Karlsruhe Institute of Technology

ETP-BACHELOR-KA/2024-13

# Automating $\mathrm{K}^0_\mathrm{S}$ Performance Monitoring at Belle II

Simon Weber

Bachelor Thesis

27th September 2024

Institute of Experimental Particle Physics (ETP)

| | |
|---|---|
| Reviewer: | Prof. Dr. Torben Ferber |
| Second reviewer: | Dr. Giacomo De Pietro |
| Advisor: | Patrick Ecker |

Editing time: 2nd May 2024 – 27th September 2024

Karlsruhe Institute of Technology

ETP-BACHELOR-KA/2024-13

# Automatisierung des $\mathrm{K}^0_\mathrm{S}$ Performance Monitorings bei Belle II

Simon Weber

Bachelorarbeit

27. September 2024

Institut für Experimentelle Teilchenphysik (ETP)

| | |
|---|---|
| Referent: | Prof. Dr. Torben Ferber |
| Korreferent: | Dr. Giacomo De Pietro |
| Betreuer: | Patrick Ecker |

Bearbeitungszeit: 2. Mai 2024 – 27. September 2024

Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

**Karlsruhe, 27. September 2024**

. . . . . .

(**Simon Weber**)

# Disclaimer

Especially in particle physics large groups work together on an experiment. Therefore, in the following list I want to mention the work of Belle II members I used in this thesis.

- This thesis uses a dataset produced and processed by the Belle II Collaboration.
- The steering file which applies the selection criteria mentioned in Section 3.3 and reconstructs the $K_S^0$ from the dataset was already implemented in Patrick Ecker's Validation Interface for the Belle II Experiment (VIBE) [1].
- The Belle II Analysis Software Framework (basf2) [2,3] was used to create the n-tuples.
- The rounding tool from Patrick Ecker was used to display the values and their uncertainties correctly [4].
- I created the website together with Arvid Kammann.

This thesis incorporates the use of Artificial Intelligence (AI) tools for assistance in grammatical and stylistic improvement of text, and program code creation.

Grammarly[1] is utilized throughout the thesis for spell and grammar checks, as well as for paraphrasing individual, selected sentences to improve clarity and precision in academic writing. I have approved all suggested changes.

Github Copilot[2] is used for repetitive program code creation, for the creation of the algorithm for FWHM calculation of the DG function, as well as for the creation of the website. I have approved and tested all suggestions to provide robust and reliable results.

[1] Grammarly: An AI writing assistant. See https://app.grammarly.com/ (Last accessed: 19. September 2024)

[2] Github Copilot: An AI coding assistant. See https://github.com/features/copilot (Last accessed: 19. September 2024)

# Contents

# 1. Introduction

The Standard Model (SM) of particle physics is one of the most well-tested theories of nature. Still, it does not provide answers to many fundamental questions, for example dark matter. Belle II is an experiment trying to solve these questions. Operating at the luminosity frontier, Belle II tries to find signatures of new particles by looking for discrepancies with the SM predictions [5].

In particle physics not only the data collected with the detector but also simulated particle collisions play an important role, as they are needed for comparison with data to verify that an observation is not simply an artifact due to the complexity of the experiment. The simulated data is also called Monte Carlo (MC) which uses repeated sampling of random numbers. The MC method, which is of course only an approximation of the real processes, starts with the event generation, simulating the physical processes, and continues with the simulation of the particle-detector interaction by using a simplified detector architecture. In the next step the results of the MC simulations are compared with available experimental data. This comparison allows us to recognize differences in the distributions and thus to be able to improve the MC parameters [6].

To improve the MC parameters it is crucial to fully understand the physics processes at Belle II. The performance monitoring is an approach to better understand these physics processes. Especially the reconstruction efficiency for $K^0_S$ and $\Lambda$ show higher discrepancies between data and MC than for other particles due to their displaced vertex. However, the statistics of the $\Lambda$ are much lower than that of the $K^0_S$ due to their smaller branching fraction [7, 8].

Therefore, this thesis is concerned with the automation of the $K^0_S$ performance monitoring at Belle II. The benefit of automating the performance monitoring lies in the fact, that a lot of time can be saved as it does not rely on the availability of person power. In addition, by displaying the results on a website it will then be easier to monitor the results because of the fast access to it.

Chapter 2 first gives an overview of the Belle II experiment and the software framework used by the Belle II Collaboration. Chapter 3 explains the steps taken to estimate the mean and width of the $K^0_S$ mass peak and also gives an explanation on how the uncertainties on these parameters are calculated. In this thesis, the fits used to describe the $K^0_S$ are performed for different conditions during data taking and in bins of different observables. The fits and the parameters obtained are discussed in Chapter 4. A conclusion and outlook is given in Chapter 5.

# 2. The Belle II Experiment

## 2.1. The SuperKEKB Collider

SuperKEKB is a 3 km long particle accelerator located at the High Energy Accelerator Research Organisation (KEK) in Tsukuba, Japan. The layout of the accelerator is shown in Fig. 2.1. In a Linear accelerator (LINAC), first electrons and positrons get accelerated up to around 7 Gev and 4 GeV respectively. The electrons then enter the High Energy Ring (HER) and the positrons enter the Low Energy Ring (LER), moving in the opposite direction as the electrons. The collision happens at the Interaction Region (IR) inside the Belle II detector, where the two beams collide almost frontally. Therefore the resulting center-of-mass energy is in the range of around 9.4 GeV to 11 GeV. Mostly the center-of-mass energy is adjusted to be at 10.58 GeV, corresponding to the mass of the $\Upsilon\,(4S)$ resonance. The $\Upsilon\,(4S)$ resonance almost always decays into a pair of B mesons. For that reason, SuperKEKB is called a B-Factory.

The asymmetry in the beam energy leads to boosted particles. This results in an increase of the distance between the decay vertices of the two B-mesons, making it possible to measure this distance more accurately. With that it was possible to prove CP-violation at the previous generation of B-factories [9].

SuperKEKB has a designed instantaneous luminosity of $6 \times 10^{35}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$. This is achieved by having a high current of electrons and positrons in the beam tubes and a high beam focus at the IR. High luminosity plays a crucial role because it enables measurements with high statistics. The given information in this section is taken from [10–12].

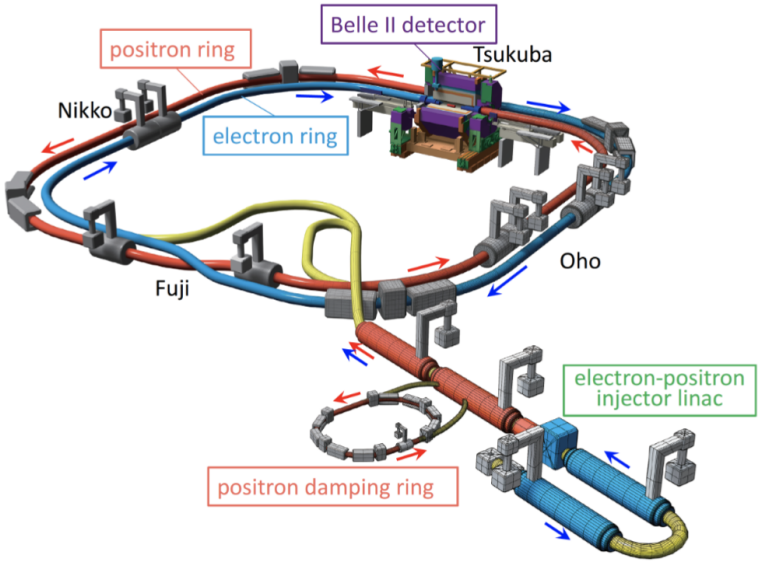



Figure 2.1.: Schematics of the SuperKEKB Collider. Image is taken from [13].

## 2.2. The Belle II detector

The Belle II detector is placed around the IR. Its goal is to identify the particle type and to measure the energy and momentum of as many of the particles produced in the collision with the highest possible precision. To accomplish this task the Belle II detector consists of multiple sub-systems arranged cylindrically around the IR.

From the inner to the outer these are the Pixel Detector (PXD), Silicon Vertex Detector (SVD), and Central Drift Chamber (CDC) to reconstruct decay vertices and the trajectory of charged particles, the Time-of-Propagation counter (TOP) and Aerogel Ring-imaging Cherenkov (ARICH) for the identification of particles, the Electromagnetic Calorimeter (ECL) for the measurement of photons, and $K_L$ and Muon Detector (KLM) for the identification of $K_L$ and muons.

The asymmetric beam energies also affect the design of the detector. Therefore the direction of the electron beam determines the forward direction. A cross-section of the detector is shown in Fig. 2.2. The given information in this section is taken from [5, 11, 14, 15].

Due to the high luminosity of SuperKEKB, the detectors close to the beam pipe face high amounts of hit rates, caused by beam-related background. Therefore the innermost detector, the PXD consists of two layers of pixel sensors, which have a much larger number of channels and therefore have a much smaller occupancy than strip sensors.

The SVD contains four layers of dual-sided silicon strip sensors, as the position is far enough away from the beam pipe. The advantage over the pixel sensors is the possibility of avoiding the huge channel count of pixels.

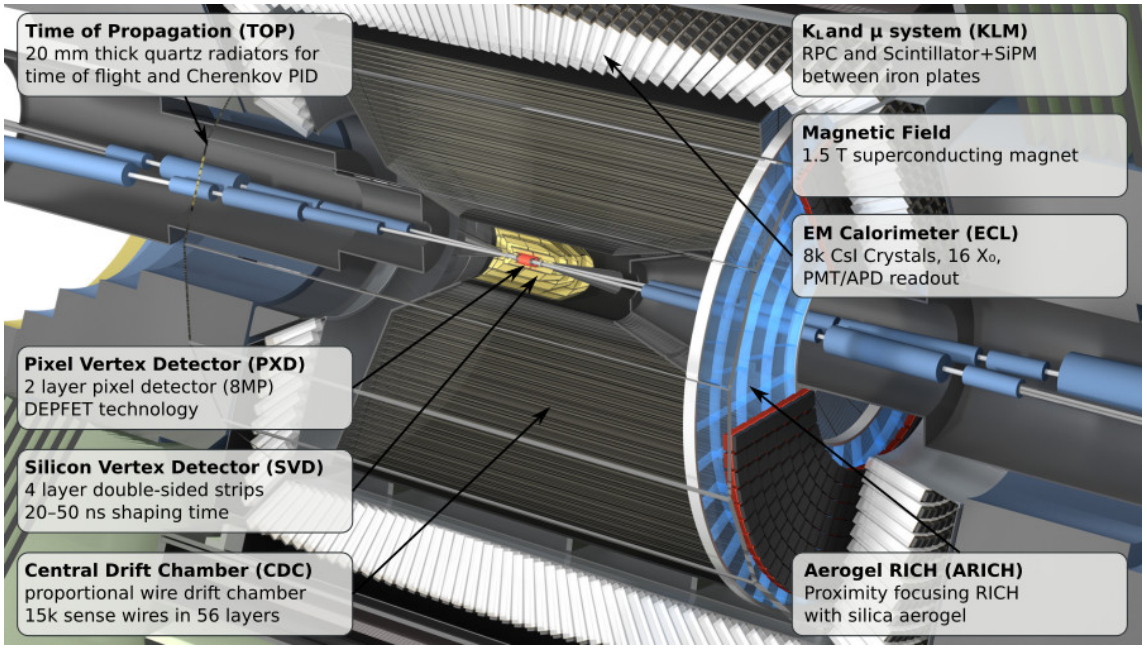



Figure 2.2.: Cross section of the Belle II detector. Image is taken from [11].

The CDC is the main tracking system for Belle II and is placed after the two Vertex detectors. The CDC consists of a total of 14336 sense wires suspended in a 50:50 gas mixture of $He - C_2H_6$. When a charged particle passes through the gas it causes ionization charges, which then drift to the wires, causing a signal. With the obtained signals it is possible to measure the trajectories of the charged particles, their momentum, and their energy loss caused by the ionization.

Almost all subdetectors contribute to particle identification but the system specifically tasked with particle identification consists of the TOP, located in the barrel region of the detector surrounding the outer wall of the CDC, and the ARICH, located in the forward end-cap of the detector. They both utilize the Cherenkov effect. When a charged particle enters the detector system it emits Cherenkov Light with a certain angle, that depends on the particle's velocity. By combining the velocity measurement with the momentum measurement in the CDC it is possible to determine the particle mass, which leads to the particle species.

The ECL is placed after the TOP and consists of a barrel as well as forward and backward end-caps. The main task of the ECL is the detection of photons and the determination of their energy and angular coordinates. It is noted that the KLM can as well detect photons, however, with a poorer performance. The ECL consists of 8736 CsI(Tl) crystals creating scintillation light when a particle flies into them. The amplitude of the light is proportional to the energy deposited in the crystal. By combining the information of the deposited energy with the length of the shower it is possible to calculate the energy the particle had when hitting the ECL.

The KLM is the outermost system of the Belle II detector containing a barrel region as well as forward and backward end-caps. The purpose of the KLM is to identify muons and

$K_L$. The muons will leave a track by passing through and the $K_L$ will leave a concentrated cluster of hits.

## 2.3. Software Framework

The Belle II Analysis Software Framework (basf2) [2,3] is used for different tasks, such as the generation of simulated data, track reconstruction, vertex fitting, and more. By writing a so-called steering file it is possible to set up some of these analysis or data-processing tasks. The software framework basf2 contains multiple modules, mostly written in C++, and each of them is designed to execute a specific task. These modules are often arranged in a path in linear order and when the steering file gets executed, the modules are executed in the same order one at a time. The linear arrangement of modules in a path can be broken, as modules can also have conditions attached to them to steer the processing flow depending on the outcome of the calculation in each module. Each module has access to the so-called Data Store, to extract the needed data and save the results after processing. Through the Data Store, the modules have access to the results of modules processed before. The module chain and its interaction with the Data Store are shown in Fig. 2.3. The image and information in this section is taken from [16].

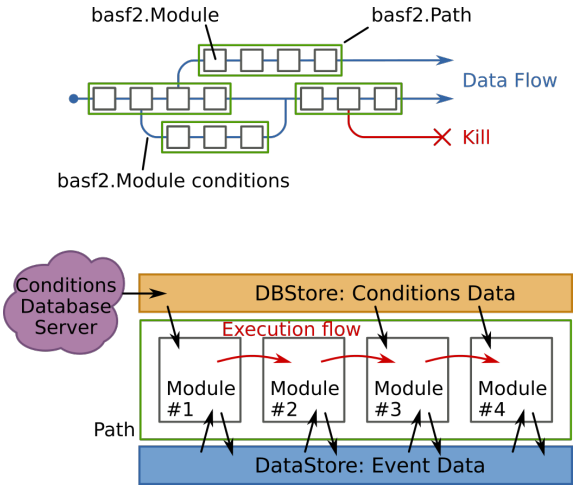



Figure 2.3.: The module chain and its interaction with the Data Store. Image is taken from [16].

# 3. Measurement

In this chapter, first some information on the used datasets, the $K_S^0$ reconstruction procedure and the selection criteria, used to target the decay chain of interest is given. From Section 3.4 ongoing I explain the steps taken to measure the $K_S^0$ performance. My program code for this measurement can be accessed under [17].

## 3.1. Dataset

For this thesis, the full dataset collected at the $\Upsilon(4S)$ resonance, corresponding to an integrated luminosity of $\int \mathcal{L}_{Data}\, dt = 364.093 \pm 0.021\,\text{fb}^{-1}$ [18], and the corresponding MC simulated dataset, equivalent to $\mathcal{L}_{MC} = 4 \cdot \mathcal{L}_{Data}$ is used. For the MC dataset, the $e^+e^- \to B\bar{B}$ part got generated with EvtGen [19] and Pythia8 [20], the $e^+e^- \to q\bar{q}$ part got generated with KKMC [21] and Pythia8. This MC dataset includes roughly 2/3 of continuum background (u, d, s, c) events and 1/3 of $\Upsilon(4S)$ resonance events.

## 3.2. Reconstruction of the $K_S^0$

The $K_S^0$ is a weak eigenstate of the $K^0$, consisting of down and strange quark, and has a Mass of $497.611 \pm 0.013\,\text{MeV}$. With a $c\tau$ of around $26.843 \pm 0.012\,\text{mm}$ the $K_S^0$ is considered a long-lived particle. It commonly decays into pions, either $\pi^+$ and $\pi^-$ or two $\pi^0$ [22, 23].

As the charged decay products of the $K_S^0$ are mostly created outside the beam pipe, it is much more difficult to find the tracks of these pions and reconstruct the $K_S^0$, as it is not possible to do this with the regular algorithm which requires the particle to come from the IR. Therefore, a so-called $V^0$ reconstruction algorithm in basf2 reconstructs the $K_S^0$ by searching the two oppositely charged particles decaying from the $K_S^0$ and pairing them. The $V^0$ reconstruction algorithm pairs all oppositely charged particle tracks and extrapolates them to the point where they are close enough to each other. If they are not compatible with originating from a common vertex they get rejected and the left pairs of oppositely charged particle tracks get processed by the vertex reconstruction package using the TreeFitter algorithm [24] for decay chain fitting [5].

## 3.3. Selection criteria

Signal events are selected by targeting the following decay chain: $D^{*+} \to D^0(\to K_S^0\pi^+\pi^-)\pi_S^+$ with $\pi_S^+$ being a low momentum pion. However, the signal part contains in general also $K_S^0$

originating from other decay chains or continuum background processes. The selections are applied in a way, so that most of the background is rejected, preferably without rejecting any signal events. The pions selections play an important role in correctly reconstructing the $D^0$ and $D^{*+}$.

Table 3.1 shows the selection criteria applied to the datasets. Here, *dr* is the transverse distance between the Point of closest approach (POCA) and Interaction Point (IP), *dz* the z distance between the POCA and IP and $significanceOfDistance$ the significance of the distance from the vertex to the IP. The significance of the distance is calculated by taking the displacement of the vertex over the uncertainty of its displacement. In turn it is important to take a "large enough" $significanceOfDistance$ to make sure that the observed particle actually has a displaced vertex.

Table 3.1.: Selection criteria applied to the datasets.

| particle | selection criteria | unit |
|---|---|---|
| $\pi^{\pm}$ | $\lvert dr\rvert < 2$ | cm |
| | $\lvert dz\rvert < 4$ | cm |
| | $p_{\text{transverse}} > 0.1$ | GeV/$c$ |
| $\pi_S^+$ | $\lvert dr\rvert < 2$ | cm |
| | $\lvert dz\rvert < 4$ | cm |
| $K_S^0$ | $0.480 < M < 0.516$ | GeV/$c^2$ |
| | $significanceOfDistance > 3$ | - |
| $D^{*+}$ | $0.144 < M_{D^{*+}} - M_{D^0} < 0.147$ | GeV/$c^2$ |
| | $p_{\text{CMSFrame}} > 1.6$ | GeV/$c$ |
| $D^0$ | $1.85 < M < 1.88$ | GeV/$c^2$ |

To allow the comparison of data and MC $K_S^0$ mass distributions, MC is rescaled to the same luminosity of data, displayed in Fig. 3.1. In contrary to data, in MC we exactly know the particles created in a collision. Therefore it is possible to split MC in signal, which means a $K_S^0$ is involved in the decay chain of interest, and background where we have no $K_S^0$. From the MC dataset, the signal and background portion can be described. The signal part shows essentially a Gaussian distribution with a peak at around $0.4975\,\text{GeV}/c^2$. However, the tail region differs from that of a Gaussian distribution due to detector effects. The background part is almost constant since it consists largely of combinatorial and beam-induced background which consists of particles whose masses are evenly distributed. The plot also shows data, which follows the same distribution as the sum of MC signal and background. Still, it can be seen, that data is higher in the tails but has a lower peak. This leads to the assumption that the signal portion in data is lower than in MC.

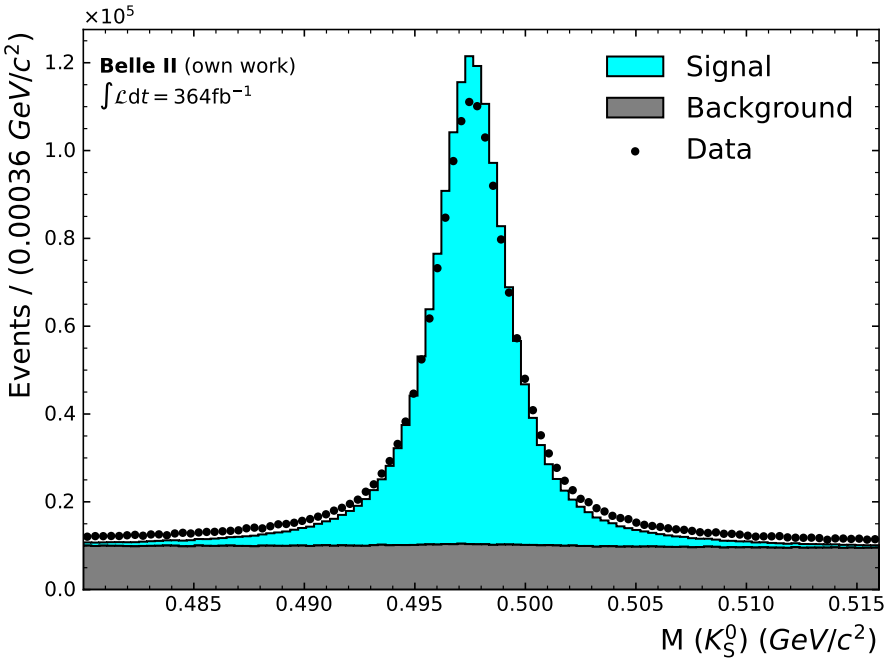



Figure 3.1.: $K^0_S$ mass distribution for data and MC, where MC is rescaled to the same luminosity of data. For MC the shape of the signal and background part can be seen. The signal part shows a Gaussian distribution with long tails. The background is almost constant. The data distribution is slightly higher in the tails but has a lower peak.

## 3.4. Fitting Method

The mean and width of the signal peak are estimated by fitting the reconstructed mass of the $K^0_S$. In this thesis, the fits are performed with the zfit python package [25]. For that reason, the following part gives some information about the general fitting procedure and which functions I chose to model the data.
The maximum-likelihood method tries to estimate a set of parameters of an assumed probability distribution for some data points. This is achieved by maximizing the likelihood with which the probability distribution would result in the observed data. However, in most cases, parameter estimation is performed using the negative logarithm of the likelihood (NLL), thus the set of best parameters is located at its global minimum. This transformation is allowed since the logarithm is a monotonic function and as a result, the global minimum of the NLL occurs at the same place as the maximum of the likelihood [26].
In the next step some algorithm must find this minimum, this is done with MIGRAD, a minimization algorithm from the MINUIT [27] software library. In order for the minimization algorithm to find the global minimum it is important, especially for functions with a lot of parameters, to define some good starting values for the fit parameters.

Based on the procedure of the maximum-likelihood method, we first need to find a probability distribution that fits the data. As we know, the data consists of a signal and background part, allowing us to define the fit function as a sum of those two functions. The signal

part consists of the $K_S^0$ mass peak and this leads to the assumption, that we should use a Breit-Wigner function as probability distribution. But as detector effects play an important role, the signal probability distribution changes to a function with tails. For this reason I use a so-called Double sided Crystal Ball (DSCB) function, consisting of a Gaussian core portion and two power-law tails. The function is defined as follows:

$$\text{DSCB} = \begin{cases} A_\text{L} \cdot \left(B_\text{L} - \frac{x-\mu}{\sigma}\right)^{-n_\text{L}}, & \text{for } \frac{x-\mu}{\sigma} < -\alpha_\text{L} \\ \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), & \text{for } -\alpha_\text{L} \leq \frac{x-\mu}{\sigma} \leq \alpha_\text{R} \\ A_\text{R} \cdot \left(B_\text{R} + \frac{x-\mu}{\sigma}\right)^{-n_\text{R}}, & \text{for } \frac{x-\mu}{\sigma} > \alpha_\text{R} \end{cases} \tag{3.1}$$

with

$$A_\text{L/R} = \left(\frac{n_\text{L/R}}{\alpha_\text{L/R}}\right)^{n_\text{L/R}} \cdot \exp\left(-\frac{\left|\alpha_\text{L/R}\right|^2}{2}\right) \tag{3.2}$$

$$B_\text{L/R} = \frac{n_\text{L/R}}{\left|\alpha_\text{L/R}\right|} - \alpha_\text{L/R}\,. \tag{3.3}$$

By taking a look at the MC signal and background parts in Fig. 3.1 it is possible to see, that a linear function as defined in Eq. (3.4) will probably be suited best to model the background. The slope of the linear function is thereby given by $m$, and the constant value is given by the number of background events $(1 - frac_\text{sig}) \cdot N$ over the number of bins.

$$\text{f}\,(x) = m \cdot x + \left(1 - frac_\text{sig}\right) \cdot \frac{N}{100} \tag{3.4}$$

The parameter $frac_\text{sig}$ specifies the signal fraction, therefore $(1 - frac_\text{sig})$ returns the background fraction, and N defines the total number of events in the dataset.
The two functions in Eqs. (3.1) and (3.4) are then summed up to the final fit-function in Eq. (3.5) by accounting the signal and background fraction.

$$\text{fit-function} = frac_\text{sig} \cdot \text{DSCB} + \left(1 - frac_\text{sig}\right) \cdot \text{f}\,(x) \tag{3.5}$$

Chosen starting points and limits of the fit parameters are listed in Table 3.2. The values for another model, which is discussed in the next section are also shown.
The parameters of the DSCB function are highly correlated. In order to make the fit more stable and to guarantee a better comparison between different fits, the parameters $n_\text{L}$, $n_\text{R}$, $\alpha_\text{L}$ and $\alpha_\text{R}$ are fixed to the initial value, without further optimization.

Table 3.2.: Starting values and limits for all parameters used by the fitting models are shown. Parameters $\alpha_{\mathrm{L/R}}$ and $n_{\mathrm{L/R}}$ are special in the way, that they are fixed to the specified initial value, as this makes the fits more stable and easier to compare. The two parameters listed at the bottom of the table are used by a fit function mentioned in Section 3.5.2, the parameters specific for the DSCB function are listed in the middle of the table, and the parameters used by both models are listed in the upper part of the table.

| Model | Parameter | initial value | limits |
|---|---|---|---|
| All | $\mu$ | 0.497 | (0.485, 0.51) |
| | $\sigma$ | 0.0013 | $\left(1 \times 10^{-5}, 3 \times 10^{-2}\right)$ |
| | $m$ | 0.1 | no limits |
| | $frac_{\mathrm{sig}}$ | 0.25 | (0, 1) |
| | $N$ | number of events in dataset | no limits |
| DSCB | $\alpha_{\mathrm{L}}$ | 1.2 | - |
| | $\alpha_{\mathrm{R}}$ | 1.3 | - |
| | $n_{\mathrm{L}}$ | 2.7 | - |
| | $n_{\mathrm{R}}$ | 2.4 | - |
| Double Gauss (DG) | $f$ | 0.7 | (0, 1) |
| | $factor$ | 2 | (1, 15) |

## 3.5. Error Calculation

The estimation of statistical and systematic uncertainties is an important part of this thesis, as this provides information in which range the parameters lie. First, the statistical $\sigma_{\mathrm{stat}}$ and systematic $\sigma_{\mathrm{syst1}}$, $\sigma_{\mathrm{syst2}}$ uncertainties on the fit parameters are calculated one by one. The total uncertainty is then computed by adding them in quadrature as shown in Eq. (3.6).

$$\sigma_{\mathrm{total}} = \sqrt{\sigma_{\mathrm{stat}}^2 + \sigma_{\mathrm{syst1}}^2 + \sigma_{\mathrm{syst2}}^2} \tag{3.6}$$

### 3.5.1. Statistical Uncertainties

The statistical uncertainties are calculated with the hesse function from zfit after making the fit. This function calculates the statistical uncertainties of all parameters used in the fit function, by estimating the covariance matrix using the inverse of the Hessian matrix.

Due to the large number of events in the dataset, the statistical uncertainties should only have a small impact on the total error, unless the fit is performed on a small subset of the dataset.

### 3.5.2. Systematic Uncertainties

Systematic uncertainties can have their origin in detector effects or theoretical assumptions. In this thesis however, the dominating part of systematic uncertainties is probably arising due to theoretical assumptions. Therefore two possible error sources based on theoretical

assumptions are taken into account to estimate the systematic uncertainties. The idea of considering these systematic error sources is coming from [7].

The first point is the choice of the function used to fit the signal part of the data because it is known, that the DSCB function is not modeling the data correctly. I want to estimate the systematic errors arising from this, by using a slightly different function, the Double Gauss (DG) defined as the sum of two Gaussians. The DG is in general performing worse than the DSCB, especially in the tails, as it can be seen in Fig. 3.2. From the pull plot on the bottom of the plots it can be seen, that both functions do not model the data correctly as a clear W-shape appears around the peak. The DG function is defined as follows:

$$\mathrm{DG} = f \cdot \mathrm{Gauss}\,(\mu,\, \sigma) + (1 - f) \cdot \mathrm{Gauss}\,(\mu,\, factor \cdot \sigma) \tag{3.7}$$

with the parameters $f$ and $(1 - f)$ being the fractions of the first and the second Gauss function, respectively, and the parameter $factor$ modifying the width of the second Gauss function. These parameters are listed in Table 3.2.

Similar to Eq. (3.5) the DG function then gets summed up with the linear function from Eq. (3.4).

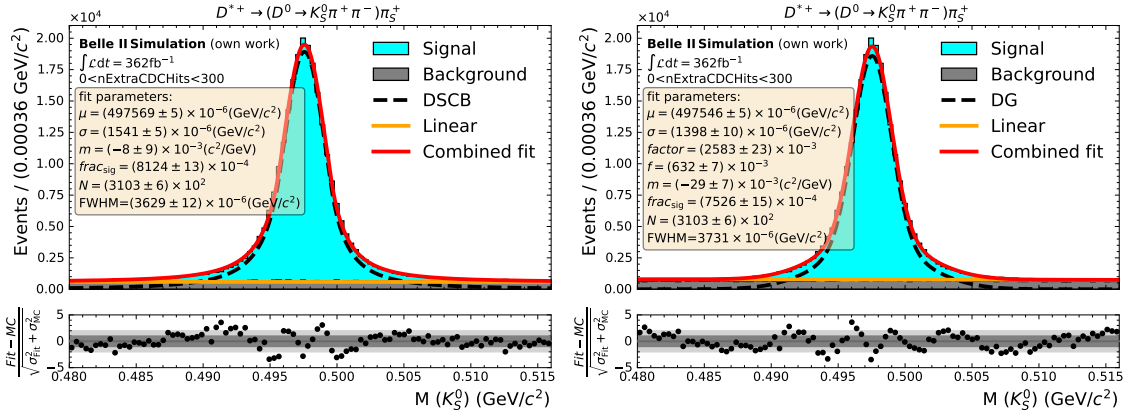
The final systematic uncertainty related to the PDF choice is then calculated by taking the absolute difference between the obtained parameter value $x_{\mathrm{original}}$ of the original model with DSCB function and the obtained parameter value $x_{\mathrm{DG}}$ of the model with the DG function. In the following chapter, the parameter $x$ will then either be the mean $\mu$ or the Full Width at Half Maximum (FWHM) of the signal function.

$$\sigma_{\mathrm{syst2}} = |x_{\mathrm{original}} - x_{\mathrm{DG}}| \tag{3.8}$$

A further systematic error source taken into account is the change in the observed $\mathrm{K_S^0}$ mass range. The resulting difference in the fit is in general smaller than the difference caused by changing the PDF and can be seen in Fig. 3.3.

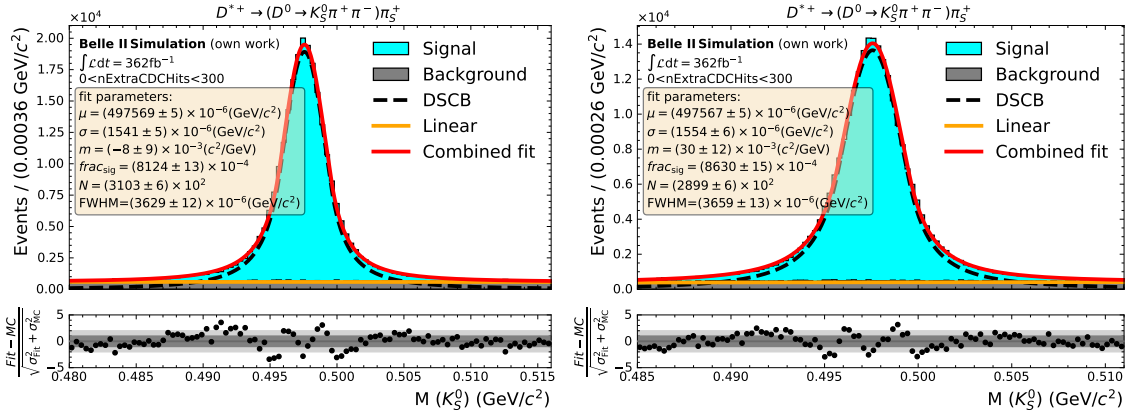
The final systematic uncertainty related to the $\mathrm{K_S^0}$ mass range choice is then calculated by taking the absolute difference between the obtained parameter value $x_{\mathrm{original}}$ of the original model with the wide $\mathrm{K_S^0}$ mass range and the obtained parameter value $x_{\mathrm{narrow}}$ of the model with the narrow $\mathrm{K_S^0}$ mass range.

$$\sigma_{\mathrm{syst1}} = |x_{\mathrm{original}} - x_{\mathrm{narrow}}| \tag{3.9}$$




(a) DSCB (b) DG

Figure 3.2.: Comparison between the DSCB and DG fit: Each of the two plots show the $K^0_S$ mass distribution of MC overlaid with the performed fit (red), the linear background part (orange) and the signal part with DSCB or DG function (dashed black line). The resulting fit parameters are also listed. Both fit results are pretty similar besides the worse description of the tails by the DG fit as can be seen from the pull plot on the bottom. The shown $K^0_S$ mass distribution does only include the MC events that lie in the given ExtraCDCHits bin. MC is rescaled to the same luminosity of data.




(a) wide (b) narrow

Figure 3.3.: Comparison between the fits on a wide and narrow $K^0_S$ mass range: Each of the two plots show the $K^0_S$ mass distribution of MC overlaid with the performed fit (red), the linear background part (orange) and the signal part with DSCB function (dashed black line). The resulting fit parameters are also listed. The difference in the fit parameters $\mu$ and FWHM caused by choosing a different $K^0_S$ mass range is almost negligible. The shown $K^0_S$ mass distribution does only include the MC events that lie in the given ExtraCDCHits bin. MC is rescaled to the same luminosity of data.

## 3.6. Full Width at Half Maximum Calculation

As the DSCB function has one parameter that describes the width of the peak but the DG function contains two parameters that influence the width it is difficult to fairly compare the widths of the two functions. To compare the width of peaks from different models the FWHM is suited best, but the way of calculating the FWHM differs for the two functions and is therefore explained in the following.

The FWHM of the DSCB function is calculated by taking the formula for FWHM calculation of a Gaussian, as the core portion of the DSCB consists of a Gaussian. The formula is given in Eq. (3.10). The statistical errors are calculated with Gaussian error propagation.

$$\mathrm{FWHM}_{\mathrm{DSCB}} = 2\sigma\sqrt{2\ln 2} \tag{3.10}$$

For the DG function the estimation of the FWHM is done by an algorithm which searches the two points at the half maximum, by interpolation of the two closest known points on each side. As we only need the FWHM values from the DG function to calculate systematic errors the statistical errors are not calculated for this function.

# 4. Data and MC comparison

In this chapter, I want to compare the resulting fit parameters from fits performed on data and MC by run conditions and by background observables. With this, it is possible to get information on how data differs from MC, how they change with different run conditions, and how they change as a function of background observables. The comparison is based on the two most important fit parameters, the peak's mean $\mu$ and the FWHM.

## 4.1. By run conditions

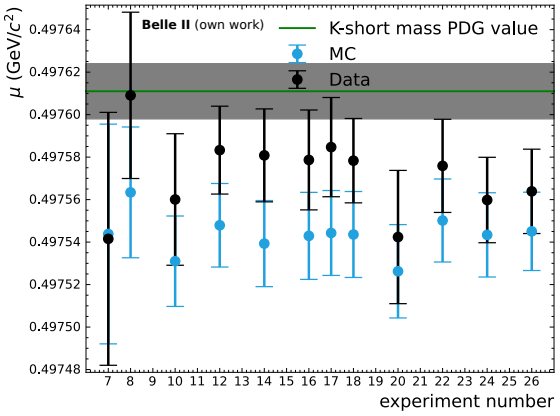
In this section, we take a look on fit parameters, where the fits are independently performed for each run condition, in the following also called experiments. The resulting fits are shown in Appendix A.2.

First, I want to mention, that the measured mass of the $K^0_S$ given by $\mu$ in Fig. 4.1a is in general slightly lower than the PDG $K^0_S$ mass value from [23]. The fact that all values are lower than the actual $K^0_S$ mass value, indicates a systematic shift. This systematic shift to lower $K^0_S$ masses is most probably introduced through a biased fit, especially as the fit parameters $n_L$, $n_R$, $\alpha_L$ and $\alpha_R$ are fixed to a chosen value mentioned in Table 3.2.

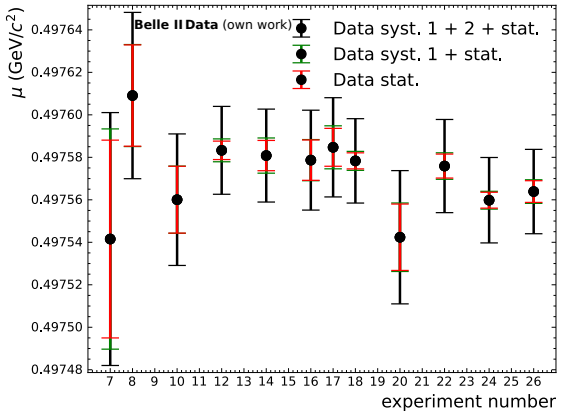
The two plots in Figs. 4.1a and 4.2a show a discrepancy between data and MC. The mean and FWHM of MC are almost always slightly below that of data indicating a systematic error. By taking a look at the plot of the $K^0_S$ mass distribution in Fig. 3.1 it is possible to give an explanation. In the tails, data is higher than MC but the peak of data is smaller than the one of MC, probably because the background portion in data is higher than in MC. As described in Section 3.4, the fit function is constructed out of a signal and background part and should therefore find the correct mean and width of the signal peak. However, the fit is very complex and the resulting fit parameters can be correlated leading to the problem that signal and background are not modeled correctly. As the background portion in data is higher than in MC this effect can have a higher impact on data, pushing the FWHM value up.

Now I want to take a look at the three sources of uncertainty on the fit parameters. These can be seen in Figs. 4.1b and 4.1c for $\mu$ over experiments, or Figs. 4.2b and 4.2c for the FWHM over experiments. The statistical uncertainty in MC is overall smaller than in data as a result of the four times higher number of simulated events. The number of events in one experiment also determines if statistical or systematic uncertainty is more important,

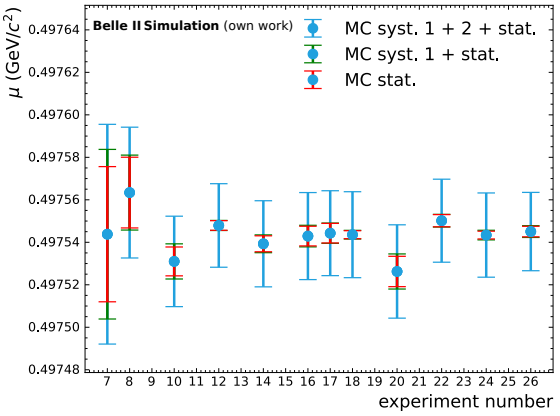
experiments 7 and 8 for example have low statistics, resulting in a dominating statistical uncertainty. However, in most cases we have high enough statistics, leading to a dominating systematic uncertainty. As mentioned in Section 3.5.2 the systematic uncertainty is set up as a combination of two parts, systematic uncertainty 1 $\sigma_{\text{syst.1}}$ considering a narrower $K^0_S$ mass range and systematic uncertainty 2 $\sigma_{\text{syst.2}}$ taking into consideration another fit function. Systematic uncertainty 1 $\sigma_{\text{syst.1}}$ is thereby always very small but the interesting part is, that it is more important for the FWHM than for $\mu$. It can be explained by considering, that the FWHM is directly proportional to $\sigma$, which will change because in the narrower mass range a small amount of extreme values in the tails will be removed. On the contrary, the mean will not be affected by that.




(a) Data and MC

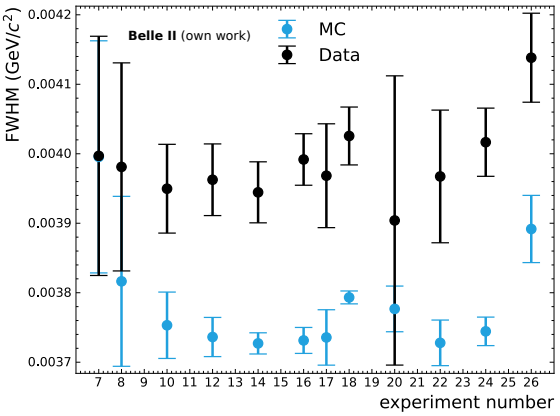



(b) Data

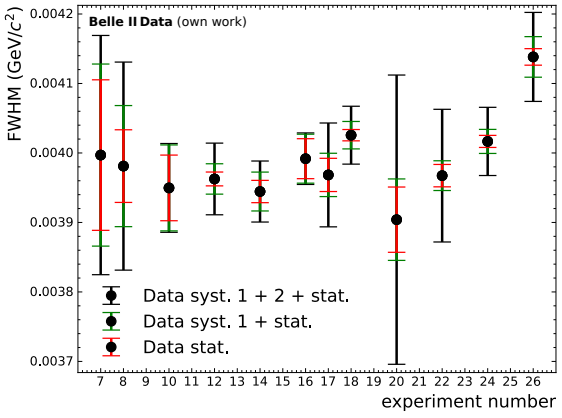



(c) MC

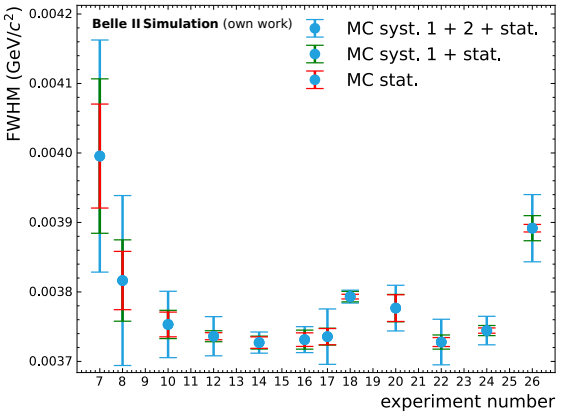
Figure 4.1.: The plots show the obtained $\mu$ values for data and MC from fits performed on experiments (run conditions). In (a), both data and MC are displayed to allow for better comparison, also the $K^0_S$ mass pdg value and its uncertainties are shown. In (b) and (c) data and MC are shown with their uncertainty portions to determine the importance of each uncertainty on the given experiment (run condition).




(a) Data and MC




(b) Data



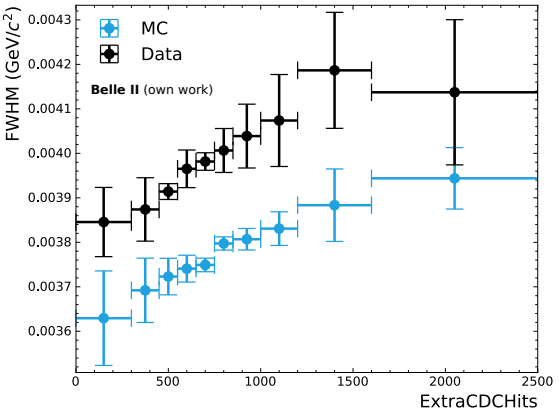

(c) MC

Figure 4.2.: The plots show the obtained FWHM values for data and MC from fits performed on experiments (run conditions). In (a), both data and MC are displayed to allow for better comparison. In (b) and (c) data and MC are shown with their uncertainty portions to determine the importance of each uncertainty on the given experiment (run condition).

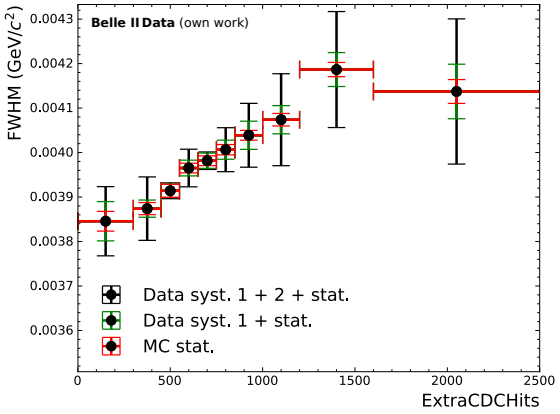
## 4.2. By observables

In this section I want to observe the behavior of the fit parameters as a function of beam background conditions. To do so, I would like to introduce the two Belle II specific variables used in this process. The ExtraCDCHits variable returns the number of CDC hits in the event not assigned to any track. The ECLOutOfTimeCrystals variable returns the number of ECL Crystals that are out of time. A higher number of ExtraCDCHits/ECLOutOfTimeCrystals therefore indicates more severe beam background conditions. However, these two variables consider different background processes, which suggests a slightly different behavior of these variables.
To do this, first the data and MC datasets need to be split in 10 different bins of ExtraCDCHits and ECLOutOfTimeCrystals. Then a fit is performed on each of these bins independently. The bins are chosen in a way that ensures the statistics of every bin are

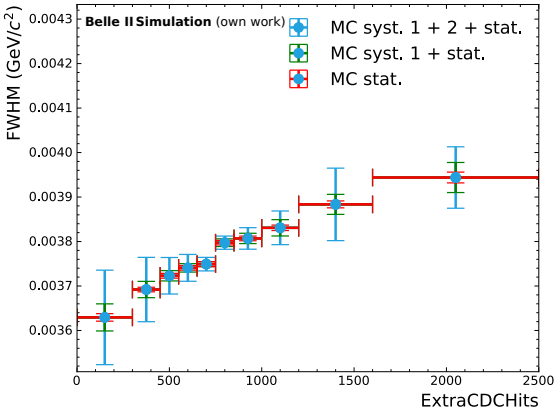
sufficiently high, resulting in different-sized bins. For this section experiments 7 and 8 were excluded from the datasets due to problems with them. In Appendices A.3 and A.4 the fits, performed on the different bins of the ExtraCDCHits and ECLOutOfTimeCrystals variable, are shown.
As already mentioned, these variables indicate the amount of background in an event. By looking at lower numbers of ExtraCDCHits/ECLOutOfTimeCrystals we will have less background and consequently less background falsely assigned to tracks. As a result, in data, the FWHM of the signal peak will increase with an increasing number of ExtraCDCHits/ECLOutOfTimeCrystals. This can be seen in Figs. 4.3a and 4.4a. MC reproduces that the FWHM increases as a function of beam background conditions. However, it is not perfect, as from the plots it looks like that the slope of the data and MC curves are different. Despite the fact, that the increase of the FWHM is not large in absolute terms, the relative increase is almost 10 % and is therefore quite significant.




(a) Data and MC

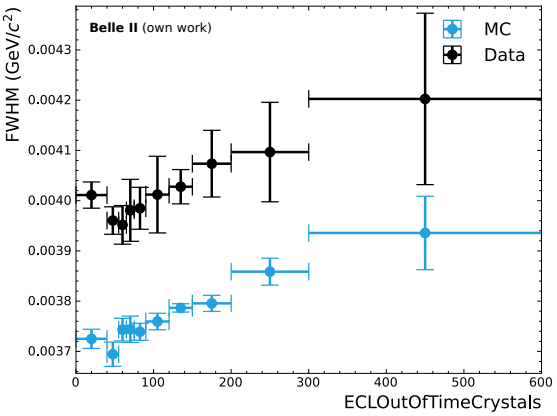



(b) Data

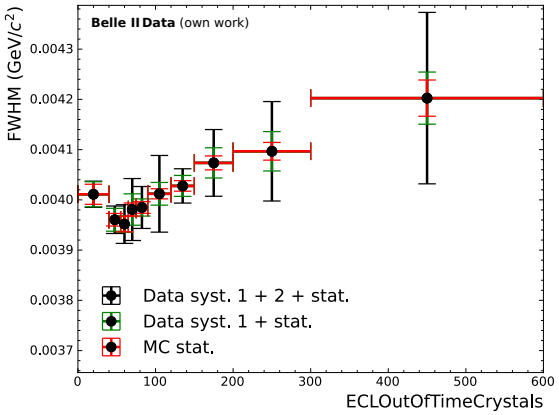



(c) MC

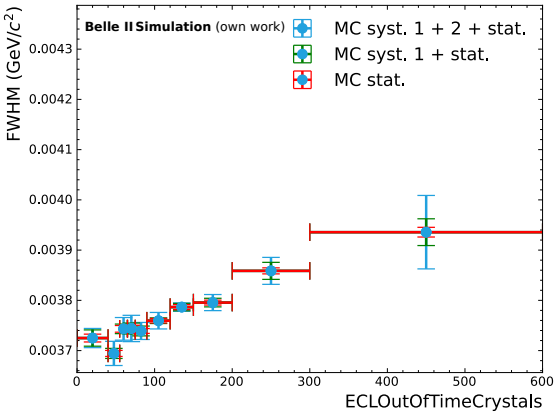
Figure 4.3.: The plots show the obtained FWHM values for data and MC from fits performed on bins of the ExtraCDCHits observable. In (a), both data and MC are displayed to allow for better comparison. In (b) and (c) data and MC are shown with their uncertainty portions to determine the importance of each uncertainty on the selected bin of ExtraCDCHits.




(a) Data and MC




(b) Data

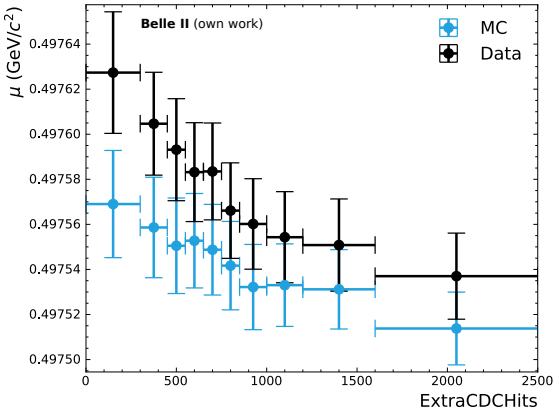



(c) MC

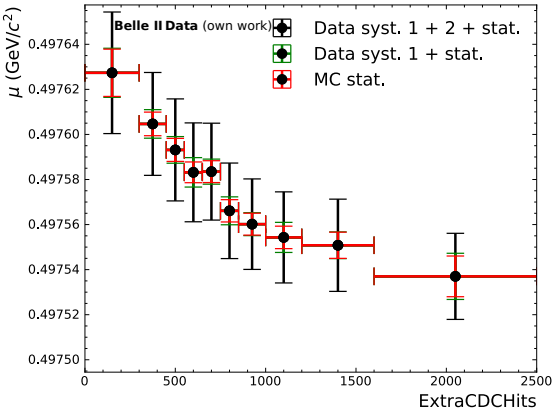
Figure 4.4.: The plots show the obtained FWHM values for data and MC from fits performed on bins of the ECLOutOfTimeCrystals observable. In (a), both data and MC are displayed to allow for better comparison. In (b) and (c) data and MC are shown with their uncertainty portions to determine the importance of each uncertainty on the selected bin of ECLOutOfTimeCrystals.

The mean value of the signal peak is decreasing as a function of ExtraCDCHits as can be seen in Fig. 4.5a. An explanation for this can be the correlation with other fit parameters, or that the background in reality has another shape causing the background to be modeled falsely. One argument for the background to be modeled falsely is underlined by Fig. 4.7, as it shows that there is a trend to a steeper negative linear function for the background with an increasing amount of background.

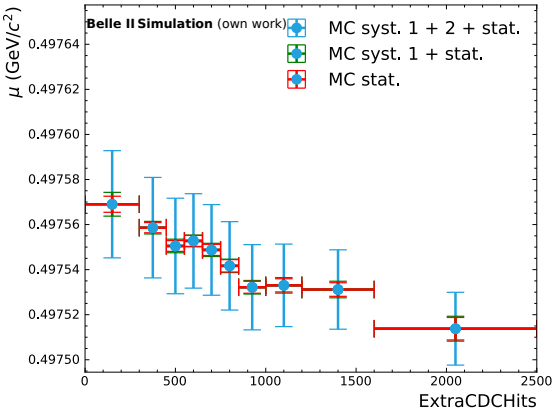
In contrary to that, the mean value of the signal peak is constant as a function of ECLOutOfTimeCrystals as can be seen in Fig. 4.6a. Also, such a trend for the slope $m$ of the linear background function from Eq. (3.4) can not be seen in Fig. 4.8. The different behavior of $\mu$ and $m$ in these two variables is probably a result of the fact, that certain background processes have different effects on the CDC than on the ECL.




(a) Data and MC

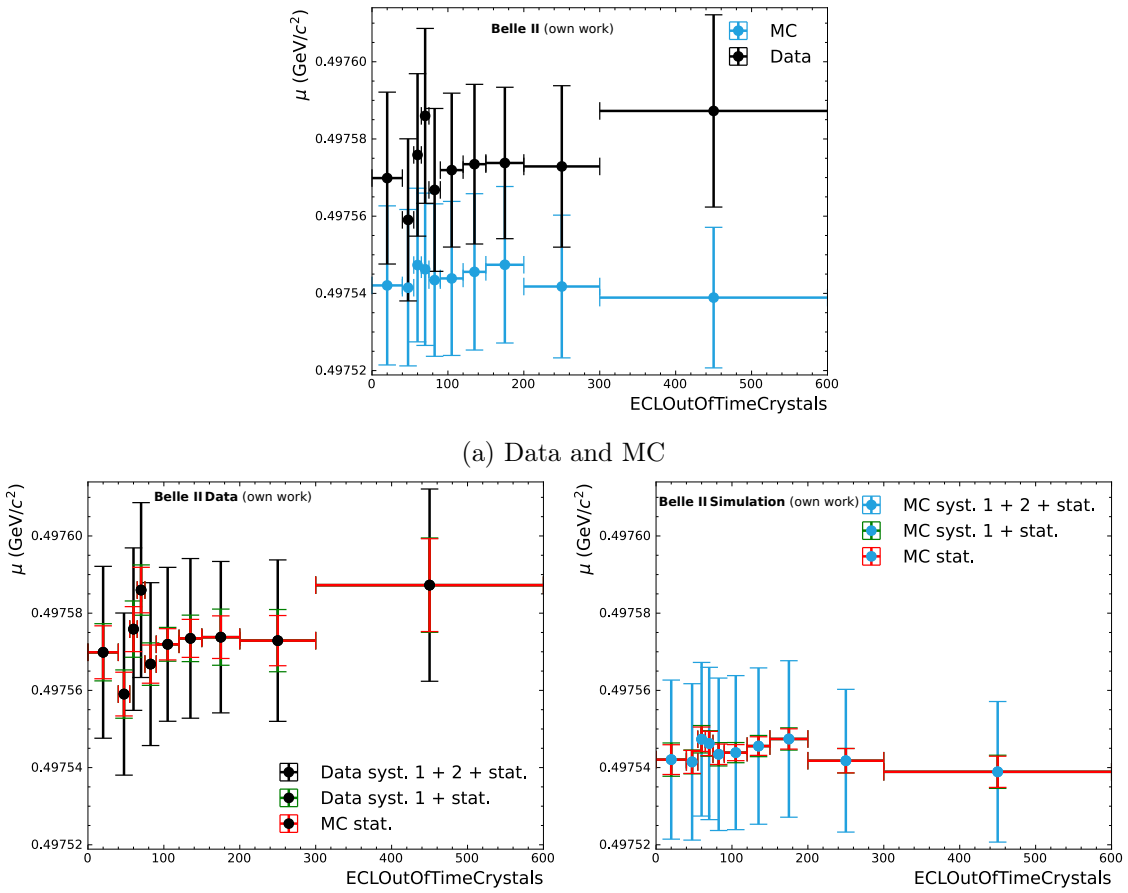



(b) Data




(c) MC

Figure 4.5.: The plots show the obtained $\mu$ values for data and MC from fits performed on bins of the ExtraCDCHits observable. In (a), both data and MC are displayed to allow for better comparison. In (b) and (c) data and MC are shown with their uncertainty portions to determine the importance of each uncertainty on the selected bin of ExtraCDCHits.




(a) Data and MC

(b) Data

(c) MC

Figure 4.6.: The plots show the obtained $\mu$ values for data and MC from fits performed on bins of the ECLOutOfTimeCrystals observable. In (a), both data and MC are displayed to allow for better comparison. In (b) and (c) data and MC are shown with their uncertainty portions to determine the importance of each uncertainty on the selected bin of ECLOutOfTimeCrystals.

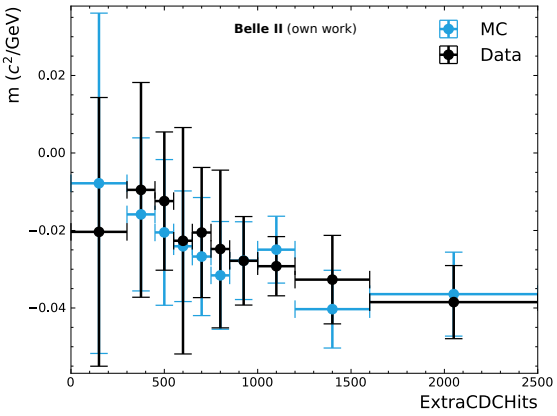



Figure 4.7.: The plot shows the obtained $m$ (slope of the linear background function) values for data and MC from fits performed on bins of the ExtraCDCHits observable.

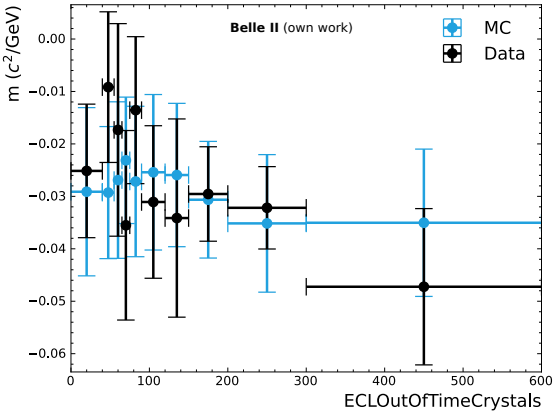



Figure 4.8.: The plot shows the obtained $m$ (slope of the linear background function) values for data and MC from fits performed on bins of the ECLOutOfTimeCrystals observable.

# 5. Conclusion and Outlook

To summarize, the task of comparing data and MC distributions of the $K_S^0$ is done by performing a fit on the $K_S^0$ mass and extracting the mean and FWHM of the according signal peak. In this thesis, the $K_S^0$ performance monitoring is based on different conditions during data taking and on observables representing the beam background conditions. We also built a website, where these results are displayed.

The plots from Figs. 4.3a and 4.4a show that the signal resolution decreases with more severe beam background conditions. This emphasizes that there are clearly observable effects due to the beam background and data taking conditions. These effects are still not perfectly characterized and understood in Belle II. Further improving the $K_S^0$ performance monitoring hopefully can help to understand the underlying physics processes. In the following, I want to give an outlook on what can be done to improve the automated $K_S^0$ performance monitoring conducted in this thesis.

As it is mentioned in Section 3.4 the fit parameters of the DSCB function are highly correlated, and as a result, some of them are fixed to a chosen value. This causes the problem of introducing a bias into the fit as it is discussed in Section 4.1. This problem can be approached by either searching for better fixed parameters for the DSCB function or by using a different fit function with less correlated parameters, that ideally provides a similar or even better description of the signal than the DSCB function.

Another improvement that can be made to the $K_S^0$ performance monitoring is to add additional information to the behavior of the parameters $\mu$ and FWHM. For expample a fit can be performed on the curves of data and MC of the FWHM over beam background conditions in Figs. 4.3a and 4.4a. The slope of these curves can then be extracted to see if there are differences in the steepness of the data and MC curves.

A very important step towards automating the $K_S^0$ performance monitoring that still needs to be implemented is the automation of the upload of newly produced results to the website.

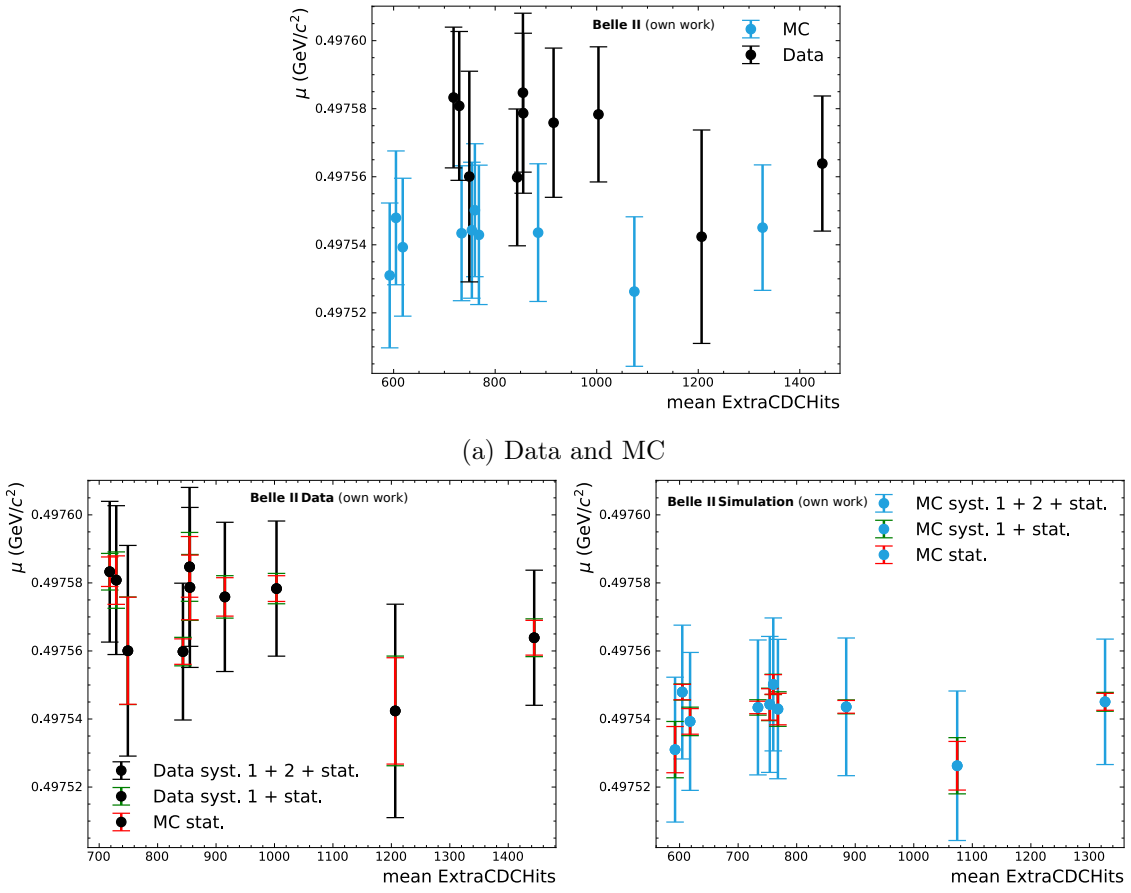
In conclusion it can be said, that interesting observations can be made by monitoring the $K_S^0$ performance, such as the quite significant relative increase of the FWHM parameter as a function of beam background conditions, discussed in Section 4.2.

# A. Appendix

## A.1. Data and MC comparison by run conditions

In this section the data and MC comparison by run conditions from Section 4.1 is evaluated a bit further. Based on the fits made for each run condition, I calculate the mean value of the number of ExtraCDCHits/ECLOutOfTimeCrystals and plot $\mu$ and the FWHM against these calculated means. This results in the following plots in Figs. A.1a, A.2a, A.3a and A.4a and can give us an idea of how the mean and width of the signal peak change when having a different amount of background. For this part experiments 7 and 8 were excluded from the dataset because of problems with them.

As can be seen from the plots, $\mu$ seems to be constant for different numbers of mean background variables. In contrast, the FWHM slightly increases with an increasing number of mean background variables.

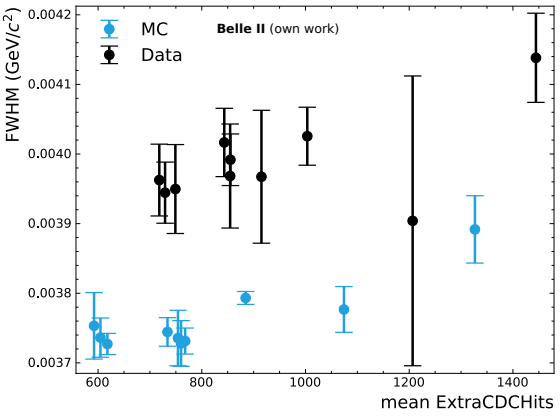



(a) Data and MC

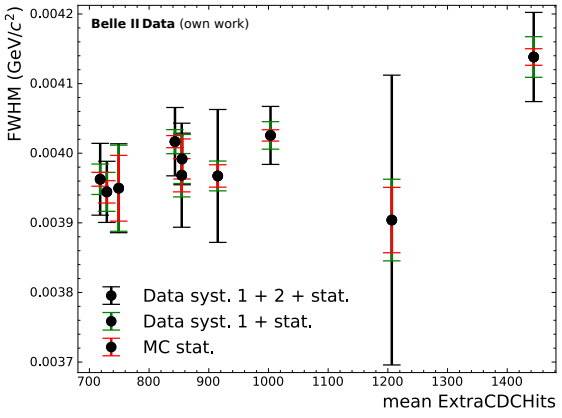
(b) Data

(c) MC

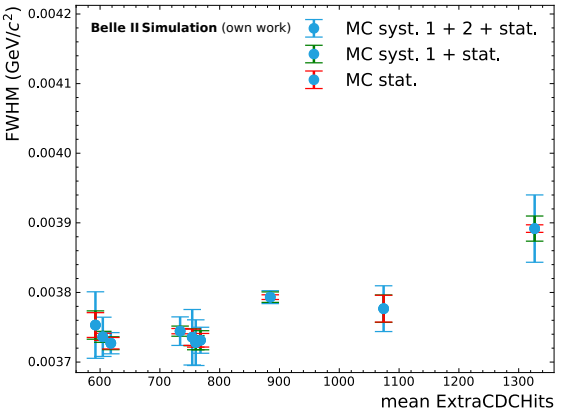
Figure A.1.: The plots show the obtained $\mu$ values for data and MC from fits performed on experiments (run conditions) over the mean value of ExtraCDCHits in the corresponding experiment. In (a), both data and MC are displayed to allow for better comparison. In (b) and (c) data and MC are shown with their uncertainty portions to determine the importance of each uncertainty on the given experiment (run condition).




(a) Data and MC

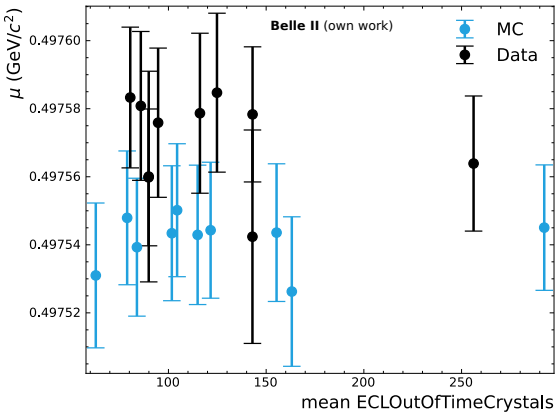



(b) Data

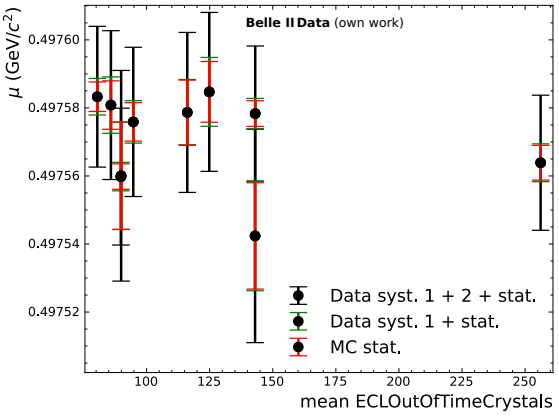



(c) MC

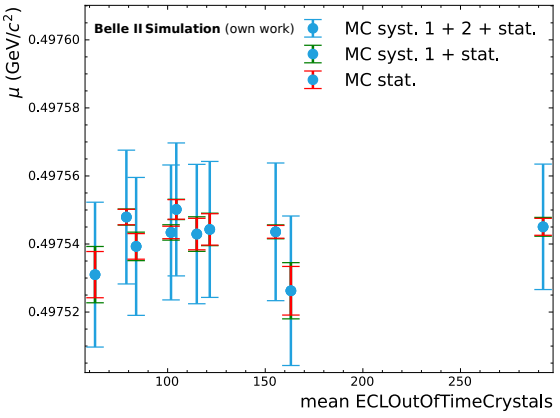
Figure A.2.: The plots show the obtained FWHM values for data and MC from fits performed on experiments (run conditions) over the mean value of ExtraCDCHits in the corresponding experiment. In (a), both data and MC are displayed to allow for better comparison. In (b) and (c) data and MC are shown with their uncertainty portions to determine the importance of each uncertainty on the given experiment (run condition).




(a) Data and MC

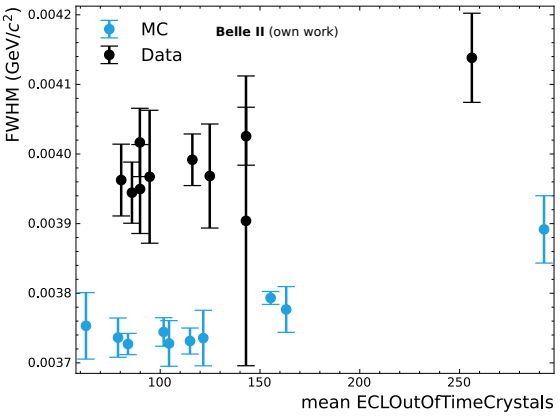



(b) Data

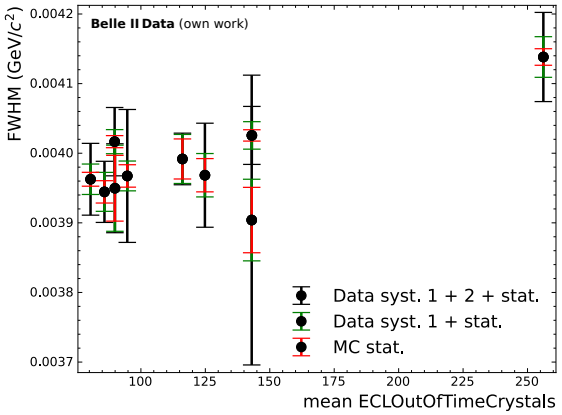



(c) MC

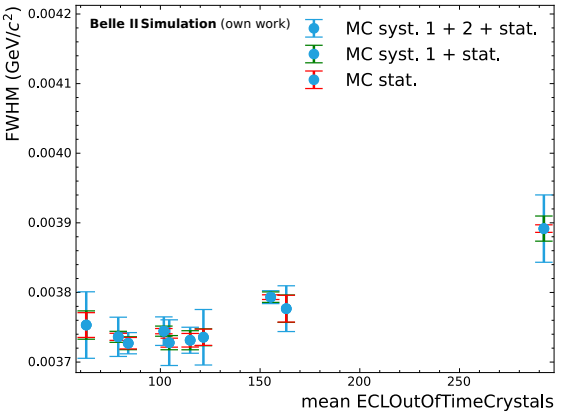
Figure A.3.: The plots show the obtained $\mu$ values for data and MC from fits performed on experiments (run conditions) over the mean value of ECLOutOfTimeCrystals in the corresponding experiment. In (a), both data and MC are displayed to allow for better comparison. In (b) and (c) data and MC are shown with their uncertainty portions to determine the importance of each uncertainty on the given experiment (run condition).




(a) Data and MC




(b) Data

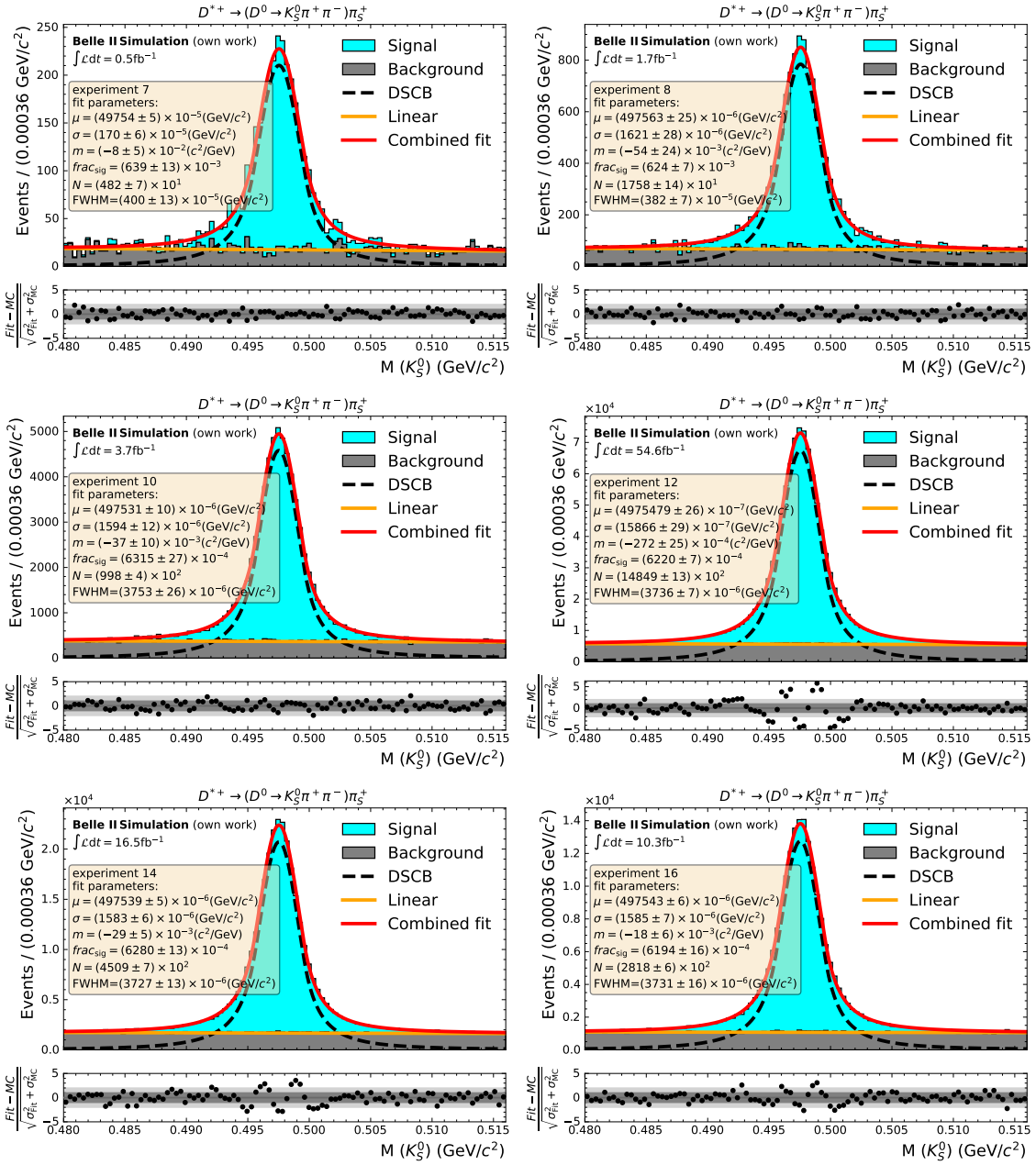



(c) MC

Figure A.4.: The plots show the obtained FWHM values for data and MC from fits performed on experiments (run conditions) over the mean value of ECLOutOfTimeCrystals in the corresponding experiment. In (a), both data and MC are displayed to allow for better comparison. In (b) and (c) data and MC are shown with their uncertainty portions to determine the importance of each uncertainty on the given experiment (run condition).

# A.2. Fits performed on experiments (run conditions)

## A.2.1. Fits on MC



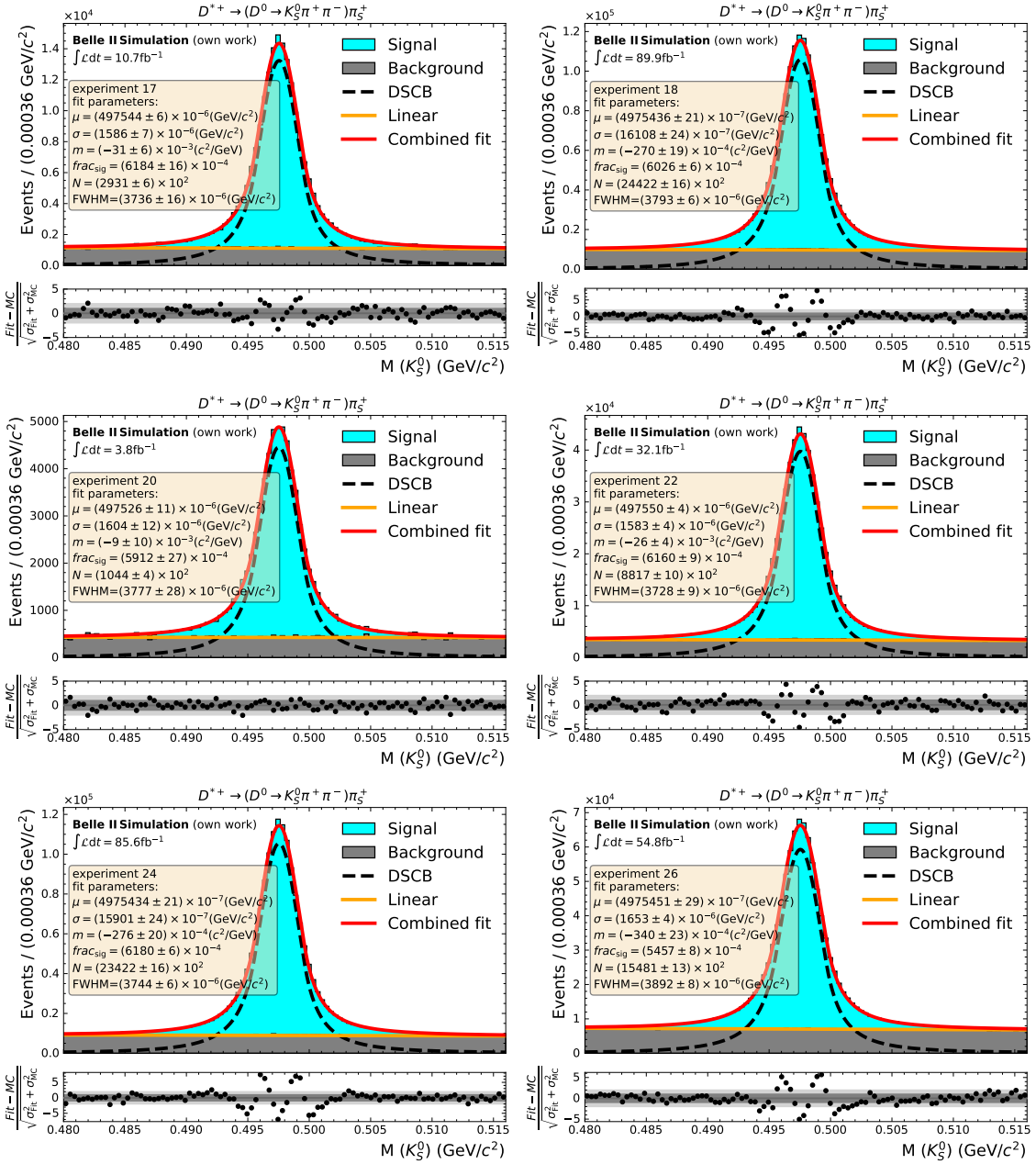
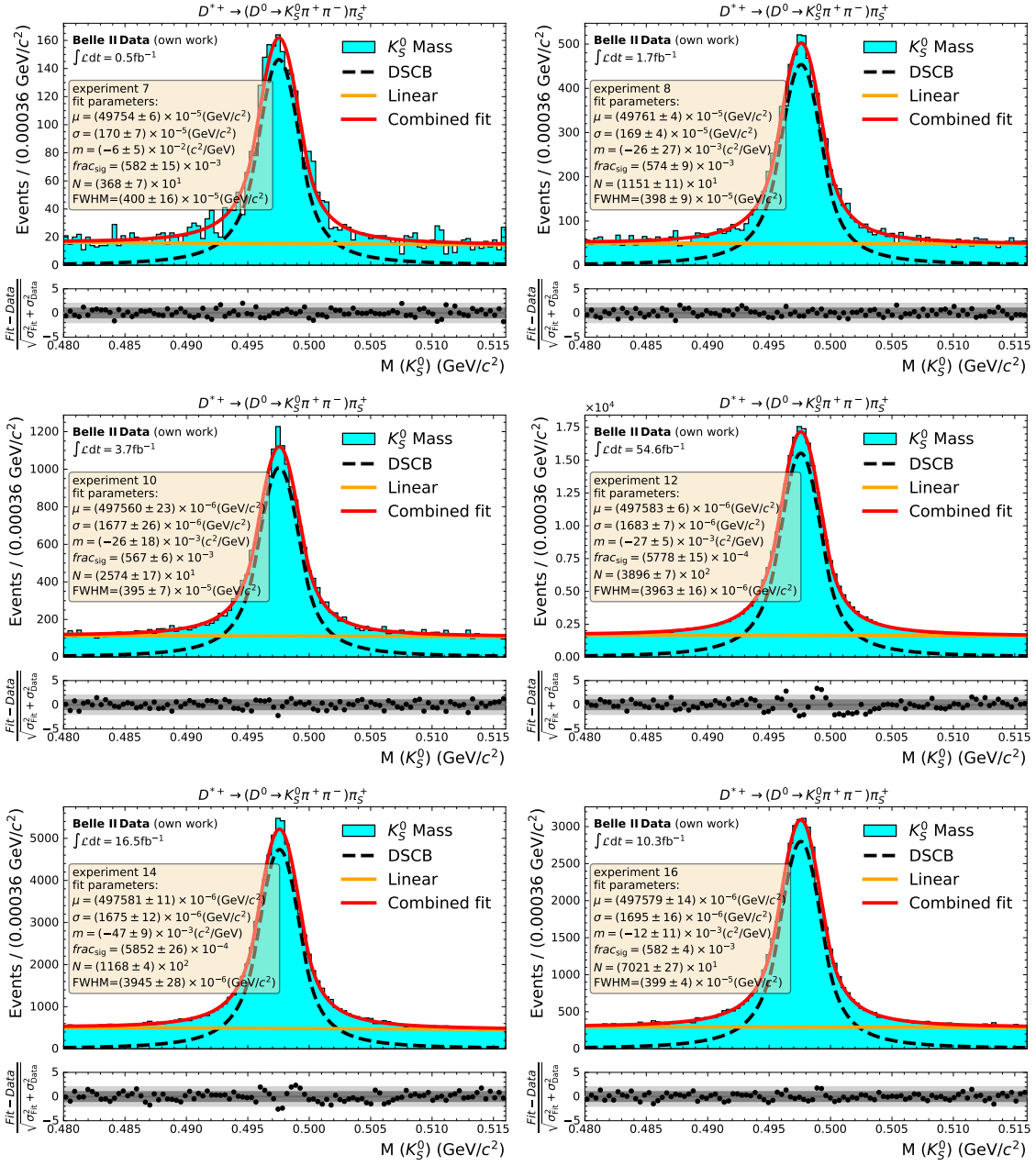



Figure A.5.: Each plot shows the $K^0_S$ mass distribution of MC overlaid with the performed fit (red), the linear background part (orange) and the signal part with DSCB function (dashed black line). The resulting fit parameters are also listed. The shown $K^0_S$ mass distribution does only include the MC events from the given experiment (run condition). MC is rescaled to the same luminosity of data.

## A.2.2. Fits on data



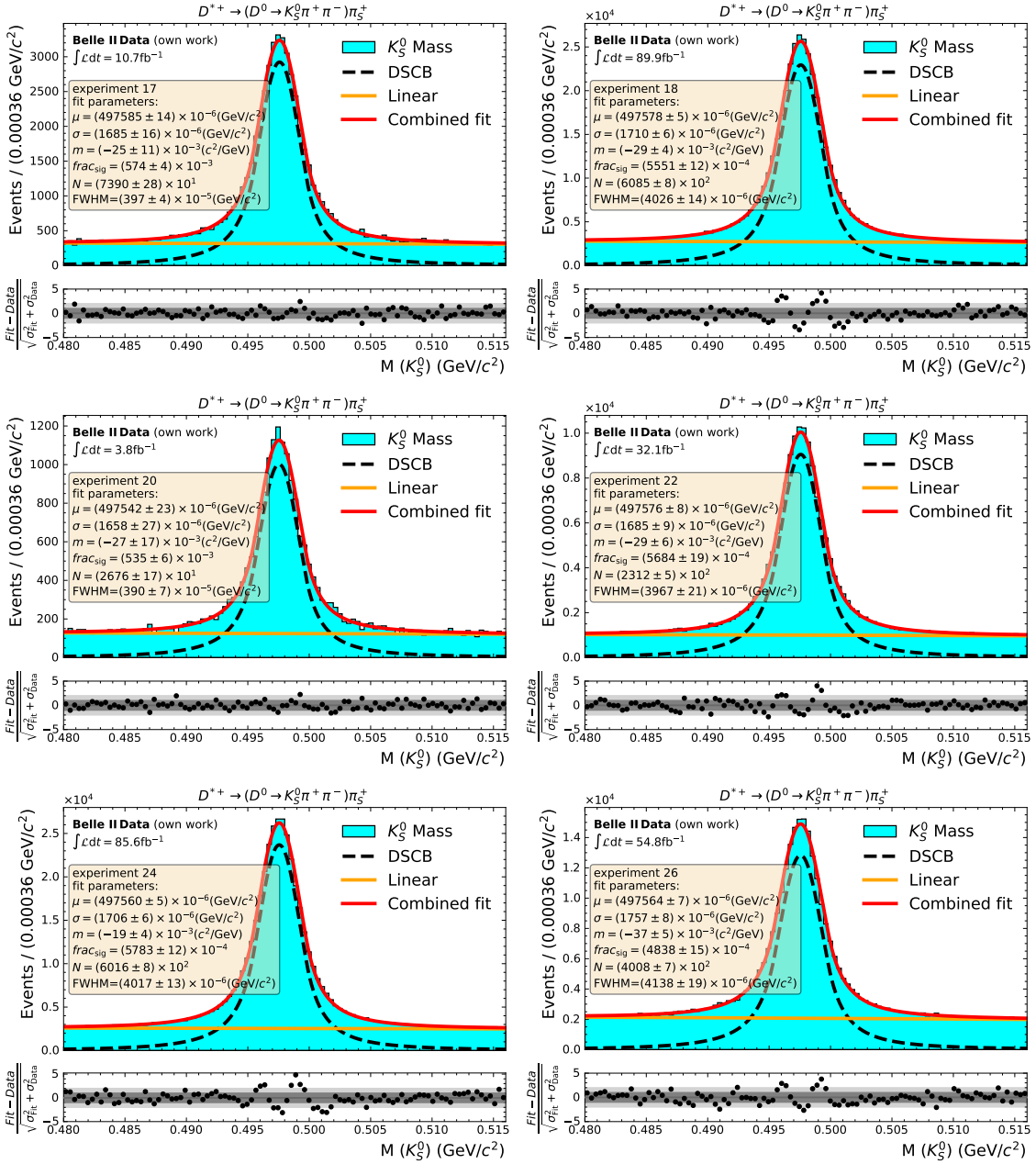
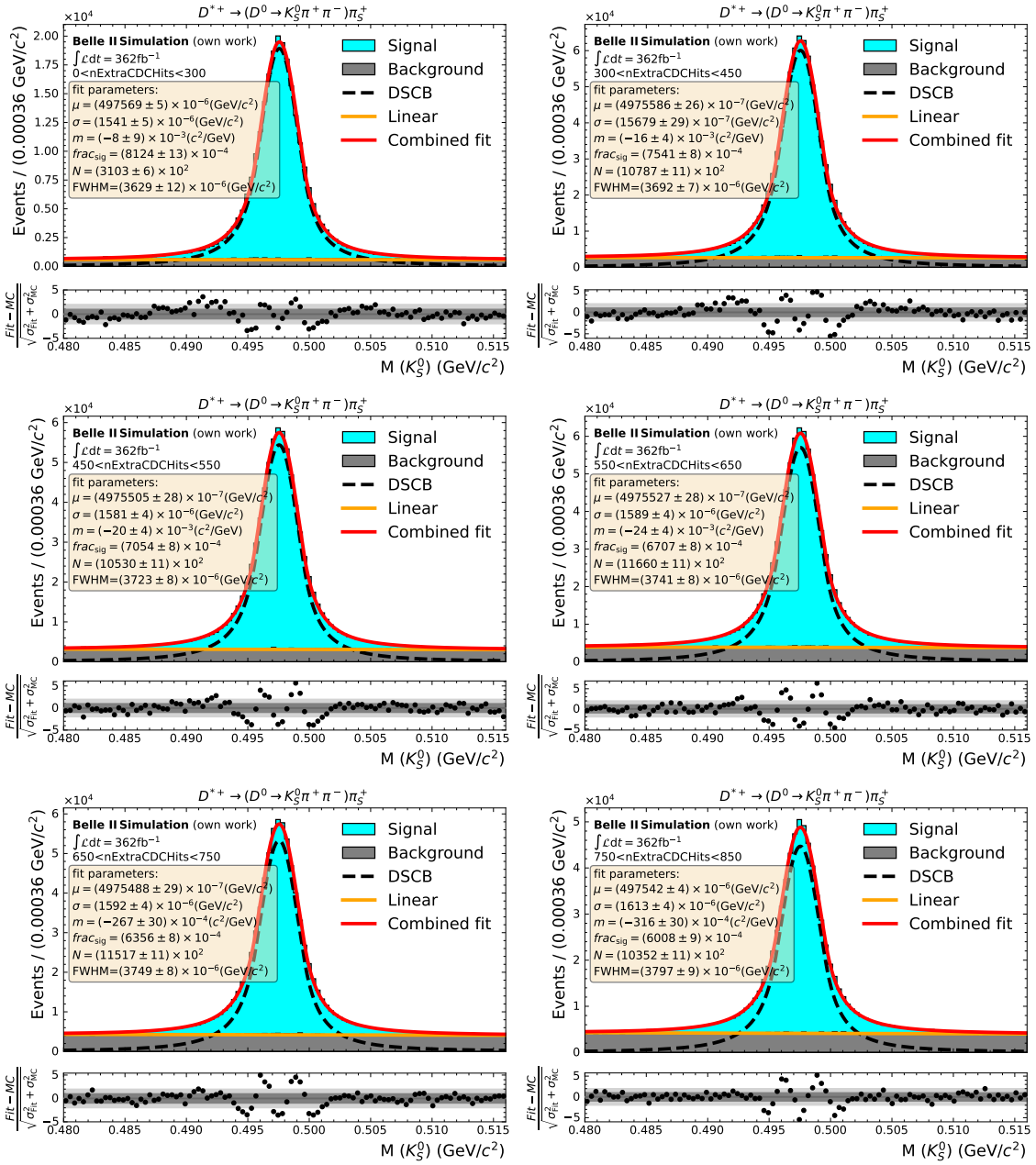



Figure A.6.: Each plot shows the $K^0_S$ mass distribution of data overlaid with the performed fit (red), the linear background part (orange) and the signal part with DSCB function (dashed black line). The resulting fit parameters are also listed. The shown $K^0_S$ mass distribution does only include the data events from the given experiment (run condition).

# A.3. Fits performed on bins of ExtraCDCHits

## A.3.1. Fits on MC



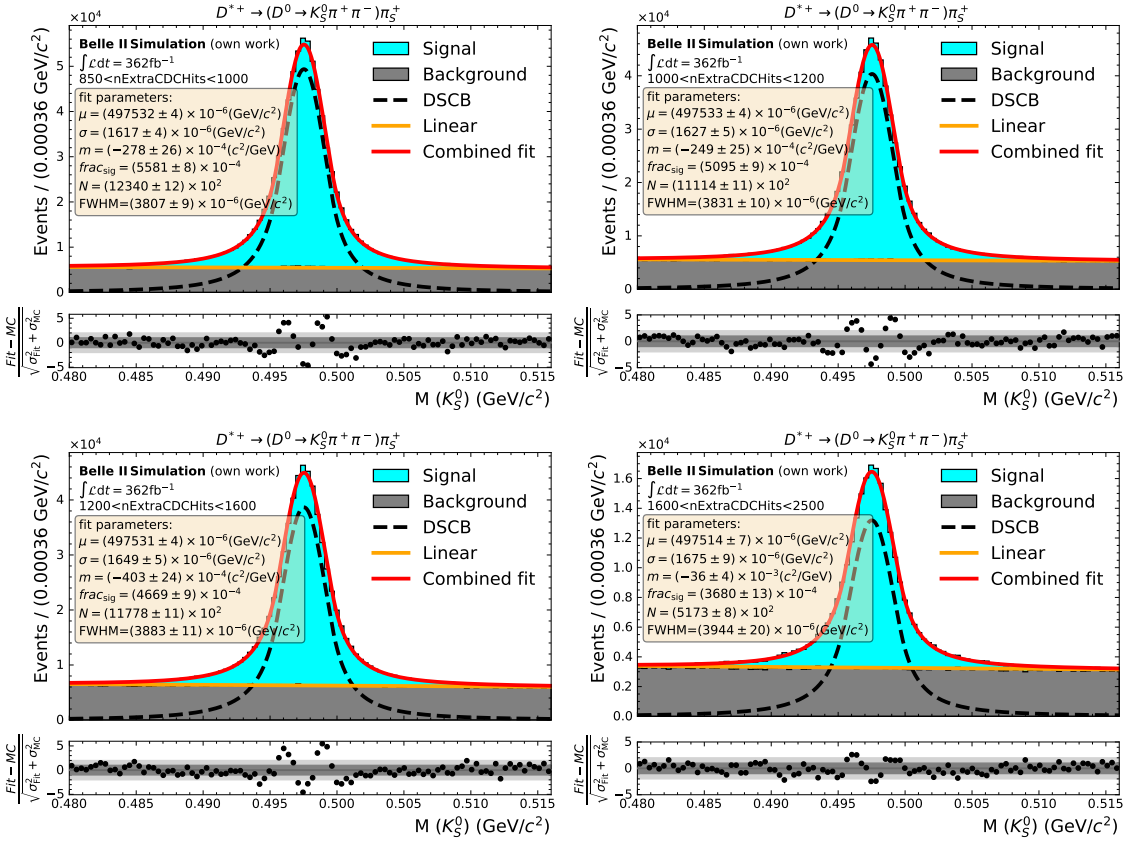
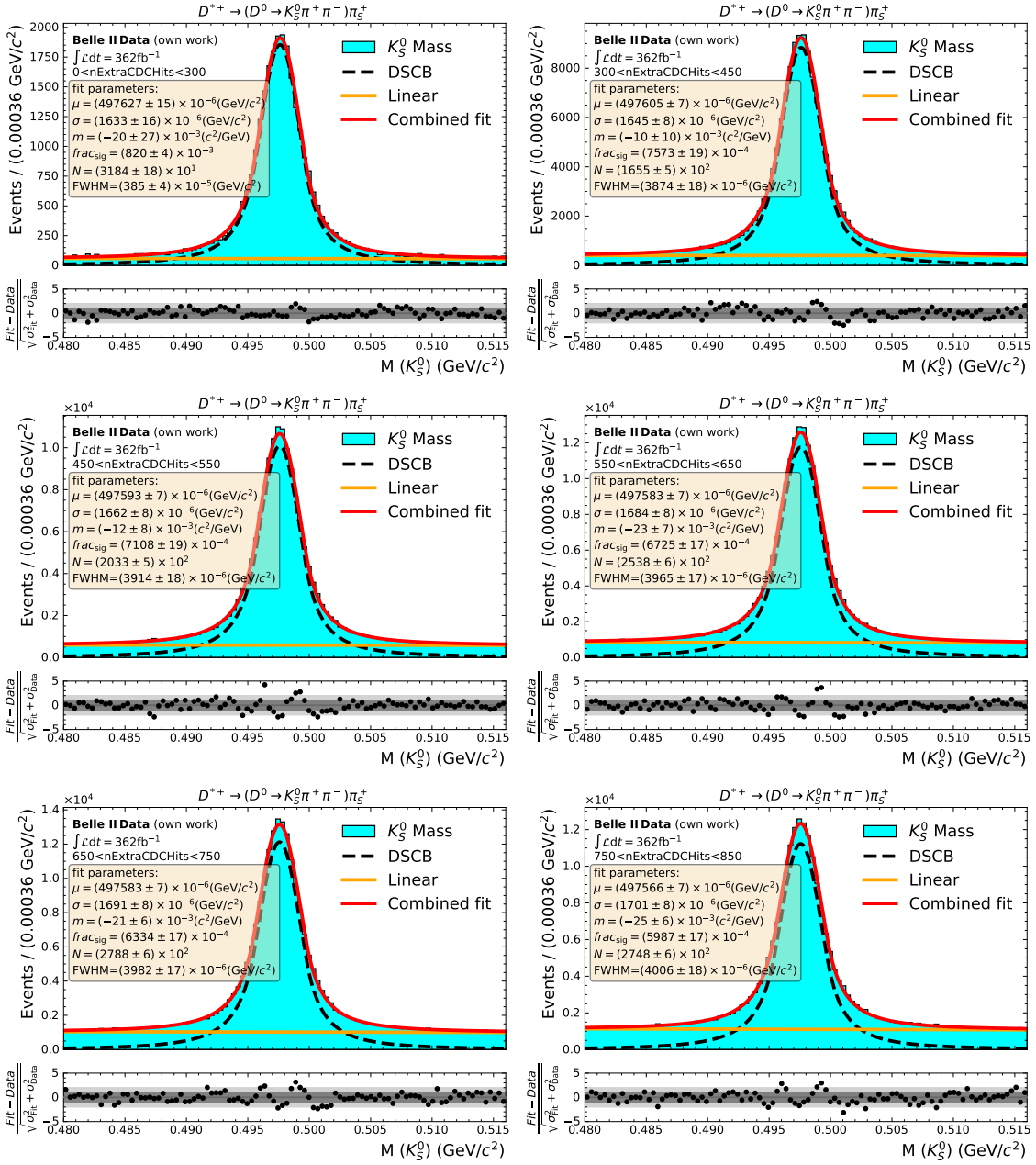



Figure A.7.: Each plot shows the $K^0_S$ mass distribution of MC overlaid with the performed fit (red), the linear background part (orange) and the signal part with DSCB function (dashed black line). The resulting fit parameters are also listed. The shown $K^0_S$ mass distribution does only include the MC events that lie in the given ExtraCDCHits bin. MC is rescaled to the same luminosity of data.

## A.3.2. Fits on data



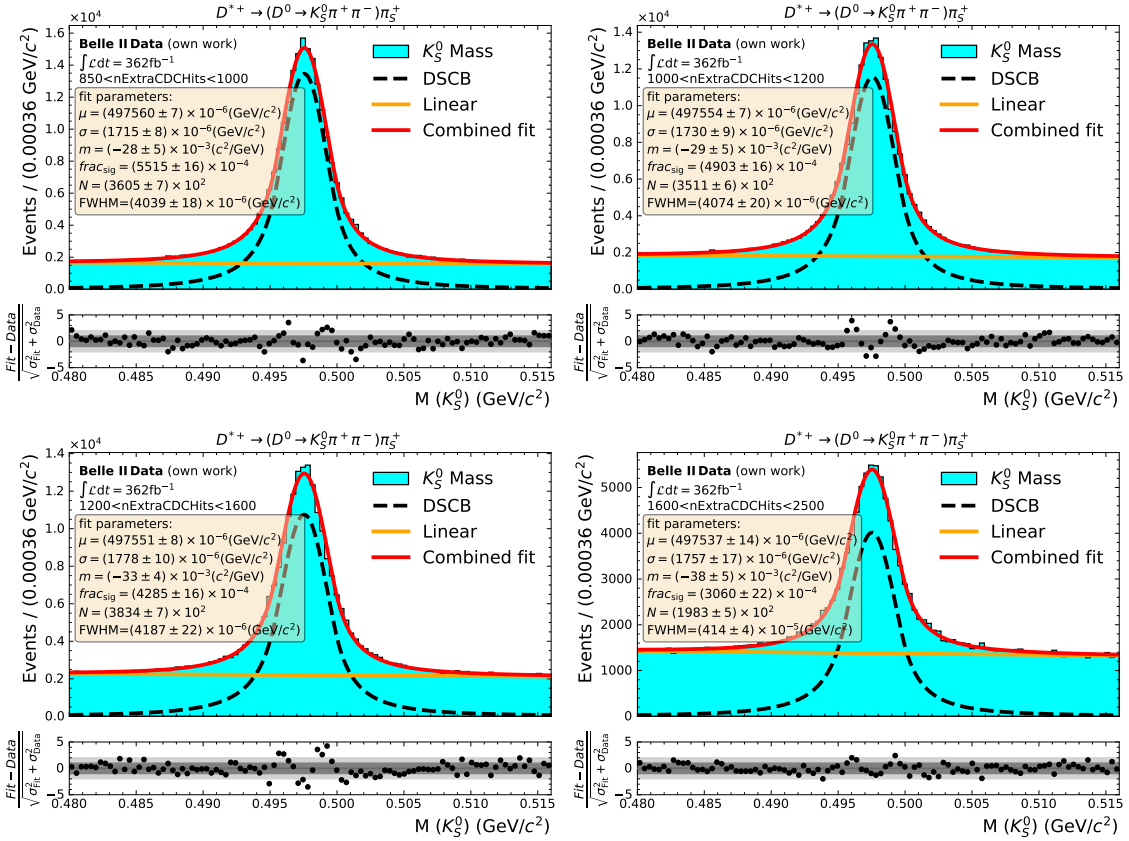
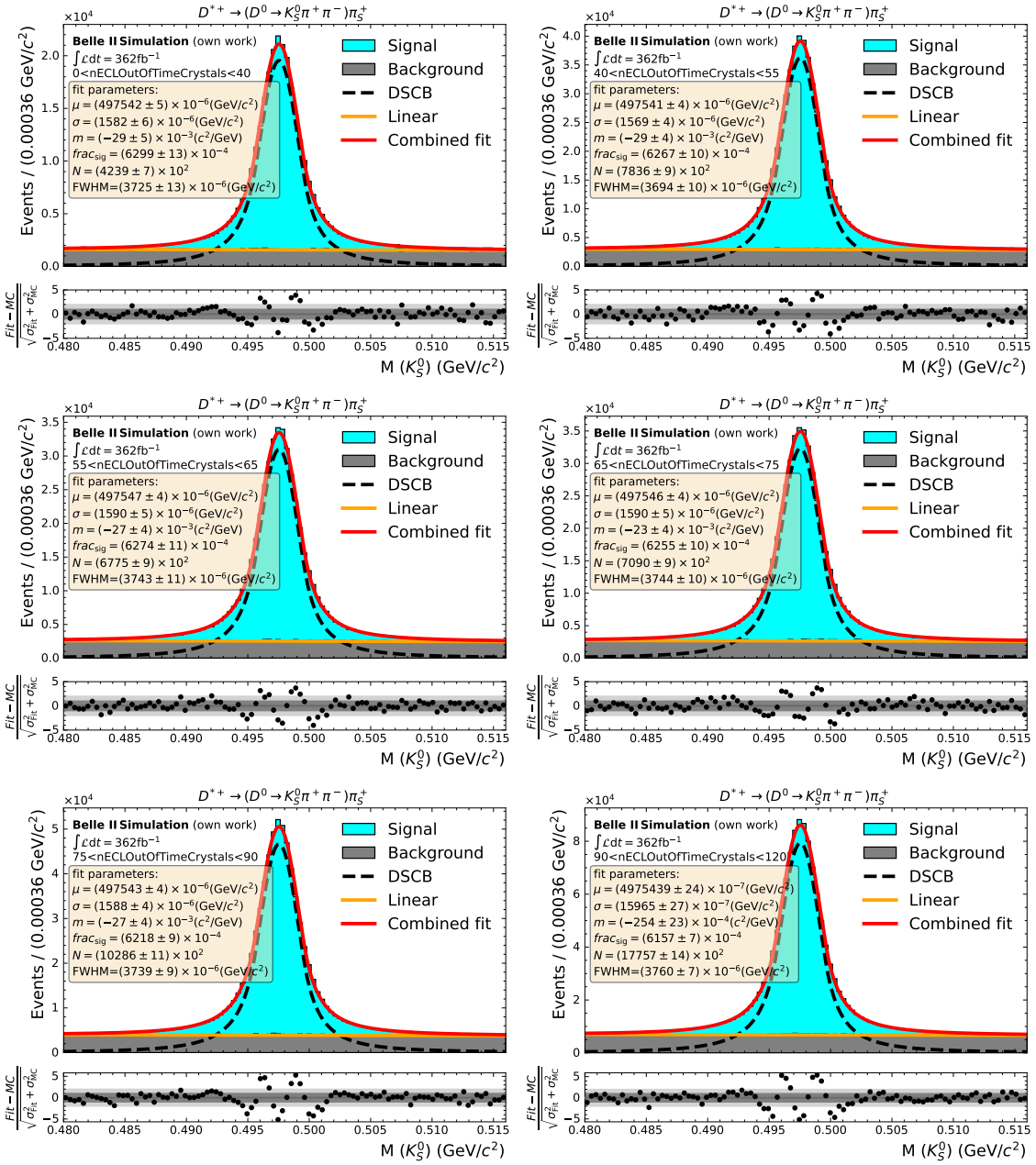



Figure A.8.: Each plot shows the $K^0_S$ mass distribution of data overlaid with the performed fit (red), the linear background part (orange) and the signal part with DSCB function (dashed black line). The resulting fit parameters are also listed. The shown $K^0_S$ mass distribution does only include the data events that lie in the given ExtraCDCHits bin.

# A.4. Fits performed on bins of ECLOutOfTimeCrystals

## A.4.1. Fits on MC



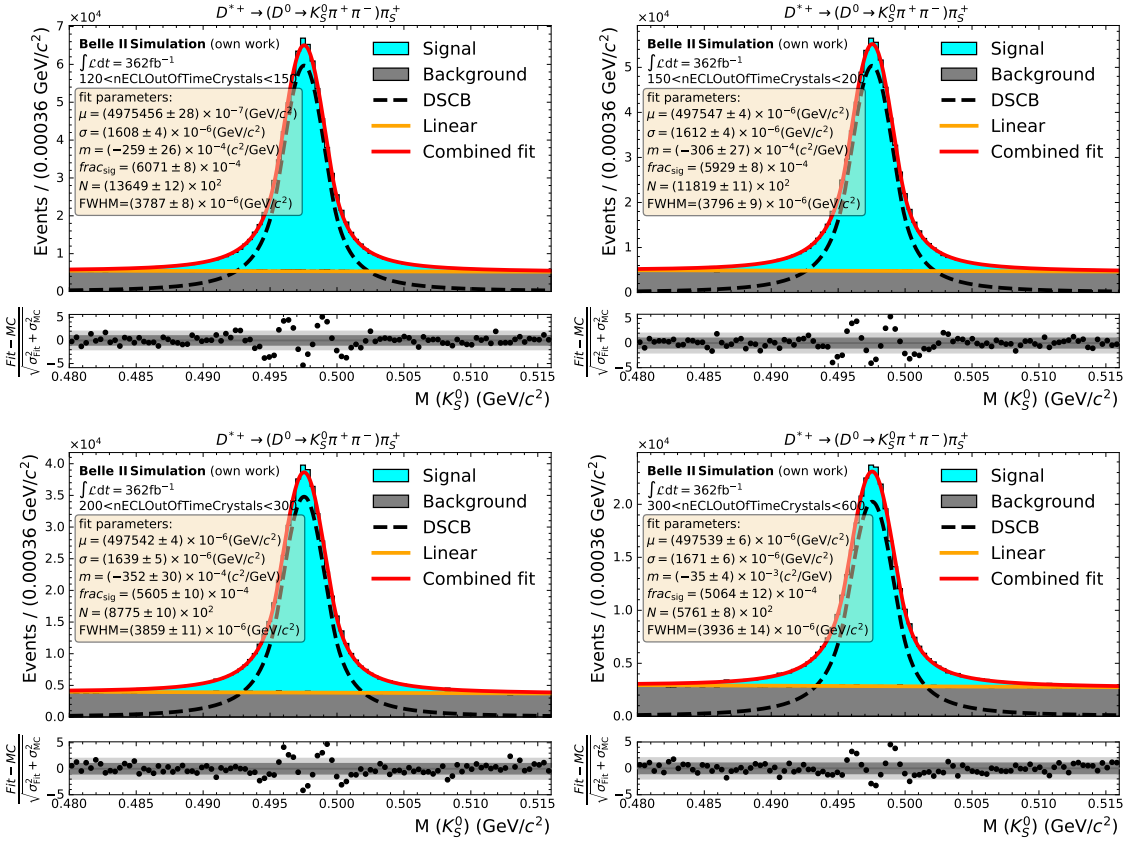
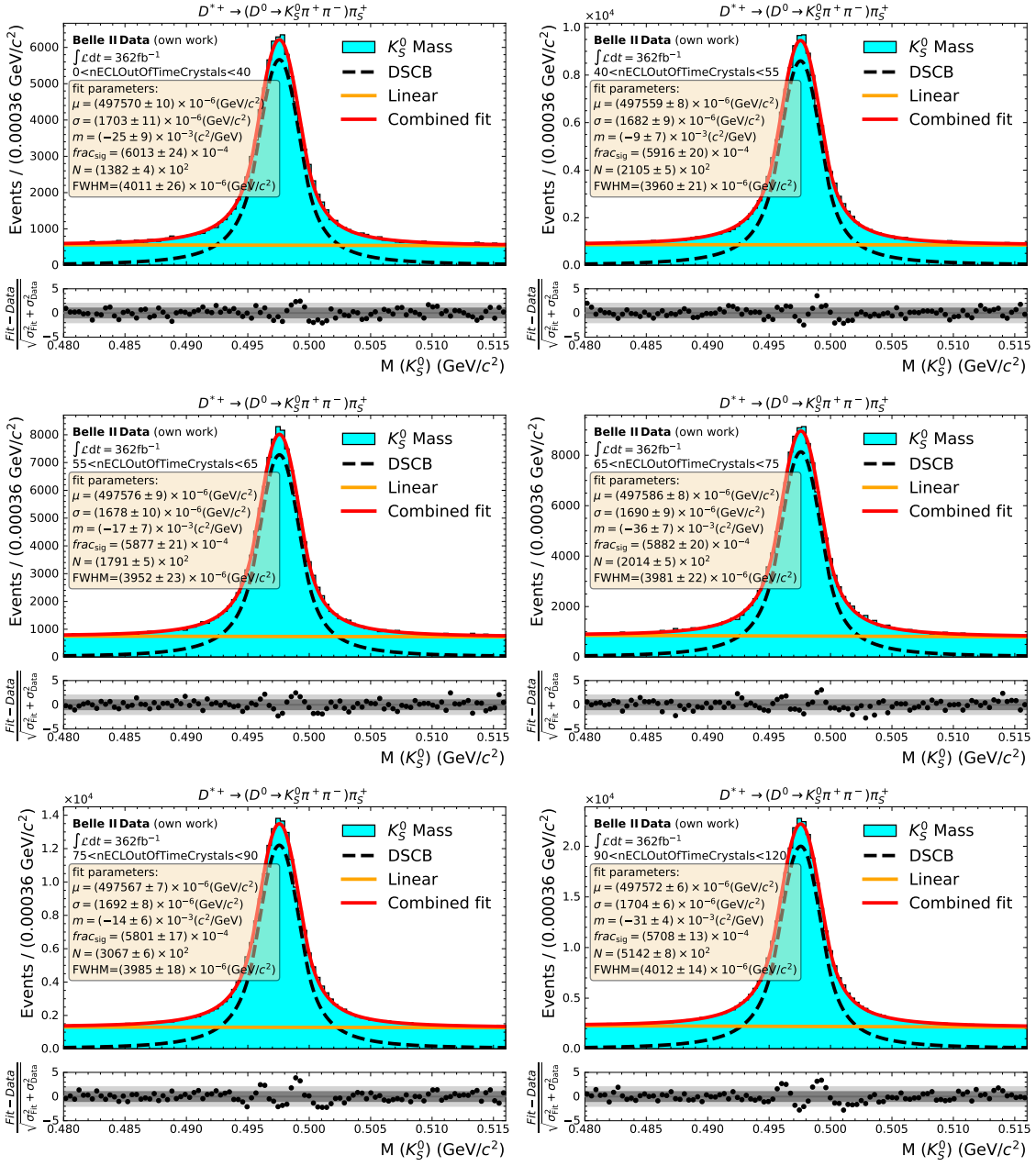



Figure A.9.: Each plot shows the $K^0_S$ mass distribution of MC overlaid with the performed fit (red), the linear background part (orange) and the signal part with DSCB function (dashed black line). The resulting fit parameters are also listed. The shown $K^0_S$ mass distribution does only include the MC events that lie in the the given ECLOutOfTimeCrystals bin. MC is rescaled to the same luminosity of data.

## A.4.2. Fits on data



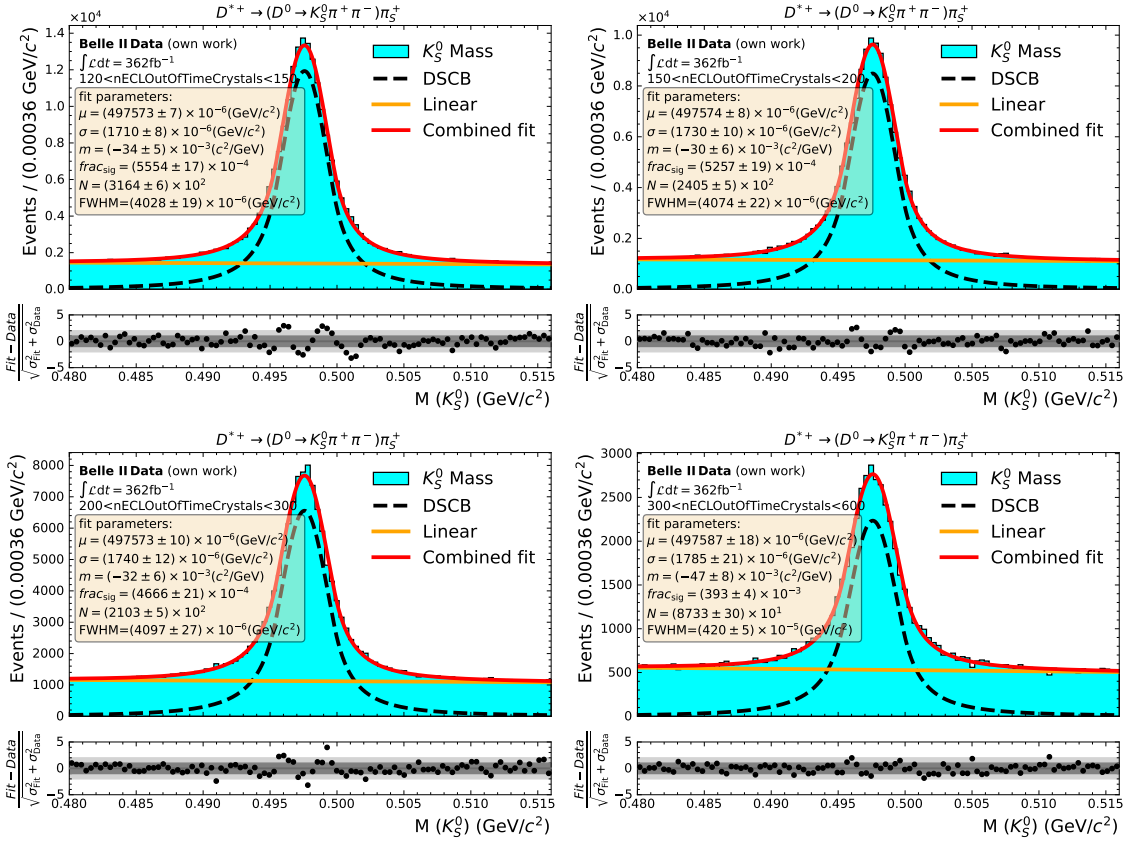



Figure A.10.: Each plot shows the $K^0_S$ mass distribution of data overlaid with the performed fit (red), the linear background part (orange) and the signal part with DSCB function (dashed black line). The resulting fit parameters are also listed. The shown $K^0_S$ mass distribution does only include the data events that lie in the given ECLOutOfTimeCrystals bin.

# Bibliography


[1] P. Ecker, "VIBE - Validation Interface for the Belle II Experiment," Oct., 2022. https://gitlab.desy.de/belle2/data-production/validation/vibe. Last accessed 25. September 2024.

[2] T. Kuhr, C. Pulvermacher, M. Ritter, T. Hauth, and N. Braun, "The Belle II Core Software," *Computing and Software for Big Science* **3** no. 1, (Nov., 2018) 1–12.

[3] **Belle II Collaboration**, "Belle II Analysis Software Framework (basf2) (release-06-00-09)," *Zenodo* (2022) .

[4] P. Ecker, J. Eppelt, G. De Pietro, "DM-Analysis-Tools." https://github.com/eckerpatrick/DM-Analysis-Tools. Last accessed 25. September 2024.

[5] E. Kou *et al.*, "The Belle II Physics Book," *Progress of Theoretical and Experimental Physics* **2019** no. 12, (12, 2019) 123C01.

[6] "Simulation: The Monte Carlo - basf2 sphinx documentation." https://software.belle2.org/development/sphinx/online_book/fundamentals/03-simulation.html. Last accessed 17. September 2024.

[7] G. de Marino and G. Casarosa, "$K_S^0$ efficiency calibration on Run1 data vs MC15," 2023. https://docs.belle2.org/record/4020. Internal Note.

[8] M. Duerr, T. Ferber, C. Garcia-Cely, C. Hearty and K. Schmidt-Hoberg, "Long-lived dark Higgs and inelastic dark matter at Belle II," *Journal of High Energy Physics* (2021) .

[9] A. J. Bevan *et al.*, "The Physics of the B Factories," *The European Physical Journal C* **74** (2014) 3026.

[10] "SuperKEKB." https://www.belle2.org/research/superkekb/. Last accessed 9. August 2024.

[11] "Data Taking - basf2 sphinx documentation." https://software.belle2.org/development/sphinx/online_book/fundamentals/02-datataking.html. Last accessed 9. August 2024.

[12] "belle2-Experiment." https://www.belle2.de/en/belle-2-experiment/superkekb. Last accessed 9. August 2024.

[13] K. Akai, K. Furukawa, and H. Koiso, "SuperKEKB collider," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **907** (2018) 188–199.

[14] **Belle II Collaboration**, T. Abe *et al.*, "Belle II Technical Design Report," 2010. https://arxiv.org/abs/1011.0352. Last accessed 26. September 2024.

[15] "The Detector | Belle II Experiment." https://belle2.jp/detector/. Last accessed 6. September 2024.

[16] "Modules and paths - basf2 light-2403-persian documentation." https://software.belle2.org/development/sphinx/framework/doc/modules_paths.html#general-modpath. Last accessed 25. September 2024.

[17] **Belle II Collaboration**, S. Weber, "K Short Monitoring," 2024. https://gitlab.desy.de/kit-etp/K-short-monitoring.

[18] **Belle II Collaboration**, I. Adachi *et al.*, "Measurement of the integrated luminosity of data samples collected during 2019-2022 by the Belle II experiment," 2024. https://arxiv.org/abs/2407.00965. Last accessed 26. September 2024.

[19] D. J. Lange, "The EvtGen particle decay simulation package," *Nucl. Instrum. Methods Phys. Res. A* **462** (2001) 152–155.

[20] T. Sjöstrand *et al.*, "An introduction to PYTHIA 8.2," *Computer Physics Communications* **191** (2015) 159–177.

[21] S. Jadach, B. F. L. Ward, and Z. Wąs, "The precision Monte Carlo event generator KK for two-fermion final states in $e^+e^-$ collisions," *Computer Physics Communications* **130** (2000) 260–325.

[22] "Kaon." https://en.wikipedia.org/wiki/Kaon. Last accessed 15. August 2024.

[23] "pdgLive - Strange Mesons." https://pdglive.lbl.gov/ParticleGroup.action?init=0&node=MXXX020. Last accessed 15. August 2024.

[24] W. D. Hulsbergen, "Decay chain fitting with a Kalman filter," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **552** no. 3, (Nov., 2005) 566–575.

[25] "zfit documentation." https://zfit.readthedocs.io/en/stable/index.html. Last accessed 9. August 2024.

[26] "kafe2 - Theoretical Foundation." https://etpwww.etp.kit.edu/~quast/kafe2/old/htmldoc/parts/mathematical_foundations.html. Last accessed 12. August 2024.

[27] F. James, "MINUIT - Function Minimization and Error Analysis," 1994. https://root.cern.ch/download/minuit.pdf. Last accessed 25. September 2024.