Daria SENINA
Promotion 2025
2024/2025

Diplôme d'ingénieur Télécom Physique Strasbourg

Mémoire de stage de 3^e année

ENHANCING BELLE II DISCOVERY POTENTIAL WITH A GRAPH NEURAL NETWORK FOR VERTEX FITTING

du 03 Mars 2025 au 30 Août 2025

IPHC
23 rue du Loess
67037 Strasbourg, France

Tuteur: Jérôme Baudot
Tél: 03.88.10.66.32
Mail: jerome.baudot@iphc.cnrs.fr

Acknowledgements

I would like to thank my supervisor, Dr. Jérôme Baudot, for introducing me to BSM physics and guiding me throughout this project with wise advice. I am also deeply grateful for the help to find the Ph.D. position at the University of Zürich and the meaningful recommendation. Additionally, I sincerely thank Dr. Giulio Dujany who provided a huge amount of help in understanding the GraFEI model and finding solutions to my technical problems.

Then, I would like to thank Prof. Yann Leroy for listening and advising me when I was lost, and for recommending me as well to the UZH for my future PhD. Thank you also Prof. Anne-Sophie Cordan for your kindness and guidance throughout the engineering school years.

I also wish to thank Corentin Santos and Merna AbuMusabh for their technical assistance and helpful discussions. Finally, I am sincerely grateful to the entire Belle II Strasbourg team for introducing me to collaborative research in particle physics and for giving me the opportunity to participate in my first scientific mission. You made me feel very welcome in this environment.

Contents

List of Figures

List of Tables

Glossary

BSM Beyond the Standard Model.

CP Charge-Parity.

FSP Final State Particle.

GNN Graph Neural Network.

GraFEI Graph Full Event Interpretation.

LCA Lowest Common Ancestor.

ML Machine Learning.

NN Neural Network.

ROE Rest Of Event.

Introduction

The Belle II collaboration is an international experiment based in Japan at the SuperKEKB accelerator, that hopes to observe new physics during the collision of an electron e^- and its anti-particle, the positron e^+. Such complex experiments require the knowledge and effort of scientists from around the world, such as the group based in Strasbourg at the IPHC. But what exactly is new physics and how do we observe it?

Today, the **Standard Model** of particle physics provides the most fundamental description of our world at the elementary level. However, it fails to correctly account for observed phenomena such as the matter-antimatter asymmetry of our universe, which is responsible for the existence of our physical world rather than a world of antimatter. Therefore, scientists in various collaborations are trying to find observational limits to this Standard Model. This is the case of Belle II that hopes to observe non-standard processes in e^+e^- collisions at an energy of 10.58 GeV and in particular with rare decays of particles called B mesons. The study of such particles is called B-physics and is very useful for observing rare "Beyond the Standard Model" (BSM) phenomena as the decay of B mesons presents large matter-antimatter asymmetry-induced properties, called Charge-Parity (CP) violation properties. This internship focuses on the study of the decay position of the B meson in the Belle II detector, referred to as the **vertex**, since precise knowledge of its position is essential for accurate asymmetry measurements.

The collaboration has already developed many tools such as a **graph neural network** tool called **GraFEI**. The GraFEI was developed with the intent of reconstructing the particle decay tree that happens after a collision from the information of the particles seen in the detector, also called the Rest Of Event (ROE). While it correctly predicts the presence of a B meson ancestor and the particles that originated from it, it does not give any information on their physical properties such as position. Therefore, the idea of this project will be to implement a new feature in the GraFEI model to reconstruct the position, and compare its performances to the existing vertex tagging tool of Belle II called TagV. A complementary test will be to implement and predict a property called the flavor of the B meson. The future goal of this project is to use the updated GraFEI to assess the potential improvements in resolution that must be made to the current

Belle II detector in order to observe the decay vertex.

The work will be explained following the methodology used to achieve these objectives: after an introduction to the IPHC and the Belle II experiment, the motivation of this project will be thoroughly explained alongside a background in B physics. Then, a dedicated chapter on machine learning in experimental particle physics, and on graph neural networks, will describe the GraFEI model. The results will be presented in Chapter 4, with additional analysis brought in Chapter 5, before discussing the performances in the last chapter. Finally, a conclusion will be made on the achievements of this new vertex tagging tool with an insight on further improvements and studies.

Chapter 1

IPHC

1 The Institute

The Institut Pluridisciplinaire Hubert Curien (IPHC) was created in 2006 by combining three scientific laboratories: the institute of subatomic research (IReS), the laboratory of analytical science and molecular and biomolecular ionical interactions, and the center of ecology and energetic physiology (CEPE). The IPHC depends both on the CNRS (National Center of Scientific Research) and the University of Strasbourg. It is located mainly on the Cronenbourg campus and is currently directed since 2021 by prof. Sandrine Courtin.
About 400 staff carry out interdisciplinary programs with a strong backbone in scientific instrumentation (approximatively 160 engineers, technicians and administrative staff).
The institute is organized into three scientific departments:

- DEPE - Department of Ecology, Physiology and Ethology
- DRS - Department of Subatomic Research
- DSA - Department of Analytical Sciences

The CNRS is a governmental organization that has an annual budget of 4 billion euros, 34 700 employees including 29 400 scientists and about 1 100 research laboratories in France and abroad. The CNRS is a leader in scientific innovation, as it ranked 4^{th} worldwide in the 2020 *Nature Index*. The organization is structured into ten national institutes. The IPHC's DRS belongs to the Institute for Nuclear and Particle Physics (IN2P3).

2 The DRS

The DRS [11] is part of CNRS–IN2P3, exploring matter from the infinitely small to the infinitely large. It also explores various fields in the middle, described in tables 1.1.

Table 1.1: DRS groups and experiments

(a) DRS groups

Name	Purpose	Location
Theory	Nuclear and particle theory.	IPHC (Strasbourg)
DNE (From the Nucleus to the Stars)	Exotic nuclear structures, spectroscopy, heavy/superheavy nuclei study; fusion reactions and nucleosynthesis in massive stars.	IPHC (Strasbourg)
Molecular Imaging & Radiobiology (IMR)	Instrumentation for cancer surgery guidance and small animal imaging.	IPHC (Strasbourg) & CYRCé platform

(b) Energy, environment and dosimetry groups

Name	Purpose	Location
DNR (Nuclear Data for Reactors)	Nuclear-data measurements and experimental development of detection systems for Gen-IV reactor research of (n,xn) reactions.	IPHC (Strasbourg)
Radiochemistry	Chemistry of actinides and lanthanides.	IPHC (Strasbourg)
DeSIs (Dosimetry, Simulation & Instrumentation)	Research in radioprotection for health, environment, and industry.	IPHC (Strasbourg)

(c) From the Big Bang to particles groups

Name	Logo	Purpose	Location
ALICE (A Large Ion Collider Experiment)		Heavy-ion collisions at the LHC to probe the Quark–Gluon Plasma and properties of strongly interacting matter.	CERN (Geneva) & IPHC
Belle II		e^+e^- collisions for flavor-physics studies, quark flavor change processes and searches for physics beyond the Standard Model.	KEK (Tsukuba, Japan) & IPHC
CMS (Compact Muon Solenoid)		General-purpose detector at the LHC, investigates a large range of physics.	CERN (Geneva) & IPHC
Neutrino		Neutrino-oscillation physics via JUNO and ESSnuSB(+).	JUNO (China), ESS (Lund, Sweden) & IPHC
OGMA		Multimessenger astrophysics combining gravitational-wave, neutrino, and electromagnetic observations.	IPHC & international observatories
PICSEL		Development of monolithic active pixel sensors (MAPS) for trackers and vertex detectors.	IPHC (Strasbourg)

3 The Organization

This internship was done in the Belle II team of the DRS. The team is currently composed of four scientists and one professor-scientist, one Post-Doc, five PhD students and one intern by counting myself.
The following Figure 1.1 is the organizational chart of the IPHC.

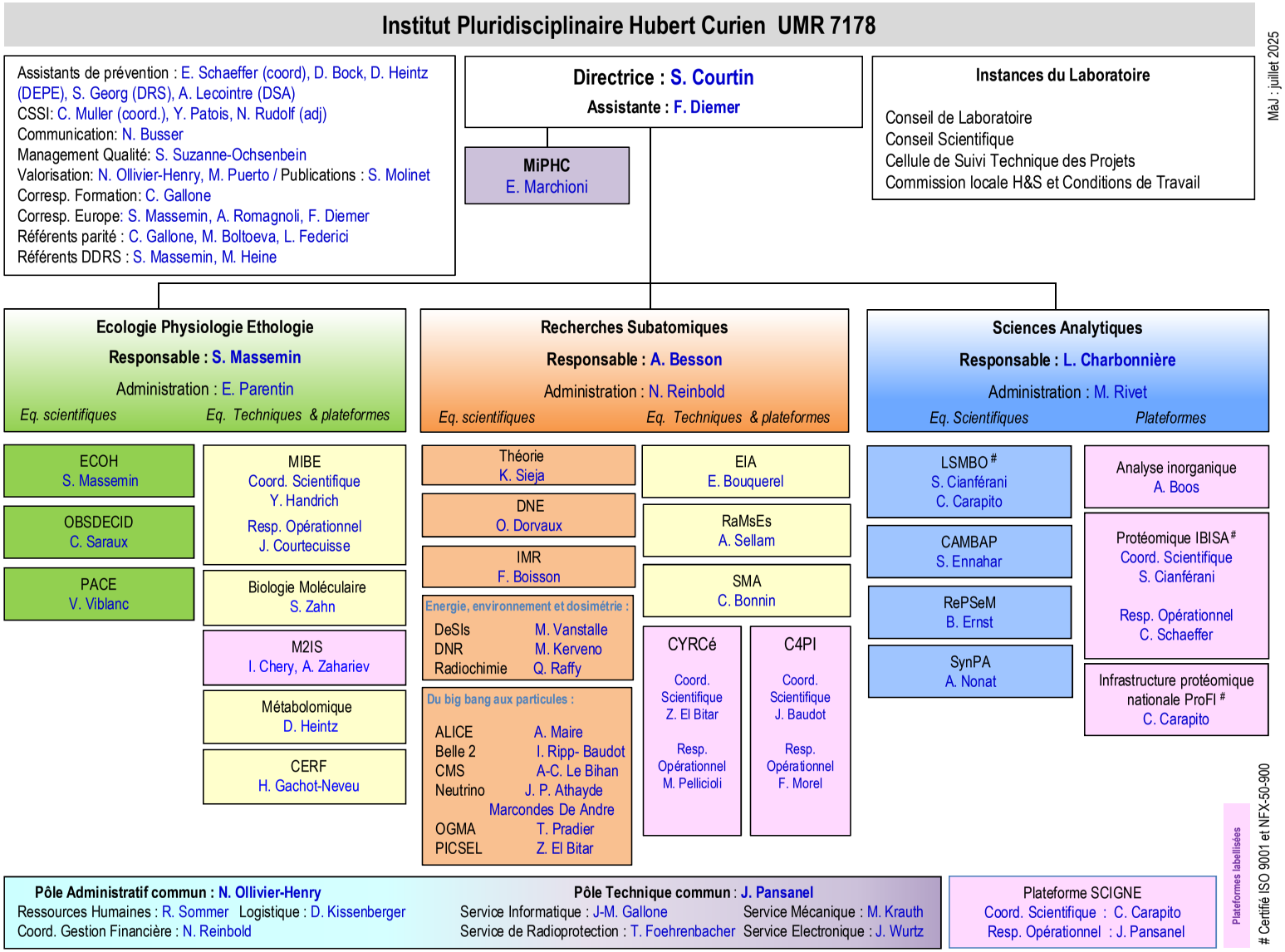


Figure 1.1: IPHC organization chart

Chapter 2

The Belle II Experiment

1 The SuperKEKB accelerator

The Belle II experiment takes place at the SuperKEKB accelerator. It is an electron-positron collider located at the High Energy Accelerator Research Organization (KEK) in Tsukuba, Japan. It is composed of an electron-positron linear injector, a positron damping ring, and it is asymmetric in energy, therefore it has two main rings with a circumference of 3 km built 11 km underground: the high energy ring (HER) for the electrons that are accelerated at the energy of 7 GeV, and the low energy ring (LER) for positrons accelerated at 4 GeV. These beams collide at the interaction point at the center of the Belle II detector. They collide at the energy of 10.58 GeV, corresponding to the energy of creation of the $\Upsilon(4S)$ particle [1], called energy of resonance. This particle almost always decays into two B mesons, boosted along the axis of the experiment. This creates a door towards new physics research in flavor physics, dark matter, and strong force bonding hadrons.
The SuperKEKB accelerator has the highest luminosity in the world, achieving $5.1 \times 10^{34}\ cm^{-2}s^{-1}$ in 2024 [21]. Luminosity is a measure of how many particle collisions an accelerator produces per unit area and per second. The higher the luminosity, the more often particles collide, which means more chances to observe rare physics events. The SuperKEKB accelerator is illustrated figure 2.1.

2 The Belle II detector

The Belle II detector illustrated in Figure 2.2 is composed of several concentric subsystems arranged around the beam tube. At the center is the beam pipe, through which the beams circulate and where the electrons and positrons collide. Surrounding it are the following layers (from the inside outward):

[1] An equal amount of $c\bar{c}$ and $\tau^+\tau^-$ are created since the three processes have a similar cross section, about 1 nb.

- The Vertex Detector (**VXD**), which consists of two sub-components: the Pixel Detector (**PXD**) and the Silicon Vertex Detector (**SVD**), provides precise tracking near the interaction point.
- The Central Drift Chamber (**CDC**), which measures the momentum of charged particles via their curvature in a magnetic field and contributes to identifying their nature via their lineic energy loss.
- The Particle Identification (**PID**) devices, including the Time-of-Propagation (**TOP**) counter in the barrel region and the Aerogel Ring Imaging Cherenkov (**ARICH**) detector in the forward endcap, which are used to distinguish between different types of charged particles.
- The Electromagnetic Calorimeter (**ECL**), which measures the energy of photons and electrons.
- The **KLM** (K_L and Muon detector), located in the outermost layers, used to detect muons and K_L^0 mesons.

The particle data from these subsystems is collected through the Trigger and Data Acquisition (**TRG** and **DAQ**) systems. After initial filtering, the data are processed, reconstructed, and stored using Belle II Analysis Software Framework (BASF2) [13].

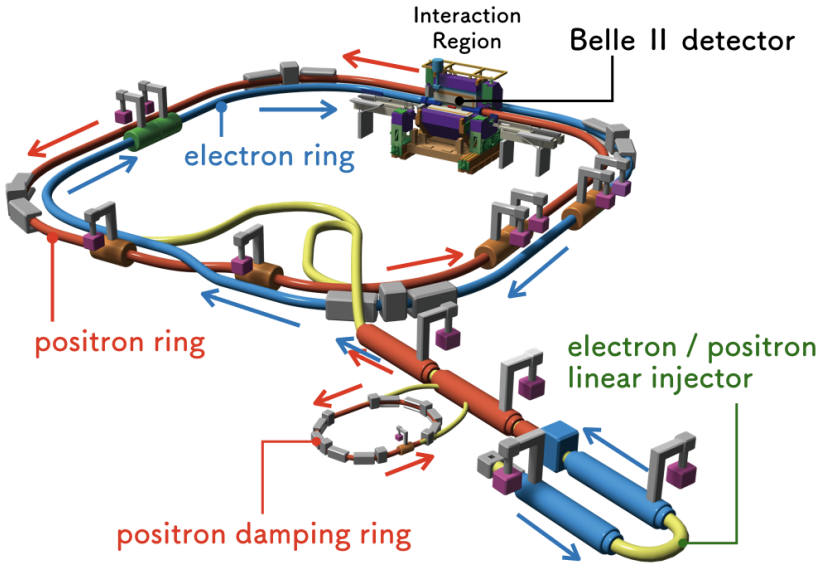


Figure 2.1: The SuperKEKB accelerator

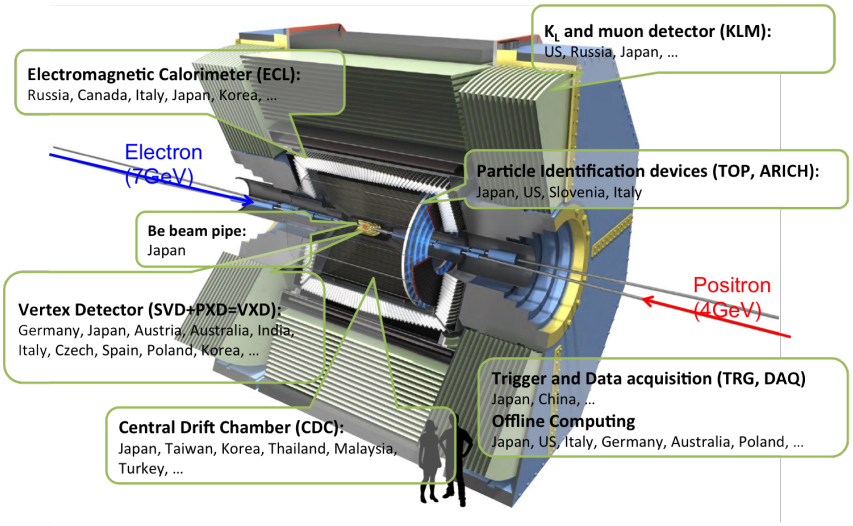


Figure 2.2: Belle II detector: an international effort

To explore physics beyond the Standard Model, it is important to recognize that finding new phenomena is easier in rare processes. Deviations coming from new physics of similar absolute amplitude will indeed appear relatively larger with respect to the probability of a rare process than compared to a more abundant one, much like looking for a flea is easier on a mosquito than on an elephant. This is why physicists focus on rare decays that allow to study more efficiently phenomena that are not fully accounted for in the Standard Model, such as the **CP violation**, which may pave the way towards new physics.

3 CP violation in B factories

Matter and antimatter annihilate each other upon contact, their existence in equal quantities would have left the universe filled with nothing but radiation. Yet, our universe is overwhelmingly filled with matter. This imbalance, known as matter-antimatter asymmetry, which most likely occurred in the early evolution of the Universe, is one of the greatest mysteries for physicists [3].

3.1 The Elementary Particles

In order to delve into particle physics and especially into B factories, it is necessary to understand the composition of the Standard Model.
The Standard Model of Particles is composed of the most elementary founding blocks of our universe. It is divided into fermions and bosons, as illustrated in Figure 2.3. The fermions constitute matter, while bosons are interaction particles. To understand this work, the electron is from the lepton class and is a fermion. The positron is its

anti-particle, meaning that it has the same properties but with an opposite charge. The quarks constitute the second class of the fermion group and they cannot be found individually in nature. Indeed, by taking the example of an atom: the electron exists on a shell around the nucleus, it can be excited and quit the atomic shell. Whereas quarks live inside of the nucleus and are bound together by the strong force. They cannot come out of the nucleus independently.

Hadron is the name given to particles constituted of quarks, such as the neutron and the proton, which are composed of two down and one up quark (ddu), and two up and one down quark (uud), respectively. But in the hadron family, there is a distinction between being constituted of two or three quarks. The particles composed of two quarks are called **Mesons**, such as the B meson, which contains a bottom quark, and the Υ meson which is constituted of the bottom quark and its anti-quark (J/Ψ found later in the document, is also a meson). The ones with three quarks like the proton and neutron are called **Baryons**.

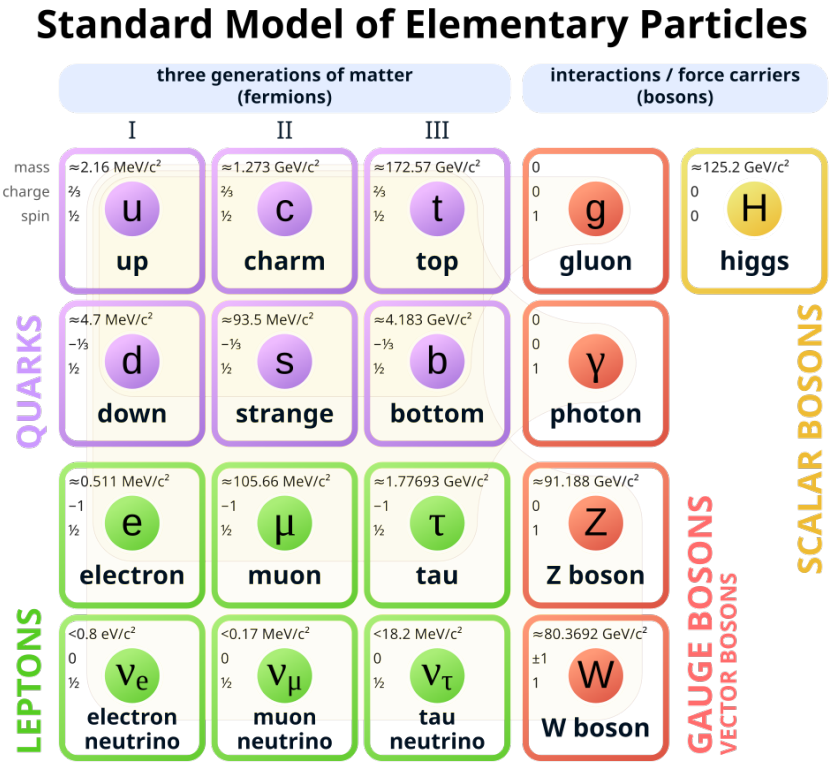


Figure 2.3: The Standard Model of Elementary Particles [7]

3.2 CP violation

In physics, the Charge conjugation C transforms a particle into its antiparticle by reversing all its charges, while the Parity transformation P flips the spatial coordinates of the wave function describing the particle. This is the CP operator, where the antiparticle is like looking at the particle in a mirror with opposite charges. If the laws of physics were CP symmetric, they would behave the same under both C and P transformations together, but they do not. Certain processes have been observed to violate the CP symmetry, meaning that nature treats matter and antimatter differently. This violation is key to explaining why our universe has more matter than antimatter - although the amount of CP violation observed so far, is not yet enough to fully account for the asymmetry [8].

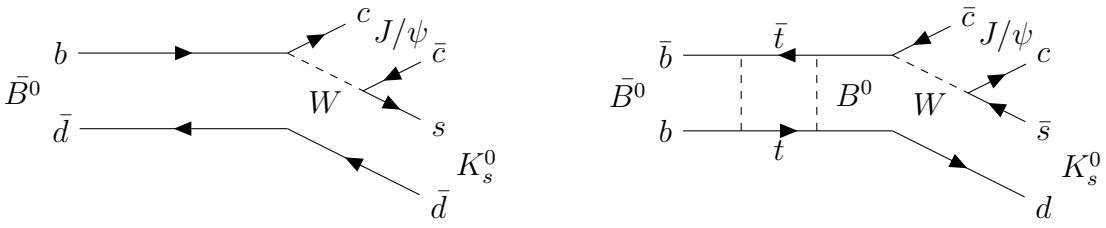
3.3 B Physics in $\Upsilon(4S) \to B\bar{B}$

The goal of B physics is to use B meson decays composed of a $\bar{b}$ antiquark and another lighter quark that determines the meson's charge (u for B^+, d for B^0, s for B_s^0 and c for B_c^0), to test the validity of the CKM mechanism for CP violation. The CKM mechanism, short for Cabibbo-Kobayashi-Maskawa, describes the way that the quark flavor can change during an interaction [5]. CP violation was first observed in the neutral Kaon system [6], and then explored in other neutral meson systems such as: $D^0 - \bar{D}^0$ and $B^0 - \bar{B}^0$. In particular, the neutral B meson system features a mixing frequency and a lifetime which generates an interference leading to **time-dependent CP asymmetries**.

According to the BaBar and the Belle experiments, in their *Physics of the B Factories* book [2], the golden observable for measuring CP violation is the time-dependent asymmetry between the decays $B^0 \to J/\psi K_s^0$ and $\bar{B}^0 \to J/\psi K_s^0$, following the production of entangled B mesons from the $\Upsilon(4S)$ resonance. The $J/\psi K_s^0$ final state being a CP eigen state (f_{CP}). This means that CP violation is observed by comparing the time-dependent decay probabilities of B^0 and $\bar{B}^0$ to the same final state, as illustrated in figure 2.4[2], and calculated in equation 2.1.

The asymmetry is computed as a function of proper time difference between the moment of decay of both mesons:

[2]To read a Feynman diagram, time flows from left to right, and the orientation of the arrow depends on the nature of the particle: for a particle, it goes along with the time, and for an antiparticle, it goes "back" in time. When a line is plain, it is a fermion, and when it is dashed, then it is a weak interaction, a W boson in this case.



(a) Direct $\bar{B}^0$ decay

(b) $B^0 - \bar{B}^0$ mixing decay

Figure 2.4: Feynman diagrams of B^0 and $\bar{B}^0$ decaying into the same final state.

$$A_{CP}(\Delta t) = \frac{\Gamma(\bar{B}^0 \to f_{CP})(\Delta t) - \Gamma(B^0 \to f_{CP})(\Delta t)}{\Gamma(\bar{B}^0 \to f_{CP})(\Delta t) + \Gamma(B^0 \to f_{CP})(\Delta t)} \propto Ssin(\Delta m_d \Delta t) - Ccos(\Delta m_d \Delta t) \tag{2.1}$$

With Γ being the probability of having such a decay as a function of time, Δm_d being the oscillation frequency of the B meson and S and C being the key parameters that quantify, respectively, the level of indirect and direct CP violation. Equation 2.1 underlines that the measurement of Δt is paramount to the evaluation of CP violation.

B factories are specially designed so that the collision is boosted along the z-axis and therefore the two B mesons fly a significant distance before decaying apart from each other. The measurement of the distance Δz between their decay positions allows to access Δt. Also, in order to know the current flavor (B^0 or $\bar{B}^0$) of the meson, we use a technique called flavor-tagging. In this technique we fully reconstruct one of the two B mesons, constituting the "Signal" side. The other B meson, constituting the "Tag" side, has its flavor inferred by analyzing its decay products, and because the two mesons are produced in an entangled state, it directly gives the flavor of the signal-side B meson.
Figure 2.5 illustrates this technique with the process used during this project, which is a $B_{sig} \to \nu\bar{\nu}$ Monte Carlo reconstructed dataset, meaning that the ROE is constituted solely from particles originating from the B_{tag} vertex since neutrinos are not detectable. Another difficulty has been represented in this figure, the B meson decaying into a D meson, because the correct decay vertex is often confused between the two of them since they have similar decay signature.

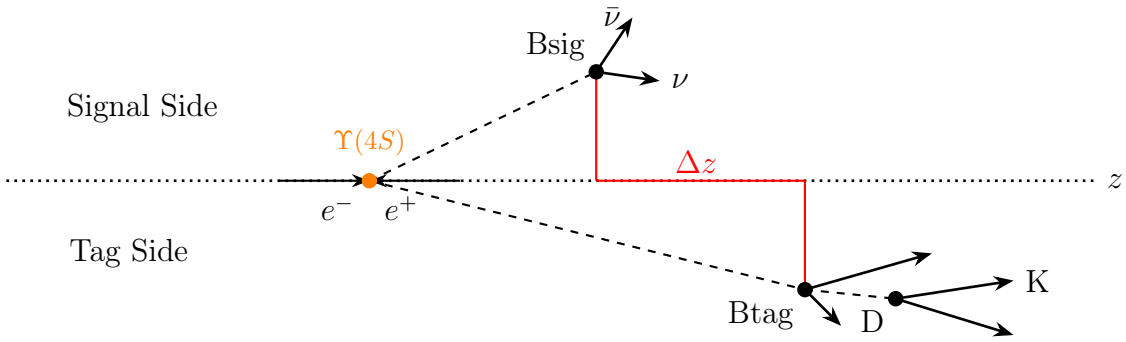


Figure 2.5: $B_{sig} \to \nu\bar{\nu}$ boosted along z decay

4 Using a Neural Network for Vertex Fitting

4.1 New model motivation

In this work, the objective is to access the $B_{tag,z}$ position in order to study Time-dependent CP violation. This is currently done with the TagV tool in Belle II by reconstructing *exclusively* the signal side, which yields high purity but low efficiency since we rely on a hard coded decay, and by reconstructing *inclusively* the tag side, using the ROE excluding the signal, and doing a global fit of the whole decay tree to predict the meson vertex position. The later inclusive reconstruction of the Tag side is highly efficient since the algorithm does not need the knowledge of the decay tree, but it also yields to inaccuracy since it can misinterpret the B and D meson signatures and count the D meson as a B. The objective is thereby to develop a new algorithm that does not rely on hard-coded processes to reconstruct the decay tree but is able to learn the decay patterns by studying the relationship between the leaves of the tree (final particles) with a graph neural network method, yielding high efficiency, better resolution, and no confusion in the B-meson position prediction.

4.2 Strategy

To develop this new model, the following strategy was used and will be explained through this report:

1. Understand the existing model structure, and the functioning of neural networks
2. Implement the vertex feature
3. Get the best resolution possible by studying the parameters
4. Compare it to the existing TagV algorithm

5. Conclude on the efficiency of this new model

This was accompanied by regular scheduled group meetings, at least twice a month, a conference in May and constant feedback from my tutor.
The following chapters will present the developed neural network model, the different training tests and problem-solving strategies, the results with a comparison with the TagV tool, and further development.

Chapter 3

Implementation of the Graph Neural Network

1 Graph Neural Networks

1.1 Definition

Algorithms capable of learning patterns from multi-dimensional data and predicting features on previously unseen inputs are called Machine Learning (ML) algorithms. They are very useful in particle physics, as analysis exploits many correlated parameters and the acquired datasets are huge. Additionally, studies focused on rare decays are very challenging to efficiently filter and extract relevant events from background noise. Since these algorithms can learn complex patterns in the data, they enable more accurate classification, anomaly detection, and reconstruction of physical processes.
The data flow of a ML algorithm is detailed in figure 3.1:

1. First the algorithm receives input data and processes it to generate a prediction. This is the **Forward pass**.

2. Then it compares the prediction to the real value by calculating the error it has made, called the **loss**. This error is therefore measured using a **loss function**.

3. Then it calculates the gradients of the loss with respect to the parameters of the model, during a step called **Backpropagation**. This shows how much and in which direction each parameter affects the error.

4. As the goal of the algorithm is to minimize the loss, it adjusts its parameters by assigning weights to the relationships between data, using a method called **gradient descent**. This is the update step. This process iteratively updates the model by moving toward local minima of the loss function, with each update controlled by a parameter known as the **learning rate**, which determines the step size. This optimization technique is illustrated in figure 3.2

5. Finally it makes a new prediction and the cycle repeats itself, for a specific number of **epochs**, which is the denomination of one cycle.

In summary, every learning algorithm is composed of three main building blocks : a loss function, an optimization criterion (minimization of the loss), and an optimization routine (gradient descent)[4].

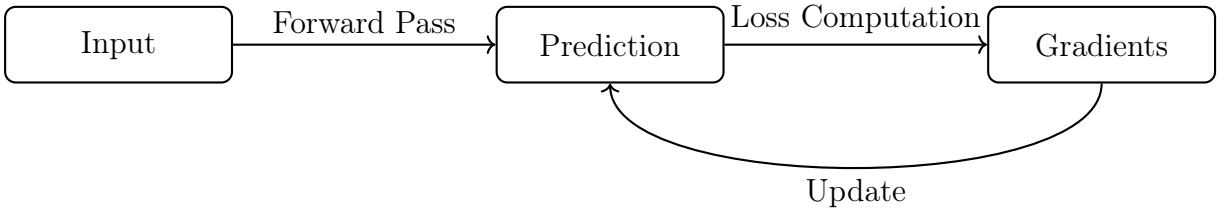


Figure 3.1: Simplified machine learning flow

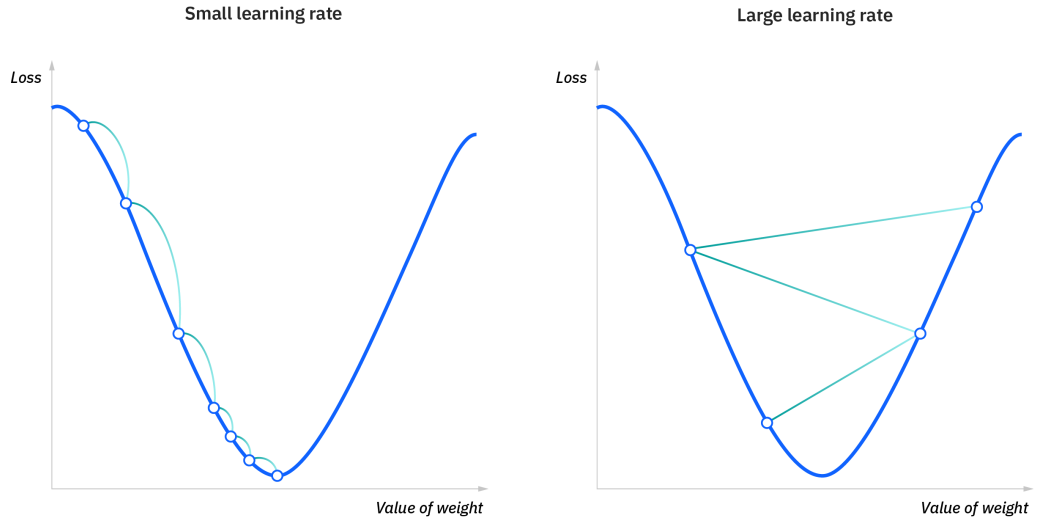


Figure 3.2: Gradient descent illustrated for two different learning rates [10]

Artifical neural networks (NN) are a specific kind of ML algorithms, which are inspired by the structure and functioning of the human brain, where information is processed through interconnected units called neurons. In a neural network, input data flows through multiple layers of neurons. Each neuron applies a linear transformation to its inputs, consisting of a weight and a bias, followed by a non-linear activation function that allows the network to model complex patterns. The output of each neuron is passed to the next layer and propagates through the network though the same ML flow seen earlier.

In standard networks, input data are handled as a vector and updates from learning change the neuron parameters (weights and biases). In contrast, Graph Neural Networks

(GNN) structure the data as a graph to take into account relations between the inputs. The graph itself is updated through layers of NN. GNN are therefore well adapted to problems where such structure exists, which is the case of a cascade of decays generated by B mesons.

1.2 The Graph Object and its layers

As stated before, GNN uses graph-structured data. As shown in Figure 3.3, a graph is composed of nodes v, edges e and global features u. The nodes represent in our case the particles and their characteristics, like charge, momentum and point of closest approach to the collision point. The edges are the relationship between particles, like the distance between two tracks or the angle between two tracks. Finally, the global features can be whatever characterizes the whole graph, such as the number of particles in total, or, as we will see, the position of the B meson decay.
A complete GNN is made of N layers of graphs G_N and $N-1$ graph network blocks GN_N used to update each graph, each with the exact same structure but with evolving values for the nodes, edges and global features.

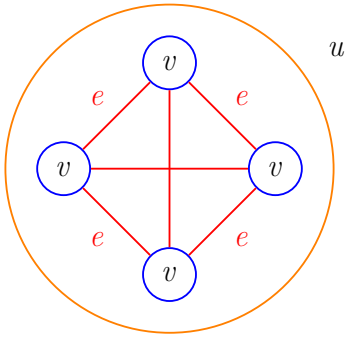


Figure 3.3: The Graph Object's Structure.

The next step is to understand how the graph G_i is connected to the graph of the next layer G_{i+1} through a graph network block made of various NNs, illustrated in figure 3.4. The different types of features are learned during three separate blocks: the edge block, the node block, and the global block.

The first feature that the neural network processes is the edges. It processes it using a Multilayer Perceptron, a common NN algorithm, taking the nodes, edges and global features as inputs. The edges are then predicted and sent to the next block after being averaged, in order to make sure that the input size corresponds to the needs of the next Perceptron. The next block takes the nodes and global features as well as the averaged

predicted edges and predicts the nodes. Finally, the input global feature and updated, averaged nodes and edges are given to the last block, predicting the global feature, giving a completely updated graph object in the end. The parameters of all NN of each block are set during the learning phase.

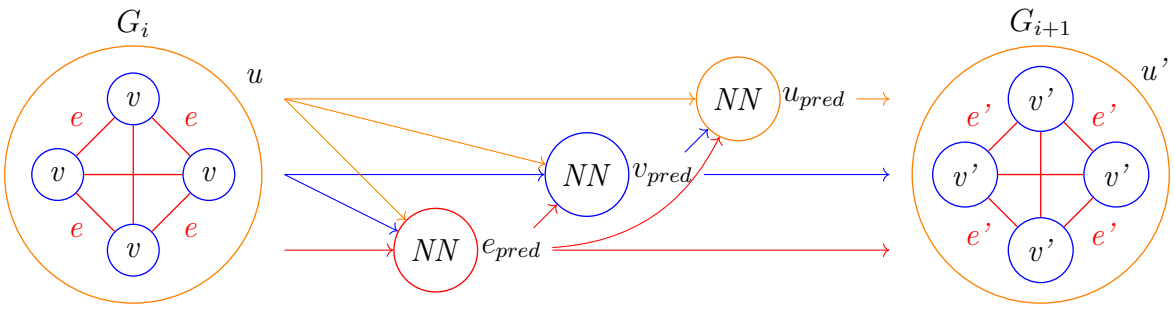


Figure 3.4: A Graph Neural Network Block

1.3 GraFEI

The currently used tagging algorithm by Belle II is TagV, which performs a global fit of the decay tree to get the position of the B-meson vertex. As seen earlier, this technique may be highly efficient, it has low signal purity since it can confuse the B meson tracks with the D meson ones. To obtain better purity, the idea is to use a GNN algorithm that would predict the decay tree without any information on intermediate particles (inclusively). This problem can simply be formulated like so: Can the structure of a tree graph be determined from its leaves alone [12]? The GraFEI (Graph Full Event Interpretation) is the solution to this question [1].

Decay trees can be seen as rooted directed acyclic graphs, where the depth can be associated to a stage or a color as seen in the illustration 3.5. Such graphs are usually represented mathematically by an adjacency matrix A, where the matrix element $a(i, j)$ is equal to 1 if the i and j nodes are connected or 0 otherwise. But the inconvenience in this representation is that it requires the knowledge of the intermediate particles. The solution is to use the Lowest Common Ancestor matrix (LCA), where the dimension of the matrix is equivalent to the number of Final State Particles (FSP) and the value of the elements corresponds to their lowest common ancestor. The ancestors are then associated with a class, which transforms the LCA matrix into a Lowest Common Ancestor Stage matrix (LCAS), where the classes are: 5 for B, 4 for D^*, 3 for D, 2 for K_s^0, 1 for π^0 or J/ψ and 0 for particles not in the decay tree.

The GraFEI reconstructs the LCA matrix using the edge features and compares its prediction using a loss function adapted to class-type data, called Cross Entropy. It is

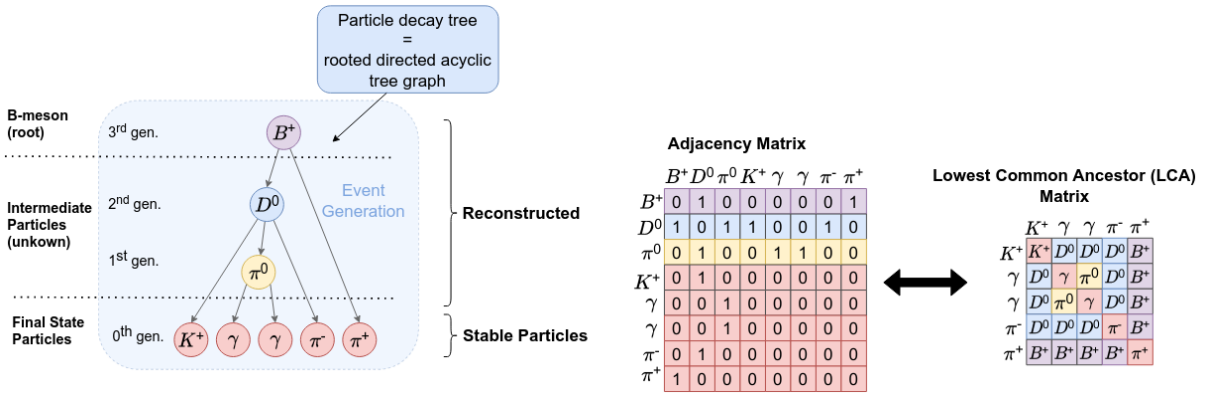


Figure 3.5: Graph to LCA matrix [17]

calculated in Pytorch [15], the ML library used, with a "mean" method of reduction:

$$\mathcal{L}_{lca} = -\frac{1}{N}\sum_{i=1}^{N}\log\left(\frac{e^{x_{i,y_i}}}{\sum_{j=1}^{C} e^{x_{i,j}}}\right) \tag{3.1}$$

where:

- N is the number of samples in the batch,
- C is the number of classes,
- $x_i \in \mathbb{R}^C$ are the raw logits (unnormalized predictions) for the i-th sample,
- $y_i \in \{0, \ldots, C-1\}$ is the target class index for the i-th sample,
- x_{i,y_i} is the logit corresponding to the true class.

In order to predict with GraFEI the B_{tag} meson decay vertex position, vertex information must be implemented in the existing graph. Since it is a feature that depends both on the particles and their relationship, it has been chosen to be implemented as a global feature.

Understanding the Belle II's framework called Basf2 [13], the functioning of the GraFEI and the successful implementation of the feature represented the most complex and substantial part of this internship. A detailed description of the code is omitted from this report.

2 Setup

In ML, there are multiple types of datasets used. Indeed, the model needs data in order to learn, to be tuned and to be tested. These datasets are called: the **training** dataset,

which is used to fit the parameters of the model, the **validation** dataset, used to tune the hyperparameter of the model, and the **test** dataset (or **inference**), used to assess the performances of the whole model [18]. The training files created with the vertex positions contain 900 000 simulated $B_{sig} \to \nu\bar{\nu}$ events, that also contain background noise. This quantity was chosen because prior work [20] identified it as offering the most effective trade-off between accuracy and computational time, and an ideal number of epochs is said to be 50. The validation dataset is made of 300 000 events, making it a 75-25 type of partition. The test dataset that will lead to all the results is also made of 300 000 events.

3 Loss Tuning

There are many possible choices of loss functions, each is specific to the type of data that are processed. For LCA training, the cross-entropy function was used, but for floating values such as vertex positions, we can use a much more intuitive function called Mean Squared Error (MSE) [16].

$$\mathcal{L}_{vtx} = \frac{1}{N}\sum_{i=1}^{N}(z_i - \hat{z}_i)^2 \tag{3.2}$$

where:

- N is the number of samples,
- z_i is the ground truth target for the i-th sample,
- $\hat{z}_i$ is the predicted output for the i-th sample.

In order to predict both LCA and vertex positions, we must combine the two loss functions and take into account that their orders of magnitude are different. Indeed, since the mean square error is in fact very close to the variance of the residual distribution seen in the result section, and knowing that we expect a standard deviation of the order of 100 μm, the squared value should be around 10^4. In contrast, it is around 1 or 10 for a class. The problem is that since the purpose of the model is minimizing the loss, if one loss is larger than the order, it will prioritize its minimization and learning, and ignore the rest. Therefore, we add the α coefficients in order to tune the loss, so we can choose what to train in priority [9].

$$L = \alpha_{lca}\mathcal{L}_{lca} + \alpha_{vtx}\mathcal{L}_{vtx} \tag{3.3}$$

Chapter 4

Model Performance

After implementing the position of the decay vertex as a global feature in the GraFEI, it was possible to train the model on three types of data : on LCA, on vertex positions, and on both simultaneously by tuning the global loss function. The results from the original model serve as reference for good loss function aspect and hyperparameter base. This chapter will present the GraFEI's results focusing on the resolution of the predicted values and the comparison with the TagV tool.

1 Classical GraFEI : a solid base

The original GraFEI model is used as a reference to evaluate the performance of the new architecture. As it has already been optimized [20], it can be considered as a solid baseline for further training and extension to vertex-position regression. For instance, figure 4.1 shows the evolution of the loss function across the epochs: it is successfully being minimized by the model, and the validation loss is superposed with the training loss at the beginning, meaning that there is no significant overfitting before a certain point. The training has been done on 100 epochs and we can observe that the loss function reaches a plateau at 30 epochs and starts to overfit, which makes it a good stopping point for the training in order to be more time-efficient. Overfitting happens when the model starts to model noise in the training data rather than the underlying pattern. Therefore, the next trainings will all be done on 30 epochs. This loss function behavior is something we aim to preserve throughout all training phases.

2 Vertex Training study

2.1 Vertex only

For the training on vertex positions, α_{lca} has been set to zero, and the learning rate has been adapted to large MSE losses. During the validation step, we observe the

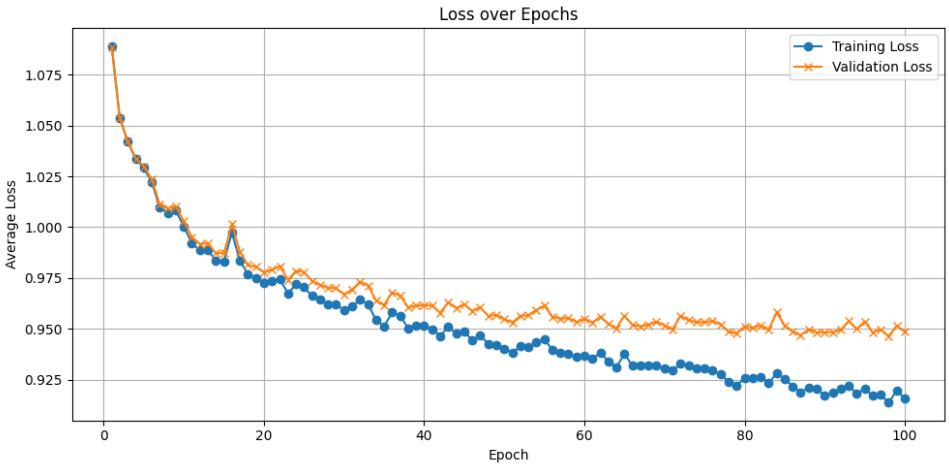


Figure 4.1: Loss over 100 epochs for LCA learning

superposition of the predicted values and the true values in figure 4.2, meaning that the model seems to learn successfully the position of the B^0_{tagz} vertex position.

After training and validation, we run the algorithm on the test dataset in order to assess the resolution of the model, in this work we call this step the *inference*. Figure 4.3 represents the residue computed as $z_{pred} - z_{truth}$. It has been fitted using a mixed gaussian. The Gaussian mixture method (GMM [14]), allows to account for deviations from normality, such as heavier tails as seen in the residue distribution, that a single Gaussian cannot represent. Equation 4.1 is the general form of a Gaussian Mixture.

$$f(x) = rN(x|\mu_1, \sigma_1) + (1 - r)N(x|\mu_2, \sigma_2) \tag{4.1}$$

with $N(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$ where:

- μ, μ_1, μ_2 are the means of the Gaussian components,
- $\sigma, \sigma_1, \sigma_2$ are the corresponding standard deviations,
- $r \in [0, 1]$ is the mixture weight.

We used the ROOT framework [19] to fit the histogram of residuals along each coordinate, optimizing the six parameters of the mixture model ($[0] = r$, $[1] = \mu$, $[2] = \sigma_1$, $[3] = \sigma_2$, $[4] = N$ to normalize) in the following function from equation 4.2, and choosing $\mu = \mu_1 = \mu_2$ as the residue is expected to be unimodal.

$$f(x) = \frac{[4]}{\sqrt{2\pi}}\left(\frac{[0]}{[2]} \times e^{-\frac{1}{2}(\frac{x-[1]}{[2]})^2} + \frac{1-[0]}{[3]} \times e^{-\frac{1}{2}(\frac{x-[1]}{[3]})^2}\right) \tag{4.2}$$

The standard deviation is then simply computed as $\sigma = r\sigma_1 + (1-r)\sigma_2$, and to propagate the uncertainties the formula becomes:

$$\delta\sigma = \sqrt{\left(\frac{\partial\sigma}{\partial r}\cdot\delta r\right)^2 + \left(\frac{\partial\sigma}{\partial\sigma_1}\cdot\delta\sigma_1\right)^2 + \left(\frac{\partial\sigma}{\partial\sigma_2}\cdot\delta\sigma_2\right)^2} \tag{4.3}$$

With:

$$\frac{\partial\sigma}{\partial r} = \sigma_1 - \sigma_2, \quad \frac{\partial\sigma}{\partial\sigma_1} = r, \quad \frac{\partial\sigma}{\partial\sigma_2} = (1-r) \tag{4.4}$$

The standard deviation of the fit is an estimator of the resolution. By computing the total standard deviation, we have $\sigma = 52.91 \pm 0.31\ \mu m$.

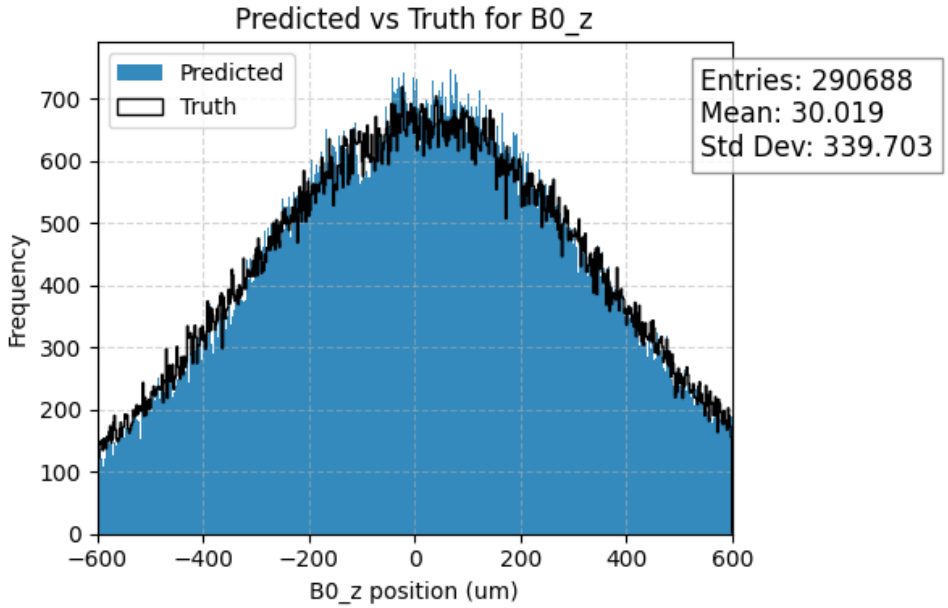


Figure 4.2: Predicted vs True B^0_{tagz} vertex position

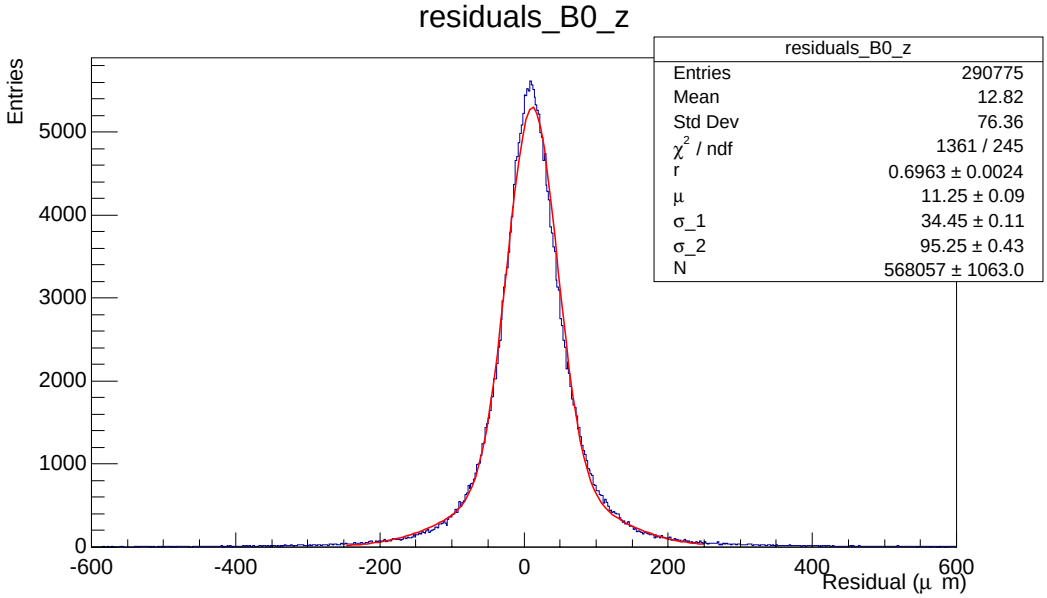


Figure 4.3: Fitted residue for training on vertex

2.2 Vertex and LCA

For the training on vertex positions and LCA, the previous setup has been kept and α_{lca} has been set to 100, 1000, 10^4, and 10^6. By increasing α_{lca} we increase the weight of the LCA loss in the model, meaning that it will increasingly focus on learning the decay tree pattern contained in the LCA matrix.
Since we now train on LCA too, it is interesting to look at the events that have perfectly reconstructed LCA matrices by the GraFEI, and see the evolution of the resolution for these events. The table 4.1 resumes the results for the different values of α_{lca}, and table 4.2 has the results for the perfectly reconstructed LCA event residues for varying α_{lca}. The distributions of the residuals for global and perfectly reconstructed LCA events are seen in figure 4.4, with the mixed gaussian fit in red on top. We can observe that the number of perfectly reconstructed LCA events is increasing with α_{lca} in the figures on the right (b),(d),(f) and (h), going from 23124 to 40061 events for $\alpha_{lca} = 10^6$.

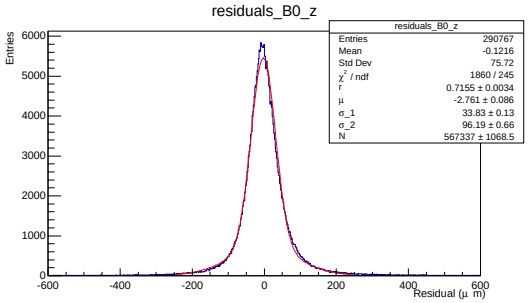
The fit has been made on more than 95 % of the data every time. A better fit may be envisioned for future improvement, as the χ^2/ndf value which represents the goodness of the fit is large (it should be close to 1 ideally).
Also, since the addition of LCA does not significantly improve the resolution as seen in Table 4.1, it was decided to do all further tests only on vertex trainings.

Table 4.1: Residue results for varying α_{lca}

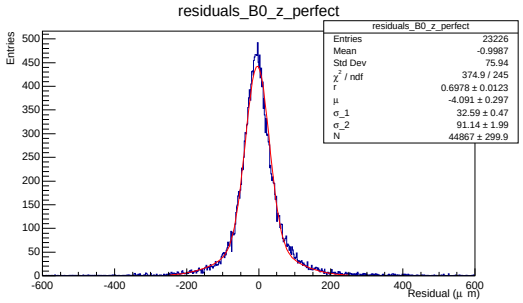
α_{LCA}	μ	σ	χ^2/ndf	r	σ_1	σ_2
0	11.25 ± 0.09	52.91 ± 0.31	5.55	0.697 ± 0.004	34.45 ± 0.15	95.25 ± 0.64
100	-2.77 ± 0.01	51.54 ± 0.29	7.59	0.715 ± 0.003	33.8 ± 0.1	96.11 ± 0.66
1000	15.69 ± 0.09	53.74 ± 0.33	9.44	0.742 ± 0.004	37.82 ± 0.15	99.43 ± 0.78
10^4	5.07 ± 0.1	55.21 ± 0.34	11.55	0.743 ± 0.004	39.51 ± 0.16	100.6 ± 0.8
10^6	21.73 ± 0.29	142.42 ± 0.42	7.94	0.070 ± 0.003	42.45 ± 1.26	149.9 ± 0.3

Table 4.2: Perfect LCA event residue results for varying α_{lca}

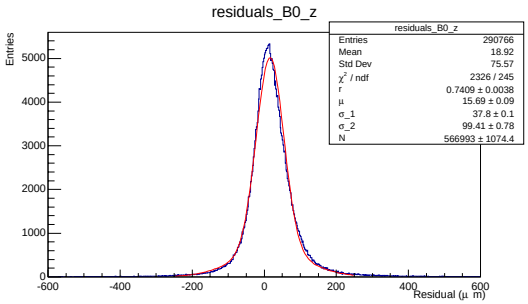
α_{LCA}	μ	σ	χ^2/ndf	r	σ_1	σ_2
100	-4.22 ± 0.3	50.06 ± 0.97	1.58	0.696 ± 0.012	32.43 ± 0.47	90.4 ± 1.9
1000	18.33 ± 0.26	51.14 ± 0.89	2.74	0.765 ± 0.011	37.59 ± 0.42	95.14 ± 2.22
10^4	-0.42 ± 0.25	51.66 ± 0.83	2.32	0.705 ± 0.012	36.5 ± 0.4	87.98 ± 1.57
10^6	10.23 ± 0.82	157.69 ± 1.05	1.29	0.046 ± 0.004	28.53 ± 2.39	163.9 ± 0.9



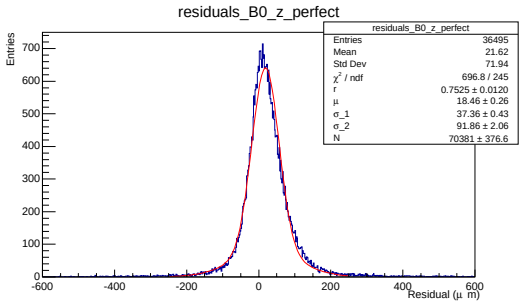
(a) Residues for $\alpha_{lca} = 100$



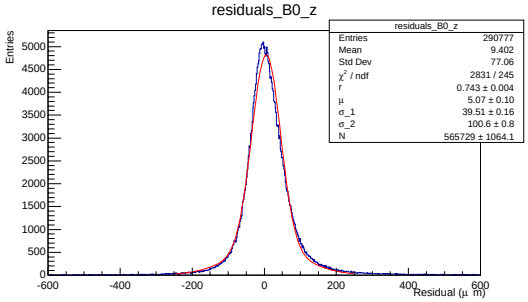
(b) PerfectLCA residues for $\alpha_{lca} = 100$



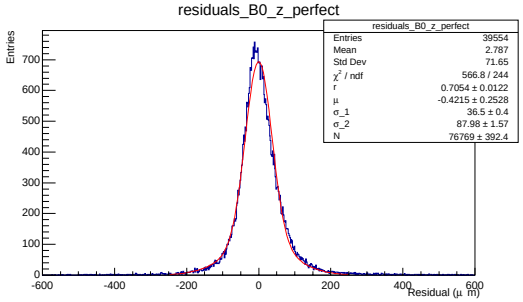
(c) Residues for $\alpha_{lca} = 1000$



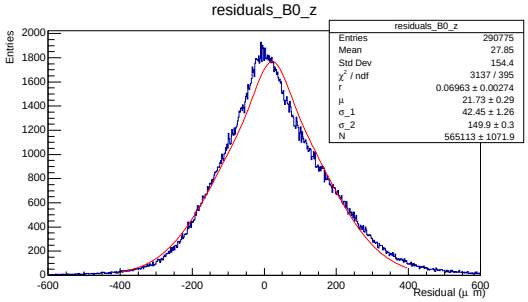
(d) Perfect LCA residues for $\alpha_{lca} = 1000$



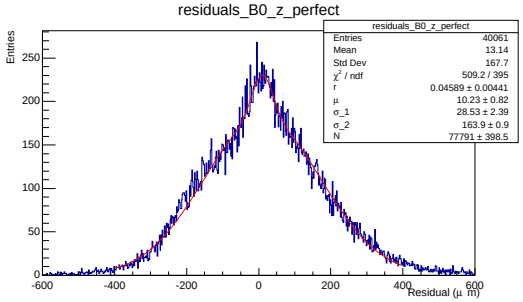
(e) Residues for $\alpha_{lca} = 10000$



(f) Perfect LCA residues for $\alpha_{lca} = 10000$



(g) Residues for $\alpha_{lca} = 1000000$



(h) Perfect LCA residues for $\alpha_{lca} = 1000000$

Figure 4.4: Residual plots and perfectly reconstructed LCA residues for varying α_{lca} values.

3 Inference and TagV

The main result of this work is the comparison of the vertex tagging GraFEI and TagV. To do so, we compare the residual distribution from Figure 4.5 of both the inference of the GraFEI training with $\alpha_{lca} = 0$ and $\alpha_{vtx} = 1$, and TagV, on the same dataset. The TagV tool was used without any constraints and the standard options [22].

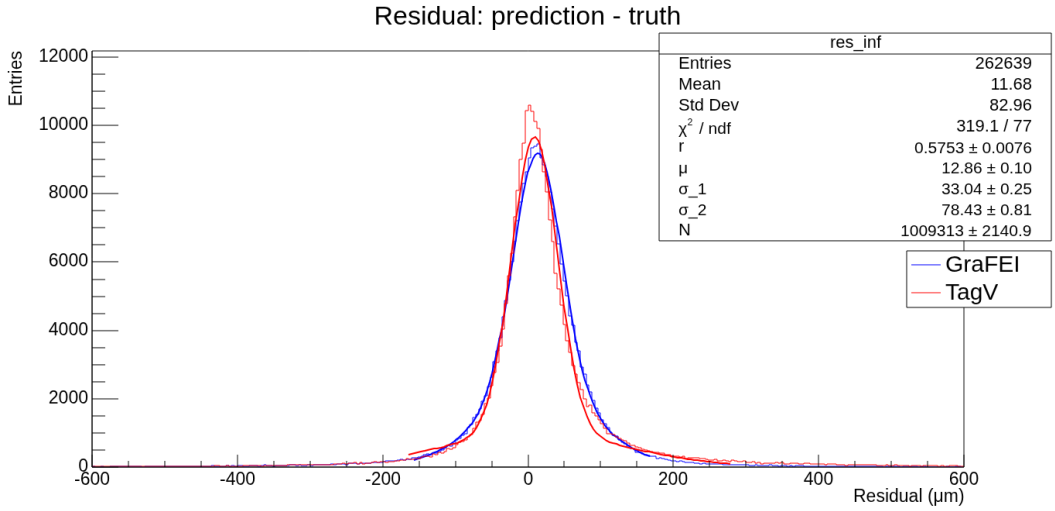


Figure 4.5: Fitted residue comparison between GraFEI and TagV

The fits were performed on 95% of the distribution. For GraFEI, between -159.66 μm and 168.48 μm, and for TagV, between -167.20 μm and 280.19 μm. To do the comparison between GraFEI and TagV, it was necessary to filter out some events that TagV did not manage to tag, or that were large outliers with residues larger than 1000 μm. From the fits visible in this figure, the resolution for the GraFEI is similar to what we have found before:

$$\sigma_{\text{GraFEI}} = 52.32 \pm 0.51\,\mu\text{m} \quad (\chi^2/\text{ndf} = 4.14)$$

$$\sigma_{\text{TagV}} = 60.44 \pm 0.47\,\mu\text{m} \quad (\chi^2/\text{ndf} = 41.44)$$

4 Another Dataset: $B \to \mu^+\mu^-$

It is interesting to see how the model is able to recognize decay patterns on a type of dataset it has never seen before, especially on a type of decay that has a signal side, unlike $B \to \nu\nu$ since the neutrinos do not interact with matter. After generating a small dataset of 10 000 $B \to \mu^+\mu^-$ events, it was possible to try both GraFEI and TagV on it by running an inference.

Figure 4.6 shows the following results for resolution:

$$\sigma_{\mathrm{GraFEI}} = 119.88 \pm 7.32\,\mu\mathrm{m} \quad (\chi^2/\mathrm{ndf} = 1.51)$$

$$\sigma_{\mathrm{TagV}} = 59.62 \pm 2.23\,\mu\mathrm{m} \quad (\chi^2/\mathrm{ndf} = 1.75)$$

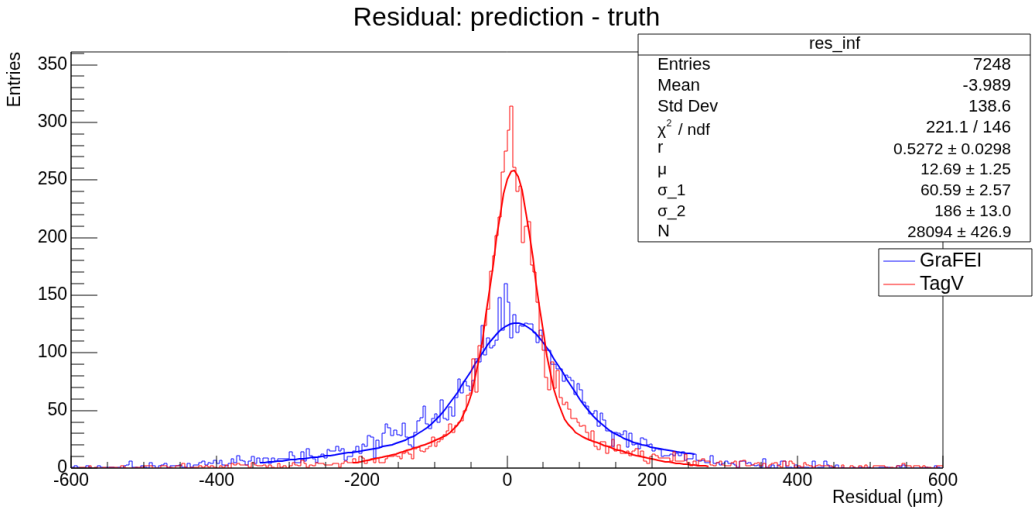


Figure 4.6: $B \to \mu^+\mu^-$ residue comparison

Chapter 5

Analysis and Improvements

1 Correlation fix

During training and validation, a clear correlation has been observed between the mean residual and the true vertex position, as shown in figure 5.1. This indicates that the model's predictions systematically deviate depending on the true z-position. The TagV model does not present any correlation. Once again, a filter on untagged values and residues smaller than 1000 μm has been applied, explaining the smaller number of entries.

Figure 5.1: Fitted residue comparison between GraFEI and TagV

The two profiles were fitted using an affine function on 95% of the distribution. The fit functions are presented in equation 5.1.

$$\begin{aligned} r_{GraFEI} &= 3.69 - 0.087 z_{truth} \\ r_{TagV} &= 21.61 - 0.002 z_{truth} \end{aligned} \tag{5.1}$$

Correcting this bias could lead to further gain in resolution, therefore, multiple methods have been investigated.

1.1 Biased learning

1.1.1 Lifetime prior

Our initial hypothesis was that the observed correlation between the mean residual and the true z position is driven by the non-uniform lifetime prior present in the training data. Because B-meson lifetimes follow an exponential distribution, short lifetimes (hence short flight distances) occur much more frequently than long ones. When optimizing a mean-squared-error loss, this imbalance may pull the regression toward the densest region of the target space, so predictions tend to be biased toward smaller decay lengths. A direct consequence is a systematic trend of the average residual with z_{truth}.

To test and mitigate this effect, we attempted to reduce the influence of the lifetime prior by reweighting the loss with the inverse of the decay law. Using the relationship shown in Fig. 5.2 and Eq. 5.2, each event with lifetime t is assigned a weight $w(t) \propto \exp(t/\tau)$, which upweights long-lived decays and in theory makes the training distribution in t closer to uniform. In principle, this reweighting should lessen the prior-induced bias and flatten the residual vs z_{truth} profile.

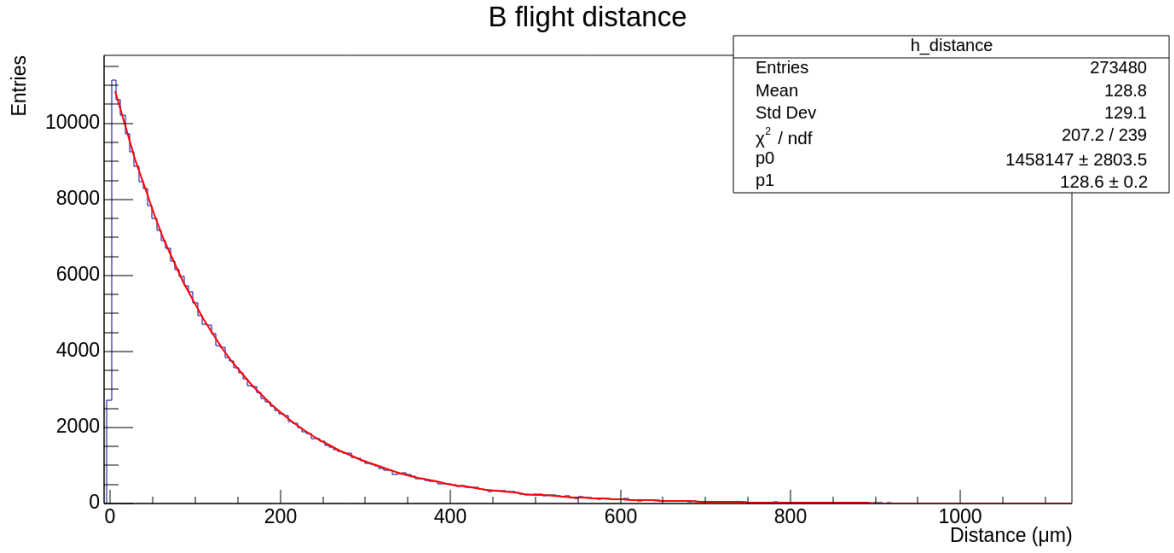


Figure 5.2: B exponential decay law

$$\exp\left(-\frac{t}{\tau}\right) = \exp\left(-\frac{\beta\gamma\, c\, t}{d}\right), \tag{5.2}$$

with the parameters defined as:

- $d = p1$: mean flight distance,
- τ: mean time of flight,
- c: speed of light,
- β: ratio of the relative velocity to the speed of light,
- γ: Lorentz factor,
- t: lifetime.

The lifetime t was added as a parameter when creating the training files so it propagates through the model and can later be used for weighting.

Figure 5.3 shows that this strategy did not fix the correlation.

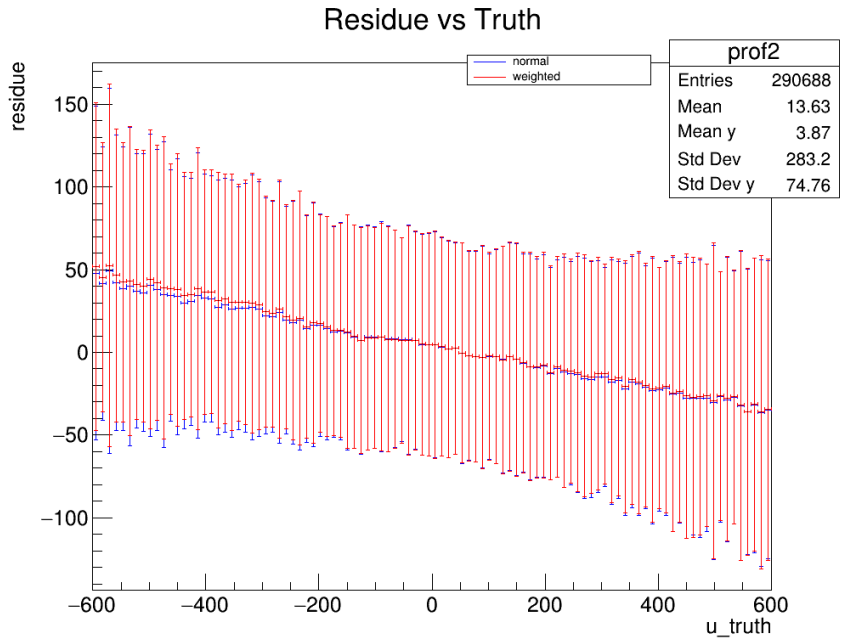


Figure 5.3: Reweighted loss profile comparison

1.1.2 Lifetime filtering

Another explored strategy is to flatten the lifetime distribution by rejection sampling. First we keep only the events that have a lifetime smaller than τ. Then, for each remaining event we draw a uniform random number between 0 and 1, and accept the event if the following condition from equation 5.3 is met.

$$rand[0,1] < \frac{\exp(-1)}{\exp\left(-\frac{t}{\tau}\right)} \tag{5.3}$$

Here exp(−1) is the the value of the exponential decay law when the lifetime is equal to τ.
This acceptance probability increases with t as short-lived events are downsamples and long-lived ones kept more often. This way, we have a uniform input of lifetimes.
Figure 5.4 with the new test called "lifecut" shows that by doing so, the bias becomes stronger.

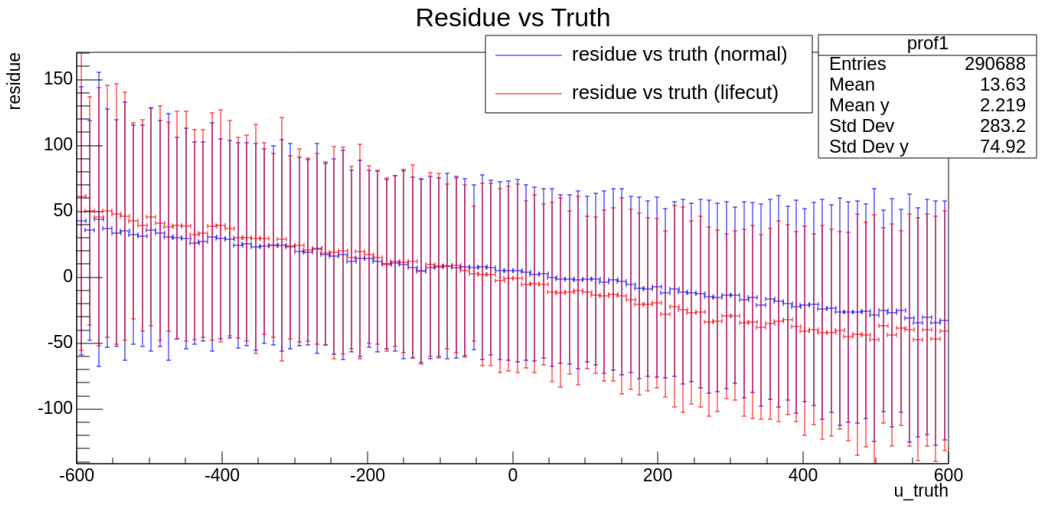


Figure 5.4: Uniform sampling profile comparison

These two strategies to fix the correlation were unsuccessful.

1.2 Calibration

The final decision was to simply calibrate the predicted values, by compensating the correlation. To do so, we fitted the true vertex position as a function of the predicted position in Figure 5.5, and used the resulting fit as a calibration function f. All predictions were then replaced by their calibrated values $f(z_{pred})$

The calibrated residual distribution is shown in Figure 5.6. After calibration, the resolution improves to $\sigma = 51.76 \pm 0.49\ \mu m$ with a better goodness of fit since $\chi/\text{ndf} = 2.97$. Figure 5.7 displays the profile fits computed on 95% of the distribution, and the fit of the GraFEI profile (equation 5.4) has flattened, indicating that some correlation has been removed.

$$r_{GraFEI} = -10.97 - 0.060 z_{truth} \tag{5.4}$$

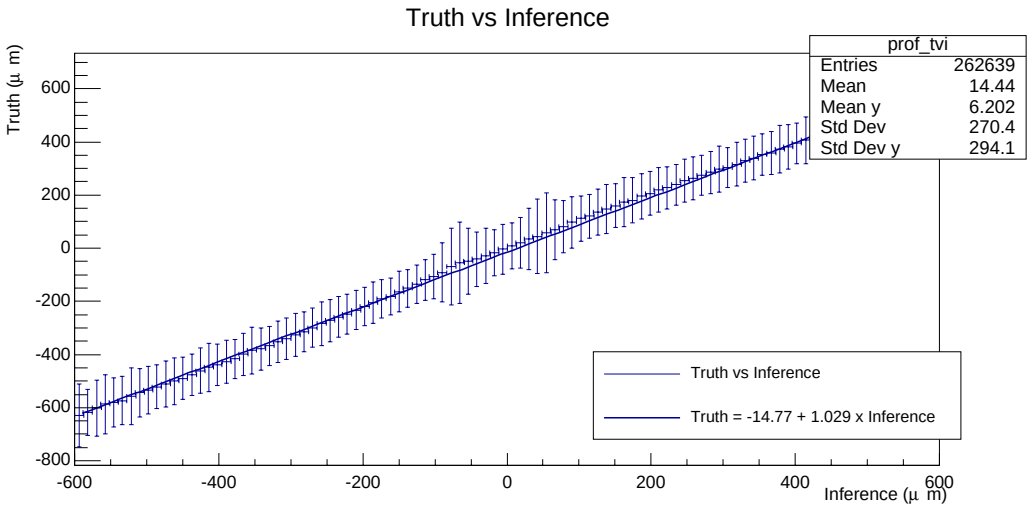


Figure 5.5: Truth vs prediction (inference) profile

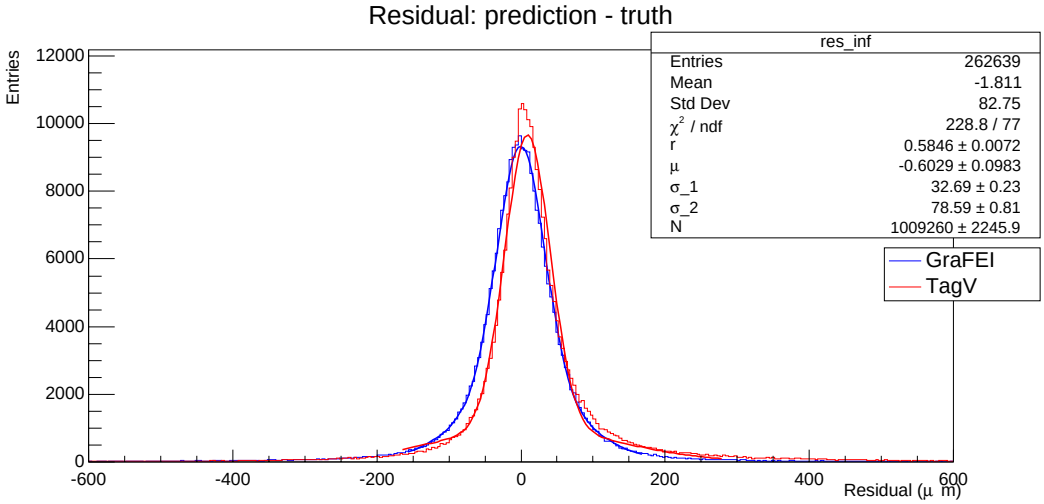


Figure 5.6: Calibrated residue result

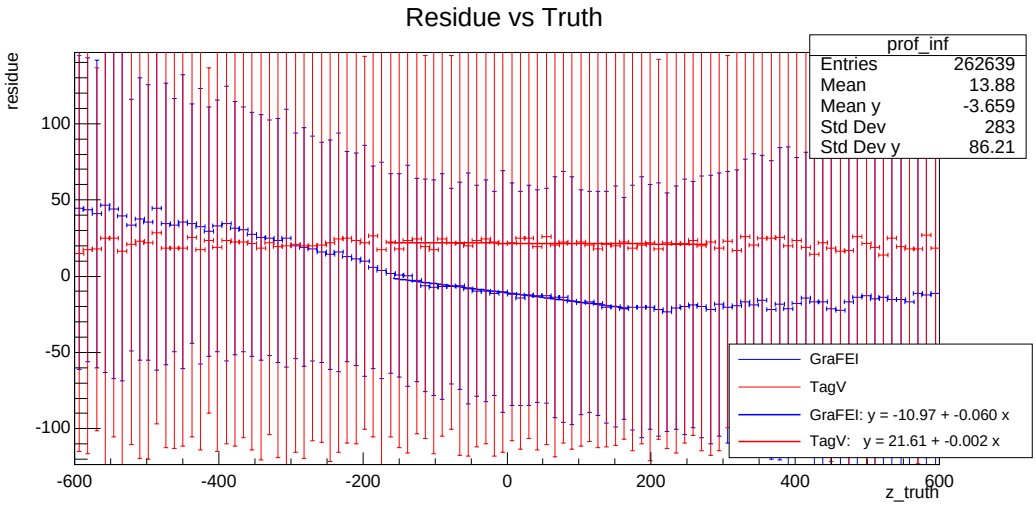


Figure 5.7: Calibrated profile result

2 Improvement tests

2.1 TagV initialization

In order to improve the performance of the GraFEI, an idea of directly giving as input the predicted vertex position by the TagV tool was considered. Contrary to expectations, this degraded the performance: the predicted values were further from the truth and the residual distribution showed a larger resolution. We therefore decided to keep a simple $[0, 0, 0]$ tensor initialization of the global layer.

2.2 Increase of model complexity

We explored the idea of adding an additional block of GNN to our model, making the network more complex. The results do not show an improvement in vertex tagging, as depicted in Figure 5.8 and Figure 5.9 where one GNN block results are in blue and two GNN block results, in red. On the contrary, the residual distribution becomes broader with heavier tails and the correlation profile exhibits an increased bias with a stronger slope. In short, increasing network depth took more computational time and decreased performances.

3 Flavor Tagging

To truly compute the CP violation asymmetry, we need to also know the flavor of the B meson (if it is the particle B or the anti-particle $\bar{B}$). Therefore, an implementation of the flavor loss has been tested, by creating a second target in the global layer of the GraFEI. The following table 5.1 summarizes the flavor prediction of four different tests. To quantify the performances of the model, we will compute the tagging efficiency ε_{tag} and the mistag probability ω, defined below.

$$\varepsilon_{\text{tag}} = \frac{N_r + N_w}{N_r + N_w + N_{\text{untagged}}} \qquad \omega = \frac{N_w}{N_r + N_w}$$

- N_r is the number of correctly tagged events,
- N_w the number of wrong.

The number of total events is 310500, and the number of tagged events is 218965. Therefore the tagging efficiency of the GraFEI is: $\varepsilon_{tag} = 0.71$, meaning 70% of events were tagged.

The results show that the model does better guesses when it learns simultaneously the LCA and vertex parameters, but it misses in more than 50% of times.

Table 5.1: Residue results for varying α_{lca}

α_{LCA}	α_{vtx}	α_{flavor}	N_r	N_w	ω
0	0	1	102633	116332	0.53
0	1	1	105052	113913	0.52
0	1	100	104938	114027	0.52
100	1	1	106068	112897	0.52

The following parameter tests were chosen, in order to rapidly inspect the impact of vertex and LCA training on flavor tagging. Here it shows that it is barely sensitive to the multitasking.

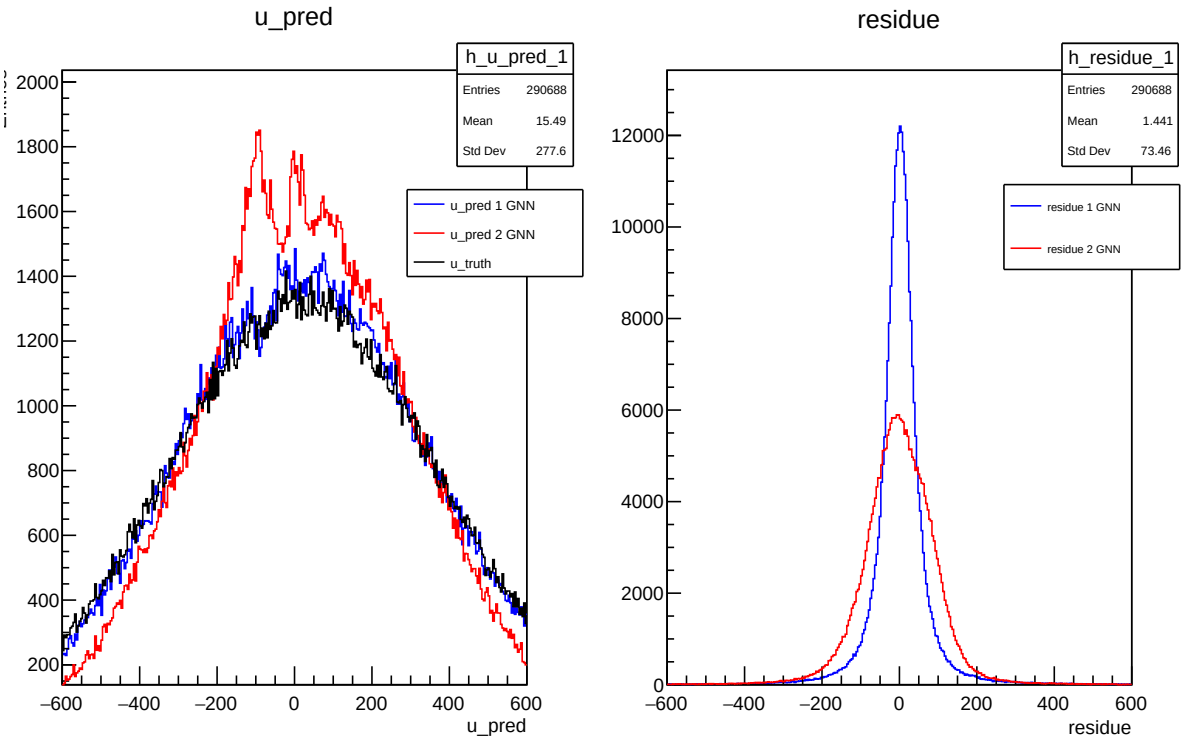


Figure 5.8: Two GNN residue comparison

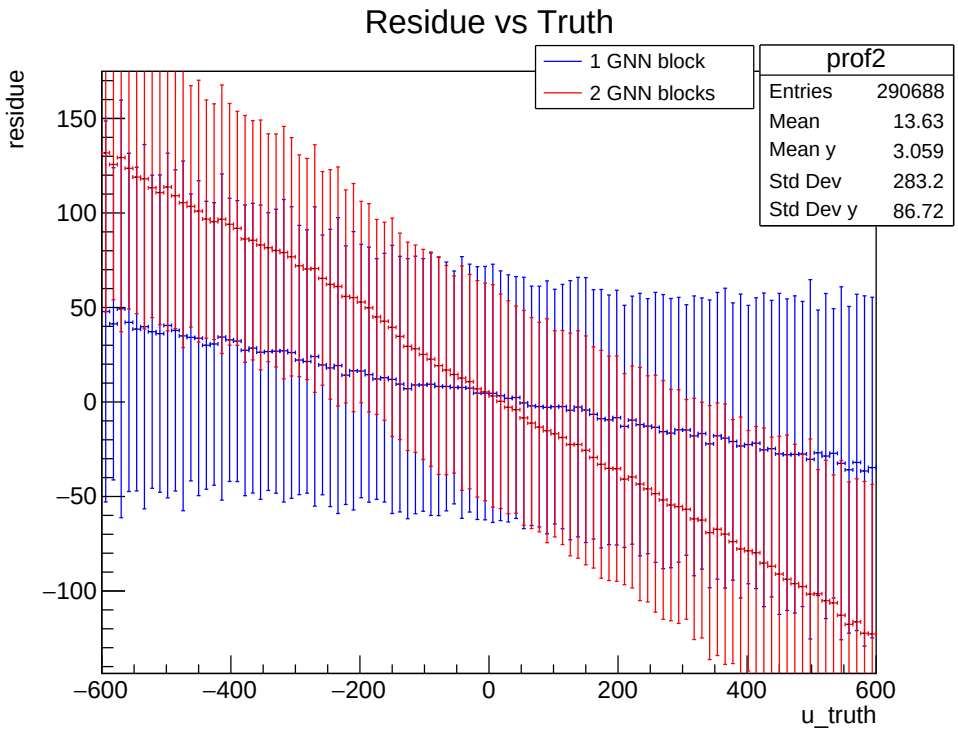


Figure 5.9: Two GNN profile comparison

Chapter 6

A Better Model for Vertex Tagging ?

1 Comparison with existing model

The results presented in Chapter 4 show that first of all, the GraFEI is able to correctly predict the vertex positions, and that on the $B \to \nu\nu$ dataset, the best resolution we can get when training solely on vertex positions is $\sigma = 51.76 \pm 0.49\ \mu m$, if we calibrate the data. This means that the GraFEI presents a competitive precision and an improvement of about 13.4% compared to TagV.

On the other hand, TagV does not present any correlation, which is good since it shows a more stable and reliable model. However it has very large errors that were necessary to filter out because they didn't allow to properly compare the profiles, which is why the two gaussian mix wasn't the best fit since it couldn't account for such large tails. Nevertheless, on unseen physics channels such as $B \to \mu^+\mu^-$, TagV is significantly better. This shows that GraFEI is more specialized for now, while TagV is more generalizable.
An important comment is that TagV was run with its default, unconstrained configuration. In Belle II analyses, TagV is usually used together with an interaction-point (beam-spot) constraint and tight track selections (e.g., for fully reconstructed signal B data we could add the constraint of tube along the tag B line of flight), which sharpen the tag-side vertex and can improve the resulting resolution. Enabling these standard constraints would likely reduce TagV residuals and narrow the gap to GraFEI.

There is still room for improvement of the GraFEI to try and train it on different data where there is a signal side and test it again on the $B \to \mu^+\mu^-$ sample.

2 LCA training impact

The table 4.1 does not suggest any significant improvement in global resolution with the inclusion of the LCA loss term. While it seems to be slightly better for $\alpha_{lca} = 100$, it coincides with an increase in the χ^2/ndf term, so a better fitting function is needed. Even for perfectly reconstructed LCA events (table 4.2), there is only a slight increase in resolution. For $\alpha_{lca} = 100$, the resolution reaches $\sigma_{perfect} = 50.06 \pm 0.29\ \mu m$, compared to the global $\sigma = 51.54 \pm 0.31\ \mu m$, an improvement of about 3%. Although, the corresponding χ^2/ndf approaches 1, indicating that the Gaussian mix fits these events well.

The researched result was to see if, when the model successfully understands the relationship between particles via the LCA, it is capable of making more accurate vertex predictions. Here we cannot make such a conclusion, as the resolution degrades when the training concentrates on LCA reconstruction.

Finally, it is important to note, that these perfectly reconstructed LCA events account for only about 8 % of the test dataset. Therefore, more input data is needed in order to have enough statistics and truly conclude on the impact of LCA-driven training.

3 A biased model

To calibrate the data in order to get rid of the observed correlation was the pragmatic solution. Ideally, a solution should be found for the model to not create a biased mapping intrinsically, as TagV effectively does in our comparison. The fact that the attempts at reweighting with the inverse lifetime prior and rejection sampling to flatten the lifetime distribution did not help and sometimes made the correlation even worse in the uniformization case, suggests that it arises perhaps from systematic biases in training rather than from the data distribution alone.

4 What about flavor tagging ?

Because of the multitasking GraFEI can do, it was thought that it could manage to do flavor tagging as well. In fact, it works well since it is capable of tagging 71% of the events. However, the mistag rate being bigger than 50% proves that the flavor prediction is no better than random guessing so there is perhaps more digging to do in the correct way to implement flavor tagging in the model.

Many possible improvements to the GraFEI exist, but as research advances, graph neural networks are maybe not the best candidates anymore for such tasks. For instance, Transformers are becoming very popular since they are used for large language models. Other teams are currently using them for flavor tagging, and are exploring the use of these new models for vertex tagging. A future comparison between GraFEI and their model could be considered.

Conclusion

This work investigated the potential of the GraFEI graph neural network to improve the resolution on the B meson decay vertex position evaluation. A resolution of $\sigma = 51.76 \pm 0.49\ \mu m$ on the position of the $B_{tag,z}$ vertex was achieved, using a mean squared error vertex-only loss. A comparison with the Belle II TagV currently used for the tag vertex resolution was conducted and showed that GraFEI could be a competitive model, with an improvement of approximately 13% on the $B \to \nu\nu$ dataset. However, the GraFEI still needs training on more data because it is not reliable on all types on datasets as TagV. Also, the model shows signs of bias, with residuals exhibiting a correlation with the true vertex position. The predictions had to be calibrated to get the final resolution, which suggests that there is still room for improvement in the model's implementation.
When trained with the Lowest Common Ancestor (LCA) loss, no further resolution improvement is observed. The resolution even degrades if the relative weight of the LCA loss becomes large with respect to the vertex-based loss. A potential explanation for this observation is that the GraFEI does not need to recognize the full B meson decay for locating its vertex.
The GraFEI has also been tested as a flavor tagging tool, but did not prove to be efficient as the predicted flavors were close to random. Maybe a different kind of implementation of this flavor feature in the model could be explored.

An additional path for the continuation of this work is to consider the planned upgrade of the Belle II vertex detector, which is expected to improve the resolution on track extrapolations towards the collision vertex. This tool could also serve as a mean to evaluate and compare detector geometries. By comparing the current Belle II detector configuration against an idealized setup with perfect resolution, it may be possible to estimate the theoretical limits of vertex reconstruction and guide future detector design. Furthermore, these developments pave the way towards more ambitious goals, such as the development of a full Tree Fitter, capable of reconstructing the entire decay tree, not just the position of the B meson vertex.
Despite the technical challenges encountered throughout the internship, this work lays a solid foundation for further work on vertex reconstruction using graph neural networks.

Abstract

Recherche de vertex à l'aide d'un réseau de neurones à graphes

Ce travail présente les résultats d'un stage de quatre mois au sein du groupe Belle II de l'IPHC de Strasbourg, portant sur l'identification de vertex de désintégration des mésons B à l'aide de réseau de neurones. Ce projet s'inscrit dans l'effort de la recherche de nouvelle physique, motivation principale de l'expérience Belle II au Japon. Le modèle developpé se base sur le réseau de neurone à graphe GraFEI développé au sein de l'équipe qui permet de reconstruire l'arbre de désintégration suite à la collision e^+e^-, à partir de la seule information des particules finales identifiées dans le détecteur. Ce stage vise à exploiter ce modèle pour reconstruire la position de désintégration du méson B du côté tag (*tag side*), en complément de la reconstruction complète de l'arbre de désintégration. Les résultats présentent une résolution compétitive avec une potentielle amélioration de 13% en résolution sur la prédiction de la position, par rapport à l'outil TagV utilisé dans Belle II, en atteignant une résolution $\sigma = 51.76 \pm 0.49\ \mu m$ sur la position du $B_{tag,z}$ vertex dans des désintégrations $B \to \nu\nu$. Des défis subsistent quant à la généralisation aux modes de désintégration comportant un côté signal (*signal side*) ainsi qu'à l'extraction de l'information de saveur du méson B, ce qui ouvre de nouvelles perspectives d'utilisation du GraFEI dans des mesures de nouvelle physique de haute précision.

Vertex fitting using a graph neural network

This work presents the results of a six-month internship in the Belle II group at the IPHC in Strasbourg, focused on identifying the decay vertices of B mesons using neural networks. This project is part of the broader effort to search for new physics, which is the primary motivation behind the Belle II experiment in Japan. The developed model is based on the graph neural network GraFEI, developed by the team, capable of reconstructing the full decay tree resulting from e^+e^- collisions, using only the information from the final-state particles identified in the detector. This internship aimed to extend the model to also reconstruct the decay position of the B meson on the tag side, in addition to the full decay tree. Results show competitive resolution with a potential 13% improvement in vertex resolution compared to the TagV tool currently used in Belle II, reaching a resolution $\sigma = 51.76 \pm 0.49\ \mu m$ on the position of the $B_{tag,z}$ vertex in $B \to \nu\nu$ datasets. Challenges remain in generalization to decay modes with an existing signal side and in the reliable extraction of flavor information, pointing to future directions for enhancing GraFEI's applicability in precise new physics measurements.

Bibliography

[1] Merna Abumusabh et al. *Graph-based Full Event Interpretation: a graph neural network for event reconstruction in Belle II*. 2025. arXiv: 2503.09401 [hep-ex]. URL: https://arxiv.org/abs/2503.09401.

[2] A. J. Bevan et al. "The Physics of the B Factories". In: *European Physical Journal C* 74 (2014). SLAC-PUB-15968, KEK Preprint 2014-3, p. 3026.

[3] Dietrich Bödeker and Wilfried Buchmüller. "Baryogenesis from the weak scale to the grand unification scale". In: *Reviews of Modern Physics* 93.3 (Aug. 2021). ISSN: 1539-0756. DOI: 10.1103/revmodphys.93.035004. URL: http://dx.doi.org/10.1103/RevModPhys.93.035004.

[4] A. Burkov. *The Hundred-page Machine Learning Book*. Andriy Burkov, 2019. ISBN: 9781999579500. URL: http://ema.cri-info.cm/wp-content/uploads/2019/07/2019BurkovTheHundred-pageMachineLearning.pdf.

[5] A. Ceccucci, Z. Ligeti, and Y. Sakai. *CKM Quark-Mixing Matrix*. Particle Data Group. Mar. 2020. URL: https://pdg.lbl.gov/2020/reviews/rpp2020-rev-ckm-matrix.pdf.

[6] J. H. Christenson et al. "Evidence for the 2π Decay of the K_2^0 Meson". In: *Phys. Rev. Lett.* 13 (4 July 1964), pp. 138–140. DOI: 10.1103/PhysRevLett.13.138. URL: https://link.aps.org/doi/10.1103/PhysRevLett.13.138.

[7] Cush. *Standard Model of Elementary Particles*. Version 2025-03-24. Wikimedia Commons. Sept. 17, 2019. URL: https://commons.wikimedia.org/wiki/File:Standard_Model_of_Elementary_Particles.svg.

[8] Wei-Shu Hou. *Source of CP Violation for Baryon Asymmetry of the Universe*. 2008. arXiv: 0803.1234 [hep-ph]. URL: https://arxiv.org/abs/0803.1234.

[9] *How to Process Multiple Losses in PyTorch*. Last updated: 23 Jul 2025. GeeksforGeeks. July 23, 2025. URL: https://www.geeksforgeeks.org/deep-learning/how-to-process-multiple-losses-in-pytorch/.

[10] IBM. *Neural Networks*. 2023. URL: https://www.ibm.com/topics/neural-networks.

[11] Institut Pluridisciplinaire Hubert Curien (IPHC). *Research at DRS*. 2025. URL: https://iphc.cnrs.fr/la-recherche/#drs.

[12] James Kahn et al. "Learning tree structures from leaves for particle decay reconstruction". In: *Machine Learning: Science and Technology* 3.3 (Sept. 2022), p. 035012. ISSN: 2632-2153. DOI: 10.1088/2632-2153/ac8de0. URL: http://dx.doi.org/10.1088/2632-2153/ac8de0.

[13] T. Kuhr et al. “The Belle II Core Software”. In: *Computing and Software for Big Science* 3.1 (2018), p. 1. DOI: 10.1007/s41781-018-0017-9. URL: https://doi.org/10.1007/s41781-018-0017-9.

[14] K.P. Murphy. *Machine Learning: A Probabilistic Perspective*. Adaptive Computation and Machine Learning series. see p.339, Section 11.2.1: Mixture of Gaussians. MIT Press, 2012. ISBN: 9780262304320. URL: https://books.google.fr/books?id=RC43AgAAQBAJ.

[15] PyTorch Contributors. *torch.nn.CrossEntropyLoss*. 2024. URL: https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html.

[16] PyTorch Contributors. *ttorch.nn.MSELoss*. 2024. URL: https://pytorch.org/docs/stable/generated/torch.nn.MSELoss.html.

[17] Lea Reuter. “Full Event Interpretation using Graph Neural Networks”. MA thesis. Karlsruhe Institute of Technology (KIT), 2022.

[18] Brian D. Ripley. *Pattern Recognition and Neural Networks*. see p.354. Cambridge University Press, 1996.

[19] ROOT Team. *ROOT Users Guide: Fitting Histograms*. Section 7.2.2.3: Creating a TF1 with a User Function. CERN. 2024. URL: https://root.cern.ch/root/htmldoc/guides/users-guide/FittingHistograms.html.

[20] Corentin Santos. “Optimization of a deep graph neural network to reconstruct generic B decays at Belle II”. M2 Internship Thesis. University of Strasbourg, 2023. URL: https://docs.belle2.org/pub_data/documents/3906/.

[21] *SuperKEKB/Belle II Complete 2024 Operations*. Institute of Particle and Nuclear Studies (KEK). Jan. 10, 2025. URL: https://www2.kek.jp/ipns/en/news/7015/.

[22] *vertex.TagV basf2 analysis documentation*. Belle II Software Group. URL: https://software.belle2.org/development/sphinx/analysis/doc/VertexWrappers.html#vertex.TagV.