ϕ with overlap equal to a fraction of the bunch length. This allows for β_y^* to be squeezed down to ~ 0.9 mm. In the future, β_y^* is expected to be reduced further down to ~ 0.3 mm in order to achieve the full design luminosity of 6.0×10^{35} cm^{-2}s^{-1}. The design parameters

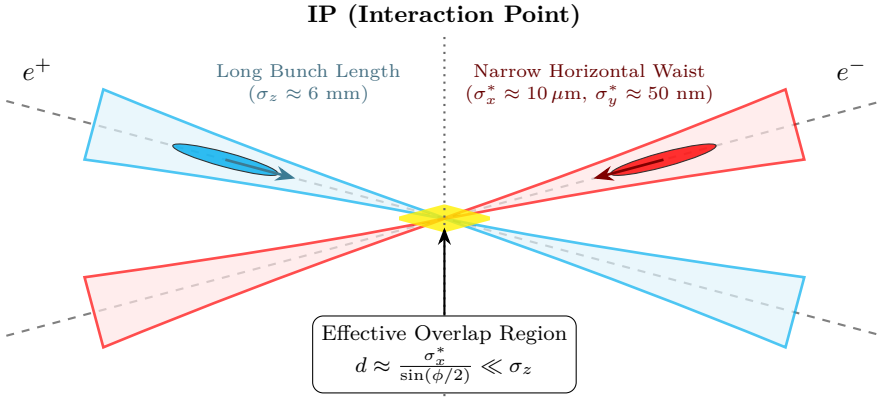


Figure 6: The nano-beam collision scheme at SuperKEKB

for SuperKEKB are shown in Table 2 and luminosity projections through 2039 are shown in Fig. 7.

3.2 The Belle II Detector

The Belle II detector is composed of seven sub-detectors, as shown in Fig. 8. The full Belle II detector is hermetic, enabling a variety of unique analyses (e.x. dark sector searches which utilize precise measurements of missing energy) which are not possible at experiments such as LHCb. In this study, we will primarily focus on the Electromagnetic Calorimeter (ECL).

Table 2: Design parameters for the SuperKEKB accelerator [22]

Parameter	**Value**
Avg. Beam Current ($\langle I \rangle$)	3.6 / 2.6 A (e^+/e^-)
Number of Bunches (N_b)	2500
Horizontal Beta Function (β_x^*)	32 / 25 mm (e^+/e^-)
Vertical Beta Function (β_y^*)	0.27 / 0.30 mm (e^+/e^-)
Luminosity ($\mathcal{L}$)	8.0×10^{35} cm^{-2}s^{-1} [a]

[a] Current target is 6.0×10^{35}cm^{-2}s^{-1}

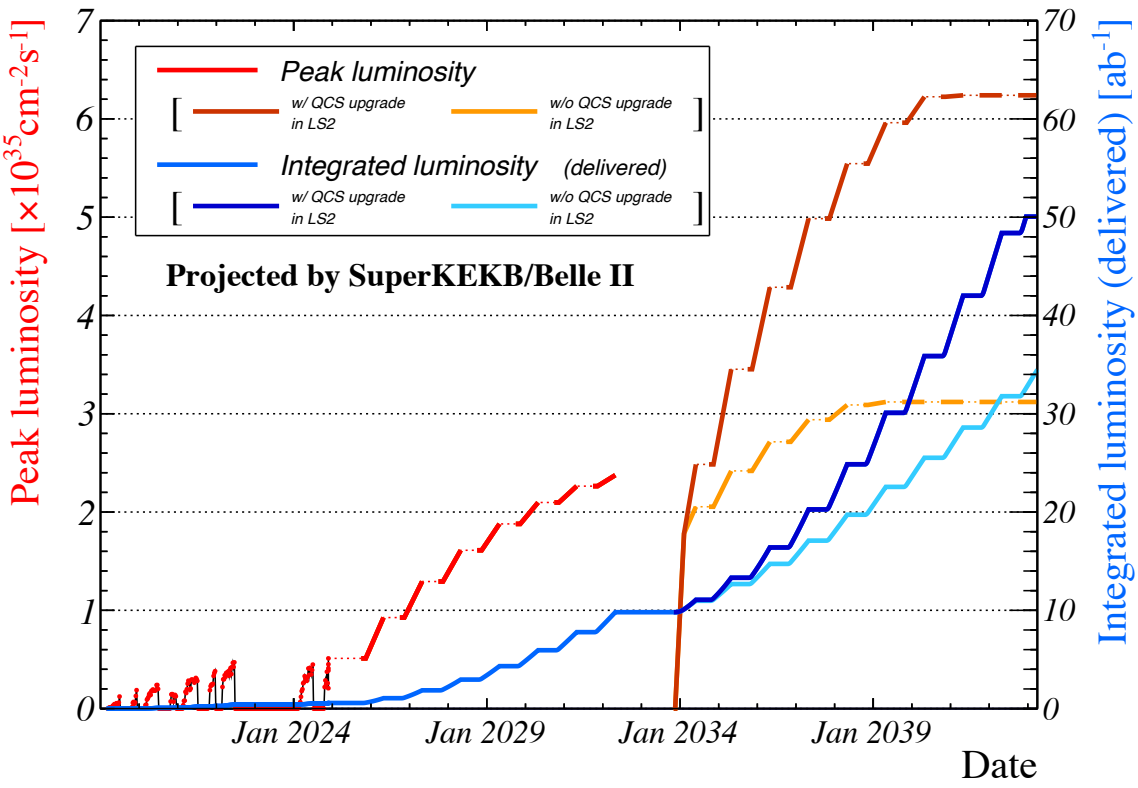


Figure 7: Luminosity projections for the SuperKEKB accelerator through 2039 [23]

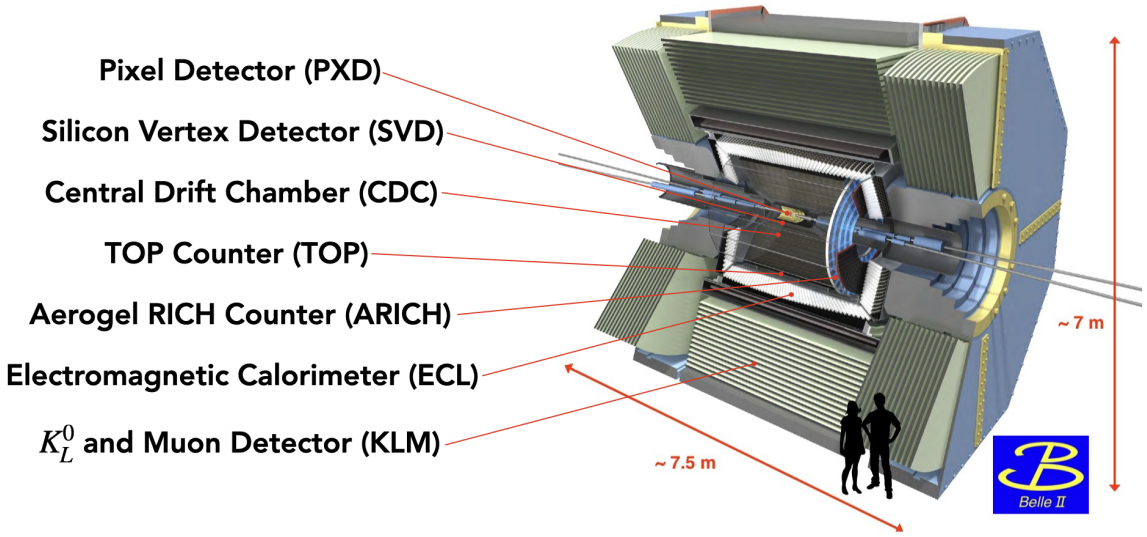


Figure 8: A schematic view of the Belle II detector and its constituent sub-detectors

3.3 Pixel Detector

The Pixel Detector (PXD) is a two-layer depleted field-effect transistor (DEPFET) pixel detector (Fig. 9). The inner layer contains 8 independent DEPFET modules, while the outer layer contains 12. Each of these modules consists of a 250×768 pixel sensor matrix, with pixel sizes ranging from $50\ \mu\text{m} \times 55\ \mu\text{m}$ to $50\ \mu\text{m} \times 60\ \mu\text{m}$ in the inner layer, and from $50\ \mu\text{m} \times 70\ \mu\text{m}$ to $50\ \mu\text{m} \times 85\ \mu\text{m}$ in the outer layer. As the innermost component of the vertex detector system, the PXD is primarily responsible for the reconstruction of the decay vertices of charged particles produced at the interaction point.

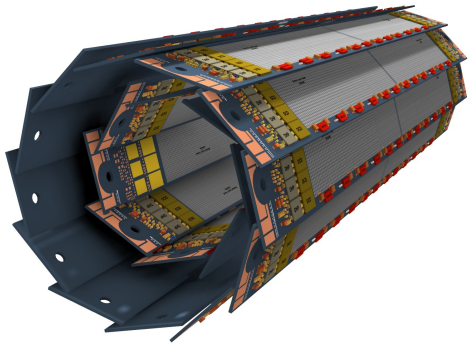


Figure 9: The Belle II Pixel Detector [24]

3.4 Silicon Vertex Detector

The Silicon Vertex Detector (SVD) is a four-layer silicon strip detector, with each layer comprised of between 7 and 16 independent Double Sided Silicon Strip Detector (DSSD) units (Fig. 10). These DSSD units are called a "ladders" and include corresponding independent readout and cooling channels. Along with the PXD, the SVD enables high-precision reconstruction of the decay vertex in each event due to its extremely high spatial granularity and proximity to the interaction point.

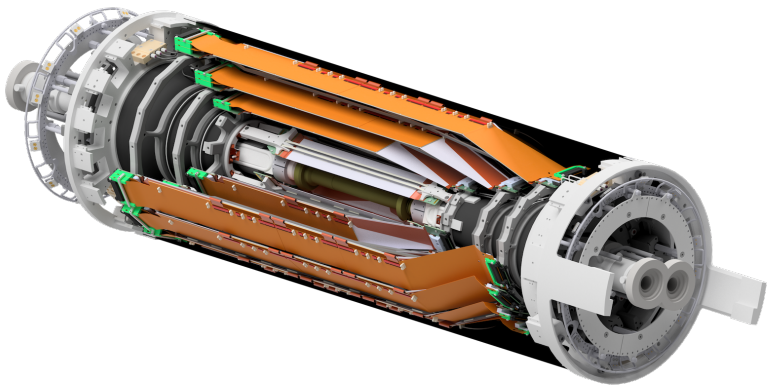


Figure 10: The Belle II Silicon Vertex Detector [24]

3.5 Central Drift Chamber

The Central Drift Chamber (CDC) is the primary tracking detector at Belle II. The CDC is a cylindrical tracking detector with an inner radius of 16 cm and an outer radius of 113 cm, consisting of coaxial aluminum and tungsten wires suspended in He-C_2H_6 gas [24]. These wires are distributed in 56 layers which partition the volume into ~ 1.8 cm $\times$ 1.8 cm cells. The cross-section of the detector in the $r - z$ plane is visualized in Fig. 11. The CDC plays a critical role in the reconstruction of charged particles, enabling precise measurement of particle momenta. Additionally, the ionization energy loss as particles traverse the He-C_2H_6 gas can be utilized for particle identification, particularly for low-momentum pions, kaons, and protons.

3.6 Time of Propagation Counter

The Time of Propagation Counter (TOP) consists of a set of 16 modules enclosing the barrel region, each consisting of a flat quartz bar with a thickness of 2 cm and a specialized photodetector called a Micro Channel Plate Photo Multiplier Tube (MCP-PMT) [25]. When a charged particle passes through the quartz bar, the emitted Cherenkov radiation propagates with total internal reflection through the length of the bar, reaching the sensor after a time dependent on the angle of the original emission. The design of one of these modules is illustrated in Fig. 12. Small differences in the angle of emitted Cherenkov photons thus allow for precise measurement of particle velocities, which can be combined with momentum information from the CDC to reconstruct particle masses. Hence, the TOP is primarily responsible for the separation of charged kaons from pions, which have overlapping ionization energy loss curves and thus cannot be easily identified using the CDC alone.

3.7 Aerogel RICH Counter

The Aerogel Ring-Imaging Cherenkov Counter (ARICH) is a conventional Cherenkov detector which consists of an aerogel radiator and photodetector, the latter of which detects the Cherenkov photons emitted by charged particles passing through the radiator [27]. The role of the ARICH is complementary to that of the TOP, improving the separation of charged kaons from pions. Unlike the TOP, however, the ARICH utilizes the ring-shaped spatial distribution of Cherenkov photons at the photodetector (rather than the time-of-propagation in the radiator) for particle identification.

3.8 Electromagnetic Calorimeter

The Electromagnetic Calorimeter (ECL) is a homogeneous calorimeter composed of 8736 caesium iodide (CsI) crystals doped with thallium (Tl). These scintillating crystals have a radiation length of

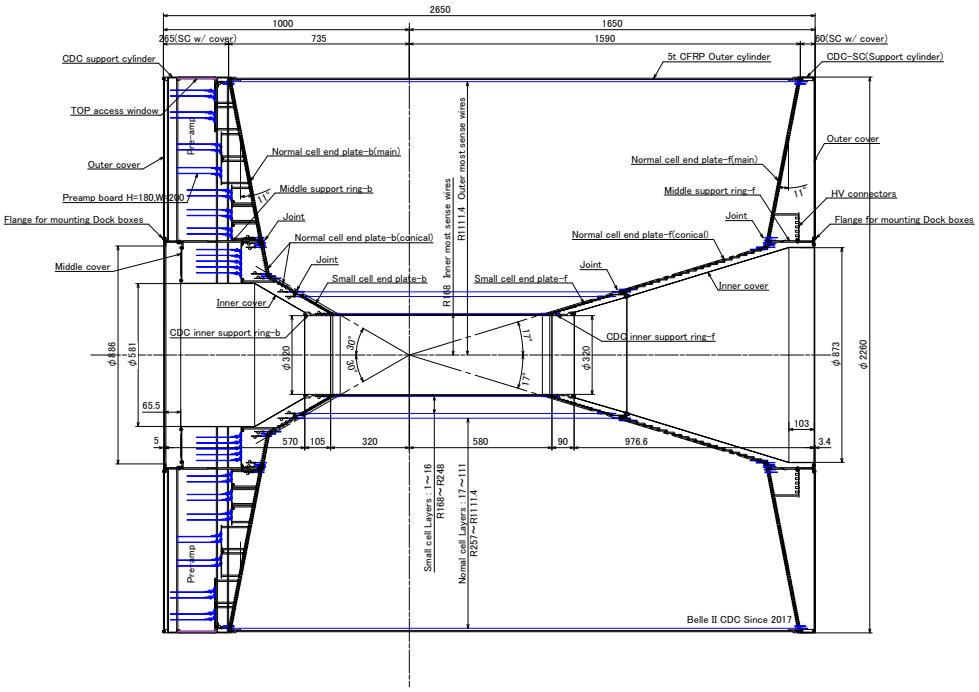


Figure 11: $r - z$ cross-section of the Belle II Central Drift Chamber [24]

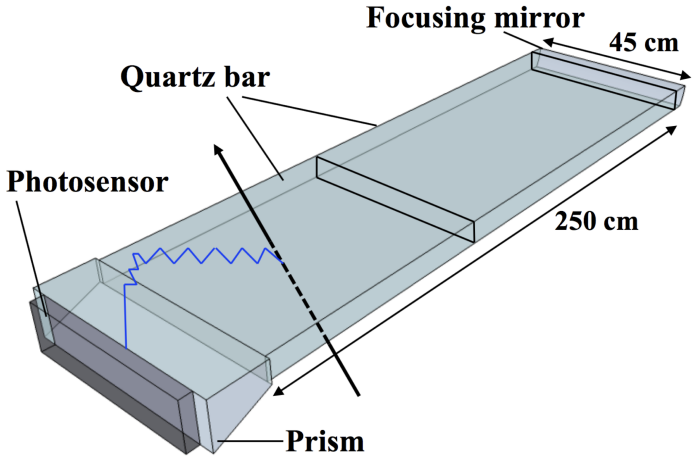


Figure 12: Overview of a Time of Propagation Counter (TOP) module [26]

$x_0 \approx 1.85$ cm and are distributed in a roughly cylindrical array centered on the interaction point as shown in Fig. 13. Each crystal is approximately 30 cm $\times$ 5.5 cm $\times$ 5.5 cm. When a particle interacts

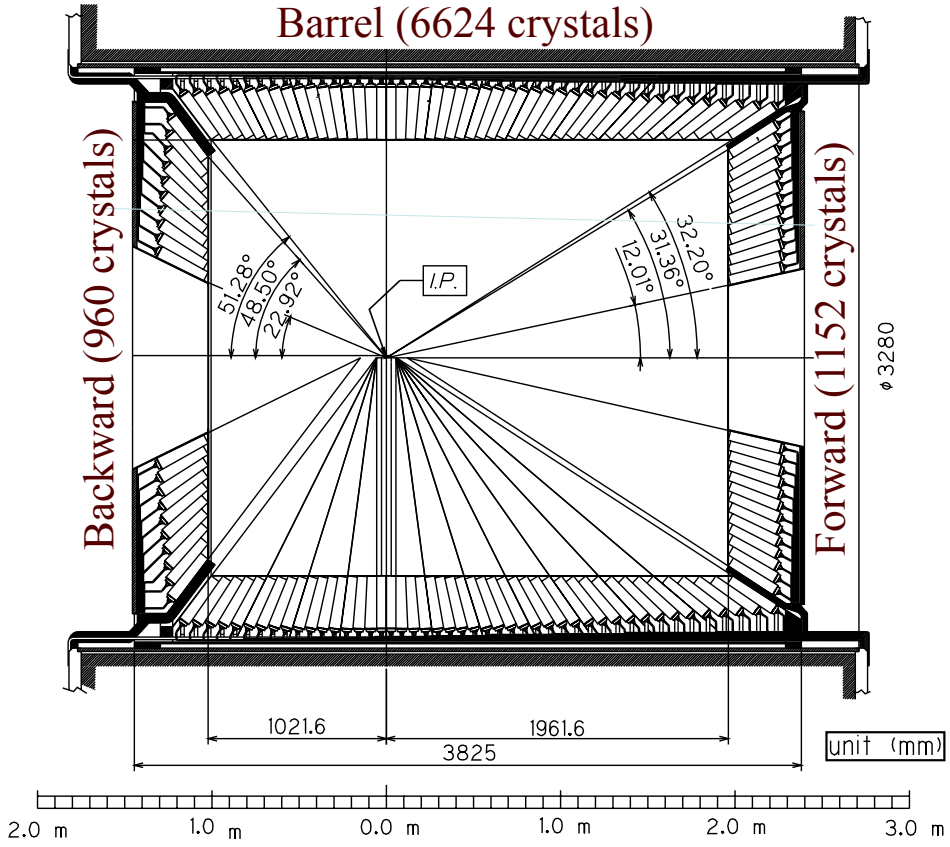


Figure 13: $r - z$ cross-section of the Belle II Electromagnetic Calorimeter [24]

with one of these crystals, the energy deposited produces photons in the visible spectrum with intensity proportional to the total energy deposited in the crystal. These photons are converted to an electrical signal by a photodiode (PIN-PD) at the rear surface of the crystal. While the hit information for each crystal is retained for offline analysis, the individual crystals are grouped and read out as 576 *Trigger Cells* ("TCs") as shown in Fig. 14 for use in online triggering due to the limited throughput of the current readout electronics.

When a particle transits a single crystal and undergoes multiple interactions, photons are emitted at multiple points in time from multiple locations in the crystal. Furthermore, even when only one interaction occurs, the photons emitted from the interaction have a finite temporal distribution due to the varying times required for the excited scintillator states to decay, as well as the differing optical path lengths experienced by photons emitted in different directions. Therefore, the current observed at the PIN-PD appears as a continuous waveform rather than a discrete step function. Waveforms from each PIN-PD are routed to the front-end electronics (FEE) at the ECL, which include a pre-amplifier and waveform shaping module ("ShaperDSP"). Each ShaperDSP module outputs two separate signals with different shaping times. The first is integrated over 0.2 μs, with the resulting waveforms sent directly to the global trigger for online triggering. The second is integrated over 1.0 μs and is routed through the full data acquisition system for offline processing. The full ECL FEE readout chain is shown in Fig. 15.

The ECL fulfills a wide variety of purposes, including reconstruction of both charged and neutral particles. While the comparatively coarse segmentation in the direction orthogonal to the beam limits

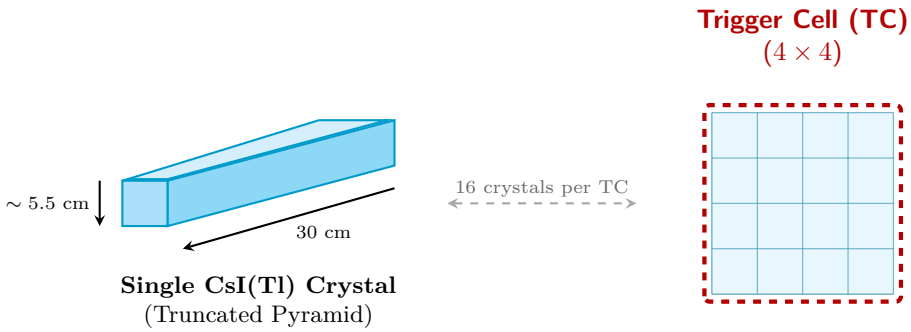


Figure 14: Geometry of a CsI(Tl) crystal (left) and the 4×4 Trigger Cell (TC) grouping (right)

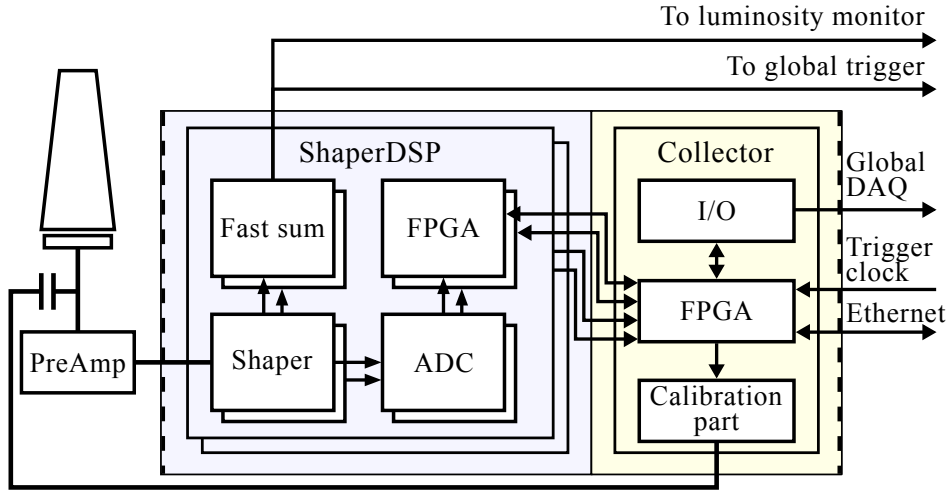


Figure 15: The front-end electronics for the Belle II Electromagnetic Calorimeter [24]

the fidelity of shower reconstruction, the high-granularity in both θ and ϕ enables high-accuracy matching of calorimeter hits to charged particle tracks in the Central Drift Chamber. The average position resolution of the ECL in the transverse plane is approximately parameterized by $\sigma_x \approx 5\ \text{mm}/\sqrt{E\ (\text{GeV})}$ [28]. On the other hand, the angular resolution ranges from ~ 13 mrad at low energies ($E \approx 100$ MeV) to ~ 3 mrad at high energies ($E > 4$ GeV). Similarly, both the timing and energy resolutions also scale with energy. The energy dependence of the energy resolution σ_E can approximated as

$$\frac{\sigma_E}{E} = \sqrt{\left(\frac{0.066\%}{E}\right)^2 + \left(\frac{0.81\%}{E^{1/4}}\right)^2 + (1.34\%)^2}, \tag{24}$$

while the timing resolution ranges from ~ 9 ns at ~ 200MeV, down to ~ 2 ns at the GeV scale [24].

3.9 K_L^0/Muon Detector

The K_L^0 and Muon Detector (KLM) is a sampling calorimeter which forms the outermost layer of the Belle II detector. The KLM consists of doped polystyrene scintillator strips in the endcap and inner barrel regions, and layered resistive plate chambers (RPCs) in the outer barrel region, as shown in Fig. 16. The KLM, as the name suggests, is primarily designed to enable the reconstruction of K_L^0 and μ particles which pass through the inner layers of the detector with minimal interaction.

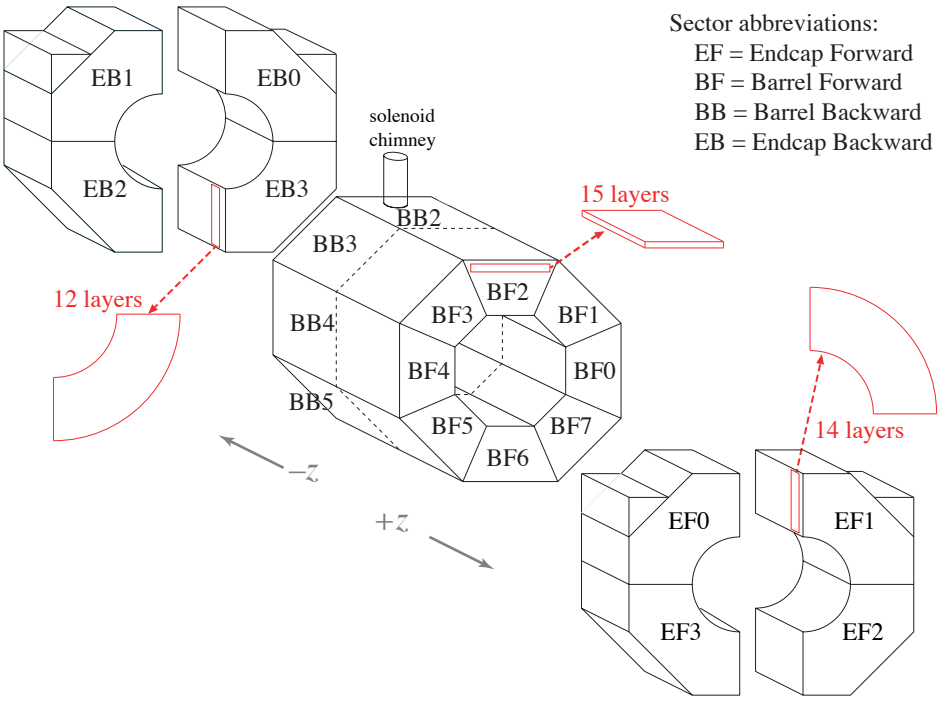


Figure 16: The Belle II K_L^0 and Muon Detector [24]

3.10 Data Acquisition System

Belle II utilizes a conventional trigger-driven Data Acquisition System (DAQ) which comprises the full synchronous data flow from each sub-detector through to the High-Level Trigger (HLT) as shown in Fig. 17. Data from each sub-detector is stored by a ring buffer on the front-end electronics (FEE) until a final trigger decision is made. The timing and trigger distribution (TTD) system distributes the Level-1 trigger signal to each FEE board on an event-by-event basis. When the Level-1 trigger

signal is received by the FEE boards, the data from each sub-detector is subsequently read out via optical link to readout modules. These readout modules are connected to readout PCs via optical links, which then forward this information to the event-building PCs using Ethernet. The data rate seen by the DAQ system is thus governed by the Level-1 trigger via the aforementioned TTD system, hence the total Level-1 trigger rate is capped by the DAQ throughput limit of 30 kHz.

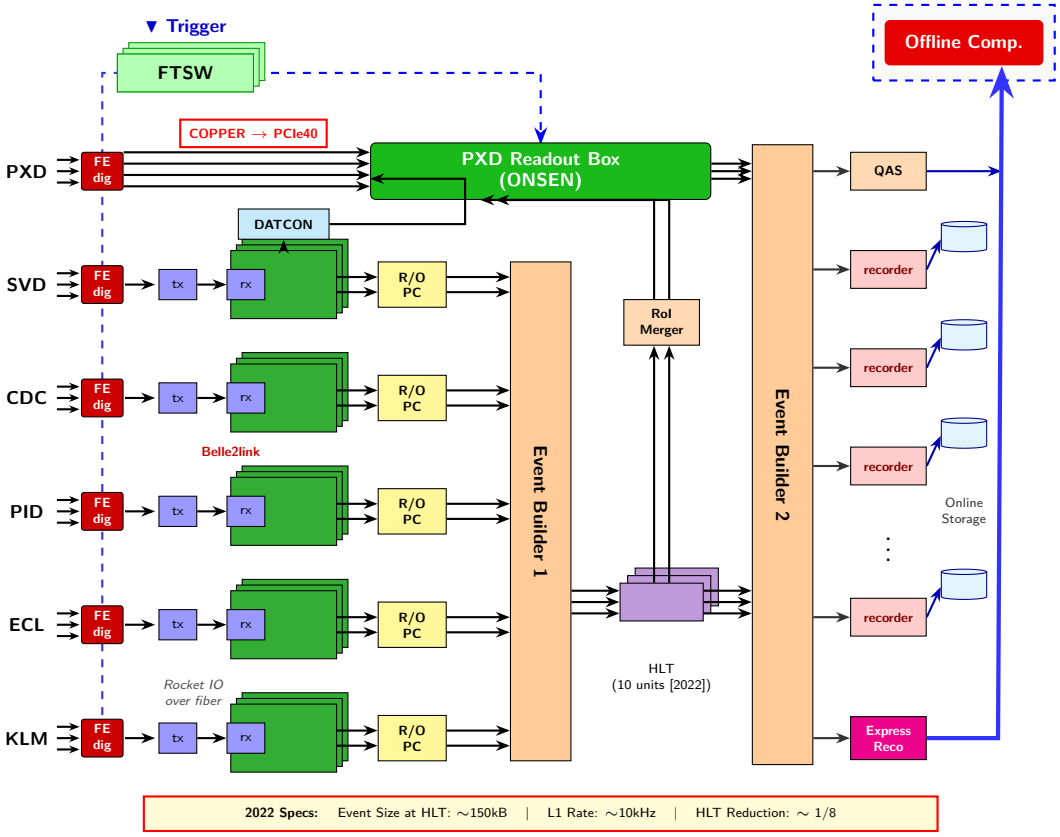


Figure 17: The Belle II Data Acquisition System [29]

3.11 The Level-1 Trigger

The Level-1 (L1) trigger comprises the entirety of the data processing system used to make real-time trigger decisions at Belle II. Unlike the High-Level Trigger which receives only data for event triggered by the Level-1 Trigger, the Level-1 trigger relies on an entirely separate data flow which utilizes a lower-fidelity data stream at *every* clock cycle in order to make the final trigger decision. Instead of performing event reconstruction using the digitized data directly from each sub-detector, the Level-1 trigger employs multiple data-reduction steps in order to minimize both data rate and latency. The data flow for the Level-1 trigger is shown in Fig. 18. The latency of the Level-1 Trigger is limited by the size of the ring buffers at each detector, which temporarily store data at the detector before receiving the L1 trigger signal. In particular, the time before the trigger decision is constrained by the time before *any* detector overflows the buffer. At Belle II, this limitation comes from the Silicon Vertex Detector, which reaches maximum buffer capacity in roughly 5 μs, hence the final trigger decision must be made within that window.

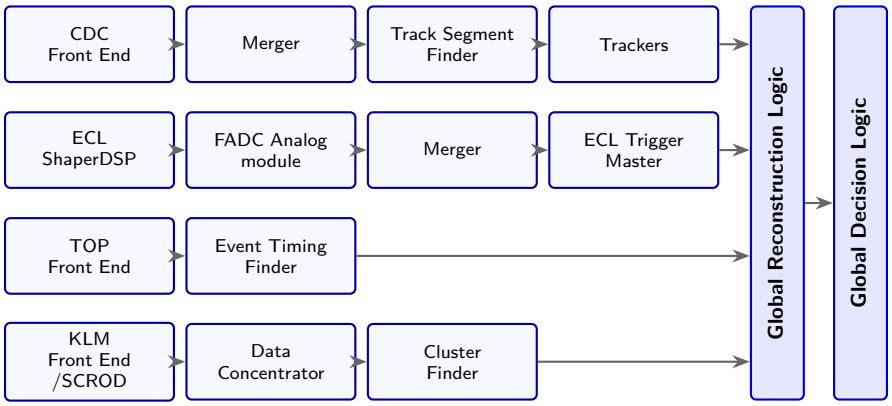


Figure 18: The Belle II Level-1 Trigger

3.11.1 Universal Trigger Board

The Belle II Universal Trigger Board 4 (UT4) is a custom electronics module designed specifically for use in the Belle II Level-1 Trigger system. Each board includes a set of high-speed optical links and a dual-FPGA architecture centered around a high-performance AMD Xilinx Virtex UltraScale FPGA to handle complex real-time trigger algorithms. Nearly all algorithms in the Level-1 trigger run on either a UT4 or the last-generation UT3 board, with at least one dedicated board for each of the discrete components illustrated in Fig. 18.

3.11.2 Data Flow and Sub-Triggers

Each of the CDC, ECL, and KLM has its own corresponding sub-trigger system which performs on-detector reconstruction with the raw signal data and outputs higher-level data objects to be utilized by later stages of the Level-1 trigger. For example, for a calorimeter such as the ECL, the sub-trigger system performs basic clustering and outputs "clusters" with associated positions and energies. These are aggregations of hits which are likely to be associated with the same primary particle at the surface of the ECL. Whereas individual hits have associated energies, positions, and hit times, clusters inherit the total energy, average position, and average hit time of their constituent hits. On the other hand, a tracking detector like the CDC performs track finding and interpolation, outputting charged-particle candidates ("tracks") with their associated positions and momenta. These data objects are then sent on to the Global Reconstruction Logic, after which the Global Decision Logic makes the final trigger decision for each event based on the collection of trigger bits provided by the GRL.

3.11.3 The ECL Sub-Trigger

While each of the aforementioned sub-detectors has its own sub-trigger algorithm, here we will focus on the sub-trigger algorithms utilized by the Electromagnetic Calorimeter (ECLTRG). The ECLTRG receives the analog sum of the (0.2 μs shaping time) ShaperDSP outputs for the 16 crystals in each TC. These waveforms are first processed by a front-end analysis module (FAM), each of which receives data from 12 TCs. The FAM includes three steps: first, the analog signal from each TC is digitized by a fast analog-to-digital converter (FADC). Then, the digitized signal is processed on an FPGA to obtain timing and energy information for each TC hit. Finally, a 100 MeV energy threshold is applied before the data is sent on to the Trigger Merger Module (TMM). The TMM performs aggregation of the data received from multiple FAMs, then routes the merged data to the ECL Trigger Module (ETM).

At the ETM, the individual TC hits are grouped into "clusters". Each resulting cluster is parameterized by 35 bits, allocated as shown in Table 3: Due to hardware limitations, only the six

Table 3: Per-cluster ECL sub-trigger outputs to GRL

Bit Index	Parameter	Least Significant Bit
$[0, 6]$	Polar Angle (θ)	$1.40625°$
$[7, 14]$	Azimuthal Angle (ϕ)	$1.40625°$
$[15, 22]$	Avg. Time (t)	1 ns
$[23, 34]$	Total Energy (E)	5.5 MeV

highest-energy clusters in each event are sent to the Global Reconstruction Logic. In a typical τ-channel decay, the expected number of clusters is ≤ 6, thus reconstruction of these events should be unaffected by this constraint. Therefore, considering all four parameters (θ, ϕ, t, and E) for each cluster, we require a maximum of $6 \times 4 = 24$ values to fully characterize the ECL response.

3.11.4 The Global Reconstruction Logic

The Global Reconstruction Logic (GRL) processes the data from all sub-trigger modules in order to reconstruct the event in real-time. The GRL assigns values to a set of boolean flags ("trigger bits") for various physics targets using on multiple independent reconstruction algorithms. The algorithms executed by the GRL include the matching of reconstructed tracks from the CDC sub-trigger with hits in the outer detectors (ECL, TOP, KLM), as well as a variety of detector-specific conditions for event selection targeting $B\bar{B}$, $\tau^+\tau^-$, and dark sector physics [30].

3.11.5 The Global Decision Logic

The Global Decision Logic (GDL) is the final step in the Belle II Level-1 trigger system. The GDL receives a set of boolean trigger bits from the GRL which are then used to make a final trigger decision. The particular algorithm applied to these trigger bits can vary, but typically involves taking combinations of logical `AND` or `OR` functions of subsets of these bits, along with `NOT` for a set of bits which target specific background processes such as Bhabha scattering. This allows for dynamic modification of trigger conditions based on the accelerator conditions by simply enabling or disabling particular trigger bits without modifying the underlying reconstruction algorithms.

3.11.6 Trigger Performance

High ($> 99\%$) trigger efficiency on the $\Upsilon(4S) \to B\bar{B}$ decay can be achieved relatively easily using standard cut-based algorithms. However, these trigger conditions are significantly less effective at identifying low-multiplicity τ decays such as $\tau \to \mu\nu_\tau\bar{\nu}_\mu$ or $\tau \to \nu_\tau + \pi^\pm + n(\pi^0)$, requiring significantly lower background rejection rates to achieve comparable efficiencies. As a result, the τ and dark sector triggers currently occupy the majority of the trigger rate while accounting for a comparatively small proportion of the expected physics rate. The rates associated with various physics targets and background processes at $\mathcal{L} = 6.0 \times 10^{35}$ cm^{-2}s^{-1} are given in Table 4. At Belle II, the dominant backgrounds come from beam backgrounds and Bhabha scattering [24]. The tree-level Feynman diagrams for Bhabha scattering are shown in Fig. 19. Beam backgrounds can be divided into two broad categories:

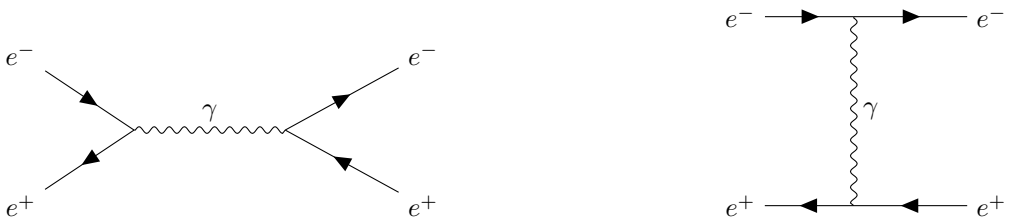
1. machine-induced ("single-beam"), and
2. luminosity-dependent ("beam-beam").

Single-beam backgrounds include synchrotron radiation, beam-gas scattering, and the Touschek effect, in which particles *within the same bunch* scatter off of each other. On the other hand, Beam-beam

Table 4: Projected physics and background event rates at Belle II

Process	**Event Rate**
e^+e^- Bunch Collision	~ 200 MHz
Beam Background	???[a]
Bhabha Scattering	~ 50 kHz
Two-Photon Processes	~ 10 kHz
$e^+e^- \to \gamma\gamma$	~ 2 kHz
$e^+e^- \to q\bar{q}$	~ 2 kHz
$e^+e^- \to \Upsilon(4S)$	~ 1 kHz
$e^+e^- \to \mu^+\mu^-$	~ 0.6 kHz
$e^+e^- \to \tau^+\tau^-$	~ 0.6 kHz
Dark Sector/New Physics	???

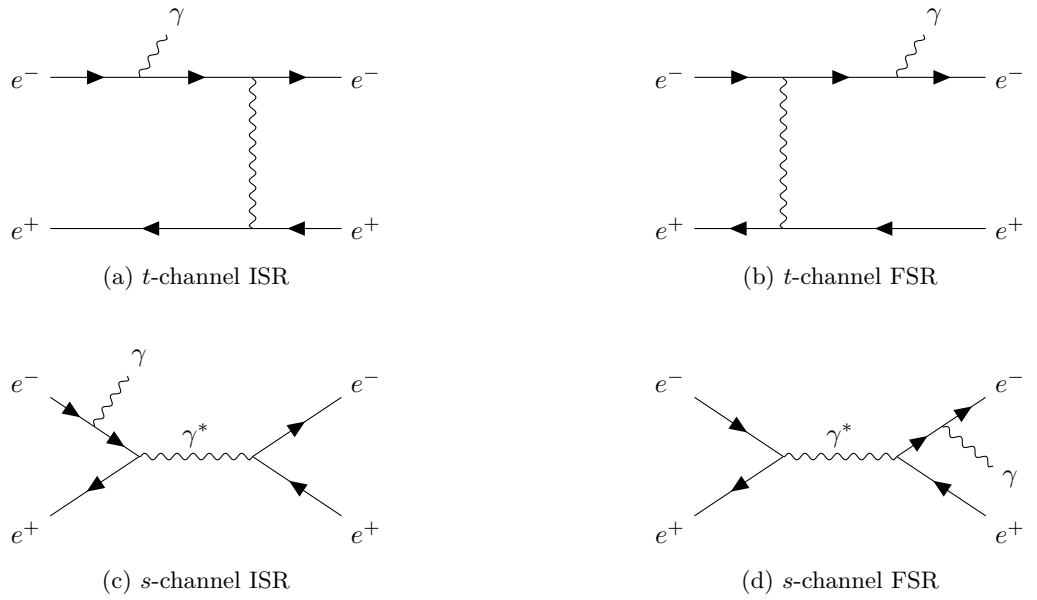
[a] ~ 300 kHz at $\mathcal{L} = 5.0 \times 10^{34}$ cm^{-2}s^{-1}



(a) s-channel (Annihilation) (b) t-channel (Scattering)

Figure 19: The tree-level Feynman diagrams for Bhabha scattering ($e^+e^- \to \gamma \to e^+e^-$)

interactions include radiative Bhabha scattering ($e^+e^- \to e^+e^-\gamma$), electron pair production via two-photon processes ($\gamma\gamma \to e^+e^-$), and hadronic backgrounds ($\gamma\gamma \to$ hadrons). At Belle II, beam backgrounds are generally the dominant backgrounds which affect event reconstruction at the Level-1 trigger, with the largest contributions coming from beam-gas interactions, the Touschek effect, and radiative Bhabha scattering [31]. The tree-level Feynman diagrams for radiative Bhabha scattering are shown in Fig. 20.



(a) t-channel ISR (b) t-channel FSR

(c) s-channel ISR (d) s-channel FSR

Figure 20: The tree-level Feynman diagrams for radiative Bhabha scattering ($e^+e^- \to e^+e^-\gamma$), including initial-state radiation (ISR) and final-state radiation (FSR) for both the t-channel and s-channel topologies

The overall requirements for the Level-1 trigger along with the present values at $\mathcal{L} = 5.0 \times 10^{34}\text{cm}^{-2}\text{s}^{-1}$ are given in Table 5 In order to maintain the requisite trigger efficiencies for both $\bar{B}B$

Table 5: Requirements for the Belle II Level-1 Trigger

Item	**Requirement**	**Present Value**
Trigger Rate	< 30 kHz[a]	~ 8 kHz[b]
Latency	5 μs	5 μs
Timing Resolution	10 ns	~ 8 ns
Efficiency	$> 99\%$ for $B\bar{B}$ pair	$> 99\%$ for $B\bar{B}$ pair
	$> 95\%$ for $\tau^+\tau^-$ pair	$> 95\%$ for $\tau^+\tau^-$ pair

[a] at $\mathcal{L} = 6.0 \times 10^{35}$ cm^{-2}s^{-1}
[b] at $\mathcal{L} = 5.0 \times 10^{34}$ cm^{-2}s^{-1}

and $\tau^+\tau^-$ events while keeping the total trigger rate below the data acquisition system limit of 30 kHz, the hardware-based Level-1 Trigger system will require significantly improved signal identification algorithms over the next several years [30]. Our goal is thus to limit the trigger rate to < 30 kHz without degrading the overall performance of the Level-1 trigger for event selection. In order to continue to support the full range of physics analyses at Belle II, we aim to achieve at least 95%

efficiency on 1- and 3-prong Standard Model τ decays.

In the current Belle II GDL, achieving these efficiencies generally requires taking the logical `OR` of multiple lower-efficiency trigger bits, further limiting the overall background rejection rate. The signal efficiencies for several of the most common trigger bits currently used to identify low-multiplicity τ decays are given in Table 6. These include a CDC-based single-track trigger (`stt`) and two-track trigger (`fyo`), and an ECL-based tau-specific trigger condition (`taueclb2b3`). The extrapolated trigger rates listed in Table 6 assume that the trigger rate for each bit scales linearly with luminosity. As shown in Table 6, `taueclb2b3` requires back to back low-energy clusters in the ECL and is expected to contribute $\sim$ 11.6 kHz to the total Level-1 trigger rate at $\mathcal{L} = 6.0 \times 10^{35}$ cm^{-2}s^{-1}. Due to the high trigger rate, this trigger bit was disabled in 2026. There are also several other ECL-based trigger conditions (`lml`) for general low multiplicity physics like $e^+e^- \to \mu\mu$. Note that several of these bits depend on the `ecl_3dbha` bit, which is defined by the following conditions:

- $165^\circ < \sum \theta_{\mathrm{CM}} < 190^\circ$, where $\sum \theta_{\mathrm{CM}}$ is the sum of the polar angles of any two clusters in the center-of-mass frame,
- $160^\circ < \Delta\phi_{\mathrm{CM}} < 200^\circ$,°, where $\Delta\phi_{\mathrm{CM}}$ is difference of the azimuthal angles of any two clusters the center-of-mass frame, and
- $E_1 > 3$ GeV $\wedge$ $E_1 > 3$ GeV $\wedge$ ($E_1 > 4.5$ GeV$||E_2 > 4.5$ GeV), where E_n is energy of cluster $n \in \{1, \ldots, 6\}$ in the center-of-mass frame.

The signal efficiency is computed on real data as the average of the efficiency across each pair combination of 1- and 3-prong Standard Model τ decays, weighted by their respective branching ratios. In the near future, both `hie` and several of the `lml` bits will need to be disabled in order to suppress the trigger rate as the accelerator luminosity increases, necessitating the development of this new algorithm as a functional replacement and potential augmentation to the existing trigger conditions. The luminosity-dependent background projections and subsequent trigger rates at the target luminosity are subject to significant uncertainty, but current estimates place the total trigger rate at $\mathcal{L} = 6.0 \times 10^{35}$ cm^{-2}s^{-1} around 27.3 kHz, thus the exclusive rate from any additional trigger bits will likely be limited to approximately 2.7 kHz [32].

Table 6: Tau Trigger Bits

Bit	**Detector**	**Definition**	**Sig. Eff.**	**Trigger Rate**[a]
`stt`	CDC	$\#\{\text{Tracks}\} \geq 1 \wedge p > 0.7$ GeV	84.5%	18.8 kHz
`fyo`	CDC	$\#\{\text{Tracks}\} \geq 2 \wedge \Delta\phi > 90^\circ$	67.2%	7.2 kHz
`hie`	ECL	$\sum_{n=1}^{6} E_n > 1$ GeV	93.7%	16.8 kHz
`hie4`	ECL	$\sum_{n=1}^{6} E_n > 4$ GeV	92.0%	11.6 kHz
`lml1`	ECL	$\#\{n \in \{1,\ldots,6\} : (E_n \geq 2 \text{ GeV}) \wedge (\theta\text{ID}_n \in \{4,\ldots,14\})\} \geq 1$	25.0%	6.9 kHz
`lml2`	ECL	$\#\{n \in \{1,\ldots,6\} : (E_n \geq 2 \text{ GeV}) \wedge (\theta\text{ID}_n \in \{2,3,15,16\})\} \geq 2$ $\wedge(\neg$`ecl_3dbha`$)$	12.2%	4.7 kHz
`lml8`	ECL	$\exists\{a,b\} \in \mathcal{P}_2(\{1,\ldots,6\})$ s.t. $(170^\circ < \Delta\phi_{\text{CM}} < 190^\circ)$ $\wedge\,(E_a > 250$ MeV$) \wedge (E_b > 250$ MeV$)$ $\wedge\,(\forall n \in \{1,\ldots,6\}\ E_n < 2$ GeV$)$	16.0%	2.2 kHz
`lml12`	ECL	$[\exists! a \in \{1,\ldots,6\}$ s.t. $(E_n \geq 0.5$ GeV$) \wedge (\theta\text{ID}_n \in \{6,\ldots,11\})]$ $\wedge\,[\forall(a \neq b)\ E_b < 300$ MeV$]$	76.3%	4.7 kHz
`eclmumu`	ECL	$\exists\{a,b\} \in \mathcal{P}_2(\{1,\ldots,6\})$ s.t. $(165^\circ < \sum\theta_{\text{CM}} < 200^\circ)$ $\wedge\,(160^\circ < \Delta\phi_{\text{CM}} < 200^\circ)$ $\wedge(E_a < 2$ GeV$) \wedge (E_b < 2$ GeV$)$	33.4%	8.1 kHz
`ecltaub2b3`	ECL	$\exists\{a,b\} \in \mathcal{P}_2(\{1,\ldots,6\})$ s.t. $(140^\circ < \sum\theta_{\text{CM}} < 220^\circ)$ $\wedge\,(120^\circ < \Delta\phi_{\text{CM}} < 240^\circ)$ $\wedge\,[(E_a > 140$ MeV$) \vee (E_b > 140$ MeV$)]$ $\wedge\,(\theta\text{ID}_a \in \{2,\ldots,16\}) \wedge (\theta\text{ID}_b \in \{2,\ldots,16\})$ $\wedge\sum\{E_n : \theta\text{ID}_n \in \{1,\ldots,17\}\} < 7$ GeV $\wedge\,(E_a > 120$ MeV$) \wedge (E_b > 120$ MeV$)$ $\wedge\,(\forall n \in \{1,\ldots,6\}\ E_n < 4.5$ GeV$)$	53.8%	11.6 kHz

[a] Projected at $\mathcal{L} = 6.0 \times 10^{35}$ cm^{-2}s^{-1}

4 Machine Learning

4.1 Basic Principles

Machine learning denotes a class of computational models which utilize statistical inference, geometric analysis, or iterative optimization to model complex relationships between data. The broad class of machine learning algorithms generally includes three distinct paradigms, often called *supervised*, *unsupervised*, and *reinforcement* learning. Among these, *supervised* learning is specifically a computational approach to *function approximation*. In supervised learning, we provide a set of inputs ("features") and the corresponding outputs ("labels") under the mapping defined by some function. The goal is then for the algorithm to learn to approximate the behavior of the function when applied to arbitrary inputs.

For most modern machine learning algorithms such as *neural networks*, this is achieved by *training* the model through the iterative process of *gradient descent* as illustrated in Fig. 21. To perform gradient descent, we provide a dataset including both features and labels, then propagate the feature vectors through the network to obtain a set of output vectors. The measure of the discrepancy between the outputs of the model and the corresponding labels is called the *loss*. This value is then used to iteratively update the parameters of the model, converging to a set of parameters for which the loss achieves a local minimum.

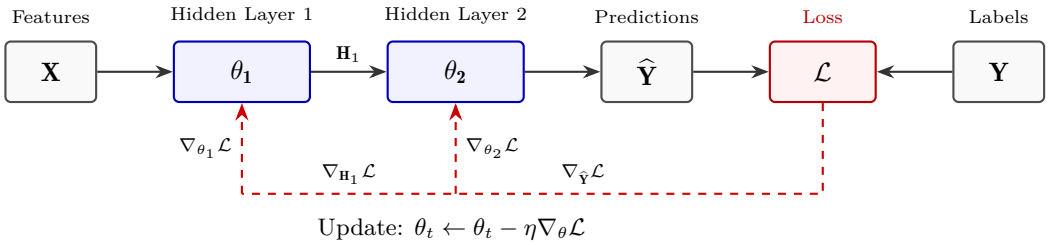


Figure 21: Gradient descent for a feed-forward neural network with learning rate η

The process of calculating the partial derivatives of the loss function with respect to each parameter is called *backpropagation*. Backpropagation utilizes the Jacobian of the operation at each layer to "pull-back" the loss function, thus determining the direction in which the model's parameters should update to minimize the loss. The magnitude of these updates is scaled by a *hyper-parameter* called the *learning rate*. Hyper-parameters are parameters which are set in advance of the learning process, governing both the behavior of the model during training as well as the structure of the model itself. This includes parameters such a the total number of layers, the number of nodes per layer, the activation functions at each layer, and more.

4.2 Neural Networks

From a purely mathematical perspective, a *fully-connected*, *feed-forward neural network* (Fig. 22) is a set of matrices

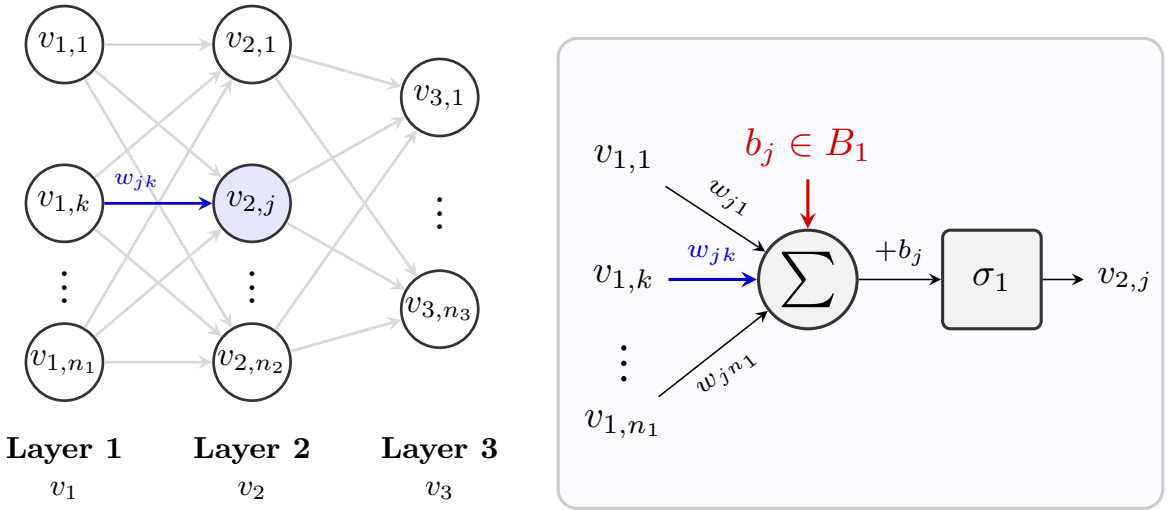
$$\left\{W_i \in \mathbb{R}^{n_{i+1}\times n_i}\right\}_{i=1}^{N} \quad \text{and} \quad \left\{B_i \in \mathbb{R}^{n_{i+1}}\right\}_{i=1}^{N} \tag{25}$$

where $N \in \mathbb{Z}^+$ is the number of *layers* in the network and n_i is the *width* of the ith layer. The entries $w_{jk} \in W_i$ are called *weights*, while the entries $b_k \in B_i$ are called *biases*. Inputs at a specific layers i are vectors $v_i \in \mathbb{R}^{n_i}$. We define the propagation of a vector through the network recursively, as

$$v_{i+1} = \sigma_i\left(\, W_i v_i + B_i\right), \quad i \in \{1, \ldots, N\}\,. \tag{26}$$

where σ is an *activation* function which is applied *element-wise* to the resultant vector. While σ_i is generally taken to be a *non-linear* function such as $\frac{1}{1+e^{-x}}$ ("Sigmoid") or $\max(0, x)$ ("ReLU"), it can

be chosen independently for each layer in the network, and is often linear at the output layer ($i = N$).



(a) Global network topology

(b) Algebraic operations at a single neuron

Figure 22: Visualization of a fully-connected, feed-forward neural network

4.3 Hardware Platforms

4.3.1 Graphics Processing Units

Most modern machine learning is performed using specialized hardware optimized for high-throughput parallel processing. The most common platforms include graphics processing units (GPUs) and tensor processing units (TPUs), each of which can contain thousands of arithmetic logic units (ALUs). Modern GPUs are also designed with dedicated tensor cores which are specifically optimized for mixed-precision matrix multiplication [33]. However, since these architectures are physically designed for massive parallelization, the hardware is significantly underutilized when performing inference on a single input, since the unused ALUs cannot be used to accelerate inference. Furthermore, GPUs rely on high-bandwith memory (VRAM) to store the model's parameters. This memory is physically separated from the compute cores, as illustrated in Fig. 23. This physical separation creates a severe latency penalty often referred to as the Von Neumann bottleneck or "memory wall". During inference on small batches, the time required to move data from external VRAM to the on-chip caches dominates the execution time.

4.3.2 Field Programmable Gate Arrays

As the name suggests, a field-programmable gate array (FPGA) is an array of configurable logic blocks (CLBs) that can be programmed using a specialized binary file called a *bitstream*. Modern FPGAs utilize a heterogeneous, column-based architecture designed to mitigate the routing overheads inherent to spatial computing. This silicon architecture interleaves fine-grained Configurable Logic Blocks (CLBs) with columns of dedicated, coarse-grained hardware primitives. The foundational resources comprising this fabric include Lookup Tables (LUTs) and Flip-Flops (FFs) for arbitrary combinational and sequential logic, Digital Signal Processing (DSP) slices for accelerated arithmetic, and a stratified embedded memory hierarchy composed of dual-port Block RAM (BRAM) and/or high-density UltraRAM (URAM).

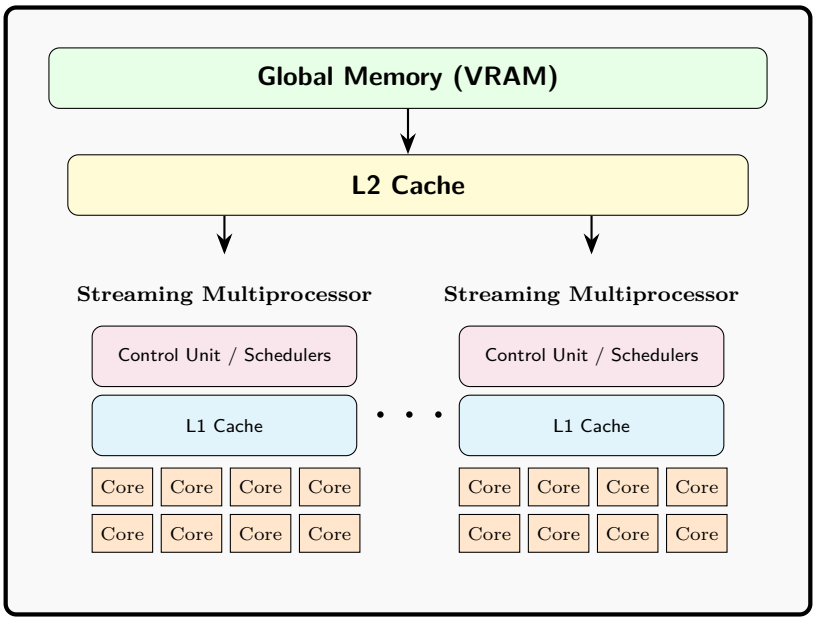


Figure 23: Schematic view of a GPU

FPGAs eliminate the centralized memory bottleneck by utilizing independent on-chip memory blocks distributed throughout the programmable logic fabric. This layout allows the hardware to be physically wired into a pipelined data path optimized for the specific computational graph. On the other hand, this severely limits the size of networks which can be implemented on a single FPGA due to limited memory and fewer logic units per-chip. Consequently, FPGAs are often used to implement relatively small models (typically containing fewer than 10^6 parameters) in contexts which require extremely low-latency inference.

4.4 Machine Learning in Particle Physics

Machine learning has a long history in experimental particle physics, beginning with the early development of neural network-based jet-tagging algorithms for the DELPHI and ALEPH experiments at CERN in the early 1990s [34, 35]. These have subsequently grown to encompass a wide range of applications from calorimeter shower reconstruction to real-time triggering and accelerator tuning [36]. The major advantage of neural networks in this domain is that they allow for much more efficient modeling of the extremely complex structure of subatomic particle interactions and decays as resolved by instruments such as calorimeters or tracking detectors. When provided with raw data from a calorimeter, physical interpretation and inference requires careful analysis of the relationships between each individual energy deposition along with spatial positions, timing information, and the material properties of the detector itself. While there are a variety of heuristics which physicists have developed to facilitate these analyses [37–40], almost all approaches which rely on these tools necessarily fail to leverage the complete informational content of the data provided by the detector, since they operate at a significantly higher level of abstraction than the raw data.

5 Event Classification Algorithm

5.1 Overview

To achieve the requisite trigger efficiency on Standard Model τ decays while maintaining a sufficiently high background rejection rate to keep the total trigger rate below 30 kHz, it is not sufficient to engineer a more complex system of cuts on individual event variables to augment those shown in Table 6. These methods are fundamentally limited in their ability to distinguish τ pair decays from the primary background contaminants such as Bhabha scattering and beam-beam interactions, as they cannot effectively utilize the complex structure of the resultant showers in the ECL to discriminate between events. For the time being, we constrain our approach to Standard Model τ decays (rather than CP-violating or LFV decays) since it allows us to train on real data, enabling more robust learning of the background distributions compared to the models used in Monte-Carlo simulation. While the accuracy of event reconstruction at the level of the GRL is limited by the data reduction

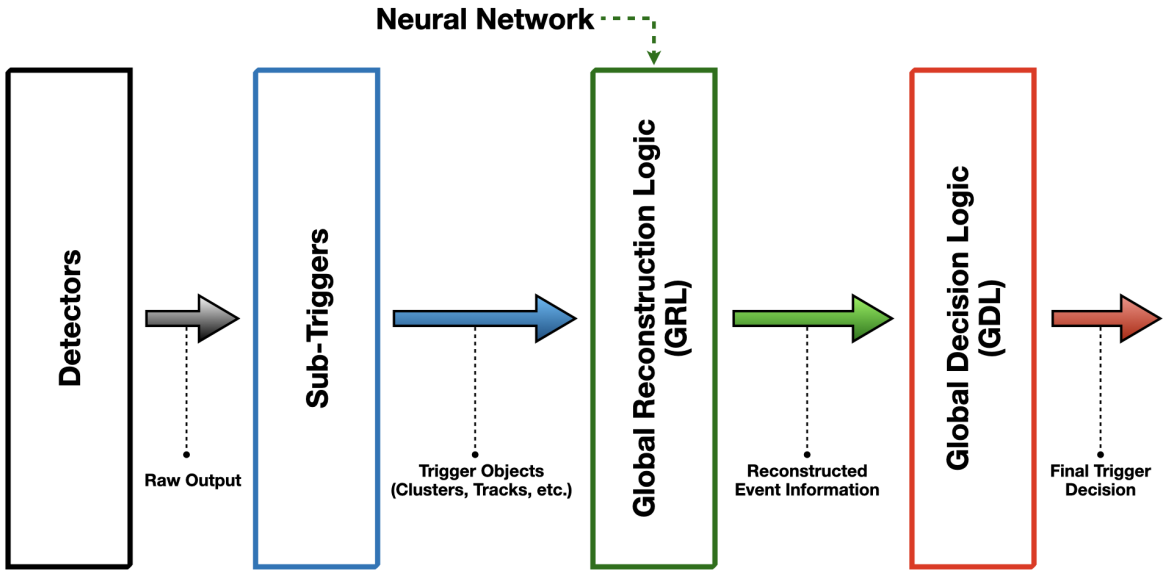


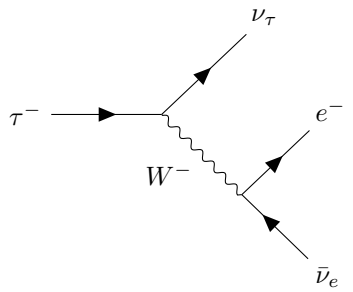
Figure 24: Implementation of neural network-based reconstruction in the Level-1 Trigger

which occurs on-detector at the ECL, the GRL still receives substantially more granular position and timing information than is currently utilized for event classification. To this end, we develop and train a neural network for τ event classification leveraging the full cluster information received by the GRL from the ECL sub-trigger system (Fig. 24). More specifically, our algorithm uses energy, timing, and position information as inputs to a fully-connected, feed-forward neural network to reconstruct these standard model τ decays and classify them among 10 possible pair-combinations as defined in 4.2. The training of the neural network is performed by using data labeled by offline analysis, and the trained model is implemented as firmware on the FPGA mounted to the GRL UT4.

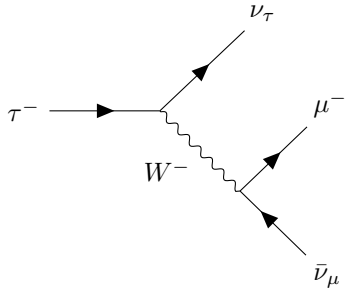
5.2 Physics Targets

In cases where the final state of a decay involves > 3 charged-particles, events can generally be identified using relatively low-complexity heuristics such as cuts on energy, track count, and/or cluster count. However, *low-multiplicity* decays (i.e. those with ≤ 3 charged particles in the final state) can be

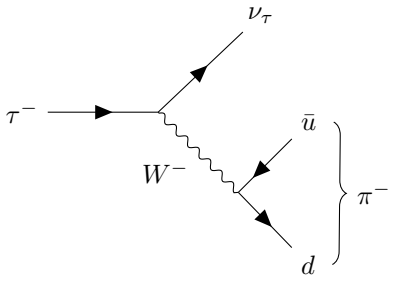
particularly challenging to identify using conventional triggers due to the sparsity and inhomogeneity of the resulting hits/tracks. Therefore, these decays are a prime candidate for reconstruction using more complex algorithms such as neural networks. In this study, we focus primarily on reconstruction of the low-multiplicity standard model τ decays, the dominant tree-level Feynman diagrams for which are shown in Fig. 25). For a pair of τ leptons produced via $e^+e^- \to \tau^+\tau^-$, this gives us 10 possible



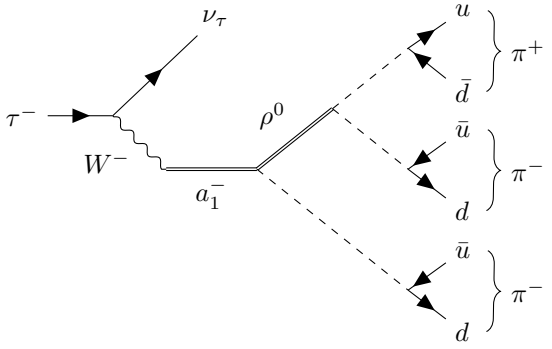
(a) Leptonic 1-Prong ($\tau^- \to \nu_\tau e^- \bar{\nu}_e$)



(b) Leptonic 1-Prong ($\tau^- \to \nu_\tau \mu^- \bar{\nu}_\mu$)



(c) Hadronic 1-Prong ($\tau^- \to \nu_\tau \pi^-$)



(d) Hadronic 3-Prong ($\tau^- \to \nu_\tau \pi^- \pi^+ \pi^-$)

Figure 25: The 1-prong (leptonic and hadronic) and 3-prong (resonance-mediated hadronic) decay modes of the Standard Model τ^- lepton

combinations of decay modes, characterized by the charged particles in the final state:

1. $e \times e$ (1×1)
2. $e \times \mu$ (1×1)
3. $e \times \pi$ (1×1)
4. $e \times 3\pi$ (1×3)
5. $\mu \times \mu$ (1×1)
6. $\mu \times \pi$ (1×1)
7. $\mu \times 3\pi$ (1×3)
8. $\pi \times \pi$ (1×1)
9. $\pi \times 3\pi$ (1×3)
10. $3\pi \times 3\pi$ (3×3)

While we perform event classification for all 10 labels, we often treat $e^+ \times e^-$ as background when making the final trigger decision due to significant signal contamination from Bhabha scattering.

5.3 Data Collection and Pre-Processing

While data obtained from Monte-Carlo simulations offers the benefit of access to ground truth information for each event, beam background modeling in the existing Belle II simulation framework is extremely limited. Therefore, we use a 36.1 pb^{-1} dataset collected with a loose trigger condition in December 2025 for the training and validation of the event classification algorithm. In this configuration, we trigger on all events which contain at least two ECL clusters or one CDC track. This ensures that the backgrounds are sampled roughly uniformly, requiring that the model learns an unbiased background distribution and limiting the discrepancy between performance in software and the eventual firmware implementation (where the model will see *every* event irrespective of the Level-1 trigger condition). The average Level-1 trigger rate during this run was 11.8 kHz with a peak luminosity of $3.05 \times 10^{34}\text{cm}^{-2}\text{s}^{-1}$.

We use the offline reconstruction tools provided in the Belle 2 analysis software framework (`basf2`) to identify low-multiplicity standard model τ decays [41]. Our selection criteria utilize several event variables such as decay vertex position, thrust, missing mass/momentum, and total visible energy, alongside neural network-based likelihood cuts for e, μ, τ, and π identification. The full set of cuts is given in Table 7. We define all events not selected by these cuts as background, comprising both continuum and beam backgrounds.

Table 7: Tau Signal Pre-Selection Cuts

Event Variable	**Definition**	**Cut(s)**
Decay Vertex	Displacement of the primary decay with respect to the IP	$-3.0 < dz < 3.0$ $dr < 1.0$
Thrust	Max. fraction of visible momentum along a single axis	$0.85 < T < 0.99$
Missing Momentum	Magnitude of the missing 3-momentum, $\lvert\vec{p} - \sum \vec{p}_{\text{vis}}\rvert$	$0.52 < p_{\text{miss.}} < 2.8$
Missing Mass	Squared invariant mass of all unobserved particles	$-0.5 < m_{\text{miss.}} < 49$
Visible Energy	Scalar sum of the energies of all reconstructed particles	$1.5 < E_v < 10.4$

The individual decay modes are distinguished using particle identification (PID) likelihood cuts (Table 8). In addition, we require that the number of reconstructed particles in the final state is limited to the expected multiplicity of the reconstructed decay (e.x. $\#\{e\} \leq 2$ for $e \times e$, $\#\{\pi\} \leq 6$ for $3\pi \times 3\pi$). We define all events not selected by these cuts as background, comprising both continuum and beam backgrounds.

Table 8: Particle Identification Pre-Selection Cuts

PID Variable	**Definition**	**Cut(s)**
Pion Likelihood	PID likelihood fraction for the π hypothesis	$L_\pi > 0.5$
Electron Likelihood	PID likelihood fraction for the e hypothesis	$L_e > 0.9$
Muon Likelihood	PID likelihood fraction for the μ hypothesis	$L_\mu > 0.9$

Due to the relative sparsity of events in each class, we employ an iterative stratification algorithm (as implemented in the `scikit-multilearn` package) with an $80:20$ split to separate the train and test data [42]. Iterative stratification is a method for stratification of multi-label data which ensures that the relative prevalence of each label remains uniform across each fold. The algorithm functions by first calculating the sample count per fold, then iterating through all label combinations and selecting the combination which maximizes the number of unique labels and minimizes the number of samples with the same combination. Unlabeled samples are then distributed randomly, weighted by remaining fold capacity. This ensures that the relative prevalence of each class is preserved in both sets [43].

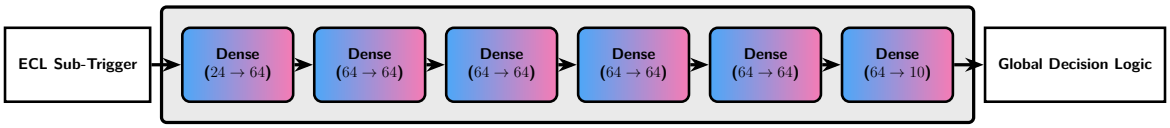


Figure 26: Structure of the τ event classification network

Due to the size of our original dataset ($N \approx 2.3 \times 10^7$ events, 10 labels), it is computationally infeasible to perform iterative stratification on the entire dataset. Therefore, we first separate the events which are labeled as background from those labeled as signal. We then use `scikit-learn` to partition the background events (also with an 80 : 20 test-train split), and iterative stratification on the signal events, before re-combining to obtain the complete train and test datasets.

5.4 Model Architecture

The event-classification algorithm developed in this study is a fully-connected, feed-forward neural network with six hidden layers as shown in Fig. 26. This model accepts all four cluster parameters $(E, \theta, \phi, t_{avg})$ from the ECL sub-trigger (Table 3) for each of the six highest-energy clusters in an event and produces ten outputs: one for each pair of the four selected τ decay modes. An overview of the model's hyperparameters is given in Table 9. In order to set a single boolean flag for this trigger

Table 9: Neural Network Architecture

Layer	Layer Type	Input Width	Output Width	Activation
1	Dense	24	64	ReLU
2	Dense	64	64	ReLU
3	Dense	64	64	ReLU
4	Dense	64	64	ReLU
5	Dense	64	64	ReLU
6	Dense	64	64	ReLU
7	Dense	64	10	Linear

bit, we program ten independent thresholds against which the outputs are compared. If any of the ten outputs exceeds the assigned threshold, we assign the flag a value of 1 ("True"); otherwise, we set it to 0 ("False").

5.5 Model Compression

In general, there are several distinct (and occasionally mutually exclusive) approaches to minimizing the size and inference time of a neural network. These include dynamic quantization, static quantization, structured pruning, unstructured pruning, and hybrid approaches which utilize hyperparameter optimization schemes alongside a combination of these methods. While these have historically been treated as independent modes of compression, necessitating separate steps for pruning and quantization respectively, modern heterogeneous quantization frameworks such as `HGQ2` effectively perform compression equivalent to static quantization and unstructured pruning simultaneously [44].

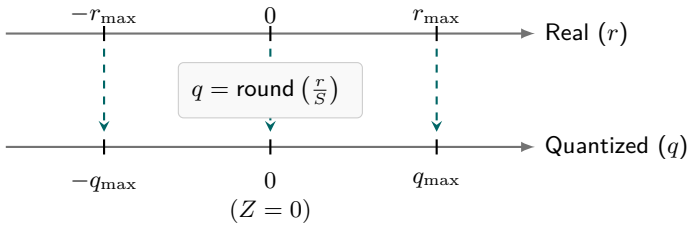
5.5.1 Quantization

Quantization reduces model size and inference time by limiting the numerical precision of the model's weights, biases, and activations. In some cases, this can even allow operations which would ordi-

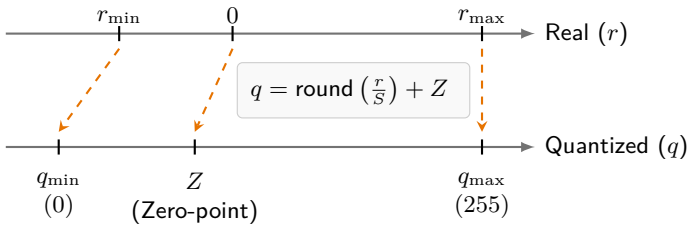
narily require more complex digital signal processing units (DSPs) to be performed using lookup tables (LUTs), which can dramatically reduce inference times for certain hardware configurations. Mathematically, for a full-precision value r, the quantized value q is given by

$$q = \text{round}(r/S) + Z, \tag{27}$$

where S is a scaling factor and Z is the "zero-point". In practice, Z determines whether a particular quantization is *symmetric* ($Z = 0$) or *asymmetric* ($Z \neq 0$). As shown in Fig. 27, symmetric quantization (Fig. 27a) simplifies computation but can result in inefficient allocation of the full bit-width if the distribution is highly skewed. Asymmetric quantization (Fig. 27b) can more efficiently utilize the full bit-width for arbitrary ranges, but requires additional overhead for the zero-point correction. In our case, we will be dealing primarily with *symmetric* quantization schemes.



(a) Symmetric Quantization



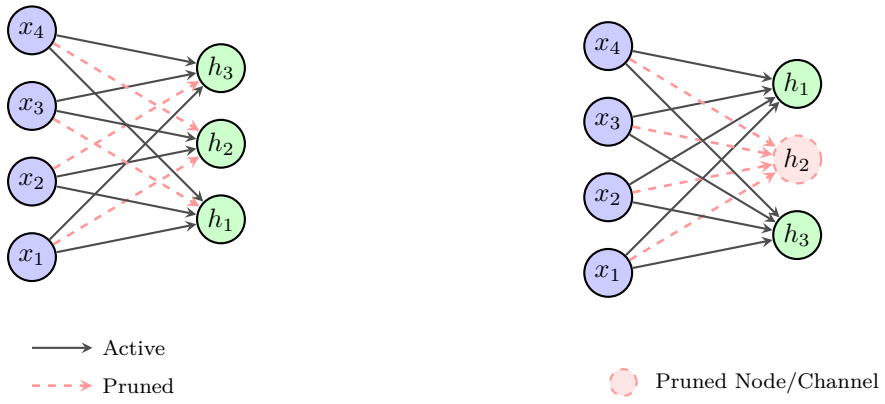
(b) Asymmetric Quantization

Figure 27: Symmetric and asymmetric quantization

When we consider the specific applications of this method for neural network compression, quantization can broadly be separated into two distinct frameworks: dynamic quantization and static quantization. While both approaches operate on the same fundamental principle of reducing the numerical precision of a model's weights and activations, the distinction lies in *when* they compute the scales for these parameters. In a dynamically quantized model, the scaling parameters for the weights are fixed ahead of time, but the activation ranges are calculated on the fly during inference. In contrast, static quantization fixes the scaling for both the weights and activations in advance. While this eliminates the runtime overhead associated with the real-time calculation of the optimal scaling for each activation, it requires calibrating the model with a representative dataset beforehand to map out the activation ranges.

5.5.2 Pruning

Pruning takes an entirely distinct approach to model compression by strategically *removing* nodes from the network entirely. This is typically done by removing the nodes which contribute the least to the output. This often (though not always) corresponds to the nodes with the lowest weights and



(a) Unstructured Pruning

(b) Structured Pruning

Figure 28: Comparison between Unstructured and Structured Neural Network Pruning techniques.

biases. Like quantization, pruning generally falls into one of two classes: unstructured pruning or structured pruning (Fig. 28).

In unstructured pruning, individual weights are pruned based on their magnitude and/or effect on the output of the model. In practice, this most commonly entails the pruning of the individual nodes with the lowest weights, though it can also involve activation-based criteria or more complex information-theoretic metrics of connectivity. However, unstructured pruning does *not* actually modify the architecture of the network (i.e. the dimensions of the weight matrices). Instead, the nodes with the lowest weights simply have their weights set to *zero*. However, this generally results in large sparse matrices, often requiring specialized hardware, data formats, and custom compute kernels for optimal performance. The result is typically a moderate reduction in model size and inference time with little to no impact on the model's accuracy.

On the other hand, structured pruning leverages the large-scale structure of the model to excise entire components of the based on their overall connectivity to the rest of the network. While fundamentally based on the same criteria, structured pruning, as the name suggests, makes *structural* changes to the network architecture, actively reshaping the weight matrices and often removing entire layers, channels, or filters. This is particularly useful when the original model is extremely sparse, as it allows the large sparse matrices to be converted to much smaller dense matrices, massively reducing computational overhead both in storing the model's parameters and in performing arithmetic operations during inference.

5.6 Training

We train the model for 500 epochs using adaptive moment estimation (ADAM) with a learning rate of $\eta = 10^{-4}$, decay rates $\beta_1 = 0.9$ and $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, and a weight decay rate of 10^{-5}. To maximize multi-label classification performance on unbalanced classes, we utilize a categorical focal cross-entropy loss with $\alpha = 0.99$ and $\gamma = 4.0$.

5.6.1 Cross-Entropy Loss

In information theory, the *cross-entropy* of a distribution A with respect to another distribution B is given by

$$H(A, B) = -\mathbb{E}_{x\sim A}\left[\log B(x)\right], \tag{28}$$

where $\mathbb{E}_{x\sim A}$ is the expected value operator with respect to A. In machine learning, the cross-entropy is commonly used to compute the loss when training models which perform discrete classification. In

particular, when the outputs of a model are normalized such that $p_t \in [0, 1]$, the loss function can be defined as the categorical cross entropy,

$$\mathcal{L}_{CE} = -\log(p_t) \tag{29}$$

where p_t is the predicted probability that the input belongs to the class.

5.6.2 Class Imbalances and Focal Loss

Generally, when there is a substantial imbalance in the number of samples from each class in a dataset, the model's learned parameters disproportionately favor configurations in which the accuracy on the dominant classes is maximized, while the accuracy on the classes which comprise a lesser proportion of the dataset can remain comparatively low. To mitigate this, we often apply a direct *weighting* to the loss function in order to ensure that classification performance on each class is treated equally across the full dataset. However, this approach suffers from a major limitation: there is an implicit assumption that the configurations which result in higher accuracy on *some* examples of a class are highly probable to result in higher accuracy on *all* examples of that class. That is, all members of a given class are structurally similar from the perspective of the classifier. Consider the following two examples:

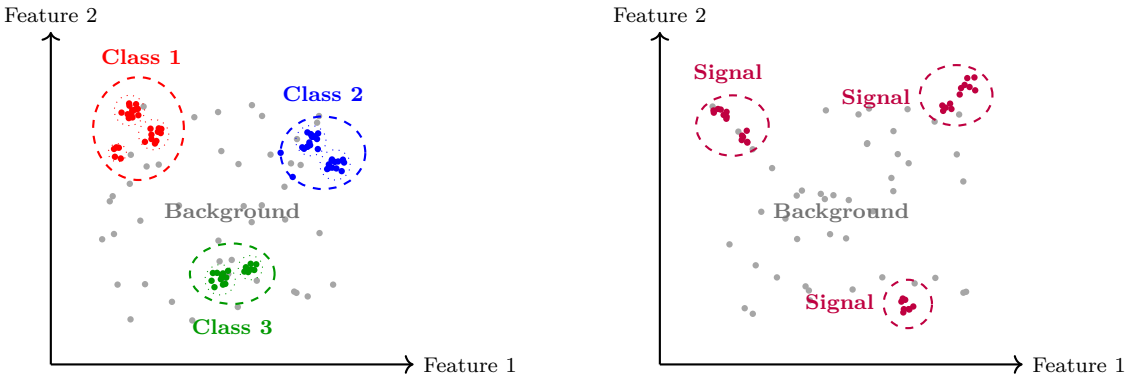
1. multi-label classification of events with the ten pair combinations of 1- and 3-prong τ decays (Fig. 29a), and
2. binary classification of Standard Model τ pair decay events (Fig. 29b).

In both cases, assume that the signal class comprises a relative minority of the training dataset. In the former case, it seems natural to assume that elements of each class comprise a single (though not necessarily uniform) cluster of points in the feature space. Therefore, as long as we weight the loss function appropriately, training the model should result in *not only* comparable performance on each class, but also comparable performance on random subsets of each class. That is, the learned information which leads to increased accuracy on some element of each class *generalizes* to the rest of the class. This is often the case when training a classifier, however it is not *always* the case. In the latter example, we have two classes; events in which there is a τ pair decay (signal) and all other events (background). There are now multiple discrete pairs of τ decay modes which are all defined as elements of the same class, hence elements of this class *do not* necessarily comprise a single cluster in the feature space. Consequently, the learned information required to accurately classify some elements of this class is considerably less likely to generalize to *all* elements, as the underlying physics may be completely distinct depending on the specific decay mode(s).

So far, this paradigm is true if we assume that the primary distinction which leads to separation in the feature space is the identity of the primary τ decay itself, but if we now consider the *secondary* decays of those daughter particles, it quickly becomes apparent that even within a particular pair of decay modes, we should still observe distinct subsets depending on the secondary decay modes. We can follow this through the entire particle shower and theoretically separate the set of all τ pair decays in any given mode into arbitrarily many subsets based on the shower substructure. However, the feature-space separation between events becomes increasingly small as we attempt to separate events further down the decay chain for two reasons; one physical, and one information-theoretic.

The physical reason is that each interaction deposits some energy in the material of the detector, resulting in a collection of secondary particles which have a total energy less than the energy of the primary particle at that vertex. Thus, phase space and kinematic constraints imply that the number of accessible future states is less than the number of states which were accessible at the previous vertex. This means that the states which can diverge the most in terms of their physical behavior (and thus in their representations in feature space) are the earliest states in the decay chain.

The information-theoretic reasoning is more straightforward; consider two events, each containing N discrete decays (time-ordered from 0 to N). For simplicity, imagine that we encode each of these



(a) Case 1: Multi-Label Classification

(b) Case 2: Binary Classification

Figure 29: Visualization of τ pair decay classes in feature space

events using one bit for each decay, such that each event is encoded by a set of ten bits, $\{v_1, \ldots, v_n\}$ and $\{w_1, \ldots, w_n\}$. The number of shared bits is given by

$$\sum_{i=1}^{N} b(v_i, w_i), \quad b(x, y) := \begin{cases} 1 & x = y \\ 0 & x \neq y \end{cases}, \tag{30}$$

hence the informational similarity between the two events is directly proportional to the number of shared decay vertices. In practice this sort of analysis is often considerably more involved, utilizing more precise measures of information such as mutual information ("MI") or Shannon entropy, but the same heuristics generally apply regardless.

Focal loss is a method for handling class imbalances in the training of neural networks without computing weights for each class in advance. Rather than calculating weights based on the relative prevalence of each class in the training dataset, focal loss utilizes a technique called *down-weighting*. This approach dynamically re-weights the loss function based on the difficulty of classifying a particular input. Formally, the focal loss is defined as

$$\mathcal{L}_{\text{FL}} = -(1 - p_t)^{\gamma} \log(p_t), \tag{31}$$

where p_t is the predicted probability that the input belongs to the class, and γ is the *focusing parameter*. Higher values of γ correspond to a more aggressive re-weighting, while $\gamma = 0$ corresponds to the standard categorical cross-entropy loss function (Eq. 29). The focal loss can be further augmented by re-inserting an overall weighting term α_t, so that

$$\mathcal{L}_{\text{FL}} = -\alpha_t (1 - p_t)^{\gamma} \log(p_t). \tag{32}$$

This is the form that we use in the training of our model, as implemented in Keras 3.

5.6.3 Quantization-Aware Training

We utilize per-weight mixed-precision quantization aware training (`HGQ2`) to train and quantize our model [44]. In quantization-aware training, the bit-widths for all parameters are masked during the forward pass to simulate the quantization (using the learned quantization parameters at the current epoch). This allows the network to optimize for the performance of the model *conditioned on the specific learned quantization parameters* at each step without artificially limiting the minimum step-size during the backward pass. The bit-widths can thus be simultaneously optimized during the training

cycle by attaching a *surrogate gradient* to them, making the bit-widths themselves differentiable parameters. In the `HGQ2` framework, the overall loss function $\mathcal{L}$ is thus given by

$$\mathcal{L} = \mathcal{L}_{\text{base}} + \beta_0 \text{ EBOPs} + \sum \text{bit-widths}, \tag{33}$$

where $\mathcal{L}_{\text{base}}$ is the original loss function, EBOPs ("Effective Bit Operations") is an estimate of on-chip resource utilization computed during training, and $\beta_0 \in \mathbb{R}^+$ determines how aggressively the model will be quantized (i.e. how to balance accuracy and on-chip resource utilization) [44]. In our case, we set $\beta_0 = 10^{-13}$.

5.7 Performance and Evaluation

The common trigger conditions currently used for τ selection generally achieve signal efficiencies between 80% and 90% in the range of trigger rates between 10 kHz and 30 kHz at the target luminosity (Table 6). In comparison, the neural-network based logic achieves signal efficiencies between 92% and 97% over the same range of trigger rates. In order to comprehensively evaluate of the model's performance, we consider four possible cases:

1. true positives T_p,

$$T_p = \#\left\{(\text{isTauSignal} == \text{True}) \wedge (\text{Output} \geq \text{Threshold})\right\}, \tag{34}$$

2. false positives F_p,

$$F_p = \#\left\{(\text{isTauSignal} == \text{False}) \wedge (\text{Output} \geq \text{Threshold})\right\}, \tag{35}$$

3. true negatives T_n,

$$T_n = \#\left\{(\text{isTauSignal} == \text{False}) \wedge (\text{Output} < \text{Threshold})\right\}, \tag{36}$$

4. false negatives F_n,

$$T_n = \#\left\{(\text{isTauSignal} == \text{True}) \wedge (\text{Output} < \text{Threshold})\right\}. \tag{37}$$

We then define the *true positive rate* as

$$\text{TPR} = \frac{T_p}{T_p + F_n}, \tag{38}$$

and the *false positive rate* as

$$\text{FPR} = \frac{F_p}{T_n + F_p}. \tag{39}$$

One way to visualize the accuracy of a classifier is by plotting the false positive rate against the true positive rate to obtain the *receiver-operating characteristic* (ROC) curve. The area under the curve (AUC) can then be used to estimate the overall performance. The weighted average of the neural network ROC across all modes is shown alongside the trigger bits defined in Table 6 in Fig. 30, with the mode-by-mode performance shown in Fig. 31.

Due to the highly unbalanced classes, it can also be useful to visualize the performance for each mode with the corresponding *precision-recall* curve. In this parameterization, we define the *precision*,

$$\text{P} = \frac{T_p}{T_p + F_p}, \tag{40}$$

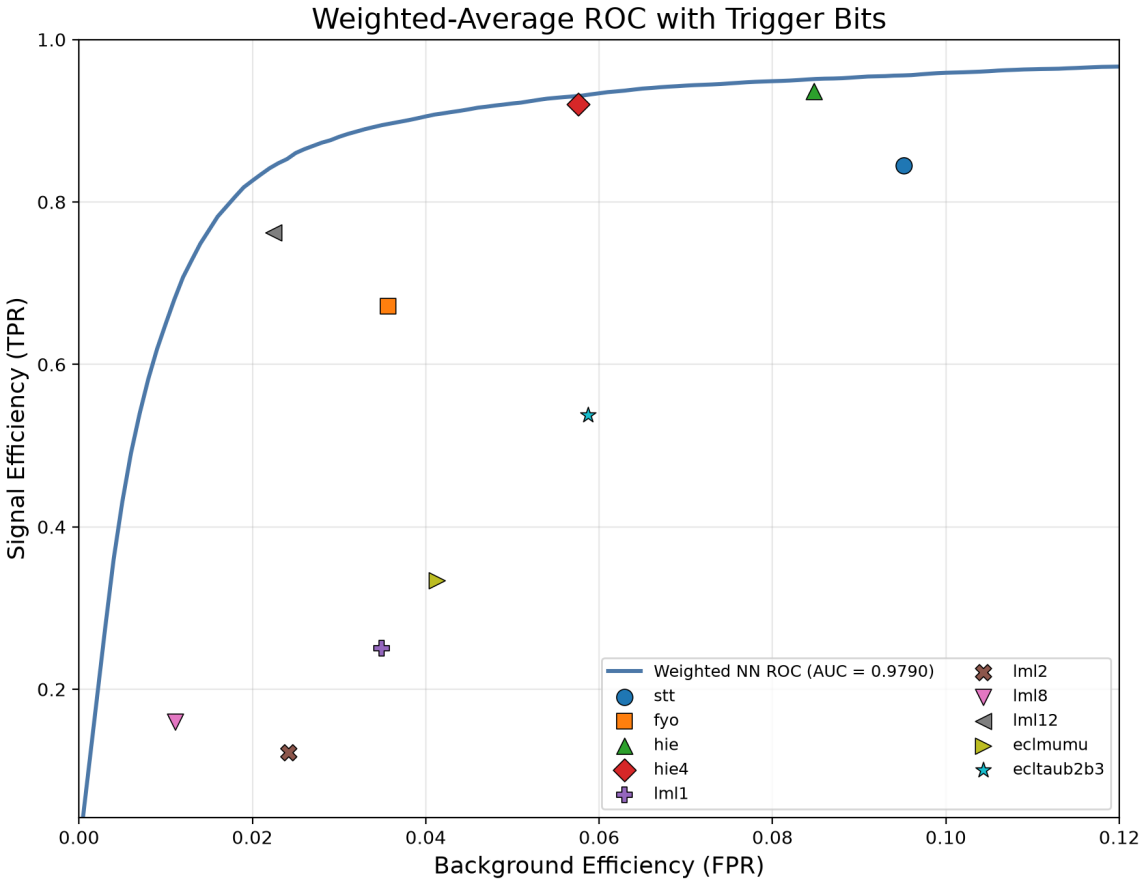
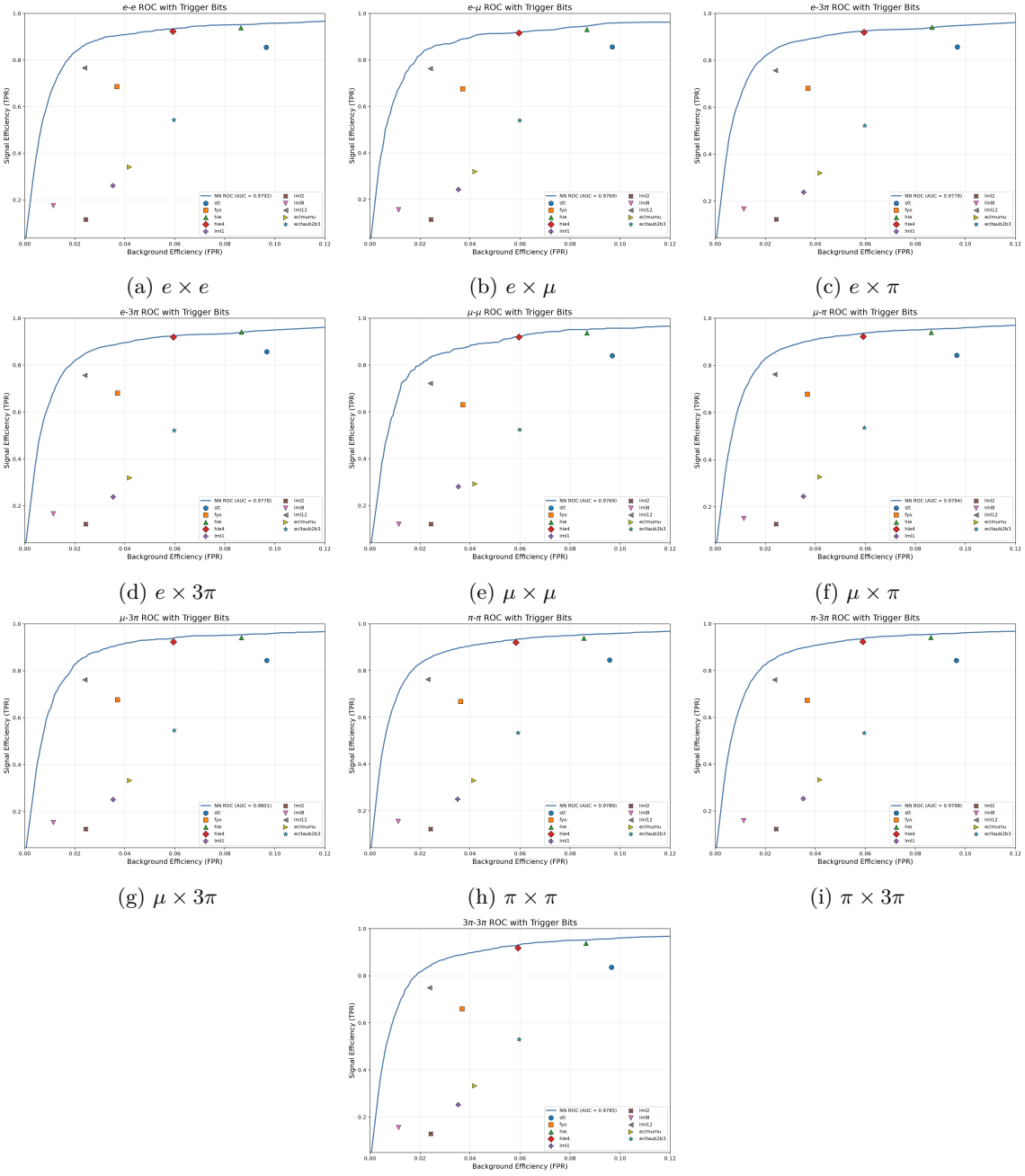


Figure 30: Comparison of the weighted average neural network ROC curve and efficiencies of the τ trigger bits defined in Table 6



(a) $e \times e$ (b) $e \times \mu$ (c) $e \times \pi$

(d) $e \times 3\pi$ (e) $\mu \times \mu$ (f) $\mu \times \pi$

(g) $\mu \times 3\pi$ (h) $\pi \times \pi$ (i) $\pi \times 3\pi$

(j) $3\pi \times 3\pi$

Figure 31: Mode-by-mode comparison of the neural network ROC curve and the τ trigger bits defined in Table 6

and the *recall* as the true positive rate (Eq. 38). The overall metric for the model's performance in this parameterization is the *average precision* (AP), which is measured as the weighted average of the precisions at each threshold,

$$\mathrm{AP} = \sum_n (R_n - R_{n-1}) P_n, \tag{41}$$

where P_n and R_n are the precision and recall at the nth threshold [45].

The precision recall curves for each decay mode are shown in Fig. 32; here we can more clearly see some discrepancies emerge. $\mu \times \mu$ remains the most challenging mode to accurately classify, requiring a relatively high false-positive rate to reach comparable signal efficiency to what an "easier" mode like $\pi \times \pi$ achieves with considerably lower background contamination. However, since the branching ratios for the more challenging modes are generally significantly lower, we can accept a larger false-positive rate in these modes in order to achieve the requisite signal efficiency without substantially increasing the total trigger rate. Additionally, we examine the activations at each the ten outputs to assess the

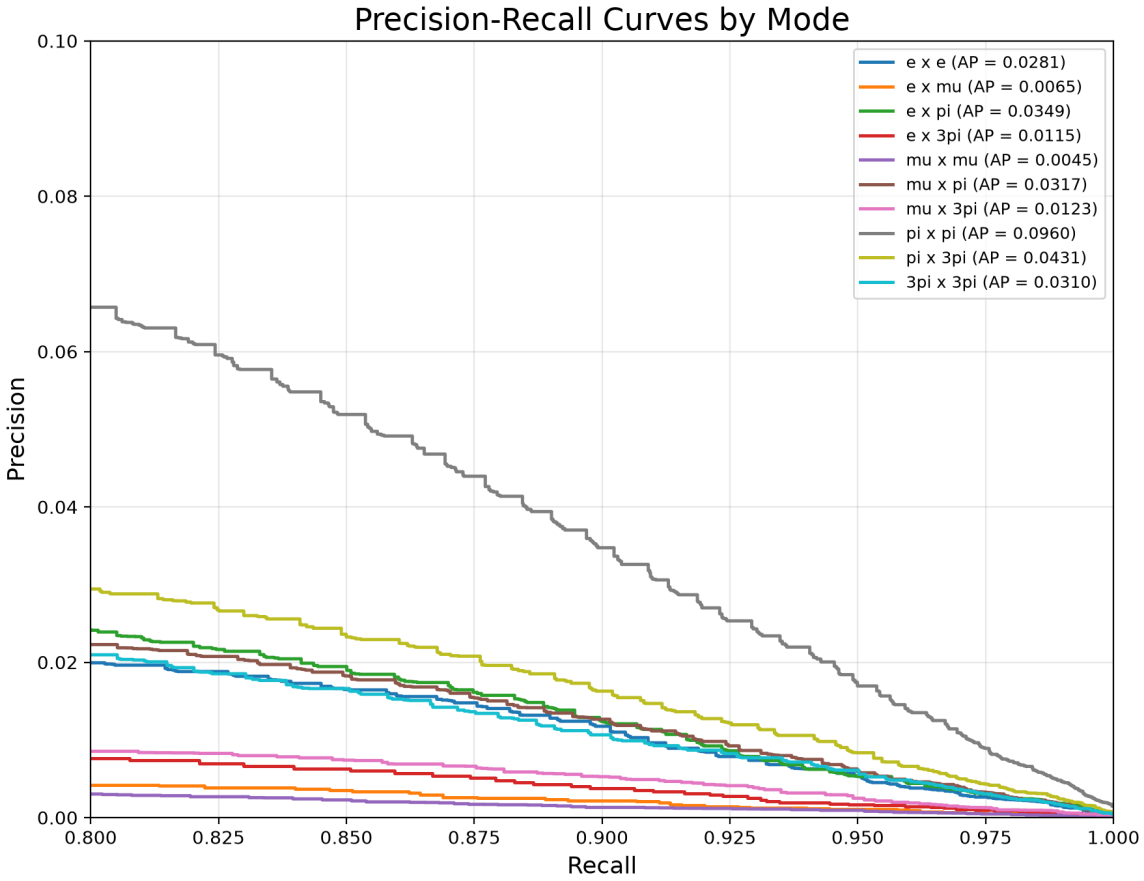
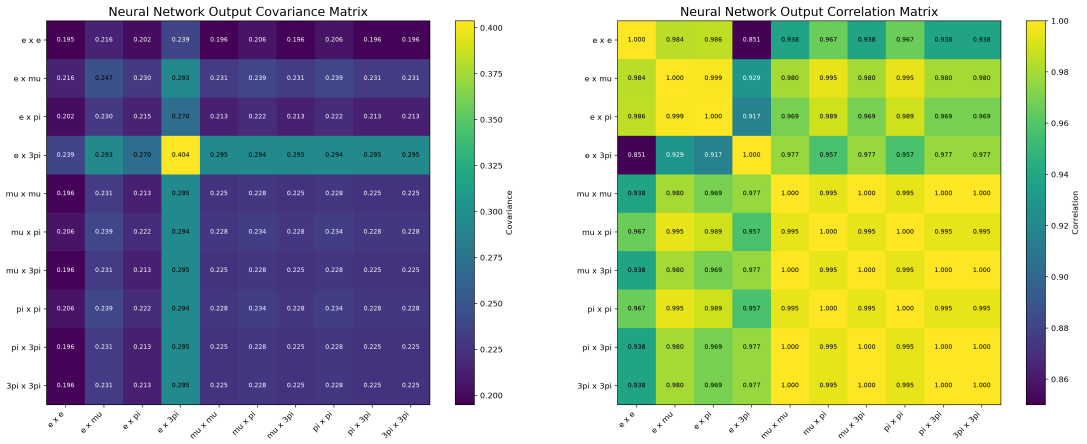


Figure 32: Precision-recall curve for for each pair of τ decay modes

degree to which the individual outputs are correlated with one another. This can be visualized as a *covariance* matrix as shown in 33a, or a normalized *correlation* matrix as in Fig. 33b. The fact that the outputs exhibit significant pairwise correlations indicates that rather than learning to effectively *separate* each of the different labels, the model is primarily learning a representation which enables it to distinguish inputs with *any* label from backgrounds. In this case, the ten outputs correspond to non-orthogonal subspaces of the activation manifold. This allows us to fine-tune the performance on



(a) Output covariance matrix

(b) Output correlation matrix

Figure 33: Neural network output correlation and covariance matrices

each label using separate cut thresholds, but not necessarily cleanly separate different decay modes. This is consistent with the way the data itself is labeled; events can, and often do have more than one label, hence the model learns to classify events non-exclusively, ultimately biasing the model's internal representations towards those which classify events primarily by whether they correspond to a τ pair decay or a background source. In this case, the naive goal of the model (event *classification*) is simply a means to an end (event *selection*).

The signal and background efficiencies for each mode across a range of cut thresholds between -3.0 and 0.0 are shown in Fig. 34 and Fig. 35. We also plot the projected trigger rate (both per-output and total) as a function of the cut threshold in Fig. 36. In the current GRL firmware, we budget a total of 10 kHz for this trigger bit. We optimize the thresholds for each mode for the highest possible efficiency which can be achieved on all modes simultaneously given this limitation. Practically, we sample signal efficiencies by scanning across a range of thresholds for each mode. For each efficiency target, we then determine the trigger rate in the case where we set the thresholds such that each mode achieves that efficiency (within a small margin of error). At this point, we can simply select the signal efficiency which results in the trigger rate closest to the desired rate, and thus determine the corresponding thresholds for each output. For a limit of 10 kHz, the corresponding accuracy across all modes is determined to be approximately 90%. If we instead scale up to the full DAQ limit of 30 kHz, the accuracy across all modes reaches approximately 97%.

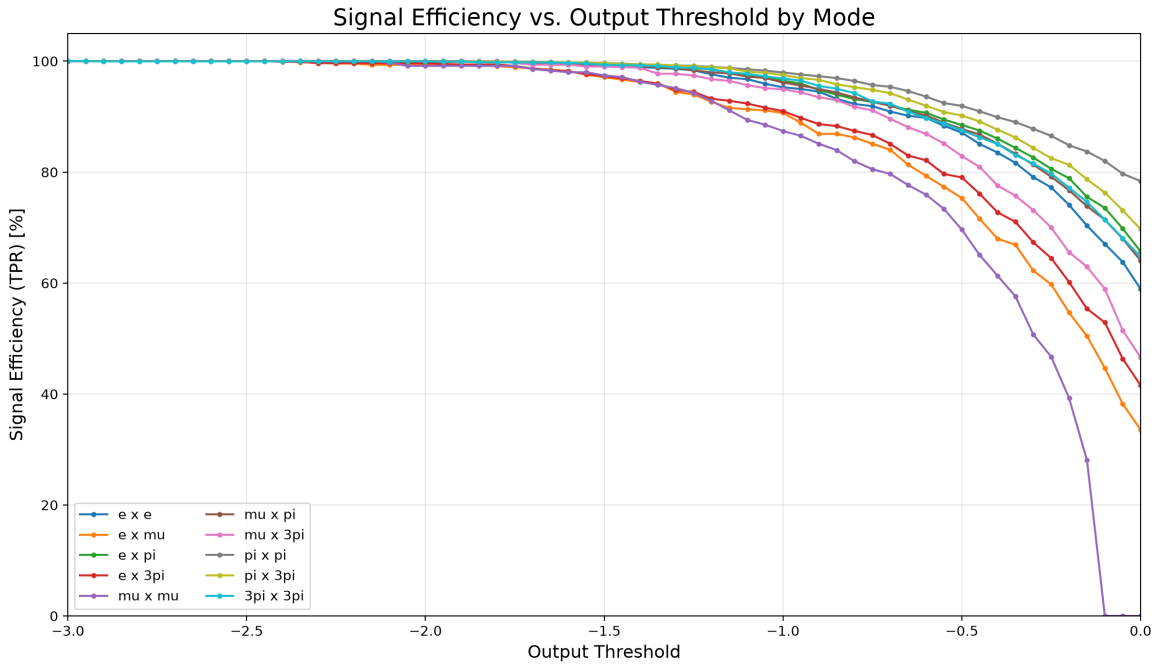


Figure 34: Signal efficiency vs. neural network output threshold for each pair of τ decay modes

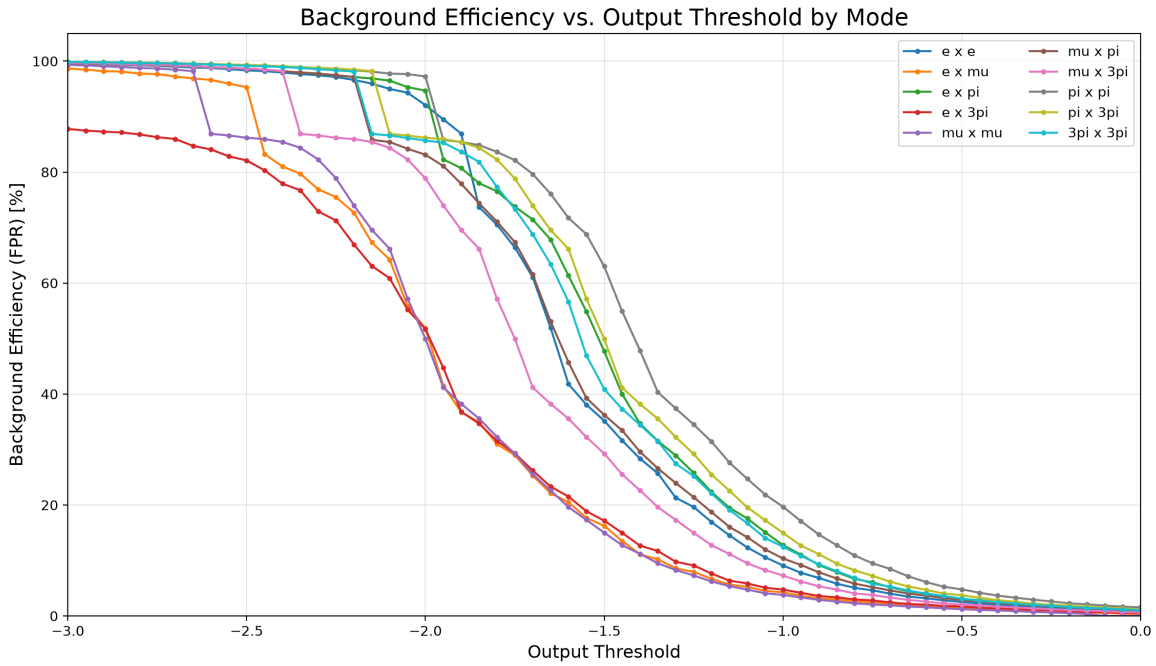


Figure 35: Background efficiency vs. neural network output threshold for each pair of τ decay modes

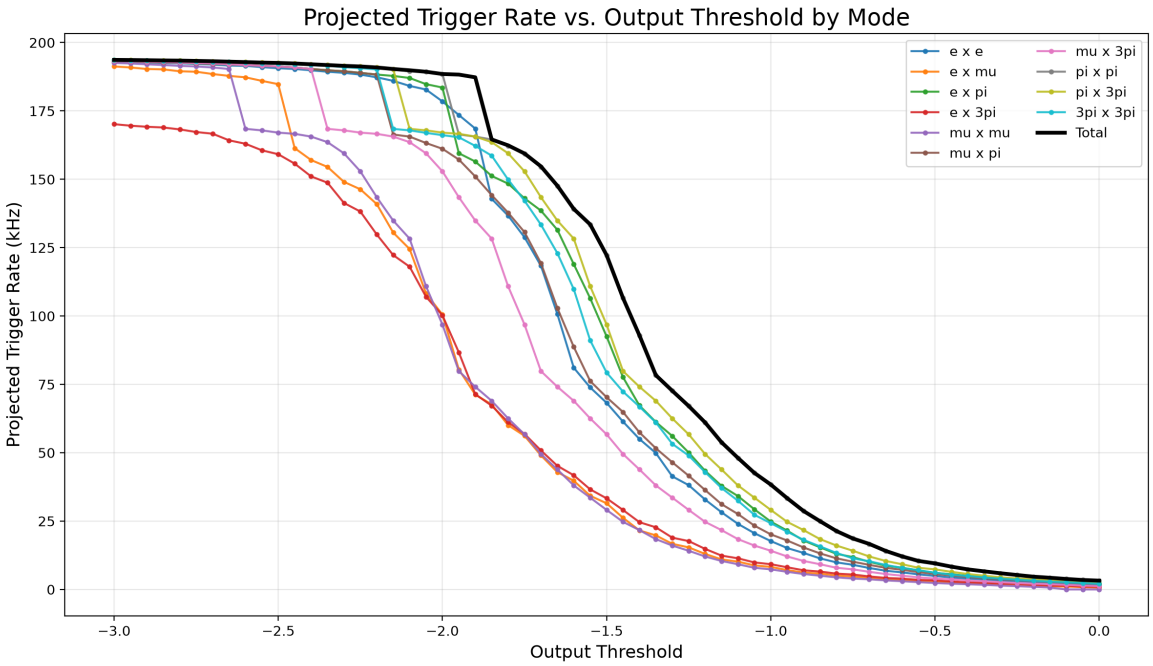


Figure 36: Projected trigger rate vs. neural network output threshold for each pair of τ decay modes

6 On-Chip Deployment

6.1 Hardware Target

The GRL algorithm is implemented on an AMD Virtex UltraScale (XCVU080) FPGA as part of the Belle II Universal Trigger Board 4 (UT4). The hardware specifications for this chip are given in Table 10. Historically, the primary bottleneck in the implementation of complex neural network-based

Table 10: Technical specifications for the AMD XCVU080 FPGA [46]

Resource	Count
Block RAM (BRAM)	2842
Digital Signal Processor (DSP)	672
Flip-Flop (FF)	891424
Lookup Table (LUT)	445712

algorithms on FPGAs has been the number of DSPs required to perform the matrix-vector multiplication operations. However, as we will discuss in Section 6.2.1, this limitation is has been rendered largely obsolete for heavily quantized models due to the development of distributed arithmetic-based synthesis tools. Therefore, the most significant restrictions on model size now come from the number of available LUTs and routing between them. In general, we require $\lesssim 50\%$ LUT utilization for the full GRL firmware in order to achieve timing closure, which corresponds to $\lesssim 20\%$ LUT utilization for the neural network. In this case, that corresponds to $\lesssim 222856$ LUTs for the full GRL firmware, or $\lesssim 89142$ LUTs for the neural network itself. Additionally, the network must satisfy latency and pipeline constraints; the total latency cannot exceed 470 ns and the throughput must be greater than four system clock cycles (127.216 MHz/4 = 31.804 MHz), which follows from the frequency of ECLTRG inputs to the GRL.

6.2 High-Level Synthesis

Standard workflows for high-level synthesis (HLS) of neural networks in particle physics frequently employ the `hls4ml` Python package for the generation of C++ HLS code from models developed with Keras or PyTorch [47–49]. This can then be used to generate HDL/VHDL code using proprietary electronic design automation (EDA) suites such as AMD Xilinx Vivado and Vitis. Recent work in this area has shown that additional optimization of the low-level implementation of vector-matrix multiplication operations can dramatically reduce the resources required by the final model, leading to the development of the `da4ml` package for high-level synthesis with *distributed arithmetic*. To this end, we opt to use `hls4ml` with the drop-in `distributed_arithmetic` strategy from `da4ml` for high-level synthesis of the `HGQ` model [50].

6.2.1 Distributed Arithmetic

As discussed in Section 4, the fundamental operation involved in neural-network inference is matrix-vector multiplication. For a vector $v \in \mathbb{R}^n$ and matrix $M \in \mathbb{R}^{m\times n}$, the product Mv involves $n \times m$ multiplication operations and $(n-1) \times m$ addition operations. At a hardware level, this operation is a implemented as stream of multiply-accumulate (MAC) operations:

$$y_i = (M_{i,1}v_1) + (M_{i,2}v_2) + \cdots = \sum_{j=1}^{n} M_{i,j}v_j. \tag{42}$$

where y_i is the ith element of the resulting vector $y \in \mathbb{R}^m$. These operations are generally implemented using DSPs, since each DSP contains a dedicated multiplier, adder, and accumulator register as shown in Fig. 37.

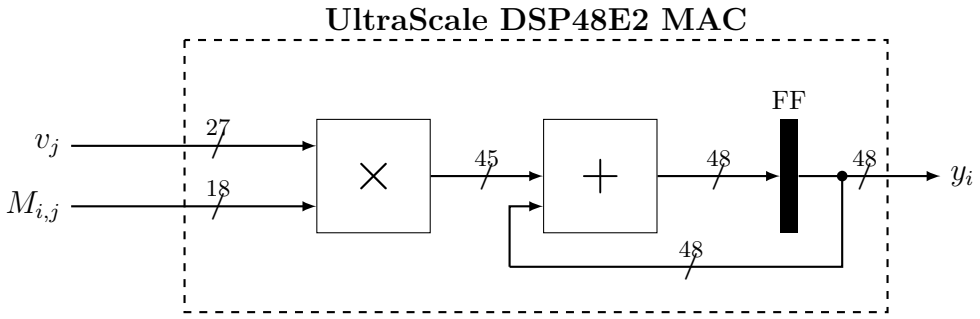


Figure 37: Hardware implementation of MAC for matrix-vector multiplication

Distributed arithmetic is a method which maps fixed-width matrix-vector multiplication operations to mathematically equivalent sequential bit-shift and addition operations (Fig. 38). Instead of operating on the full input vector v_j, distributed arithmetic operates on *bit slices* $v[k]$, defined as is the kth bit taken from all n elements of v simultaneously. By eliminating the need for direct multiplication of the elements $M_{i,j}$ and v_j, the fundamental operations are reduced to those which can be efficiently implemented using LUTs, shift registers, and adders, rather than DSPs. Since the number of DSPs

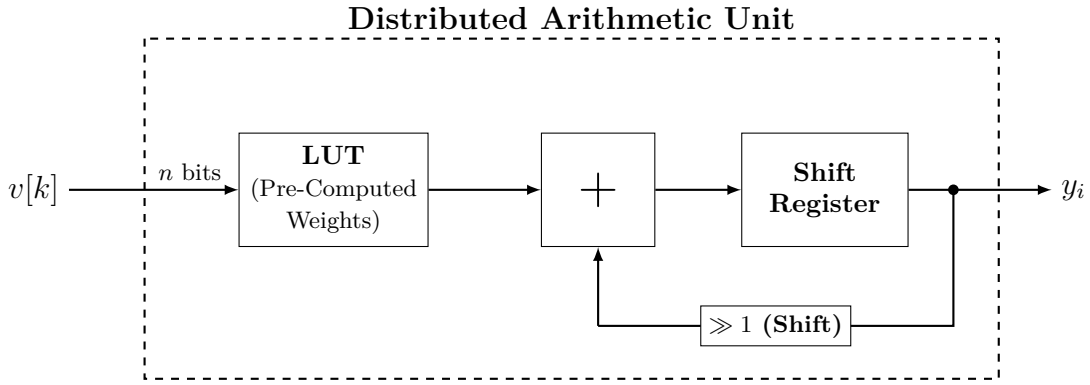


Figure 38: Hardware implementation of distributed arithmetic for matrix-vector multiplication

available on most FPGAs is typically far smaller than the number of LUTs, this significantly increases the size of model which can be implemented on a given chip. The `distributed_arithmetic` synthesis strategy from `da4ml` implements this algorithm within the `hls4ml` high-level synthesis framework. In practice, this typically reduces the number of DSPs used by the model to 0 as long as the original model was quantized appropriately.

6.3 Firmware Implementation

We instantiate the neural network itself as a Vivado Intellectual Property (IP) core within the full GRL firmware structure. IP cores are pre-designed modular logic blocks which can be seamlessly integrated into a larger firmware design. As mentioned in Section 3.11.3, the GRL receives ECL cluster information directly from the ETM at each clock cycle with a frequency of 8 MHz. This cluster information is not sorted by default, thus we first sort the clusters by energy using an odd –

even transposition sorting algorithm. This sorting process is pipelined such that each stage operates as an independent comparator level of a descending sorting network. The process is distributed across five synchronous pipeline stages, where each energy comparison synchronously swaps the corresponding θ, ϕ, and t_{avg} data. Once the inputs are fully sorted, the top six clusters are extracted, padded so that all parameter bit-widths match, and concatenated into a 288-bit vector.

Since the sorting logic and neural network run at a clock frequency of 127 MHz, data is read at each clock cycle irrespective of whether there is a valid new event at that cycle or not. Therefore, we evaluate the presence of a valid event by checking if the sorted cluster with the highest energy is non-zero. To prevent the IP from repeatedly processing the same event in consecutive clock cycles, we assign a state flag which ensures that when a new non-zero event is detected, the module generates a one-cycle pulse which prompts the neural network to perform inference on the current input vector. We also require that the original input features are synchronized with the neural network outputs for the Belle II Link (B2L) readout [51]. The input vector is held in a synchronous delay line until inference is completed, at which point a three-stage pipeline is sequentially triggered.

The first stage registers the network outputs directly from the IP. Stage 2 then compares these outputs against ten distinct thresholds (one for each class) to produce a set of 10 boolean values. Stage 3 takes the logical OR of these values; if any individual threshold is exceeded, the τ decision flag is set to 1. If none of the thresholds are exceeded, the flag is set to 0. Finally, the fully pipelined neural network outputs, the boolean values for each class, the τ flag, and the delayed original inputs are simultaneously captured by the B2L registers. The data sent to the GDL is simply the value of the τ flag for each event, which is then used to make the final Level-1 trigger decision. The data flow for the full τ trigger logic is shown in Fig. 39.

6.4 Latency and Utilization Estimates

The post-synthesis resource utilization and latency estimates results for the standalone neural network IP are reported in Tables 11 and 12. We also compare these results to those obtained using `hls4ml` *without* distributed arithmetic (Tables 13 and 14). In this particular case, since the model is already aggressively quantized, the majority of the observed improvements are seen in LUT utilization and latency. For larger models which utilize more DSPs by default, the impacts of distributed arithmetic-based HLS strategies are even more pronounced. The post-synthesis resource utilization for the full GRL firmware (including the new IP) is reported in Table 15, while the post-implementation values are given in Table 16.

6.5 Behavioral Simulation

We first compare the model outputs in register-transfer level (RTL) behavioral simulation with those obtained from the original Keras model for a sample of 10^3 random inputs (Fig. 40). This is a low-level simulation of the physical circuit as routed on the target hardware. We find that the outputs from the RTL simulation match the software results for all tested inputs. We also perform the same comparison for a sample of 10^3 real inputs captured during a physics run (Fig. 41), with the Keras model continuing to show perfect agreement with the RTL simulation for all events.

6.6 Firmware Validation

After completing the preliminary validation steps in simulation, the new trigger bit was enabled to allow for a thorough analysis of performance and efficiency on collision data gathered during Belle II physics runs in June 2026. To validate the performance of the FPGA firmware after installation to the Belle II UT4, we capture the inputs and outputs to the neural network during a physics run. The captured inputs are then fed back into the Keras model and the resulting outputs are compared to the captured outputs from the firmware implementation. The results of this comparison are shown in Fig. 42. For the 10^4 events in the sample, the firmware outputs match those from the corresponding

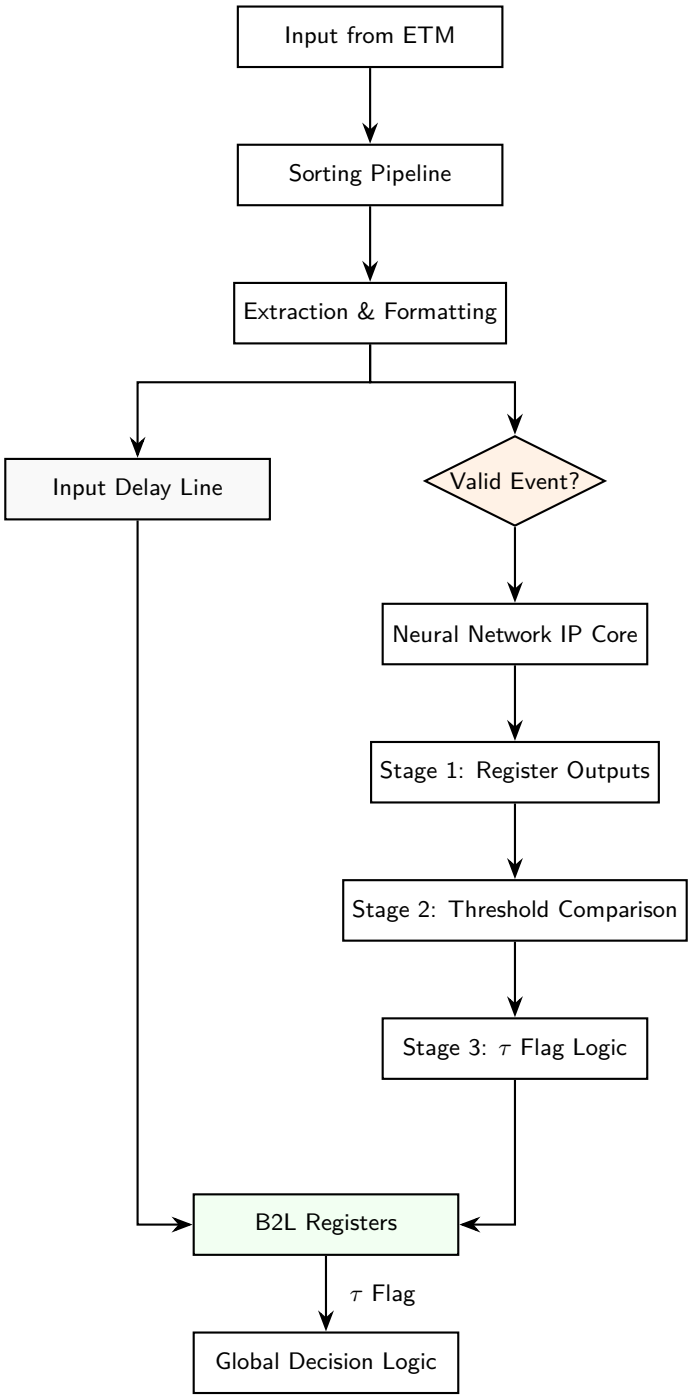


Figure 39: Structure of the τ event classification algorithm in the Global Reconstruction Logic

Table 11: Post-synthesis resource utilization estimates for the neural network IP (with distributed arithmetic)

Name	BRAM	DSP	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	-	0	5522	-
FIFO	-	-	-	-	-
Instance	-	-	0	1286	-
Memory	-	-	-	-	-
Multiplexer	-	-	0	36	-
Register	-	-	1144	-	-
Total	0	0	1144	6844	0
Available	2842	672	891424	445712	0
Utilization (%)	0	0	0	1	0

Table 12: Post-synthesis latency estimates for the neural network IP (with distributed arithmetic)

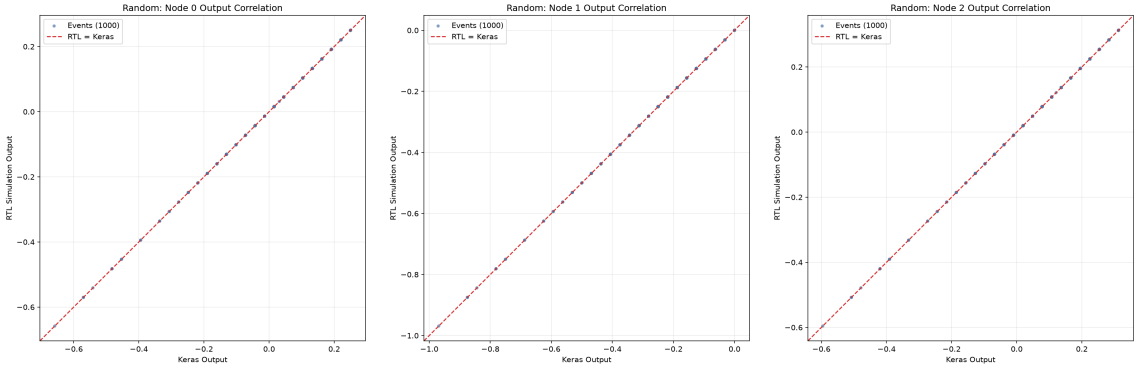
Measure	Latency (Min.)	Latency (Max.)
Clock Cycles	6	6
Absolute	47 ns	47 ns

Table 13: Post-synthesis resource utilization estimates for the neural network IP (without distributed arithmetic)

Name	BRAM	DSP	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	-	0	944	-
FIFO	-	-	-	-	-
Instance	-	1	486	10638	-
Memory	-	-	-	-	-
Multiplexer	-	-	-	36	-
Register	-	-	1051	-	-
Total	0	1	1537	11618	0
Available	2842	672	891424	445712	0
Utilization (%)	0	0	0	2	0

Table 14: Post-synthesis latency estimates for the neural network IP (without distributed arithmetic)

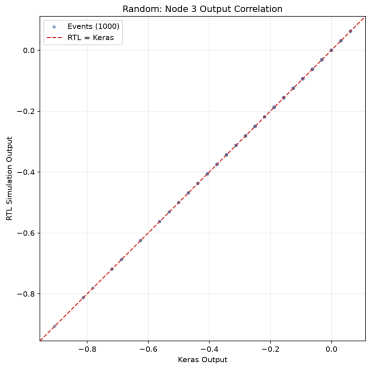
Measure	Latency (Min.)	Latency (Max.)
Clock Cycles	9	9
Absolute	71 ns	71 ns



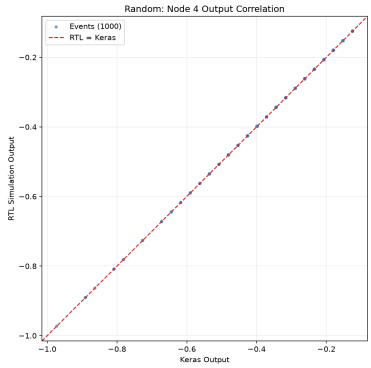
(a) $e \times e$

(b) $e \times \mu$

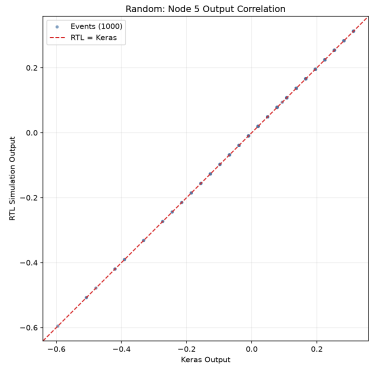
(c) $e \times \pi$



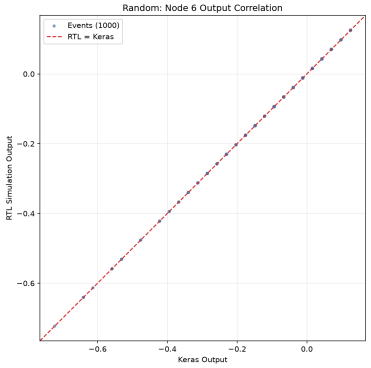
(d) $e \times 3\pi$



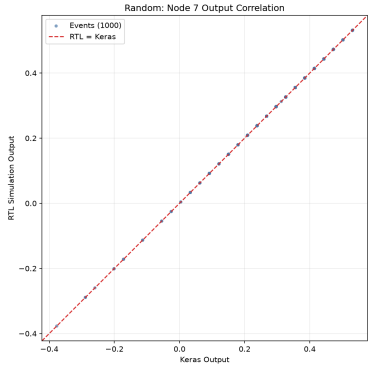
(e) $\mu \times \mu$



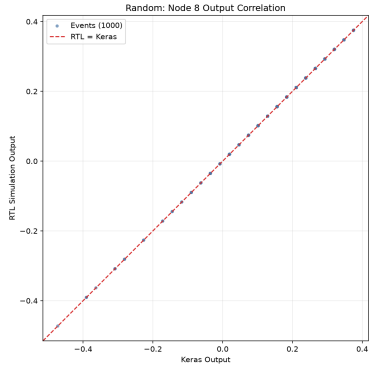
(f) $\mu \times \pi$



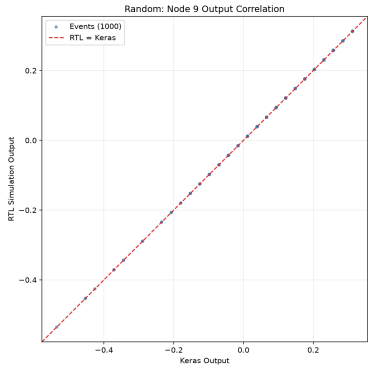
(g) $\mu \times 3\pi$



(h) $\pi \times \pi$

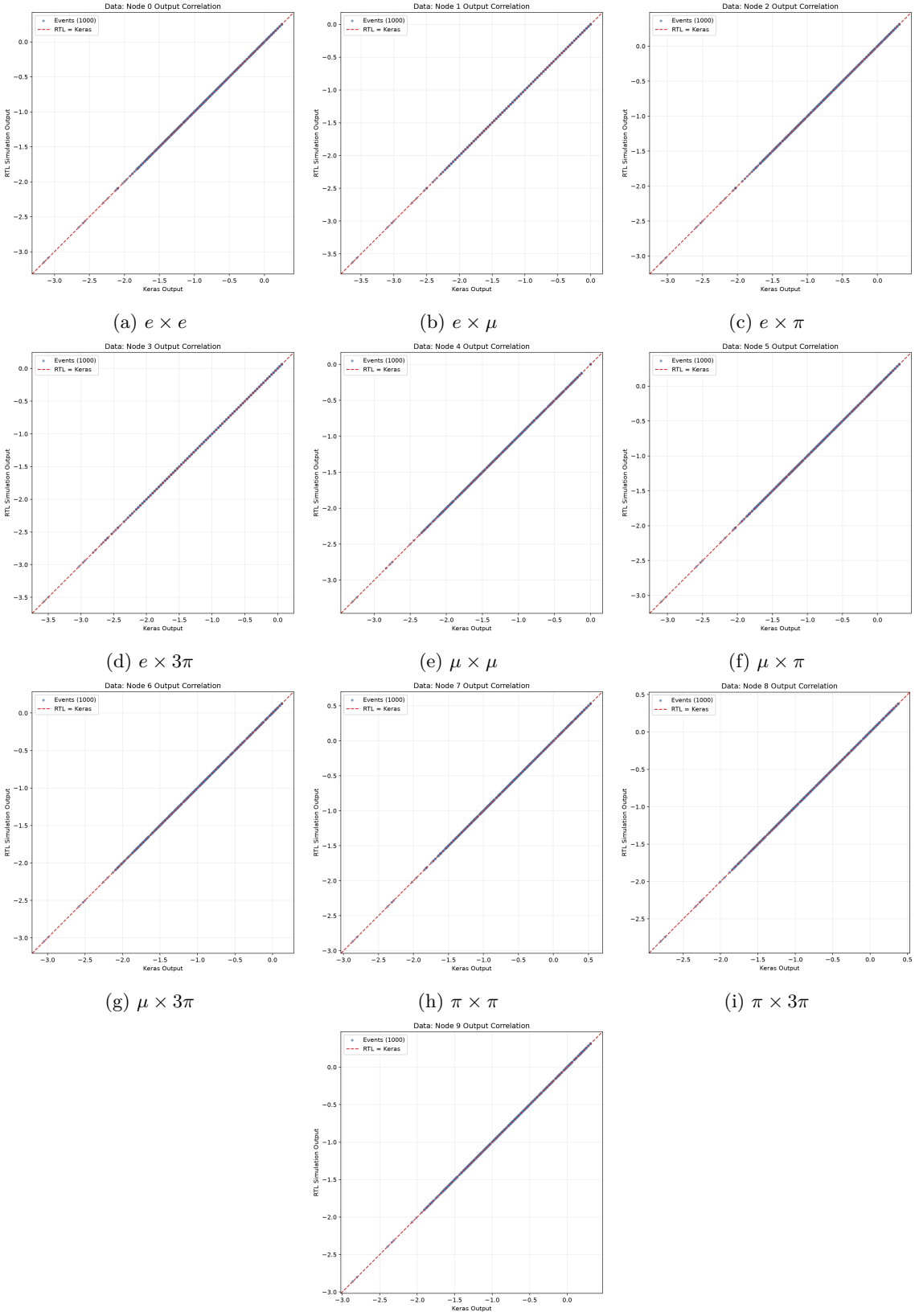


(i) $\pi \times 3\pi$



(j) $3\pi \times 3\pi$

Figure 40: A comparison of the outputs from Vivado RTL simulation and Keras for a sample of 10^3 random event-like inputs



(a) $e \times e$ (b) $e \times \mu$ (c) $e \times \pi$

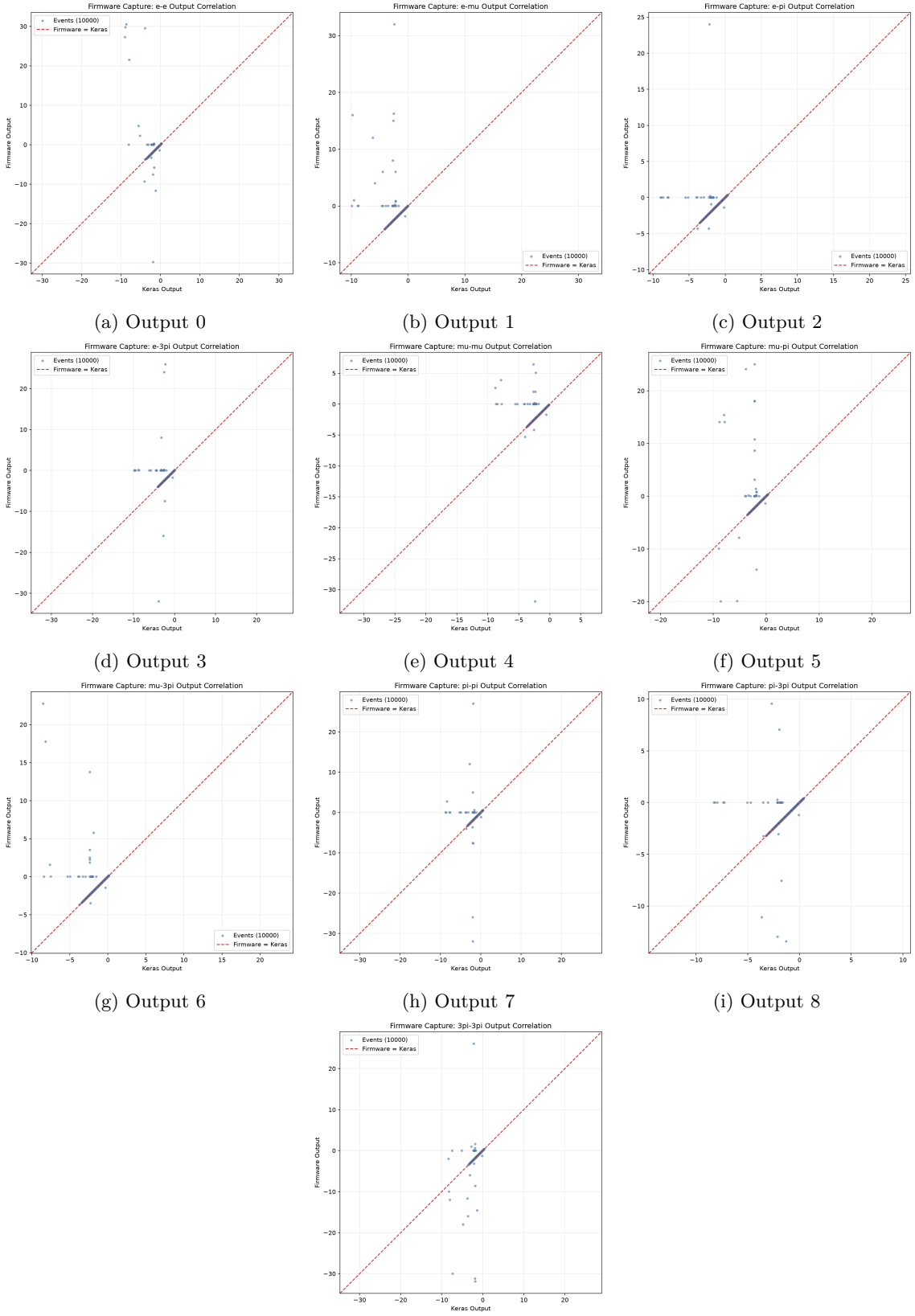
(d) $e \times 3\pi$ (e) $\mu \times \mu$ (f) $\mu \times \pi$

(g) $\mu \times 3\pi$ (h) $\pi \times \pi$ (i) $\pi \times 3\pi$

(j) $3\pi \times 3\pi$

Figure 41: A comparison of the outputs from the Vivado RTL simulation and Keras for a sample of 10^3 inputs captured during physics runs

Keras model in 9968 (99.68%). This is consistent with the expected behavior of a quantized neural network, with the discrepancies attributable to values which fall outside the range of the calibration sample (likely due to run-by-run differences in accelerator conditions).



(a) Output 0

(b) Output 1

(c) Output 2

(d) Output 3

(e) Output 4

(f) Output 5

(g) Output 6

(h) Output 7

(i) Output 8

(j) Output 9

Figure 42: A comparison of the outputs captured from the GRL firmware and Keras for a sample of 10^4 events

Table 15: Post-synthesis resource utilization estimates for the full GRL firmware

Name	**BRAM**	**DSP**	**FF**	**LUT**	**URAM**
Total	235	0	160974	168232	0
Available	2842	672	891424	445712	0
Utilization (%)	8.27	0.00	18.06	37.74	0

Table 16: Post-implementation resource utilization for the full GRL firmware

Name	**BRAM**	**DSP**	**FF**	**LUT**	**URAM**
Total	1006	0	167474	159820	0
Available	2842	672	891424	445712	0
Utilization (%)	35.40	0.00	18.79	35.86	0

7 Physics Performance

7.1 Trigger Rate

We monitored the behavior of the new trigger bit (`ecltaunn`) during a Belle II physics run on June 28, 2026. During this run, SuperKEKB achieved a peak luminosity of 4.66×10^{34} cm^{-2}s^{-1} with an average delivered luminosity of 4.38×10^{34} cm^{-2}s^{-1}. The average Level-1 trigger rate during this run was 10.3 kHz, with a maximum of 12.3 kHz. The average trigger rate of the `ecltaunn` bit over the same period was 1.8 kHz, with a maximum of 2.6 kHz. The trigger rate for the `ecltaunn` bit is plotted against the total Level-1 in Fig. 43. When evaluated alongside the other trigger bits utilized

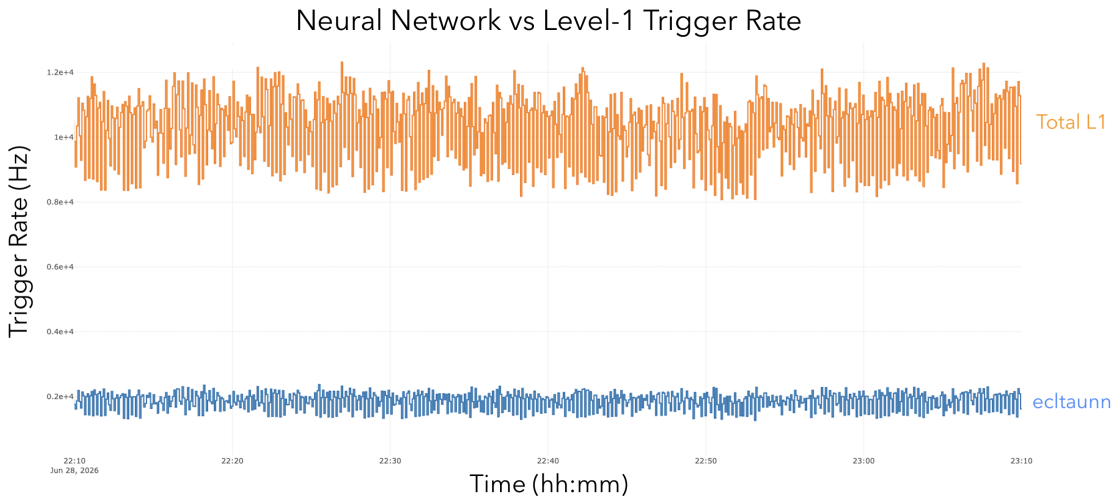


Figure 43: Trigger rate of the `ecltaunn` bit and the total Level-1 rate over a 1-hour period with an average luminosity of 4.38×10^{34} cm^{-2}s^{-1}

for τ event selection as defined in Table 6, approximately 35% of the average rate associated with the `ecltaunn` bit is exclusive (i.e. not triggered by any of other bits), while the remaining $\sim$ 66% are associated with events already triggered by existing bits. With `hie` and `ecltaub2b3` disabled (as will be the case in future physics runs), the exclusive rate from `ecltaunn` increases slightly to around 38% of the total. The inclusive and exclusive distributions for each of the ECL cluster parameters E, θ, ϕ, and t_{avg} are shown in Figs. 44 - 47.

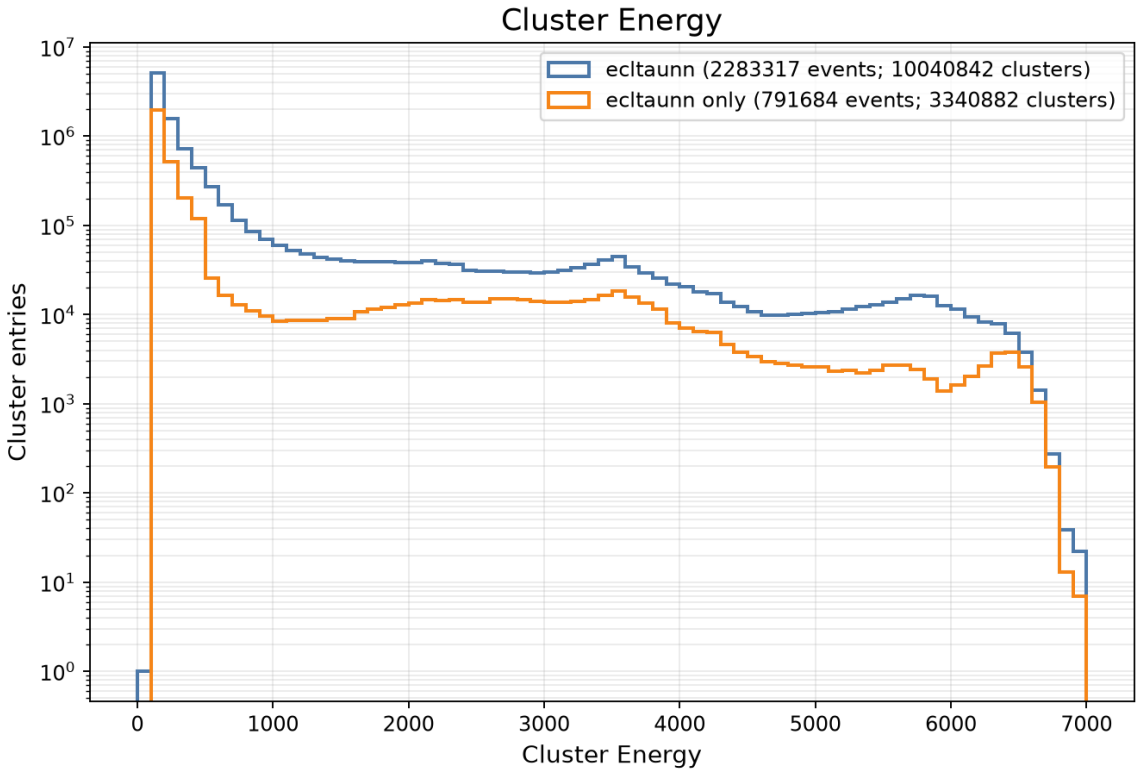


Figure 44: Inclusive and exclusive distribution of the cluster energy E for all clusters in a sample of $2,283,317$ events selected by the `ecltaunn` trigger

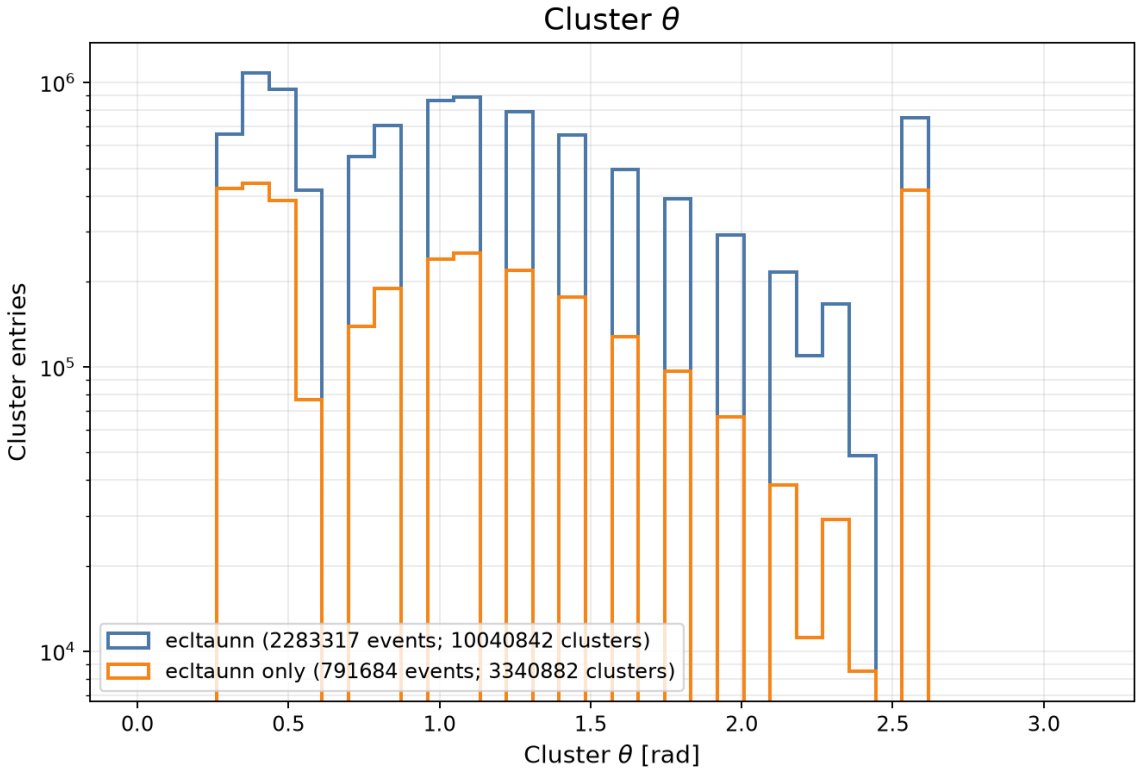


Figure 45: Inclusive and exclusive distribution of the polar angle θ for all clusters in a sample of $2,283,317$ events selected by the `ecltaunn` trigger

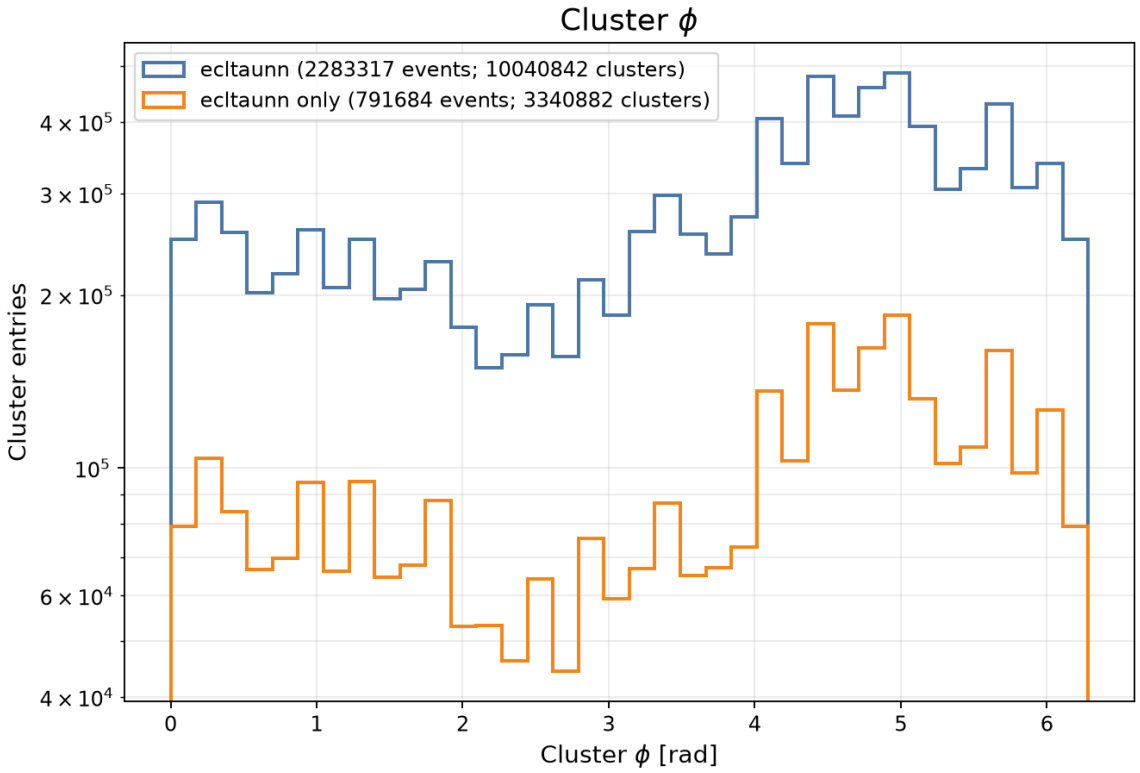


Figure 46: Inclusive and exclusive distribution of azimuthal angle ϕ for all clusters in a sample of $2,283,317$ events selected by the `ecltaunn` trigger

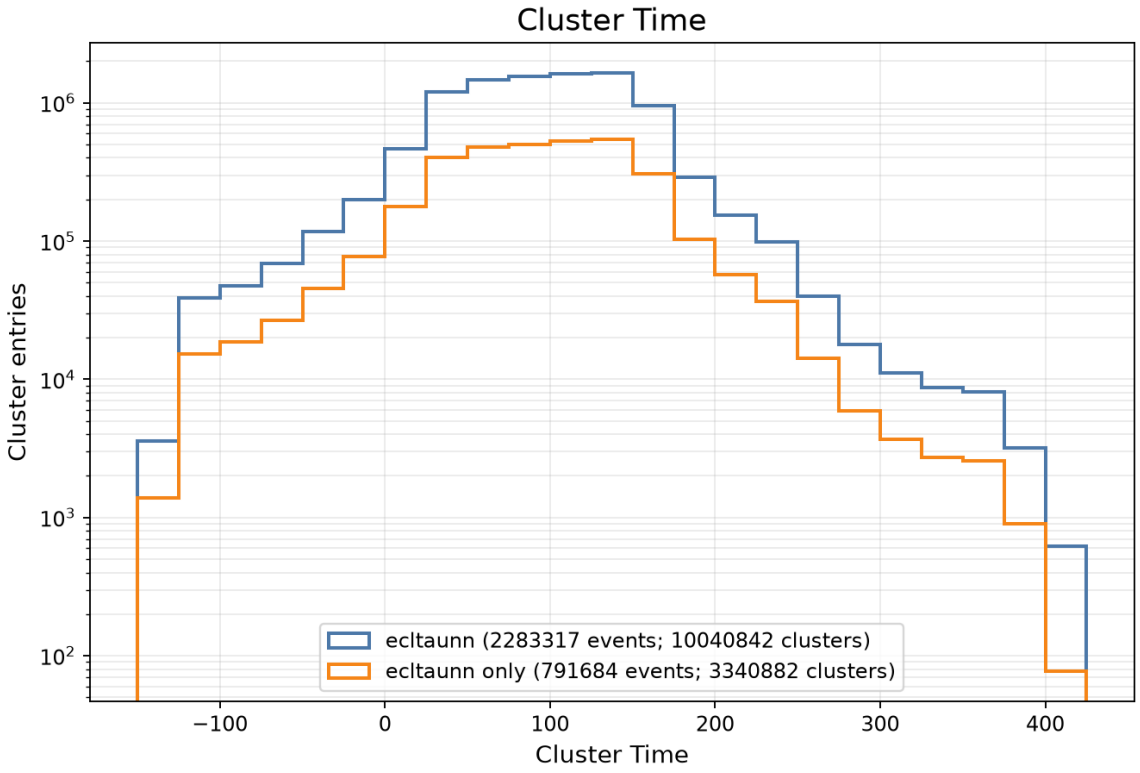


Figure 47: Inclusive and exclusive distribution of the average cluster hit time t_{avg} for all clusters in a sample of $2,283,317$ events selected by the `ecltaunn` trigger

7.2 Relative Trigger Efficiency

Let X be the set of all events, $Y \subset X$ the set of events containing a τ pair decay, and $Z \subset X$ be the subset of events meeting a specific reference selection criteria targeting τ decays. The inclusive efficiency of this selection criteria is

$$e_Z := \frac{|Z|}{|Y|}. \tag{43}$$

Next, we define two triggers, `bit1` and `bit2` with *true* efficiencies e_1 and e_2 on the set of τ pair events Y. The set of τ events selected by `bit1` and `bit2` are subsets $B_1 \subseteq X$ and $B_2 \subseteq X$, where

$$|B_1 \cap Y| = e_1|Y| \quad \text{and} \quad |B_2 \cap Y| = e_2|Y| \tag{44}$$

respectively. Evaluated on the set Z, we can define efficiencies e_1' and e_2' such that

$$|B_1 \cap Z| = e_1'|Z| \quad \text{and} \quad |B_2 \cap Z| = e_2'|Z|. \tag{45}$$

These efficiencies clearly depend on the selection criteria used to define the set Z. If the triggers `bit1` and `bit2` are perfectly statistically independent of the selection criteria,

$$\epsilon_{Z|B1} = e_Z \implies e_1' = e_1. \tag{46}$$

where

$$\epsilon_{Z|B1} := \frac{|Z \cap B_1|}{|Y \cap B_1|}. \tag{47}$$

This implicitly assumes that Z acts as a perfectly uniform random scalar reduction on B_1. In practice, $Z \cap Y$ is generally a kinematically biased subset of Y, therefore the triggers are *not* independent of the selection criteria.

Now consider the conditional probability $P(\texttt{bit1}|\texttt{bit2})$ on the set Z:

$$P(\texttt{bit1}|\texttt{bit2})\Big|_Z := \frac{|(B_1 \cap B_2) \cap Z|}{|B_2 \cap Z|}. \tag{48}$$

This is the proportion of events in $|B_2 \cap Z|$ that are *also* in B_1. If the two triggers are statistically independent, then

$$P(\texttt{bit1}|\texttt{bit2})\Big|_Z = e_1'. \tag{49}$$

More generally, we can draw a number of conclusions regarding this quantity from the efficiencies e_1' and e_2' in the case where the triggers are *not* perfectly independent:

if $e_1' = 1$, then $P(\texttt{bit1}|\texttt{bit2})\big|_Z = 1$,

if $e_2' = 1$, then $P(\texttt{bit1}|\texttt{bit2})\big|_Z = e_1'$,

if $e_1' + e_2' \leq 1$, then $\min\left\{P(\texttt{bit1}|\texttt{bit2})\big|_Z\right\} = 0$,

if $e_1' + e_2' > 1$, then $\min\left\{P(\texttt{bit1}|\texttt{bit2})\big|_Z\right\} = \frac{e_1'+e_2'-1}{e_2'} > 0$,

if $e_1' \leq e_2'$, $\max\left\{P(\texttt{bit1}|\texttt{bit2})\big|_Z\right\} = \frac{e_1'}{e_2'}$, and

if $e_1' > e_2'$, $\max\left\{P(\texttt{bit1}|\texttt{bit2})\big|_Z\right\} = 1$.

Written more compactly, we have

$$\max\left(0, \frac{e_1' + e_2' - 1}{e_2'}\right) \leq P(\texttt{bit1}|\texttt{bit2})\Big|_Z \leq \min\left(1, \frac{e_1'}{e_2'}\right). \tag{50}$$

In order to express this in terms of the *true* efficiencies e_1 and e_2, note that

$$e_1' = \frac{\epsilon_{Z|B1}}{e_Z} e_1 \quad \text{and} \quad e_2' = \frac{\epsilon_{Z|B2}}{e_Z} e_2, \tag{51}$$

therefore

$$\max\left(0, \frac{\epsilon_{Z|B1} e_1 + \epsilon_{Z|B2} e_2 - e_Z}{\epsilon_{Z|B2} e_2}\right) \leq P(\texttt{bit1}|\texttt{bit2})\Big|_Z \leq \min\left(1, \frac{\epsilon_{Z|B1} e_1}{\epsilon_{Z|B2} e_2}\right). \tag{52}$$

We evaluate the performance of the `ecltaunn` bit using combinations of reference bits which do not utilize the ECL cluster information for reconstruction. In this case, we use the existing CDC single- and two-track bits. From 2026 onward, `stt` and `fyo` were replaced by `sttz` and `fzo`, which also depend on the number of reconstructed tracks in the CDC, but include a tighter z-vertex cut to prevent triggering on tracks not originating from the IP. Hence we can evaluate the `ecltaunn` bit against this combination to indirectly estimate the efficiency of `ecltaunn`. In practice, we compute the relative efficiency of each ECL-based trigger bit as

$$P(\texttt{bit}|\texttt{fzo} \vee \texttt{sttz}) := \frac{\#\{\texttt{bit} \wedge (\texttt{fzo} \vee \texttt{sttz})\}}{\#\{\texttt{fzo} \vee \texttt{sttz}\}}, \tag{53}$$

evaluated on the set of events selected by the original signal selection criteria. We then plot this quantity as a function of event-level variables obtained from offline reconstruction: the total energy E deposited in the ECL, the mean transverse momenta p_t of the τ^+ and τ^-, and the mean angle θ between the decay products in the primary τ decay. These distributions are shown in Figs. 48, 49, and 50.

The most significant dependence on an event-level variable is seen for the total energy deposition in the ECL, where the number of events triggered by `fzo` $\vee$ `sttz` is more than five times that of `ecltaunn`$\wedge$(`fzo`$\vee$`sttz`) at very low energies. In this region, the signal efficiencies of both `ecltaunn` and `fzo`$\vee$`sttz` also independently drop below 50%. Thus, at high energies, the `ecltaunn` and track triggers provide highly redundant coverage, triggering on the exact same subset of signal events. Conversely, at very low energy, the decreasing correlation suggests that the conditions capture complementary and independent regions of the kinematic phase space as their respective signal efficiencies decrease.

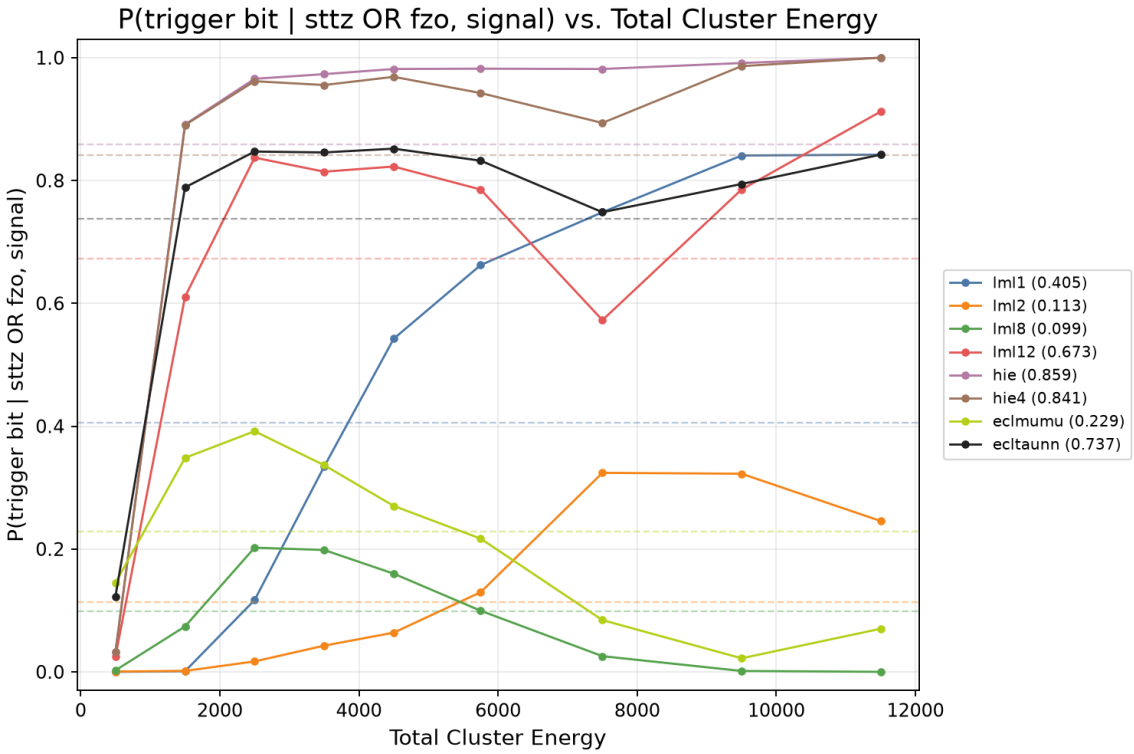


Figure 48: $P(\texttt{ecltaunn}|\texttt{fzo} \vee \texttt{sttz})$ as a function of E for all signal events

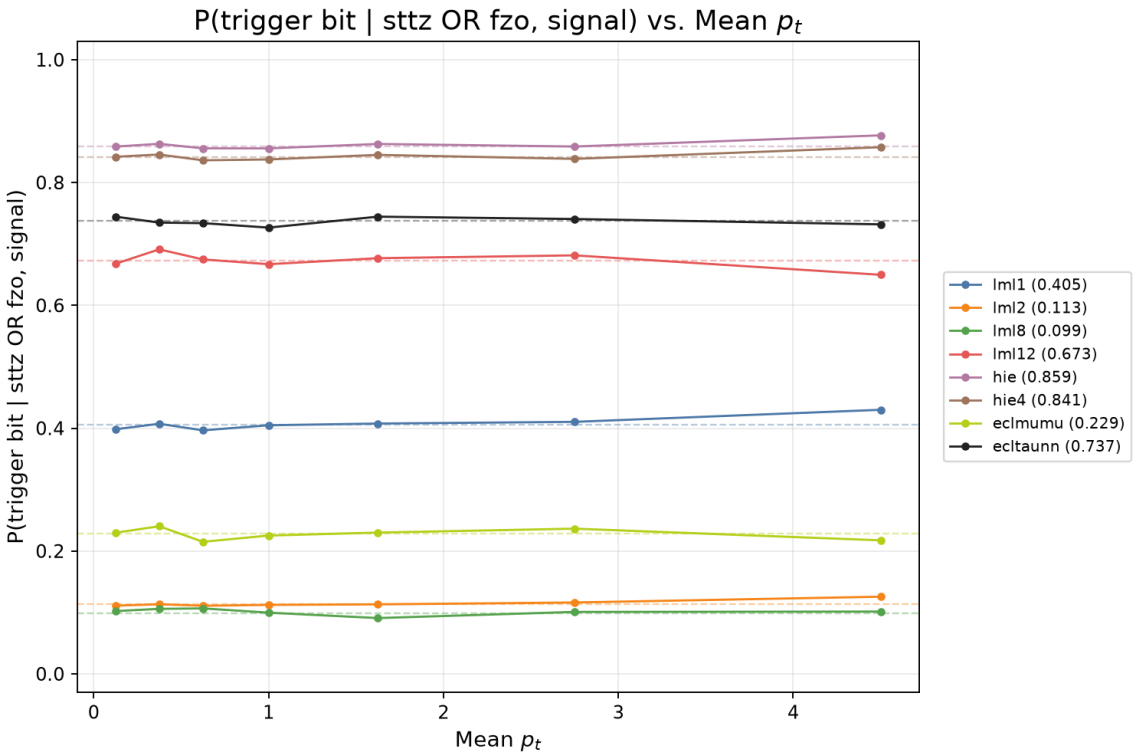


Figure 49: $P(\texttt{ecltaunn}|\texttt{fzo} \vee \texttt{sttz})$ as a function of mean p_t for all signal events

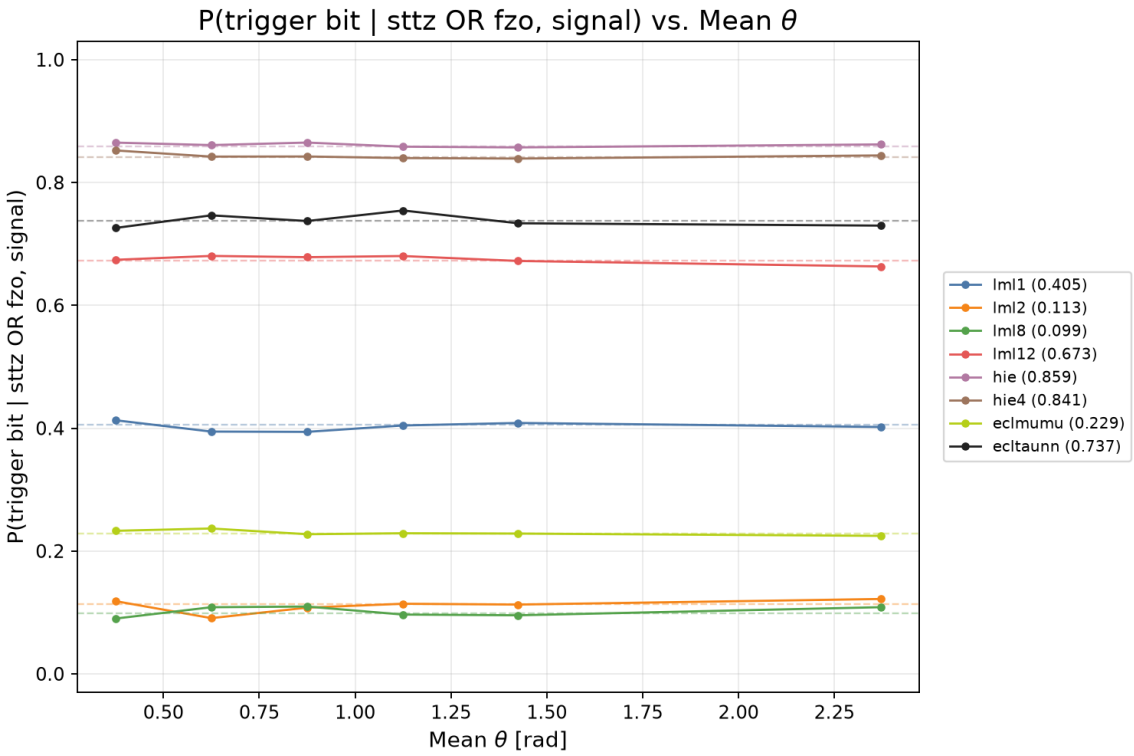


Figure 50: $P(\texttt{ecltaunn}|\texttt{fzo} \vee \texttt{sttz})$ as a function of mean θ for all signal events

8 Summary and Future Work

8.1 Summary

The Belle II experiment is expected to continue operation as the luminosity at SuperKEKB increases from 5.2×10^{34} cm^{-2}s^{-1} to 6.0×10^{35} cm^{-2}s^{-1}. While the existing conditions for τ event-selection by the hardware-based Level-1 trigger are sufficient to achieve high signal efficiencies under the current accelerator conditions, the prospect of an order-of-magnitude increase in luminosity demands the development of new trigger algorithms to suppress backgrounds and maintain a total trigger rate below the < 30 kHz limit of the Data Acquisition System. To that end, we develop a low-latency machine learning-based τ event selection algorithm utilizing high granularity quantization and high level synthesis for on-chip deployment in the Belle II Global Reconstruction Logic. We show that this new algorithm consistently outperforms existing trigger bits on low-multiplicity standard model τ decays, maintaining signal efficiencies as high as 97% across all modes at $\mathcal{L} = 6.0 \times 10^{35}$ cm^{-2}s^{-1} while respecting the 30 kHz limit. We validate the logic in register-transfer level behavioral simulations, showing consistent behavior in both software and FPGA firmware, before finally enabling the new trigger bit for Belle II physics runs from Summer 2026 onward. Outputs from the new trigger logic in subsequent physics runs show strong agreement with the aforementioned simulations, indicating successful implementation of the model in FPGA firmware. Direct evaluation of the `ecltaunn` trigger performance using the CDC track triggers `fzo` and `sttz` as a reference also suggests several directions for ongoing study and refinement of the neural network algorithm.

8.2 Future Work

There are two notable avenues for future development of the neural network-based τ trigger trigger, both of which are likely to significantly improve the performance of the existing model. The first is the extension of the existing neural network to accept raw ECL hit information. The primary bottleneck identified in this study is the ECLTRG clustering, which reduces the hit information from all 576 ECL TCs to ≤ 6 clusters parameterized by $\{E, \theta, \phi, t_{\text{avg}}\}$. Even when substantially increasing the complexity of the network itself, the accuracy remains largely unchanged. This is even more evident when the model is trained with quantization-aware training, as any increase in the number of layers or nodes per-layer is offset by more aggressive quantization of each individual parameter, since there is no increase in the model's accuracy to offset the downward pressure of the surrogate gradient Eq. 33 on the bit-widths. This is a strong indication that the model is already utilizing essentially all available information to the extent possible within the existing paradigm.

The second, and likely more immediately actionable direction for continued study focuses on improving the labeling of the training data and subsequent separation of different decay modes. As noted in Section 5.7, the outputs of the model remain highly correlated, particularly for μ and π final states. Future iterations may benefit from more restrictive selection criterion to limit the multiplicity of signal candidates in each event, reducing the number of events with multiple labels. This could involve fine-tuning of the selection criteria to directly restrict the number of charged particle tracks in an event using rest-of-event reconstruction in `basf2`, and/or ranking candidates such that each event can be labeled with only the mode corresponding to the best candidate. While this is almost certain to improve measured separation between different modes by directly reducing the degeneracy of labels in the training set, it remains an open question as to whether this would improve the event *selection* efficiency across all modes, and requires further empirical investigation. Full validation of the offline event selection criteria will also require detailed study in Monte-Carlo simulations, which is currently ongoing.

More broadly, future studies may also focus on extending this machine learning-based event selection framework to other detectors at Belle II. Since the basic methods developed in this thesis are effectively agnostic to the structure of the inputs, the same model can easily be re-trained using inputs from any other sub-trigger system available at the GRL. While not all detectors may provide sufficient

information to meaningfully identify τ events in particular, both the physics targets and inputs can be set on a case-by-case basis for those detectors which *do* allow for robust event selection. Another plausible extension along these lines involves extending the physics targets to non-Standard Model modes such as CP-violating or LFV decays. These targets would require the use of Monte Carlo simulations to generate training data, thus a more thorough consideration of background modeling would likely be necessary to ensure that the trained model's performance on simulated data effectively transfers to real experimental background conditions.

References

[1] S. Navas *et al.* (Particle Data Group), Phys. Rev. D **110**, 030001 (2024).

[2] I. Adachi *et al.* (Belle II), Journal of High Energy Physics **2024**, 205 (2024).

[3] E. Kou *et al.* (Belle II), Progress of Theoretical and Experimental Physics **2019**, 123C01 (2019), https://academic.oup.com/ptep/article-pdf/2019/12/123C01/32693980/ptz106.pdf .

[4] L. D. Landau, Nuclear Physics **3**, 127 (1957).

[5] M. Pospelov and A. Ritz, Annals of Physics **318**, 119 (2005).

[6] C. Abel *et al.* (nEDM), Physical Review Letters **124**, 081803 (2020).

[7] M. Kobayashi and T. Maskawa, Progress of Theoretical Physics **49**, 652 (1973).

[8] J. Charles *et al.* (CKMfitter Group), The European Physical Journal C **41**, 1 (2005), [Updated results and plots available at: http://ckmfitter.in2p3.fr].

[9] B. Aubert *et al.* (BaBar), Phys. Rev. Lett. **87**, 091801 (2001), arXiv:hep-ex/0107013 .

[10] K. Abe *et al.* (Belle), Phys. Rev. Lett. **87**, 091802 (2001), arXiv:hep-ex/0107061 .

[11] R. Aaij *et al.* (LHCb), JHEP **12**, 141, arXiv:2110.02350 [hep-ex] .

[12] R. Aaij *et al.* (LHCb), Nature Phys. **11**, 743 (2015), arXiv:1504.01568 [hep-ex] .

[13] K. Abe *et al.* (T2K), Nature **580**, 339 (2020).

[14] M. B. Gavela, P. Hernandez, J. Orloff, O. Pene, and C. Quimbay, Nuclear Physics B **430**, 382 (1994).

[15] S. Davidson, E. Nardi, and Y. Nir, Physics Reports **466**, 105 (2008).

[16] J. P. Lees *et al.*, Physical Review D **85**, 10.1103/physrevd.85.031102 (2012).

[17] V. Andreev *et al.* (ACME), Nature **562**, 355 (2018).

[18] T. S. Roussy *et al.*, Science **381**, 46 (2023).

[19] G. W. Bennett *et al.* (Muon g-2), Phys. Rev. D **80**, 052008 (2009), arXiv:0811.1207 [hep-ex] .

[20] K. Inami *et al.*, Physics Letters B **551**, 16 (2003).

[21] S. Banerjee, B. Pietrzyk, J. M. Roney, and Z. Wąs, Physical Review D **77**, 054012 (2008).

[22] Y. Ohnishi *et al.*, PTEP **2013**, 03A011 (2013).

[23] Belle II Collaboration, Luminosity status and projection plan, `https://www.belle2.org/info/luminosity_status/index_html#plan` (2026), accessed: 2026-07-06.

[24] T. Abe *et al.* (Belle II), (2010), arXiv:1011.0352 [physics.ins-det] .

[25] J. Fast (Belle-II Barrel Particle Identification Group), Nucl. Instrum. Meth. A **876**, 145 (2017).

[26] B. Wang, The construction and commissioning of the belle ii itop counter (2017), arXiv:1709.09938 [physics.ins-det] .

[27] S. Nishida *et al.*, Nucl. Instrum. Meth. A **766**, 28 (2014).

[28] A. Abashian *et al.* (Belle), Nucl. Instrum. Meth. A **479**, 117 (2002).

[29] S. Yamada *et al.*, IEEE Trans. Nucl. Sci. **62**, 1175 (2015).

[30] Y.-T. Lai *et al.*, Nuclear Instruments and Methods in Physics Research A **1078**, 170577 (2025), arXiv:2503.02192 [hep-ex] .

[31] A. Natochii *et al.*, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **1055**, 168550 (2023).

[32] Y. Xue, T. Koga, and J. Yin, (2024), bELLE2-NOTE-TE-2024-023.

[33] S. Markidis, S. W. D. Chien, E. Laure, I. B. Peng, and J. S. Vetter, in *2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)* (IEEE, 2018) pp. 522–531.

[34] L. Lönnblad, C. Peterson, and T. Rögnvaldsson, Physical Review Letters **65**, 1321 (1990).

[35] P. Abreu *et al.* (DELPHI), Physics Letters B **295**, 383 (1992).

[36] A. Radovic, M. Williams, D. Rousseau, M. Kagan, D. Bonacorsi, A. Himmel, A. Aurisano, K. Terao, and T. Wongjirad, Nature **560**, 41 (2018).

[37] A. J. Larkoski, G. P. Salam, and J. Thaler, Journal of High Energy Physics **2013**, 108 (2013).

[38] J. Thaler and K. Van Tilburg, Journal of High Energy Physics **2011**, 15 (2011).

[39] M. Aaboud *et al.* (ATLAS), The European Physical Journal C **77**, 490 (2017).

[40] A. M. Sirunyan *et al.* (CMS), Journal of Instrumentation **12** (10), P10003.

[41] The Belle II Collaboration, Belle ii analysis software framework (basf2) (2025).

[42] P. Szymański and T. Kajdanowicz, ArXiv e-prints (2017), arXiv:1702.01460 .

[43] K. Sechidis, G. Tsoumakas, and I. Vlahavas, in *Machine Learning and Knowledge Discovery in Databases*, edited by D. Gunopulos, T. Hofmann, D. Malerba, and M. Vazirgiannis (Springer Berlin Heidelberg, Berlin, Heidelberg, 2011) pp. 145–158.

[44] C. Sun, Z. Que, T. Aarrestad, V. Loncar, J. Ngadiuba, W. Luk, and M. Spiropulu, in *Proceedings of the 2026 ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, FPGA '26 (Association for Computing Machinery, New York, NY, USA, 2026) p. 79 – 91.

[45] Scikit-learn Developers, Scikit-learn user guide and documentation, `https://scikit-learn.org/stable/documentation.html` (2026), accessed: 2026-06-24.

[46] Advanced Micro Devices, Inc., *UltraScale FPGA Product Tables and Product Selection Guide*, AMD (2026), available online; accessed June 2026.

[47] FastML Team, fastmachinelearning/hls4ml (2026).

[48] J. Duarte *et al.*, JINST **13** (07), P07027, arXiv:1804.06913 [physics.ins-det] .

[49] J.-F. Schulte *et al.*, ACM Trans. Reconfigurable Technol. Syst. **19**, 19 (2026), arXiv:2512.01463 [cs.AR] .

[50] C. Sun, Z. Que, V. Loncar, W. Luk, and M. Spiropulu, ACM Trans. Reconfigurable Technol. Syst. 10.1145/3777387 (2025).

[51] D. Sun, Z. Liu, J.-z. Zhao, and H. Xu, Physics Procedia **37**, 1933 (2012), tIPP 2011: Technology and Instrumentation in Particle Physics 2011.

Glossary

Belle II Link (B2L) The high-speed optical link system used by the Belle II experiment to transmit data from the Front-End Electronics to the Data Acquisition System.

Belle II Analysis Software Framework (basf2) A software framework developed by the Belle II collaboration which provides tools for offline event reconstruction and data analysis.

ECL Sub-Trigger (ECLTRG) The on-detector algorithms which convert hit information from the Electromagnetic Calorimeter to high-level data objects ("clusters") for use in online event reconstruction.

ECL Trigger Module (ETM) The final stage of the ECL Sub-Trigger, which groups individual Trigger Cell hits into "clusters" for use in online event reconstruction.

Front-End Analysis Module (FAM) The first stage of the ECL Sub-Trigger. Each module digitizes and process waveform data from a subset of crystals in the Electromagnetic Calorimeter to obtain hit energy and timing information.

Front-End Electronics (FEE) On-detector electronics which perform the initial processing and buffering of the raw data in each event.

Global Reconstruction Logic (GRL) The second-to-last step in the Belle II Level-1 Trigger system, which assigns a set of boolean flags called "trigger bits" based on event-level reconstruction algorithms which utilize data from all detector sub-trigger systems.

Global Decision Logic (GDL) The final step in the Belle II Level-1 Trigger system, which uses the set of boolean trigger bits from the Global Reconstruction Logic to make a final trigger decision.

ShaperDSP A waveform shaping module used for real-time integration of waveform data from the Electromagnetic Calorimeter. Each module outputs two signals with different shaping times to be used for online and offline reconstruction respectively.

Trigger Merger Module (TMM) The second stage of the ECL Sub-Trigger, which aggregates data from multiple Front-End Analysis Modules.

Trigger Cell (TC) A 4×4 group of CsI(Tl) crystals in the Electromagnetic Calorimeter that are read out collectively for online triggering.

Universal Trigger Board 4 (UT4) A custom electronics module designed for the Belle II Level-1 Trigger system, featuring high-speed optical links and a dual-FPGA architecture.

Acknowledgments

I would first like to thank my advisor, Professor Takeo Higuchi, for his careful supervision and for entrusting me with the independence and autonomy to pursue a topic at the leading edge of experimental particle physics over the last two years. I would also like to thank the members of the Belle II TRG group, particularly Taichiro Koga, Yu Nakazawa and Hanwook Bae, for their patient guidance as I became acquainted with the Belle II experiment and trigger system. I am also deeply grateful to Laura Helgeson, whose devout partnership and enduring support allowed me to take risks and pursue my dreams without hesitation. Finally, I would like to thank my parents for encouraging my studies and my aspirations wherever they may take me.

This research was supported by a Global Research Graduate Course (GSGC) Scholarship from The University of Tokyo.