Measurement of the width of $D_{s0}^*(2317)$ in $\overline{\mathrm{p}}\mathrm{p} \rightarrow D_s^{\pm} D_{s0}^*(2317)^{\mp}$ reactions close to production threshold

K. Götzen

July 25, 2008

Abstract

In the last couple of years various charmonium like states as well as open charm systems have been discovered mainly by the B-Factories, where many of the resonances properties are only known to poor accuracy. In order to understand the true nature of these states higher precision for the essential quantities like mass, width and angular momenta is mandatory. The aim of this analysis performed on simulated data for the PANDA experiment is to reconstruct events of the reaction $\overline{\mathrm{p}}\mathrm{p} \rightarrow D_s^{\pm} D_{s0}^*(2317)^{\mp}$ and determine the width of the recoiling $D_{s0}^*(2317)^{\pm}$ by measuring the line shape of the energy dependent production cross section close to threshold.

1 Motivation

2 Analysis Strategy

Main goal of this analysis is the measurement of the width Γ of the $D_{s0}^*(2317)$. Therefore the analysis generally has to be performed in two separate steps:

1. Reconstruction of the signal:
   - determination of efficiency of signal
   - estimate background level (signal to noise ratio S/N)
2. Simulation of energy scan:
   - generation of expected obeservable distributions according to results from above
   - determination of the line shape of the exitation function
   - measurement of width and mass of $D_{s0}^*(2317)$

3 Reconstruction of Signal

3.1 Inclusive reconstruction

Figure of merit is the detection of the number reactions of the type

$$\overline{\mathrm{p}}\mathrm{p} \rightarrow D_s^{\pm} D_{s0}^*(2317)^{\mp} \tag{1}$$

to finally determine the energy dependent production cross section for this channel.

A straight forward analysis strategy usually is a full exclusive reconstruction of the signal in a particular decay chain. Nevertheless this approach is not persued here because of the arguments given below.

First of all one has to make a choice about the reconstructed decay channel. The $D_s^{\pm}$ meson has a lot of possible decays. One of them known to be reconstructable with a reasonable efficiency is $D_s^{\pm} \to \phi\pi^{\pm}$ with $\phi \to K^+K^-$ with the combined branching ratio

$$f_{\mathcal{B},D_s} = \mathcal{B}(D_s^{\pm} \to \phi\pi^{\pm}) \cdot \mathcal{B}(\phi \to K^+K^-) = 0.044 \cdot 0.492 = 0.022. \quad (2)$$

Since the only known decay channel of $D_{s0}^*(2317)$ is the isospin violating $D_{s0}^*(2317) \to D_s^{\pm}\pi^0$ (with unknown branching fraction) one also needs to reconstruct the $\pi^0 \to 2\gamma$ and the second $D_s^{\pm}$ in the above channel for a full exclusive reconstruction resulting in the total branching ratio factor

$$f_{\mathcal{B},\text{excl}} = \underbrace{\mathcal{B}(D_{s0}^*(2317) \to D_s\pi^0)}_{\text{unknown!}} \cdot \mathcal{B}(D_s^{\pm} \to \phi\pi^{\pm})^2 \cdot \mathcal{B}(\phi \to K^+K^-)^2 \cdot \mathcal{B}(\pi^0 \to 2\gamma) < 4.6 \cdot 10^{-4}. \quad (3)$$

Under the asumption of a signal cross section in the order of $\sigma = 1$nb at threshold, an integrated luminosity of about $\mathcal{L} = 9000$/nb per day and efficiency $\epsilon \approx 0.2$ this leads to an expected number of efficiency and branching ratio corrected signal reactions of

$$N_{\text{excl}} = \sigma \cdot \mathcal{L} \cdot \epsilon \cdot f_{\mathcal{B},\text{excl}} < 9000 \cdot 0.3 \cdot 4.6 \cdot 10^{-4} = 0.8 \text{ detected signals/day.} \quad (4)$$

Even with reasonable low background level this would reflect in at least a couple of hundreds of days beam time for this particular measurement which is for sure inacceptable.

The strategy followed here instead is an inclusive reconstruction of only the recoiling $D_s^{\pm}$. To ensure that the $D_s^{\pm}$ really recoils of a $D_{s0}^*(2317)$ kinematic correlations in the event are exploited which will be discussed in detail later. In this case the expected number of reactions which can be detected can be estimated to

$$N_{\text{incl}} = \sigma \cdot \mathcal{L} \cdot \epsilon \cdot f_{\mathcal{B},D_s} < 9000 \cdot 0.2 \cdot 0.022 = 40 \text{ detected signals/day,} \quad (5)$$

so to collect a reasonable number like 500 events only around two weeks beam time would be required. In addition the anticipated efficiency will most probable be higher since only the three tracks from the $D_s^{\pm}$ decay have to be detected in the event.

3.2 Simulation and Datasets

The results presented in this writeup are based on simulations performed with the BaBar like software framework [?] which then have been analyzed with the framework internal analysis toolset `Simple Compositions`.

Since a true energy scan cannot be performed in the simulation due to inaccuracies of the generator for production below threshold all events have been generated ≈ 5 MeV above threshold at

$$\sqrt{s} = c^2 \cdot [m(D_s) + m(D_{s0}^*(2317))] + 5 \text{ Mev } = (1968.5 + 2317.3 + 5) \text{ MeV } \approx 4291 \text{ MeV} \quad (6)$$

under the assumption that efficiencies as well as background will be constant for the scanned region. The values determined will therefore be used as constant input to the determination of the width $\Gamma(D_{s0}^*(2317))$.

Signal events have been generated which the event generator `EvtGen`[?] with the intrinsic width of the $D^*_{s0}(2317)$ set to $\Gamma = 0.1$ MeV. To account for the inclusive reconstruction, half of the events (=Signal 1) were generated in the following way:

$$\bar{p}p \rightarrow D_s^{\pm} D^*_{s0}(2317)^{\mp} \tag{7}$$
$$D_s^{\pm} \rightarrow \phi\pi^{\pm}, \quad \phi \rightarrow K^+K^- \tag{8}$$
$$D^*_{s0}(2317)^{\mp} \rightarrow \text{anything} \tag{9}$$

The second half of the signals (=Signal 2) was generated completely inclusive, i.e.

$$\bar{p}p \rightarrow D_s^{\pm} D^*_{s0}(2317)^{\mp} \tag{10}$$
$$D_s^{\pm} \rightarrow \text{anything} \tag{11}$$
$$D^*_{s0}(2317)^{\mp} \rightarrow \text{anything} \tag{12}$$

to cross check the efficiency determined with the dataset above.

In order to estimate the background level several specific decay channels with similar kinematics have been investigated. In particular all channels have also a recoiling D_s meson decaying to the same final state like those in the signal events. Considered have been channels which comprise a second (non-resonant produced) D_s or D_s^* together with light mesons like one or two pions or gammas to fill up the available phase space of

$$\sqrt{s} - 2 \cdot m(D_s) \cdot c^2 = 4291 \text{ MeV } - 2 \cdot 1968.5 \text{ MeV } = 354 \text{ MeV}. \tag{13}$$

In addition to that generic hadronic background produced with the event generator `DpmGen` based on the dual parton model [?] has been analyzed. Table 1 summarizes the datasets considered.

Channel	Number of events
$\bar{p}p \rightarrow D_s^{\pm} D^*_{s0}(2317)^{\mp}$ (Signal 1)	40 000
$\bar{p}p \rightarrow D_s^{\pm} D^*_{s0}(2317)^{\mp}$, $D_s^{\pm} \rightarrow$ any (Signal 2)	40 000
$\bar{p}p \rightarrow D_s^{\pm} D_s^{\mp} \pi^0$	40 000
$\bar{p}p \rightarrow D_s^{\pm} D_s^{\mp} 2\pi^0$	40 000
$\bar{p}p \rightarrow D_s^{\pm} D_s^{\mp} \pi^+\pi^-$	40 000
$\bar{p}p \rightarrow D_s^{\pm} D_s^{*\mp}$	40 000
$\bar{p}p \rightarrow D_s^{\pm} D_s^{*\mp} \pi^0$	40 000
$\bar{p}p \rightarrow D_s^{\pm} D_s^{\mp} \gamma$	40 000
$\bar{p}p \rightarrow D_s^{\pm} D_s^{*\mp} \gamma$	40 000
DPM generic	5 000 000

Table 1: Datasets

3.3 Selection

The first step on the way to determine the number of signal reactions the $D_s^{\pm}$ mesons have to be reconstructed and isolated from background as good as possible. The procedure to create $D_s^{\pm}$ candidates was:

1. Select kaon candidates from charged tracks with `veryLoose` PID criterion[1] (will be tightend later for finding an optimum)

[1] PID selection is based on a global likelihood function LH. Available criteria are: `veryLoose` ($LH > 0.2$), `loose` ($LH > 0.8$), `tight` ($LH > 0.95$), `veryTight` ($LH > 0.99$)

2. Create a list of ϕ candidates by forming all combinations of a negative with a positive charged kaon candidate
3. Kinematic fit of the single ϕ candidates with vertex constraint[2]
4. Select pion candidates from charged tracks with `veryLoose` PID criterion
5. Combine ϕ candidates with pion candidates to form $D_s^\pm$ candidates
6. Kinematic fit of the $D_s^\pm$ candidates with vertex constraint

On the so preselected candidates the following requirements have been applied in addition:

1. Probability of ϕ vertex fit: $P_\phi > 0.001$
2. Probability of $D_s^\pm$ vertex fit: $P_{Ds} > 0.001$
3. ϕ mass window: $|m(K^+K^-) - m_{\mathrm{PDG}}(\phi)| < 10$ MeV/c^2
4. ϕ decay angle[3]: $|\theta_{\mathrm{dec}}| > 0.5$
5. $D_s^\pm$ mass window: $|m(\phi\pi^\pm) - m_{\mathrm{PDG}}(D_s^\pm)| < 30$ MeV/c^2

Fig. 1 shows the corresponding distributions according to the upper selection criteria for the particular case of very loose kaon selection.

Plot (a) shows the spectrum of the invariant K^+K^- mass, in (b) the invariant $K^+K^-\pi^\pm$ mass is displayed, which is in fact the mass of the $D_s^\pm$ candidates. In plot (c) the decay angle of the ϕ candidates is shown. As expected in a decay of a polarized[4] vector meson into two pseudoscalar mesons the distribution of $\cos(\theta_{\mathrm{dec}})$ follows $f(x) = a \cdot x^2$. Finally figure (d) shows the missing mass of the recoil $D_s^\pm$ which is defined as

$$m_{\mathrm{miss}} = |P_{\mathrm{init}} - P_{Ds}| \tag{14}$$

where P_{Ds} denotes the 4-momentum of the D_s candidate and P_{init} the 4-momentum of the initial $\bar{\mathrm{p}}\mathrm{p}$ system given by

$$P_{\mathrm{init}} = (p_x, p_y, p_z, E \cdot c) = (0, 0, 8824, 9812) \text{ MeV} \tag{15}$$

for the chosen center-of-mass energy of $E_{\mathrm{cms}} = 4291$ MeV, assuming the beam going in positive z direction. As expected a peak around $m = 2317$ MeV/c^2 appears matching the mass of the $D_{s0}^*(2317)$.

The black crosses in the figures represent all reconstructed candidates whereas the shade area corresponds to candidates failing the so called Monte Carlo Truth (MTC) match[5] The slight peaking originates from $D_s^\pm$'s resp. ϕ's appearing in the $D_{s0}^*(2317)$ decay tree which by chance decay to the same channel $D_s^\pm \to \phi\pi^\pm$ or $\phi \to K^+K^-$ as in the signal. The MCT match of

[2] vertex constraint: fit candidates under assumption, that all daughter trajectories originate from a common point in space time.

[3] The decay angle is defined as the angle between the direction of motion of the reconstructed ϕ candidate in the laboratory frame and the direction of motion of one of the kaons in the frame of the ϕ.

[4] Since the ϕ originates from a decay of a pseudoscaler (the D_s) itself together with another pseudoscalar (the $\pi^\pm$), its spin orientation is polarized.

[5] The Monte Carlo Truth match is the check whether a decay tree has been exactly reconstructed as it was generated.

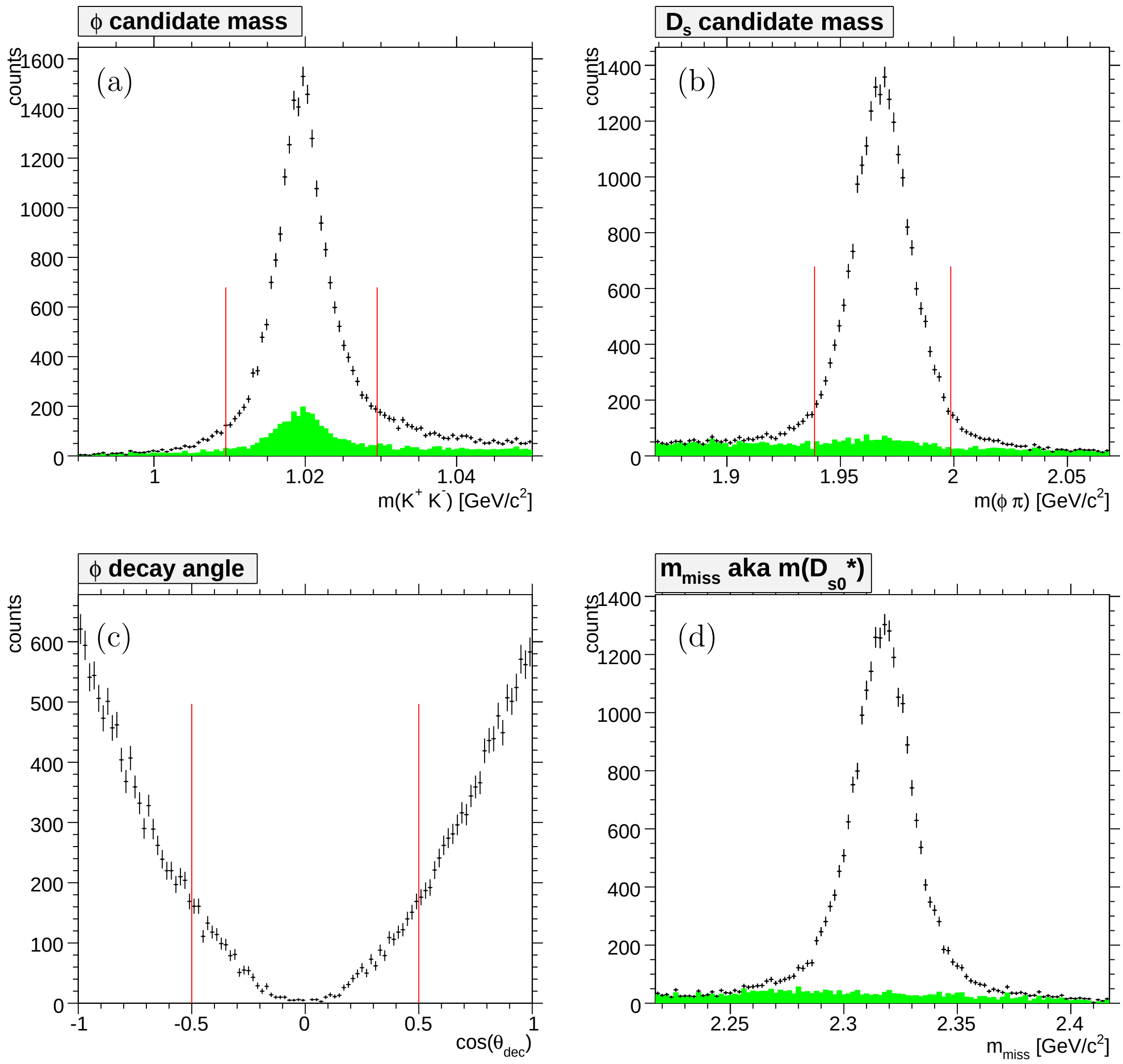

Figure 1: Distributions for reconstructed signal events of type 1 (Signal 1; see text). (a) Spectrum of the invariant mass $m(K^+K^-)$, (b) invariant mass $m(K^+K^-\pi^\pm)$, (c) distribution $\cos(\theta_{\text{dec}})$ of the ϕ candidates' decay angle, (d) missing mass for the $D_s^\pm$ candidates according to the initial 4-momentum of the $\bar{p}p$ system. Black histograms correspond to all reconstructed combinations, the shaded area represents combinations failing the MCT match (see text).

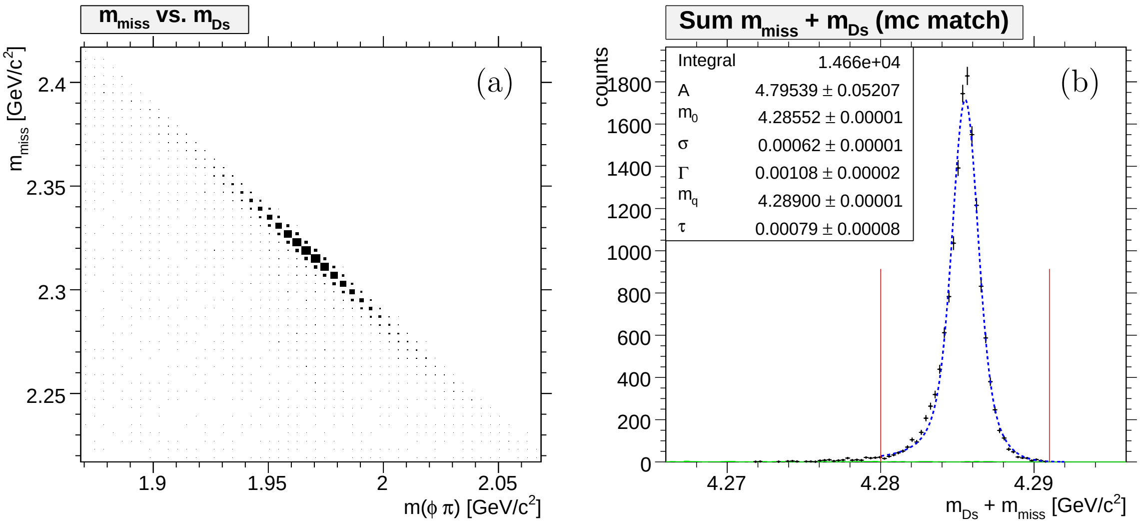

Figure 2: Reconstruction of signal events type 1. (a) Correlation of missing mass and invariant $D_s^\pm$ mass. (b) The sum $m_{\text{miss}} + m(D_s^\pm)$ as observable for counting signal events.

course only accepts correctly reconstructed $D_s^\pm$ from the recoil side (so that the rest of the event really corresponds to an unreconstructed $D_{s0}^*(2317)$ decay.) The red vertical lines symbolize the selection requirement defined in the list above.

It is clearly visible that the signal events can be successfully reconstructed on one side, on the other side the resolutions of all appearing signal peaks is of the order 10-15 MeV/c^2 which will make background suppression difficult due to wide mass windows. To enhance to signal to noise ratio when counting the number of signal events, the kinematic correlation of the missing mass and the mass of the $D_s^\pm$ candidate is exploited.

When in general performing exclusive analysis, the signal quality can be enhanced significantly by performing a so called 4 constraint fit to the reconstructed decay tree. This takes into account that the sum of all 4 momenta in the decay tree have to add up to the 4 momentum of the intial system which is basically defined by the precisely know beam energy[6].

Since this measurement is based on inclusive reconstructed D_s mesons it is not possible to fit with 4 constraints because not all particles in the event have been reconstructed. Nevertheless what is the case is that the (still well known) initial 4 momentum is splitting into the two particles D_s and $D_{s0}^*(2317)$ very close to their production threshold. When computing the invariant mass $m(D_s)$ with e.g. a strong deviation to higher or lower values due to poor resolution, the mass $m(D_{s0}^*(2317))$ which is actually computed as the missing mass $m(D_{s0}^*(2317)) \equiv m_{\text{miss}} = |P_{\text{init}} - P_{Ds}|$ has basically the same deviation to the opposite direction.

Fig. 2 (a) shows how the correlation of $m(D_s)$ and m_{miss} looks like.

Both the missing mass and the invariant mass $m(D_s^\pm)$ have a relative poor resolution whereas the correlation of those two is big which results in a very narrow ellipsis. This can be exploited by considering the sum $m_{\text{sum}} = m_{\text{miss}} + m(D_s^\pm)$ which corresponds to a projection of the 2 dimensional plot to a decreasing 45° diagonal or $f(x) = -x$. This leads to plot (b) exhibiting a very narrow peak with a resolution of the order ≈ 1 MeV/c^2. Therefore during this analysis

[6]Depending on the HESR operating mode the projected beam resolution will be either $dp/p = 10^{-4}$ or 10^{-5}.

above sum of masses m_{sum} will be considered as the observable to count the number of signal events. It will be demonstrated later that background channels show a different behaviour and therefore can be reasonably well separated from signal in this projection. In particular there is almost no combinatoric background within signal events reflected by the almost invisible shaded area correspondig to canidates with MCT match failed.

In Fig. 2 (b) an additional line shape fit has been performend. The partly empirical fit model is chosen to be a convolution of a ordinary (non-relativistic) Breit Wigner function $BW(m)$ with a Gaussian $G(m)$[7] with a Fermi like damping:

$$f(m) = A \cdot \left[\int_{-\infty}^{+\infty} G(x'; m_0, \sigma) * BW(m - x'; m_0, \Gamma)\, dx' \right] \cdot \frac{1}{1 + \exp\left(\frac{m - m_q}{\tau}\right)} \tag{16}$$

with intensity A, running variable m, resonance pole mass m_0, resonance width Γ, reconstruction resolution σ, phase space limit m_q and decay parameter τ (all except A in [GeV/c^2]).

The goodness of the fit (i. e. appropriateness of fit model) is of no relevance for this particular study since event counts for efficiency determination can be extracted by counting histogram entries due to absence of background. Therefore a signal region is defined by

$$4280\ \text{MeV}/c^2 < m_{\text{sum}} < 4291\ \text{MeV}/c^2 \tag{17}$$

which is marked by the two vertical lines in Fig. 2 (b). This region will be used to count numbers of signals as well as residual background candidates for the purpose of computing efficiencies.

For the particular example here with very loose kaon identification we find $S = 14490$ entries in the signal region corresponding to an efficiency of $\epsilon = 36.2\%$

As a cross check for the signal quality as well as background level arising within signal events the same analysis chain has been applied to totally inclusive signal events (Signal 2; see above), where also the recoiling D_s decays generically. Under the assumption of the same efficiency as above one expects for this configuration a number $S'_{exp} \approx S \cdot f_{Ds} = 314$ entries in the signal box. Counting in Fig. 3 (f) results in $S' = 320$ which agrees very well with the expectation. It can be concluded that there is no systematic effect in efficiency measurements due to the specific decay channel of the $D_s^{\pm}$ meson in signal events.

[7] Voigtian distribution: Convolution of a Gaussian with a Breit-Wigner, also known as real part of the Faddeeva function resp. complex error function [?].

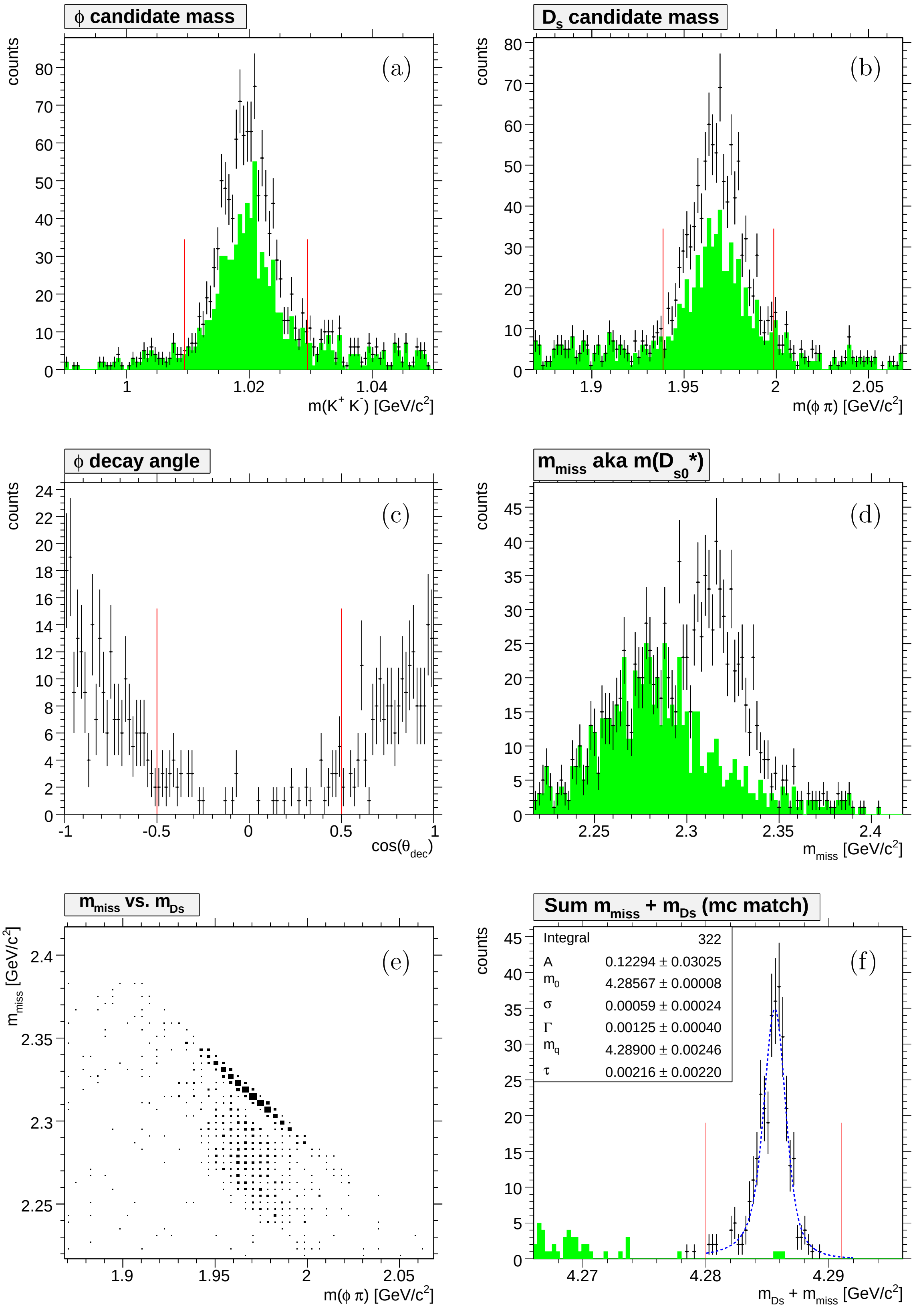

Figure 3: Reconstruction of full inclusive signal events. Description of plots: see Fig. 1 and Fig. 2

.

3.4 Backgrounds

This sections shows how much residual background is populating the signal region. The plots here are shown for `veryLoose` kaon identification like for the signal channels. The summary of the optimization for different identification criteria can be found in Table 2. The distributions shown for all background channels in Reffigfig:bg1 – Fig. 11 are

1. D_s candidate mass $m(\phi\pi)$ (upper left)
2. missing mass m_{miss} (upper right)
3. 2-dimensional correlation of latter two (lower left)
4. the sum m_{sum} of both (lower right)

Fig. 4 – Fig. 10 show the distributions for the specific channels, whereas Fig. 11 show the results for generic background events generated with the DPM generator. It can be seen clearly that the missing mass is shaped either as a step function or exhibits a peaking behaviour for the specific background channels due to the limited available phase space, resulting in the correlation plots as a blop in the signal region.

This also emphasizes the choice of m_{sum} as a good observable, since the distribution follows a phase space background shape for all channels. The background model chosen to determine background levels is the so called Argus function[**?**] defined by

$$f_{\text{bg}}(m) = A_s \cdot m \cdot \sqrt{1-(m/m_0)^2} \cdot \exp\left[c \cdot \left(1-(m/m_0)^2\right)\right] \tag{18}$$

with amplitude parameter A_s, phase space limit m_0 and shape parameter c. Again the shaded area in the plots corresponds to candidates failing the Monte Carlo Truth match. Certainly in Fig. 4 – Fig. 10 most of the candidates are reconstructed correctly since there are 'true' recoil $D_s^{\pm}$. The DPM generator is know to produce an unrealistic low level of charm quarks reflected in the complete absence of correctly reconstructed $D_s^{\pm}$ mesons in Fig. 11.

3.4.1 BG 1: $\bar{p}p \rightarrow D_s^{\pm} D_s^{\mp} \pi^0$

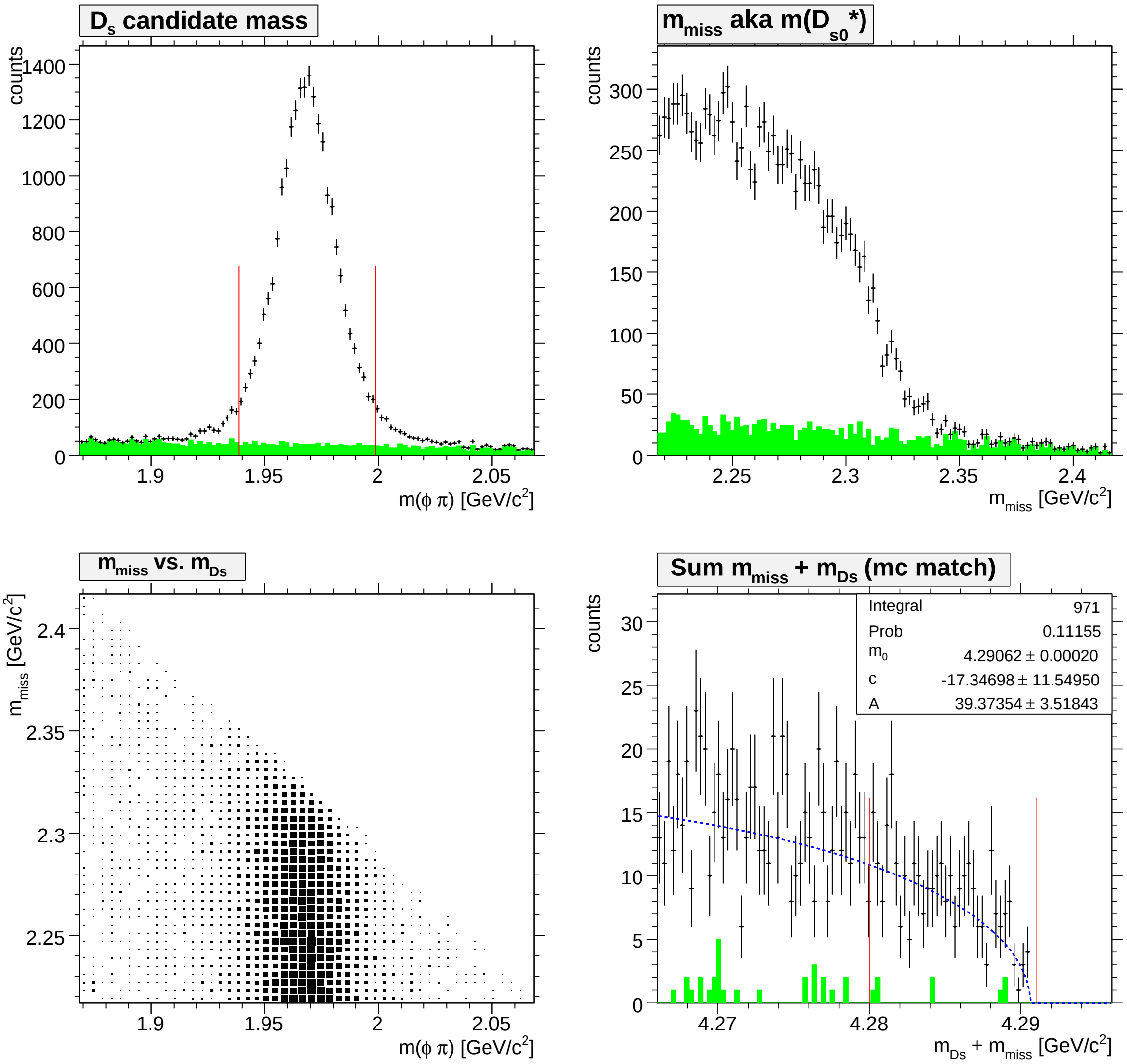

Figure 4: Background channel 1: $\bar{p}p \rightarrow D_s^{\pm} D_s^{\mp} \pi^0$

3.4.2 BG 2: $\bar{p}p \rightarrow D_s^{\pm} D_s^{\mp} 2\pi^0$

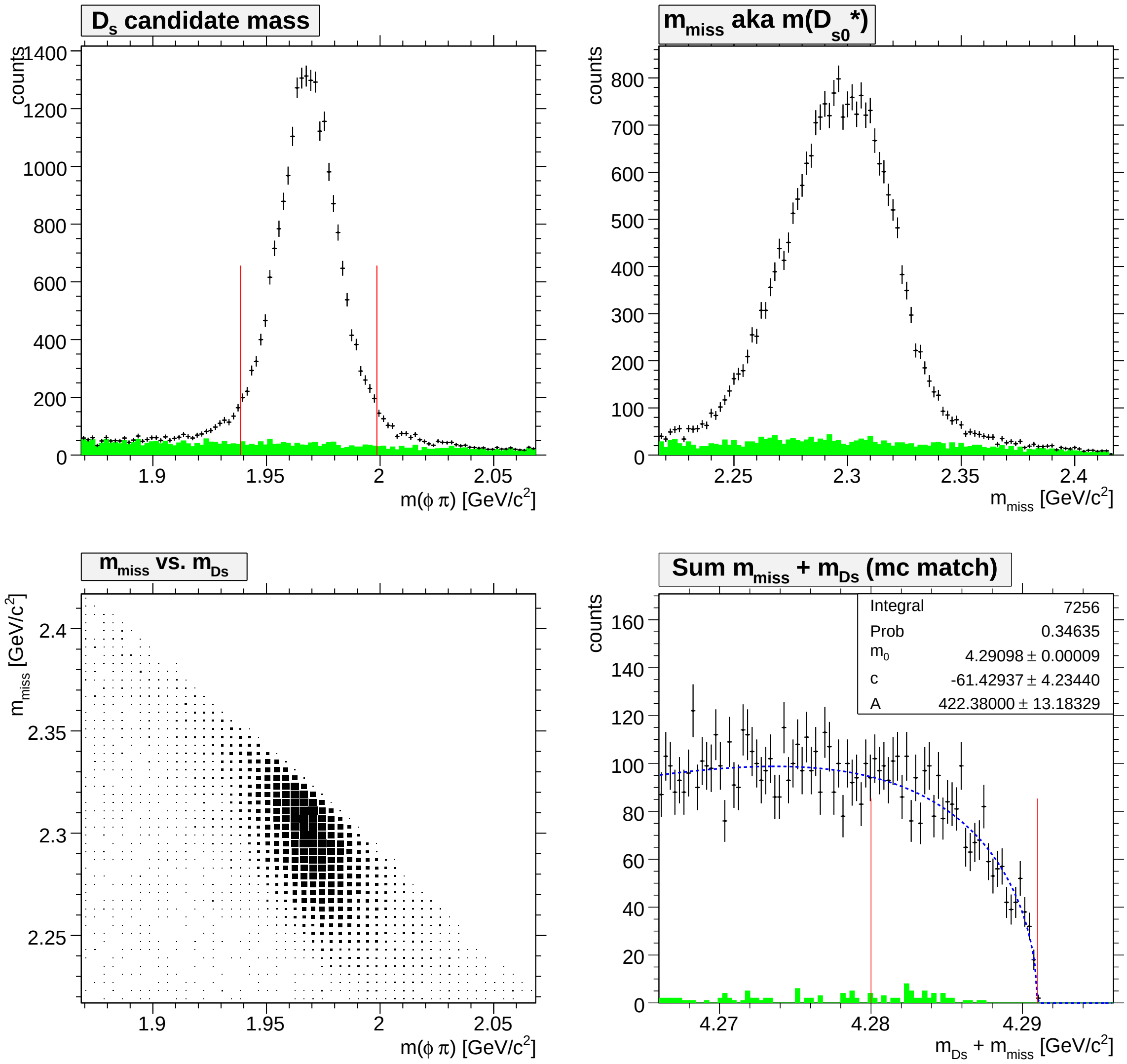

Figure 5: Background channel 2: $\bar{p}p \rightarrow D_s^{\pm} D_s^{\mp} 2\pi^0$

3.4.3 BG 3: $\bar{p}p \to D_s^{\pm} D_s^{\mp} \pi^+ \pi^-$

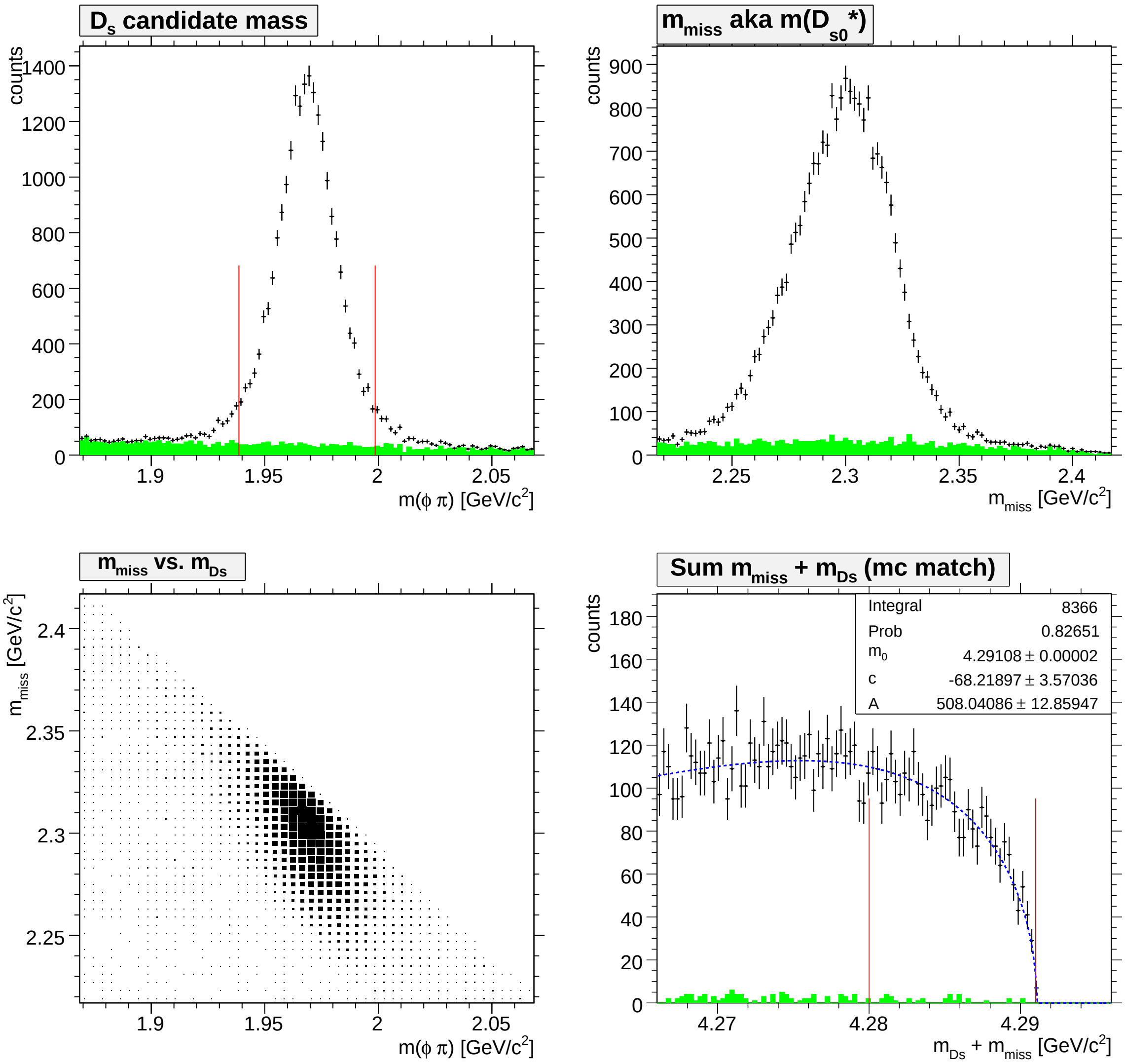

Figure 6: Background channel 3: $\bar{p}p \to D_s^{\pm} D_s^{\mp} \pi^+ \pi^-$

3.4.4 BG 4: $\bar{p}p \rightarrow D_s^{\pm} D_s^{*\mp}$

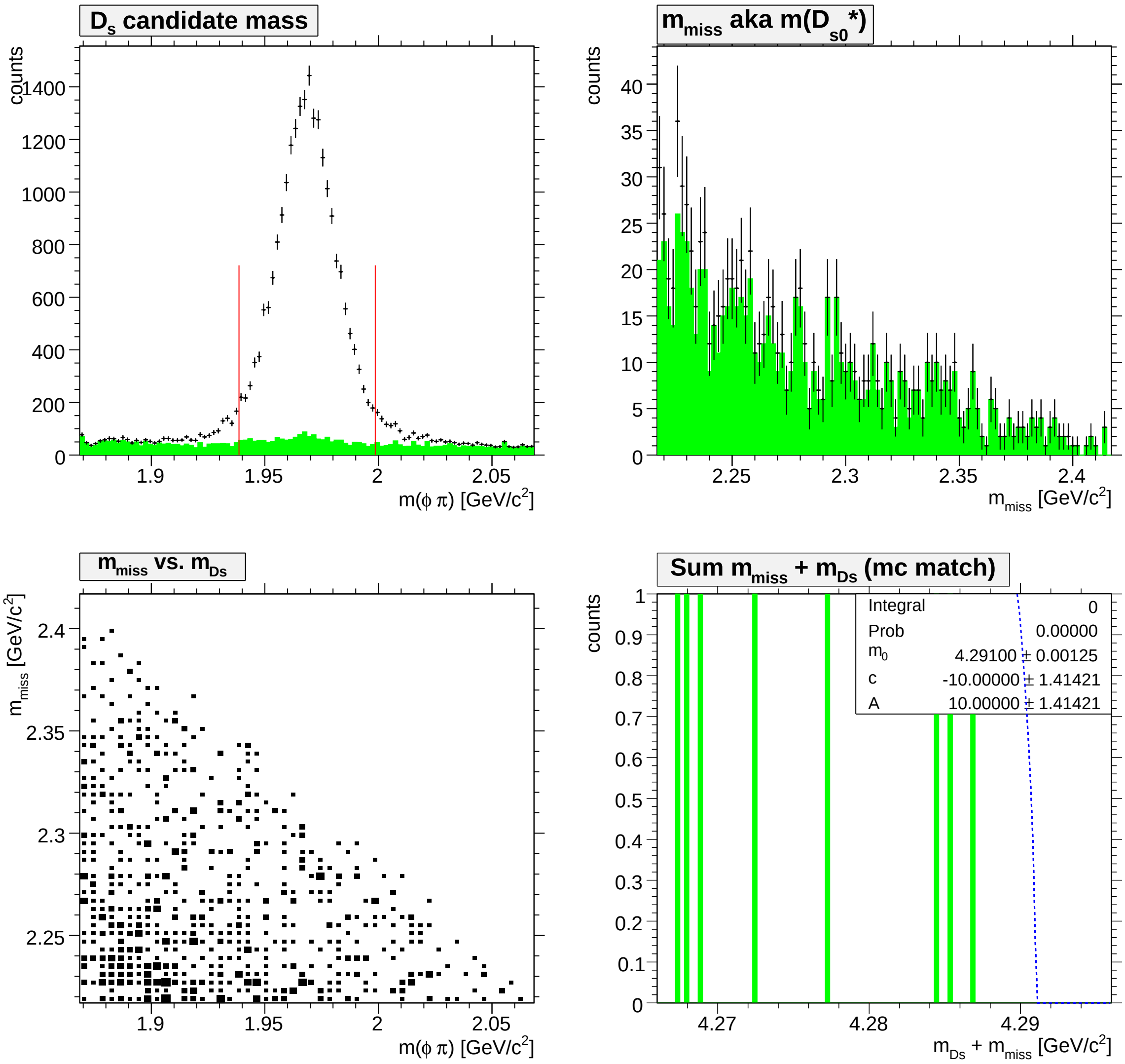

Figure 7: Background channel 4: $\bar{p}p \rightarrow D_s^{\pm} D_s^{*\mp}$

3.4.5 BG 5: $\bar{p}p \to D_s^{\pm} D_s^{*\mp} \pi^0$

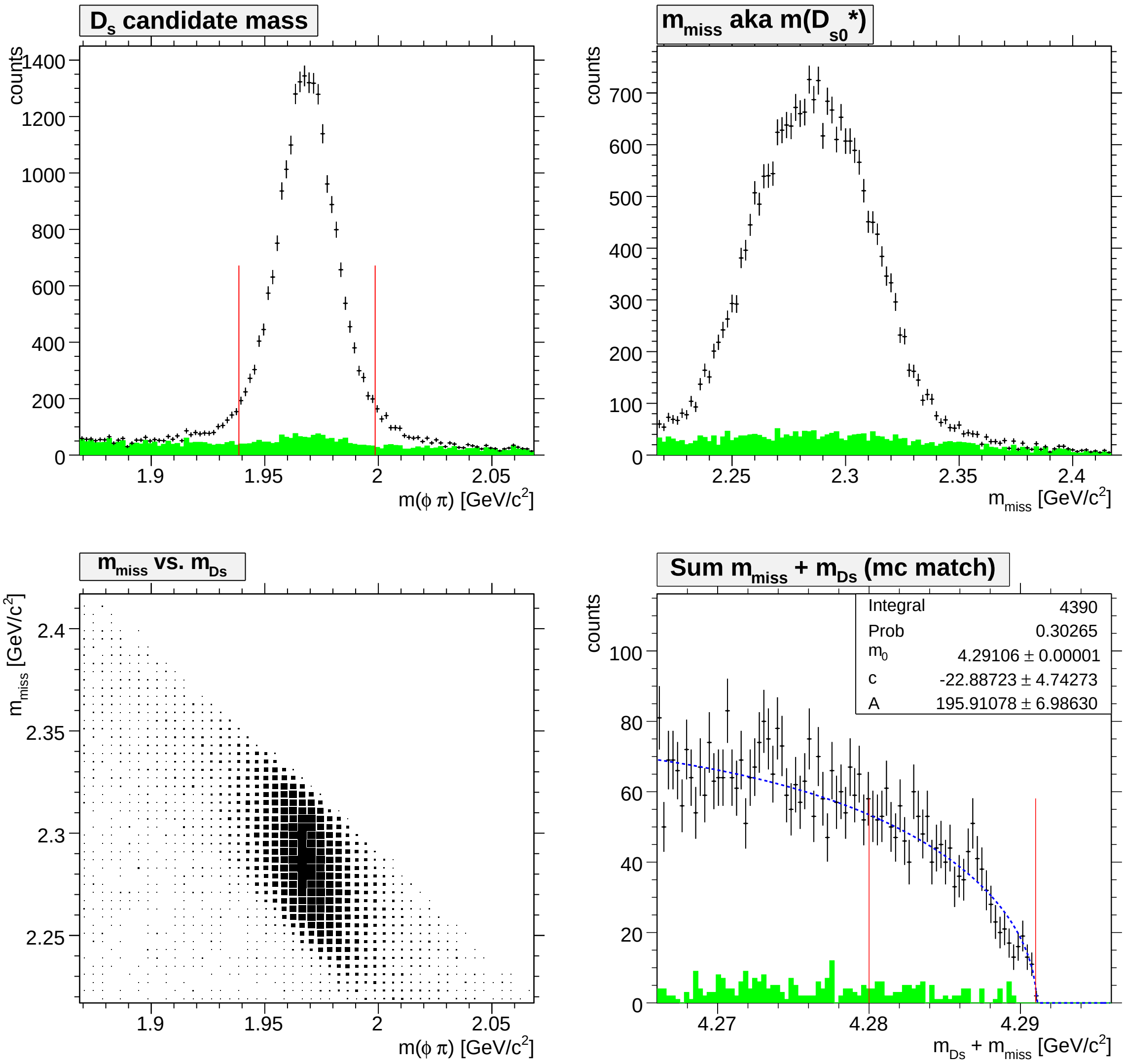

Figure 8: Background channel 5: $\bar{p}p \to D_s^{\pm} D_s^{*\mp} \pi^0$

3.4.6 BG 6: $\bar{p}p \to D_s^{\pm} D_s^{\mp} \gamma$

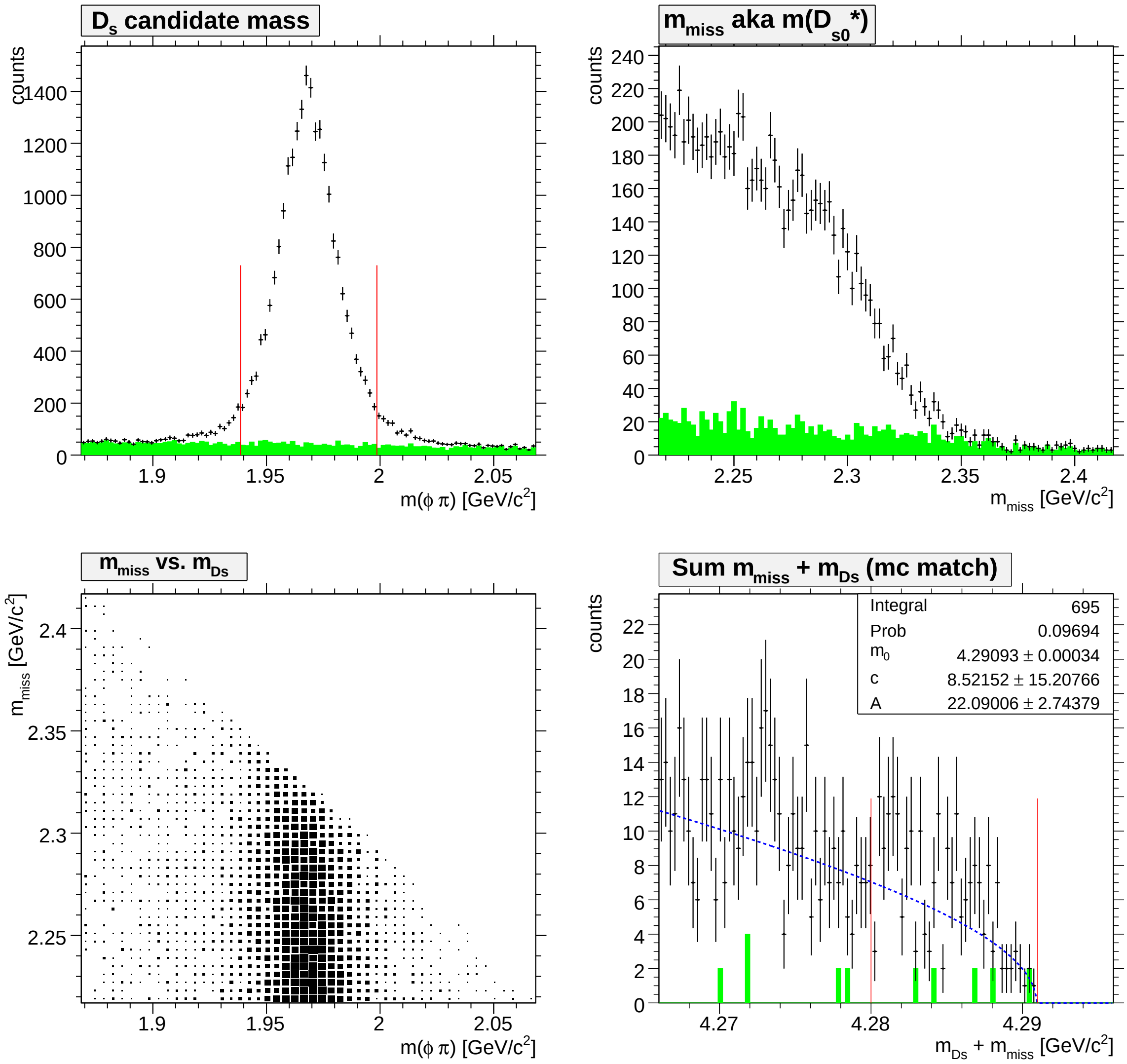

Figure 9: Background channel 6: $\bar{p}p \to D_s^{\pm} D_s^{\mp} \gamma$

3.4.7 BG 7: $\bar{p}p \rightarrow D_s^{\pm} D_s^{*\mp} \gamma$

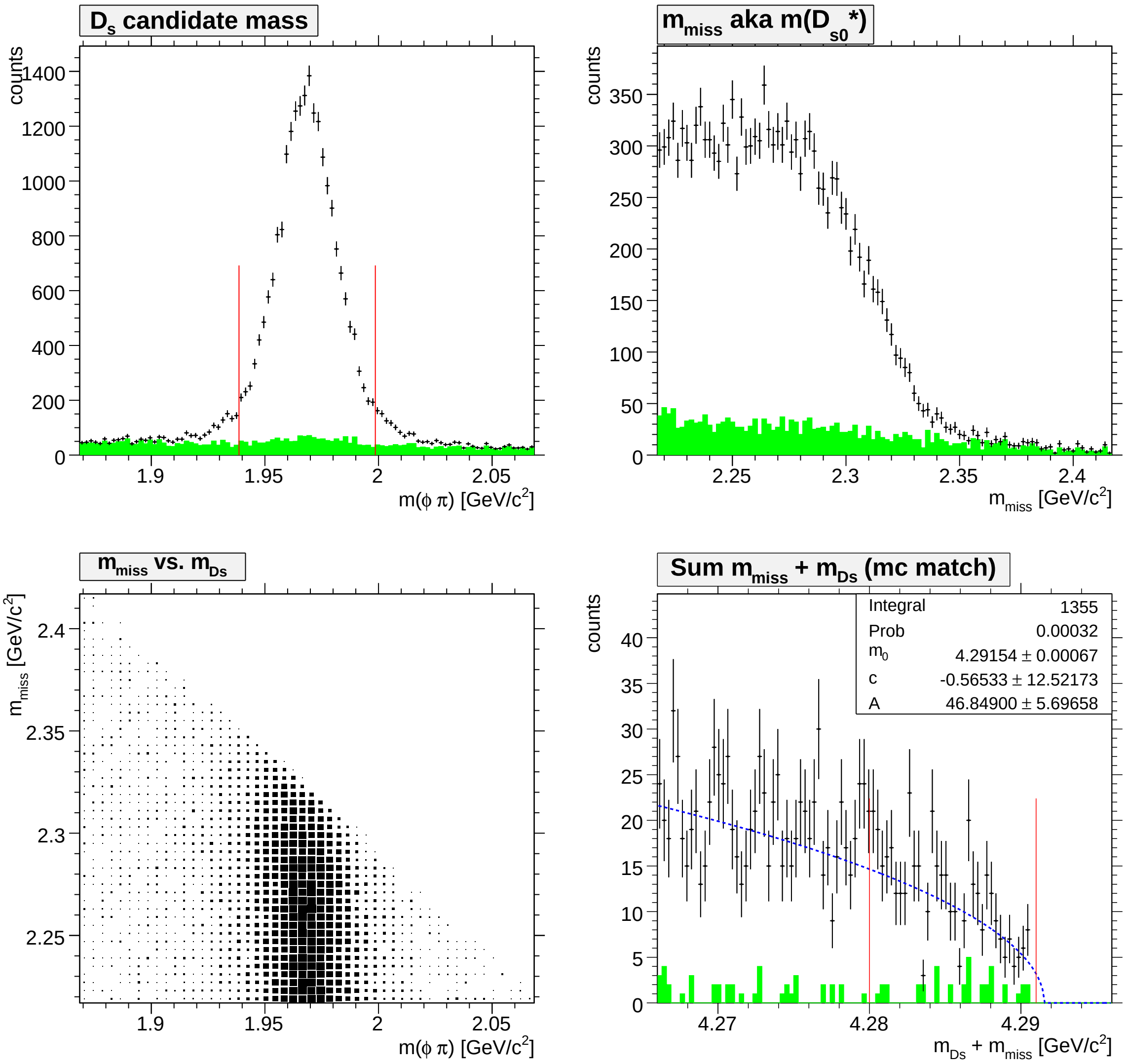

Figure 10: Background channel 7: $\bar{p}p \rightarrow D_s^{\pm} D_s^{*\mp} \gamma$

3.4.8 BG 8: DPM generic

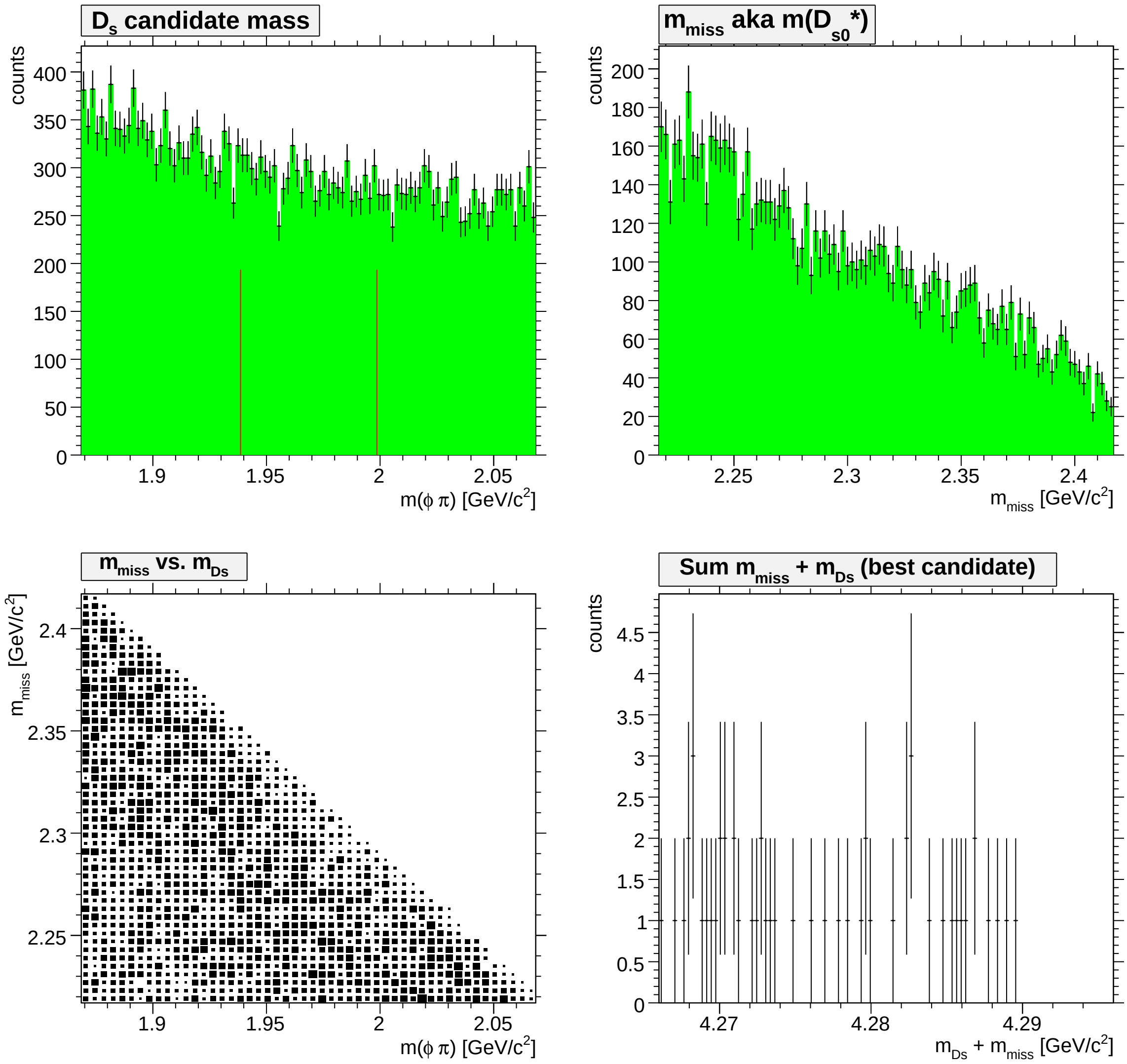

Figure 11: Background channel 8: Generic hadronic background (DPM generator)

3.5 PID Optimization

In principle to find an optimal selection for supressing background and thus optimizing the significance of the signal there are a lot of parameters like all the kinematic cuts and mass windows, which have to be varied. Up to now this procedure has only been performed for the kaon identification quality.

There are four different selection criteria available based on a global PID likelihood function LH which are:

- very loose (VL): $LH > 0.2$
- loose (L): $LH > 0.8$
- tight (T): $LH > 0.9$
- very tight (VT) : $LH > 0.95$

All these four criteria for kaon identification have been applied to the selection procedure and the number of residual candidates for signal and the different backgrounds in the signal region Eq. 17 of m_{sum} have been counted. Assuming values for the cross sections of these backgrounds relative to that of the signal one can compute the expected signal-to-noise ratio r_{SN} depending on PID.

For all hadronic modes the cross sections are assumed to be in the same order of magnitude, the electromagnetic modes involving a γ were arbitrarily scaled down by a factor 10 (factor α/α_s perhaps would have been more realistic but neither is known exactly nor seems to be of big relevance). The relative cross section of generic backgrounds here is ad hoc assumed to be a factor 10^6 higher than the signal. This assumption can be significantly wrong. The total cross section for $\overline{\mathrm{p}}\mathrm{p}$ collisions in the energy region relevant for the analysis performed here according to Eq. 15 is of the order

$$\sigma(\overline{\mathrm{p}}\mathrm{p} \to \text{ anything}) \approx 60 \text{ mb} \tag{19}$$

which would lead to a relative cross section of

$$\frac{\sigma(\overline{\mathrm{p}}\mathrm{p} \to \text{ anything})}{\sigma(\overline{\mathrm{p}}\mathrm{p} \to D_s D_{s0}^*(2317))} \approx \frac{60 \text{ mb}}{1 \text{ nb}} = 6 \cdot 10^8 \tag{20}$$

resulting in a modified estimate for the signal-to-noise ratio in the next to last line of Table 2 in the 5th column (ϵ(T)[%]) of $r_{SN} > 1 : 27000$ which is quite a limited statement. Thus in order to improve the conclusive power one would need at least a factor of 1000-10000 more generic events which is unfeasible for the time being.

The results of this study, i.e. the effciencies are given in Table 2. In cases where no candidate was reconstructed in the signal region, the efficiency values are given as an upper limit computed with the assumption that one entry would have been observed.

It is obvious that the expected signal-to-noise ratio will be completely governed by the generic hadronic background since the expected values for r_{SN} in case of ignoring the contributions from the DPM generator are constant at about $r_{SN} \approx 2$. Due to limited Monte Carlo statistics for the generic background it cannot be made a strong statement about feasibility or a realistic signal-to-noise ratio so far.

Apparently there nevertheless is an optimum for kaon identification with criterion `tight` indicating r_{SN} better than 1:45 for the current scenario.

Channel	rel. X-sec	ϵ(VL)[%]	ϵ(L)[%]	ϵ(T)[%]	ϵ(VT)[%]
Signal	1	36.2	28.1	21.0	19.0
$\bar{p}p \to D_s^{\pm} D_s^{\mp} \pi^0$	1	0.8	0.6	0.5	0.4
$\bar{p}p \to D_s^{\pm} D_s^{\mp} 2\pi^0$	1	6.9	5.2	4.0	3.6
$\bar{p}p \to D_s^{\pm} D_s^{\mp} \pi^+\pi^-$	1	8.1	6.1	4.6	4.2
$\bar{p}p \to D_s^{\pm} D_s^{*\mp}$	1	0.0	0.0	0.0	0.0
$\bar{p}p \to D_s^{\pm} D_s^{*\mp} \pi^0$	1	3.7	2.8	2.1	1.9
$\bar{p}p \to D_s^{\pm} D_s^{\mp} \gamma$	0.1	0.6	0.4	0.3	0.3
$\bar{p}p \to D_s^{\pm} D_s^{*\mp} \gamma$	0.1	1.1	0.9	0.6	0.6
DPM generic	10^6	$1.8 \cdot 10^{-2}$	$1.9 \cdot 10^{-3}$	$< 9.4 \cdot 10^{-4}$	$< 9.4 \cdot 10^{-4}$
r_{SN} (w/ DPM)	–	1 : 495	1 : 68	> 1 : 45	> 1 : 50
r_{SN} (w/o DPM)	–	1.86	1.90	1.89	1.88

Table 2: PID optimization summary

4 Simulation of Energy Scan

4.1 Excitation Function

Like mentioned in the introduction the observable to determine the properties of the $D_{s0}^*(2317)$ resonance (in particulay its width) basically is the dynamic behaviour of the cross section for its production together with a recoiling D_s meson close to the threshold energy $E_{\text{thr}} = [m(D_s^{\pm}) + m(D_{s0}^*(2317))] \cdot c^2$.

Generally the cross section for the production of two particles with spectral functions according to

$$\rho_i(m) = \frac{1}{\pi} \cdot \frac{\Gamma_i/2}{(m - m_{R_i})^2 + (\Gamma_i/2)^2} \tag{21}$$

with resonances pole mass and width m_{R_i} and Γ_i is given by the integral

$$\sigma(s) = |M|^2 \int_{-\infty}^{+\infty} dm_1 \int_{-\infty}^{+\infty} dm_2 \; \rho_1(m_1)\rho_2(m_2) \cdot p \cdot \Theta(\sqrt{s} - m_1 - m_2). \tag{22}$$

Here m_1 and m_2 are the running masses, $\sqrt{s}$ is the total center-of-mass energy, M the (usually unknown) matrix element of the scattering process and p the breakup momentum of the two resonances in the cms frame with

$$p^2(\sqrt{s}, m_1, m_2) = \frac{\left[s - (m_1 + m_2)^2\right] \cdot \left[s - (m_1 - m_2)^2\right]}{4s}. \tag{23}$$

For the process considered in this analysis the width of one of the resonances, namely for the D_s meson, can be taken as zero. Thus its spectral function degenerates to a delta function leading to the simplified integral (with $m_d \equiv m(D_s)$):

$$\begin{aligned} \frac{\sigma(s)}{|M|^2} &= \int_{-\infty}^{+\infty} dm_1 \int_{-\infty}^{+\infty} dm_2 \; \rho_1(m_1)\delta(m_2 - m_d) \cdot p \cdot \Theta(\sqrt{s} - m_1 - m_2) && (24) \\ &= \int_{-\infty}^{+\infty} dm \, \rho_1(m) \cdot p \cdot \Theta(\sqrt{s} - m - m_d) && (25) \\ &= \int_{-\infty}^{\sqrt{s}-m_d} dm \, \rho_1(m) \cdot p && (26) \end{aligned}$$

Substituting with the terms from Eq. 23 and Eq. 21 under the assumption that M is approximately constant in this energy region, the line shape of the cross section for this process looks like

$$\frac{\sigma(s)}{|M|^2} = \frac{\Gamma}{4\pi\sqrt{s}} \int_{-\infty}^{\sqrt{s}-m_d} dm\, \frac{\sqrt{[s-(m+m_d)^2]\cdot[s-(m-m_d)^2]}}{(m-m_R)^2+(\Gamma/2)^2}. \tag{27}$$

4.2 Scan Procedure

In reality the measurement reported about will be performed in the following (or a similar) way:

1. Data taking at n energy values around threshold energy $E_{\text{thr}} = [m(D_s^{\pm}) + m(D_{s0}^*(2317))] \cdot c^2$, e. g.

$$E_{\text{thr}} - \Delta E_{\max} < E + i\cdot\Delta E < E_{\text{thr}} + \Delta E_{\max}, \qquad 0 < i < n \tag{28}$$

   for step number i with step size ΔE and width of energy region $2\cdot\Delta E_{\max}$.

2. Perform selection and reconstruction resulting in a distribution m_{sum} consisting of signal events reflected in a resonance peak close to the phasespace limit and phasespace distributed background events.
3. Fit an appropriate model to the line shape, e.g. consisting of functions like Eq. 16 and Eq. 18 to describe the signal and background shape
4. Determine the number of signal events S_i for each of these mass histograms
5. Enter all the pairs (E_i, S_i) in a graph and fit the line shape of the resulting function of the energy dependent cross section described by Eq. 27
6. Calculate the mass and width $m(D_{s0}^*(2317)) \pm \Delta m(D_{s0}^*(2317))$ and $\Gamma_{Ds0} \pm \Delta\Gamma_{Ds0}$ from the fit parameters
7. Significance of this measurement is then given by $\Gamma/\Delta\Gamma$

Since there is no realible possibility to generate signal below threshold the scenario of an energy scan will be effectively simulated as realistic as possible by replacing steps 1. and 2. of the above procedure by 'guessing' an according spectrum of the mass m_{sum}:

i. Chose the following parameters (some reasonable values in paranthesis):

   - Width Γ_{Ds0} of the $D_{s0}^*(2317)$. (≈ 0.05 MeV – 3 MeV; PDG: $\Gamma < 3.8$ MeV, CL=95%)
   - Beam time T_{beam} for the complete measurement; assumptions are signal cross section of $\sigma_S = 1$ nb at threshold energy E_{thr} and an integrated luminosity of $\mathcal{L} = 9\,\text{pb}^{-1}/\text{day}$.
   - Energy region $\Delta E_{\max}$ for scan. ($\approx 2\Gamma - 3\Gamma$)
   - Number n of energy steps ($\rightarrow \Delta E = 2\cdot\Delta E_{\max}/n$) (7 – 15)
   - Signal-to-Noise ratio r_{SN}
   - Efficiency factor $\epsilon\cdot f_{\mathcal{B},D_s}$

ii. Compute the number of expected signal events at energy E_i as

$$S_i = \sigma(E_i/c^2) \cdot \frac{\mathcal{L}_{\text{tot}}}{n} \cdot \epsilon \cdot f_{\mathcal{B},D_s} = \sigma_S \cdot \frac{f_{\text{ex}}(E_i + \delta E)}{f_{\text{ex}}(E_{\text{thr}})} \cdot \mathcal{L} \cdot \frac{T_{\text{beam}}}{n} \cdot \epsilon \cdot f_{\mathcal{B},D_s} \qquad (29)$$

where δE is related to the uncertainty of the beam energy δp and is gaussian distributed around 0 with $\sigma_E = 150$ keV. [8]

iii. Compute the number of expected background events at energy E_i. In principle this number is assumed to be constant, but it has to be taken into account the shift of the phase space limit (i. e. the energy E_i) which will shift the position of the distribution relative to the signal. Therefore setup the background model function Eq. 18 with fixed (and more or less arbitrary) parameter values according to those found for the specific background channels, except parameter m_q, which has to be set to the energy E_i. The parameter r_{SN} defined above is assumed to apply for the highest energy value $E_n \equiv E_{\text{thr}} + \Delta E_{\text{max}}$ considered. Thus for E_n and the region defined in Eq. 17 with width ΔE_{SR} it is $B'_n = r_{SN} \cdot S_n$. For all lower energies $E_i < E_n$ the region $E_i - \Delta E_{SR} < m \cdot c^2 < E_i$ has to contain this number of entries. The number of entries B_i finally filled in the histogram with lower limit m_{min} is then given by

$$B_i = \frac{\displaystyle\int_{m_{\text{min}}}^{E_i/c^2} f_{\text{bg}}(m)\, dm}{\displaystyle\int_{(E_i - \Delta E_{SR})/c^2}^{E_i/c^2} f_{\text{bg}}(m)\, dm} \cdot r_{SN} \cdot S_n \qquad (30)$$

iv. For each energy E_i create a histogram and fill it with S_i randomly generated entries according to Eq. 16 and B_i generated entries according to Eq. 18.

v. Follow the procedure from above beginning at step 3.

Of course to do systematic studies for finding a global optimum of the procedure, there is high dimensional parameter space to be investigated. This has not been done up to now.

Fig. 12 shows the mass histograms as a result for the upper procedure with the parameters

- $T = 14$ d, $r_{SN} = 1/3$, $\Gamma = 1$ MeV$/c^2$, $\Delta E_{\text{max}} = 2$ MeV, $n = 12$,

with increasing energy from left to right and top to bottom.

In Fig. 13 (a) the generated excitation function is show with a fit revealing exactly the parameters which have been put in as a cross check. In Plot (b) the reconstructed excitation function consisting of the event numbers retrieved from the individual fits on the histograms shown in Fig. 12 is displayed. The fit in this example would give the result:

$$\Gamma = (1.0 \pm 0.3)\ \text{MeV} \qquad \rightarrow\ \text{Significance} = \frac{1.0}{0.3} = 3.3\sigma \qquad (31)$$

The same procedure with parameters

- $T = 28$ d, $r_{SN} = 1/30$, $\Gamma = 0.5$ MeV$/c^2$, $\Delta E_{\text{max}} = 1$ MeV, $n = 12$

[8] The appropriate way to count for the beam uncertainty would certainly be to convolute the excitation function f_{ex} with a Gaussian instead, which has not been performed for the time being.

results in mass histograms shown in Fig. 14 and a line shape fit presented in Fig. 15. Here the significance and the accuracy is as expected lower:

$$\Gamma = (0.89 \pm 0.91)\ \mathrm{MeV} \qquad \rightarrow\ \text{Significance}\ = \frac{0.88}{0.91} = 1.0\sigma \tag{32}$$

It should be stressed here that the latter result is not necessarly representative for fits with such a high background level, so worse results can be expected with a low signal-to-noise ratio.

In general the systematic effects of the procedure still have to be investigated to draw serious conclusions from the method and analysis results presented.

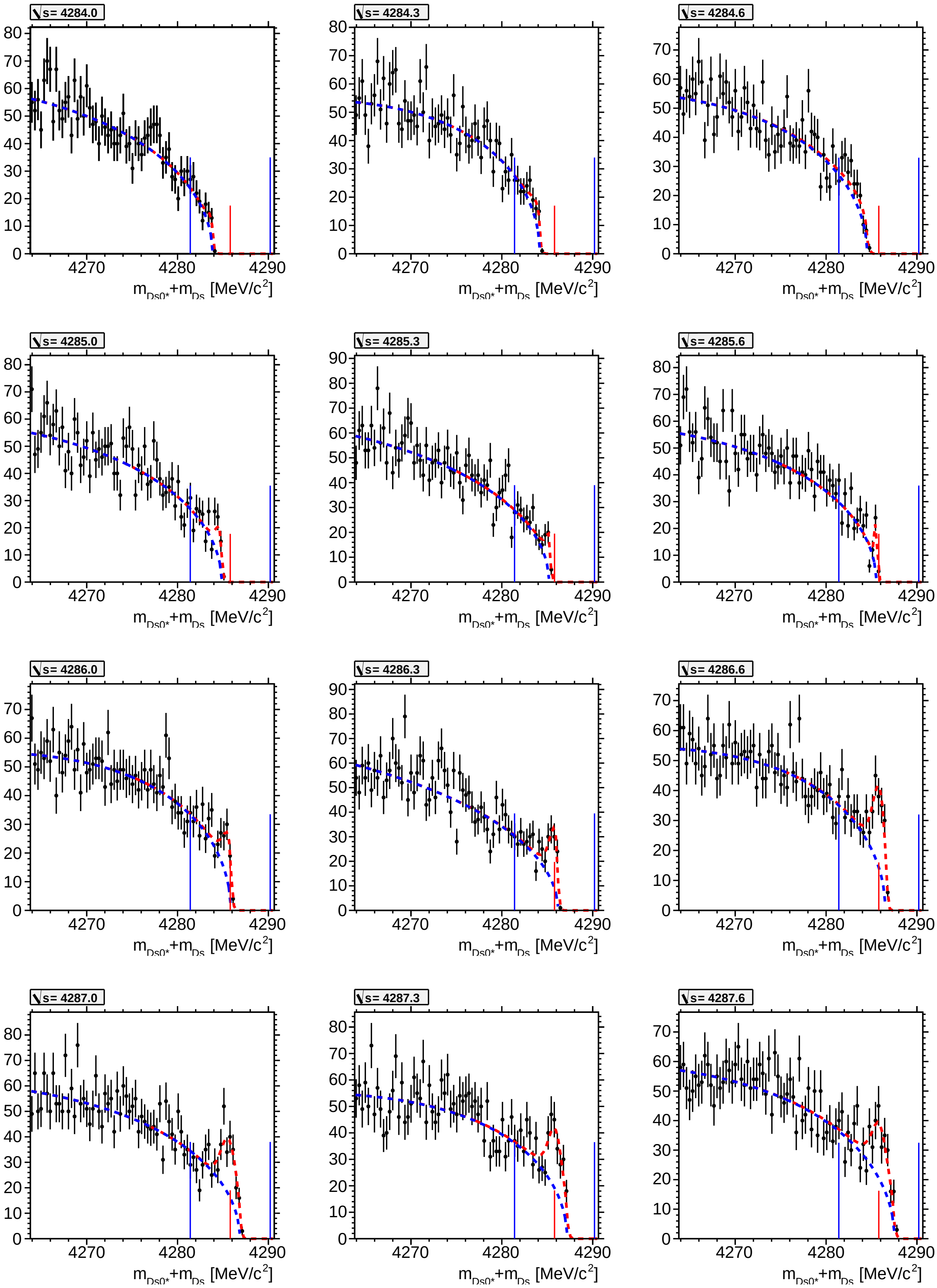

Figure 12: The individual mass distributions m_{sum} generated for energies E_i (Scan 1)

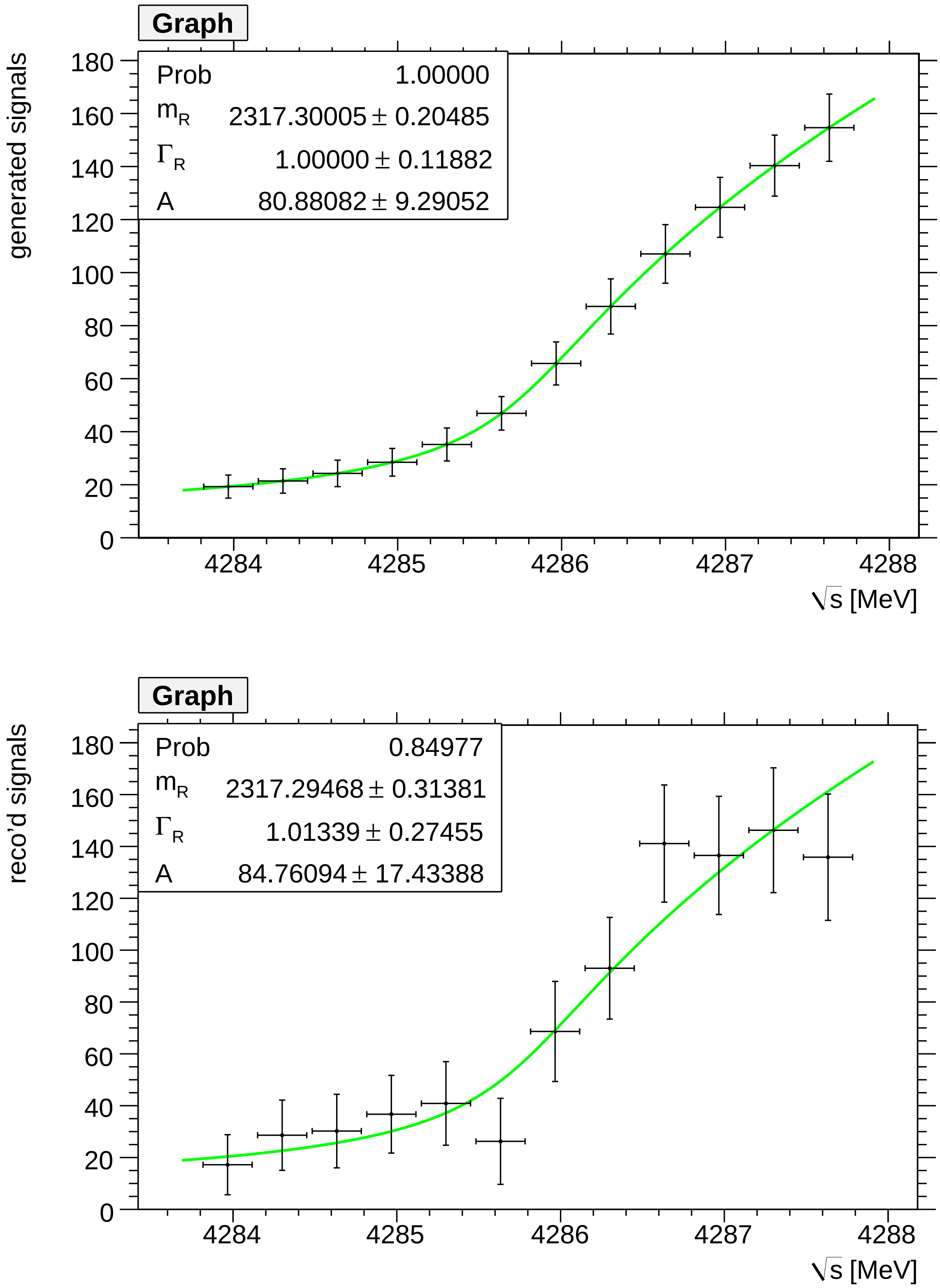

Figure 13: Fit to the excitation function for (a) generated and (b) fitted numbers. (Scan 1)

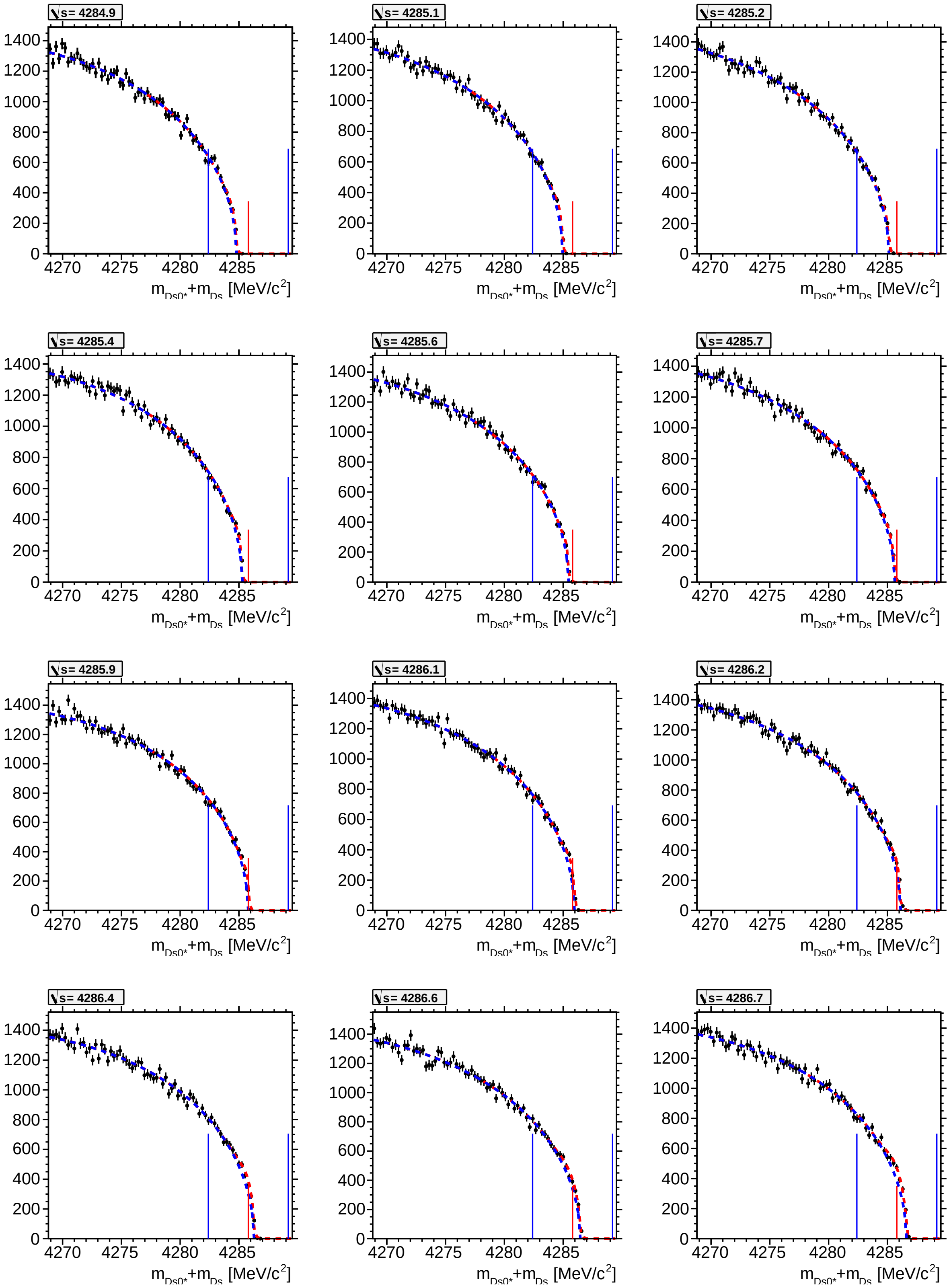

Figure 14: The individual mass distributions m_{sum} generated for energies E_i (Scan 2)

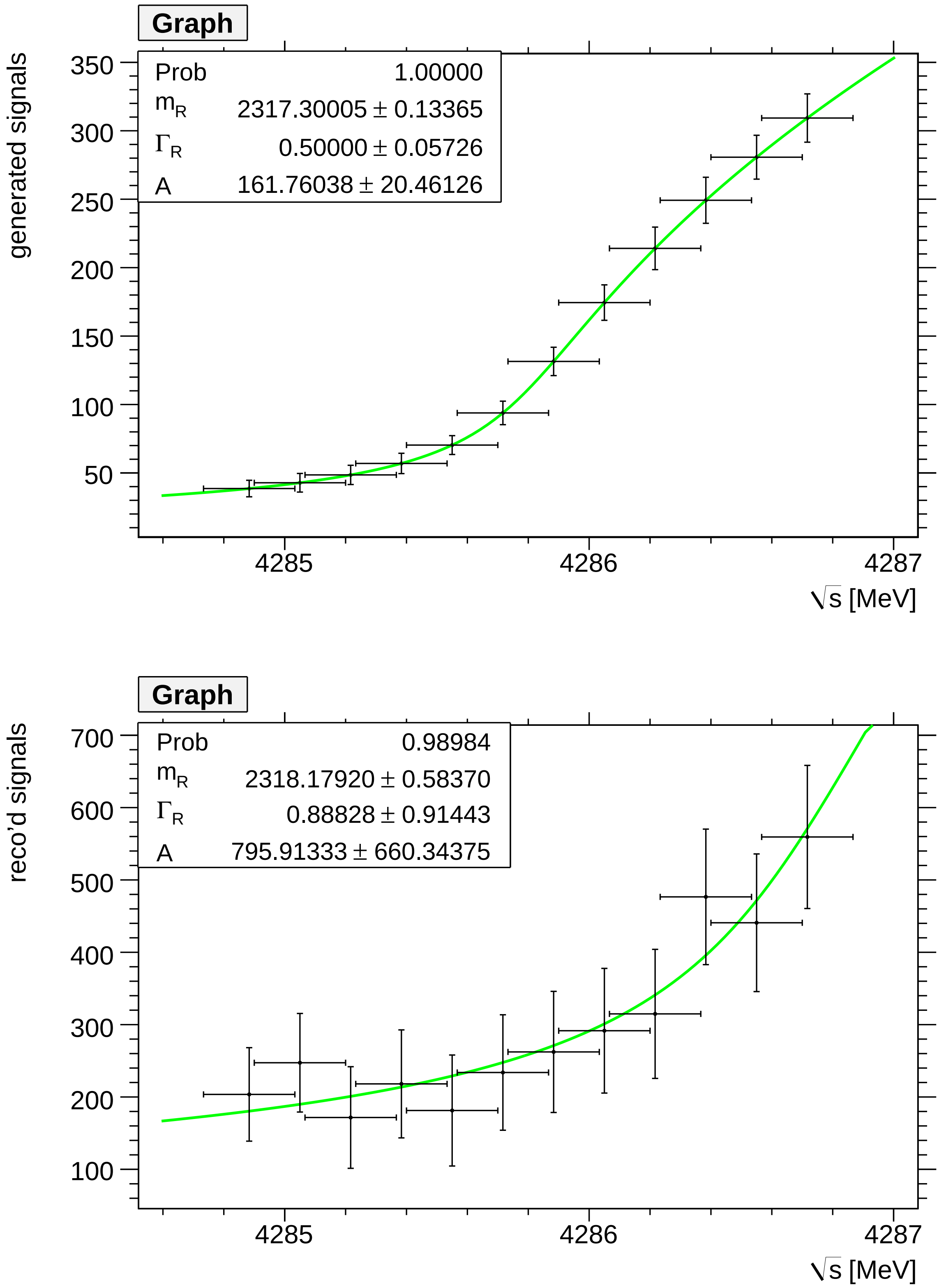

Figure 15: Fit to the excitation function for (a) generated and (b) fitted numbers. (Scan 2)

5 Limitations

In this section some final remarks and limitations are listed which will have to be investigated on the long run, i.e. either for the 'large' Physics Book or the real world analysis.

- Systematic investigation of the influence of the simulated width of the $D^*_{s0}(2317)$ on the distribution of the observable m_{sum}. Is its shape and width behaving 'gentle'?
- Production of simulated data at different energies (also closer to threshold or even slightly below). Does that spoil the method? Is the shape of the mass distribution strongly distorted? Is the efficiency constant?
- Larger sample of generic background to get better estimate of the expected signal-to-noise ratio r_{SN}. Might be crucial for feasibility in general.
- Perhaps tighten selection criteria.
- Find more appropriate model to describe the signal. Maybe relativistic Breit-Wigner with Blatt-Weisskopf damping. Examine systematic uncertainties originating from wrong fit model. Do the same for background shape.
- To take into account the beam jitter accordingly convolute the excitation function Eq. 27 with the appropriate beam resolution model (e.g. a Gaussian in the simplest case).
- Perform systematic studies for the energy scan with different parameter settings. What is the optimum scan region? Number of scan points? Systematic shift of the fit results under particular circumstances?
- Determine dependency of the significance from parameter settings. Detector limitations for the significance of measurement (something like 'smallest width one will be able to measure').

A Source Codes

A.1 Simulation of Energy Scan

```
#include "TF1.h"
#include "TH1F.h"
#include "TMath.h"
#include <iostream>
#include "TCanvas.h"
#include "TRandom3.h"
#include "TGraphErrors.h"
#include "TMinuit.h"
#include "TLine.h"
#include "TGraphErrors.h"
#include "TSystem.h"
#include "time.h"
#include "TList.h"

void config_histo(TH1 *h, TString tx, TString ty)
{
    h->SetLineWidth(2);

    h->GetXaxis()->SetTitleOffset(1.2);
    h->GetXaxis()->SetTitleColor(1);
    h->GetXaxis()->SetLabelSize(0.06);
    h->GetXaxis()->SetTitleSize(0.06);
    h->GetXaxis()->SetNdivisions(505);

    h->GetYaxis()->SetTitleOffset(1.7);
    h->GetYaxis()->SetTitleFont(42);
    h->GetYaxis()->SetLabelSize(0.06);
    h->GetYaxis()->SetTitleSize(0.06);

    h->SetXTitle(tx);
    h->SetYTitle(ty);

}

Double_t dampvoigt(Double_t *x, Double_t *par)
{
    double result=par[0]*TMath::Voigt(x[0]-par[1],par[2],par[3],4)*(1/(1+exp((x[0]-par[4])/par[5])));
    return result;
}


Double_t argusBG(Double_t *x, Double_t *par)  // voigtian = convolution of gauss with breit wigner
{
    double t=x[0]/par[1];

    if (t>=1) return 0;

    double u=1-t*t;

    return x[0]*par[0]*TMath::Power(u,par[3])*exp(par[2]*u);
}

Double_t dampvoigtArgus(Double_t *x, Double_t *par)
{
    return dampvoigt(x,par)+argusBG(x,&(par[6]));
}


Double_t voigt(Double_t *x, Double_t *par)  // voigtian = convolution of gauss with breit wigner
{
    return (par[0]*TMath::Voigt(x[0]-par[1],par[2],par[3],4)+par[4]);
}

Double_t resonance(Double_t *y,Double_t *par)
```

```
{
    Double_t crosss = 0, lambda;
    Double_t vf = 0;
    if (par[0] != 0)
    {
        lambda = (y[0] - 2*par[0]) / par[1]; //y[0]=sqrt(s)
        vf = sqrt(par[0] * par[1]);                    //par[0]=mr,par[1]=gamma

        static TF1 *f = new TF1("f","sqrt(([2] - 2 * [0]) / [1] - x) / ( x * x + 1)");
        f->SetParameters(par[0],par[1],y[0]);
        crosss = f->Integral(-1000,lambda);
        crosss = par[2]*crosss * vf / TMath::Pi()+par[3];
        //cout <<lambda<<endl;
    }
    return crosss;
}

Double_t resonance2(Double_t *y,Double_t *par)
{
    Double_t crosss = 0, lambda;

    double mds=1968.5;
    double mr=par[0];
    double gamma=par[1];
    double mrd=mds+mr;
    double dmrd=mr-mds;


    Double_t vf = 1;
    if (par[0] != 0)
    {
        lambda = 2/gamma*(y[0]-mrd);
        //vf = sqrt((mrd*mrd-dmrd*dmrd)/(mrd*gamma));

        static TF1 f("f","sqrt(2/[1]*([2]-[0])-x)/ ( x * x + 1)");
        f.SetParameters(mrd,gamma,y[0]);
        crosss = f.Integral(-1000,lambda);
        crosss = par[2]*crosss * vf / TMath::Pi();//+par[3];

        //cout <<crosss<<endl;
    }
    return crosss;
}

void fit_graph_new2(double nTotDays=1.0, double NS=1, double Gamma=1.0, double win=2., int nbins=15, bool equal=true)
// N     = number of signal events
// NS    = Noise-to-signal ration -> num of background evts B= NS*N
// Gamma = width of the DsJ
// win   = window around threshold for the scan
// nbins = number of scanpoints
{
    //TCanvas *c1 =new TCanvas("c1","c1",700,500);
    TCanvas *c3 =new TCanvas("c3","c3",650,900);
    c3->Divide(1,2);

    TCanvas *c2=new TCanvas("c2","c2",650,900);
    if (nbins<=9) c2->Divide(3,3);
    else if (nbins<=12) c2->Divide(3,4);
    else c2->Divide(3,5);

    TCanvas *c4=new TCanvas("c4","c4",650,900);
    if (nbins<=9) c4->Divide(3,3);
    else if (nbins<=12) c4->Divide(3,4);
    else c4->Divide(3,5);

    //TCanvas *c1=new TCanvas("c1","c1",700,500);

    // ***********
    // Parameters + constants
```

```
// ***********

bool equalscanpoints =  equal;  // use equally distributed points or more dense scan points around threshold

double nDays   = nTotDays/nbins;
double sigXsec = 1.0;              // cross section [nb]
double lumiDay = 8800;             // int lumi/day [nb-1]
double effFact = 0.25*0.044*0.492;  // efficiency factor times BR factors

double sigma   = 0.6;              // resolution of the detector (for the Dsj signal peak); don't mix up with beam resolut
double beamerr = 0.15;             // beam uncertainty in [MeV]
double mdsj    = 2317.3;           // mass of the DsJ [MeV]
double mds     = 1968.5;           // mass of the Ds [MeV]

double widthFactor=4.0;

double thresh=mdsj+mds;   // threshold for scan

double sigwinup=thresh+(sigma+Gamma/2)*widthFactor;
double sigwinlow=thresh-(sigma+Gamma/2)*widthFactor;
double sigwin=sigwinup-sigwinlow;

double mlow=thresh-2.5*sigwin;      // lower limit of mass histos
double mhigh=thresh+0.55*sigwin;     // upper limit of mass histos

double low = thresh-win;  // lower limit of scan
double up= thresh+win;    // upper limit of scan

//double BGwidthFactor = (mhigh-mlow)/(sigma+Gamma)/6;


double upshift=0.2;


int hbins=80;             // number of bins for mass histos

TH1F *mass[20];  // the mass histos (we only need nbins of them)
TH1F *hbg[20];   // histos for bg only (for automated fitting)

TRandom3 rand;   // a random generator
rand.SetSeed(time(0));


// ************************
// define the functions
// ************************

// the voigtian (= convolution of gauss with breit wigner) to fit the mass histo
TF1 *fsigbg=new TF1("fsigbg",dampvoigtArgus,mlow,mhigh,10);
fsigbg->SetParNames("A_{s}","m_{0}","#sigma","#Gamma","m_{q}","#tau","A_{b}","m_{t}","c","p");
fsigbg->SetParameters(10,thresh,sigma,Gamma,4286,0.1,100,4286,-30,0.5);
fsigbg->SetLineStyle(2);
fsigbg->SetLineWidth(2);
fsigbg->SetLineColor(2);

// signal part intergrating the peak in each histo
TF1 *fsig=new TF1("fsig",dampvoigt,mlow, mhigh,6);
fsig->SetParNames("A_{s}","m_{0}","#sigma","#Gamma","m_{q}","#tau");
fsig->SetParameters(10,thresh,sigma,Gamma,4286,0.1);
fsig->SetLineStyle(2);
fsig->SetLineWidth(2);
fsig->SetLineColor(2);

// background part in mass histo
TF1 *fbg=new TF1("fbg",argusBG,mlow,mhigh,4);
fbg->SetParNames("A_{b}","m_{t}","c","p");
fbg->SetParameters(100,4286,-30,0.5);
fbg->SetLineStyle(2);
fbg->SetLineWidth(2);
```

```
    fbg->SetLineColor(4);

    // the exitation function
    TF1 *fexi = new TF1("fexi",resonance2,low,up,3);
    fexi->SetParNames("m_{R}","#Gamma_{R}","A");
    //fexi->SetParLimits(0,2317-0.5,2317+0.5);
    fexi->SetParameters(mdsj,Gamma,100.0);
//     fexi->SetParameter(0,mdsj);
//     fexi->SetParameter(1,Gamma);
//     fexi->SetParameter(2,1.0);//*1.5/1000*sigXsec*lumiDay*nDays*effFact);
    fexi->SetLineWidth(2);
    fexi->SetLineColor(3);
    fexi->SetNpx(100);

    c3->cd(1);
    //fexi->Draw();
    c3->Update();
    //func2->FixParameter(3,0);

    // the coordinates and errors for the TGraphErrors object below
    double *x=new double[nbins];    // the scan positions sqrt(s)_i
    double *y=new double[nbins];

    double *ex=new double[nbins];
    double *ey=new double[nbins];

    double *gx=new double[nbins];    // the scan positions sqrt(s)_i
    double *gy=new double[nbins];

    double *gex=new double[nbins];
    double *gey=new double[nbins];

    int i,j;
    double step=(up-low)/(double)nbins;
    //double sum=0;

    // this loop is to calculate the scan positions
    for (i=0;i<nbins;i++)
    {
        if (equalscanpoints) x[i]=step*i+0.5*step+low;
        else {
            if ((i<1) || (i>(nbins-2)) ) x[i]=step*i+0.5*step+low;
            else x[i] = (i-(nbins+1.)/2.+0.5)*step/2.+thresh;
        }
        ex[i]=beamerr;         // the error of beam (250 keV)
        //y[i]=fexi->Eval(x[i]);     // store the value of exitation function at x_i

        // estimate signals; nominal lumi is at threshold
        y[i] = fexi->Eval(x[i]+rand.Gaus(0,ex[i]))/fexi->Eval(thresh)*sigXsec*lumiDay*nDays*effFact;
        //sum+=y[i];                  // needed for normalization later

        char tmp[200],tmp2[200],tmpbg[200];
        sprintf(tmp,"mass%d",i);
        sprintf(tmp2,"#sqrt{s}= %6.1f",x[i]);
        sprintf(tmpbg,"hbg%d",i);

        // create the histograms
        mass[i]=new TH1F(tmp,tmp2,hbins,mlow,mhigh);
        mass[i]->Sumw2();
        mass[i]->SetMinimum(0);
        mass[i]->SetTitleSize(0.1);
        config_histo(mass[i],"m_{Ds0*}+m_{Ds} [MeV/c^{2}]","");//"entries / MeV/c^{2}");
        mass[i]->SetLineWidth(1);
        mass[i]->SetStats(0);

        hbg[i]=new TH1F(tmpbg,tmpbg,hbins,mlow,mhigh);
        hbg[i]->SetLineWidth(1);
        hbg[i]->Sumw2();
        //mass[i]->SetMaximum(500);
```

```
}

// creat sum histogram
//mass[nbins]=new TH1F("msum","sum histogram",hbins,mlow,mhigh);
//mass[nbins]->Sumw2();
//config_histo(mass[nbins],"m_{Ds0*}+m_{Ds} [MeV/c^{2}]","");//"entries / MeV/c^{2}");

cout <<"Scan point postions + expected signals:"<<endl;

for (i=0;i<nbins;i++)
{
    //y[i]=y[i]/sum*N;    // here we calculate the number of expected signal events for every scan point
    ey[i]=sqrt(y[i]);   // the error of #signals

    gy[i]=fexi->Eval(x[i])/fexi->Eval(thresh)*sigXsec*lumiDay*nDays*effFact;
    gey[i]=ey[i];
    gx[i]=x[i];
    gex[i]=ex[i];

    cout<<"E"<<i<<" = "<<x[i]<<" -> S_exp_"<<i<<" = "<<y[i]<< " +- "<<ey[i]<<endl;
}

c3->cd(1);
TGraphErrors *gengr=new TGraphErrors(nbins,gx,gy,gex,gey);
gengr->GetXaxis()->SetTitle("#sqrt{s} [MeV]");
gengr->GetXaxis()->SetTitleColor(1);
gengr->GetXaxis()->SetTitleOffset(1.4);
gengr->GetYaxis()->SetTitleFont(42);
gengr->GetXaxis()->SetNdivisions(505);
//gengr->SetLineColor(6);
//gengr->SetMarkerColor(6);

gengr->GetYaxis()->SetTitle("generated signals");

fexi->SetParameters(mdsj+0.2*rand.Rndm()-0.1,Gamma+0.2*rand.Rndm()-0.1,100.0);
gengr->Draw("AP");
c3->Update();
gengr->Fit("fexi");
/*gengr->Fit("fexi","m");
gengr->Fit("fexi","m");*/
c3->Update();

// in this loop we fill the histograms with signal and background events
//int B=y[nbins-1]*NS*BGwidthFactor;    // the num of bgk events is NS*Signals for highest sqrt(s) (=highest signal peak)
//cout <<y[nbins-1]<<" "<<NS<<" "<<BGwidthFactor<<endl;

TLine l1(thresh,0,thresh,500);
l1.SetLineColor(2);
TLine l2(thresh,0,thresh,500);
l2.SetLineColor(4);

int gensum=0;

fbg->SetParameter(1,x[nbins-1]);

double regionwidth=x[nbins-1]-mlow;
double Bfactor=fbg->Integral(mlow,x[nbins-1])/fbg->Integral(sigwinlow,sigwinup);


for (i=0;i<nbins;i++)
{
    fsig->SetParameter(4,x[i]);
    //fbg->SetParameter(1,x[i]);
    c2->cd(i+1);
    // the number of signals to generate; poisson random value of expectation value
    int nev=rand.PoissonD(y[i]);
    gensum+=nev;
    for (j=0;j<nev;j++)
        mass[i]->Fill(fsig->GetRandom(mlow,mhigh));
```

```
    //double Bfactor=(fbg->Integral(mlow,x[i])/fbg->Integral(x[i]-sigwin,x[i]));
    int B=y[nbins-1]*NS*Bfactor;

    int nevb=rand.PoissonD(B);     // the number of bkg evts to generate ('')
    cout <<"S="<<y[i]<<"  -->  "<< nev<<"   B="<<B<<"  -->  "<<nevb<<endl;
    for (j=0;j<nevb;j++)
    {
        //double bgent=fbg->GetRandom(mlow,x[i]);
        double bgent=fbg->GetRandom(mlow,x[nbins-1]);
        mass[i]->Fill(bgent-x[nbins-1]+x[i]);
        hbg[i]->Fill(bgent-x[nbins-1]+x[i]);

    }
    mass[i]->Draw();
    l1.DrawLine(thresh,0,thresh,mass[i]->GetMaximum()*0.25);
    l2.DrawLine(sigwinup,0,sigwinup,mass[i]->GetMaximum()*0.5);
    l2.DrawLine(sigwinlow,0,sigwinlow,mass[i]->GetMaximum()*0.5);
    c2->Update();

    c4->cd(i+1);
    hbg[i]->Draw();
    l2.DrawLine(x[i]+upshift,0,x[i]+upshift,hbg[i]->GetMaximum()*0.5);
    l2.DrawLine(x[i]+upshift-sigwin,0,x[i]+upshift-sigwin,hbg[i]->GetMaximum()*0.5);
    c4->Update();

}

/*
// sum all histos
for (i=0;i<nbins;i++) mass[nbins]->Add(mass[i]);
mass[nbins]->SetMinimum(0);

c2->cd(nbins+1);
mass[nbins]->Draw();

// fit the voigtian to the sum histo
//fv.SetParLimits(0,1,10000);
fv.SetParameter(0,mass[nbins]->GetBinContent(hbins/2)-mass[nbins]->GetBinContent(2));
fv.SetParameter(4,mass[nbins]->GetBinContent(2));
mass[nbins]->Fit("fv","","",2.307,2.327);
mass[nbins]->Fit("fv","m");

fv.ReleaseParameter(0);


// fix the parameters mean, Gamma, sigma for all fits
fv.FixParameter(1,fv.GetParameter(1));
fv.FixParameter(2,fv.GetParameter(2));
fv.FixParameter(3,fv.GetParameter(3));
*/
double nsum=0;


//gMinuit->SetMaxIterations(10);

fbg->SetParameters(100,x[0],-30,0.5);
fbg->FixParameter(3,0.5);
c2->cd(1);

mass[0]->Fit("fbg","");
mass[0]->Fit("fbg","m");



double thdev=0.1;

fsigbg->SetParLimits(1,thresh-thdev,thresh+thdev);
fsigbg->SetParLimits(2,0.1,2.);
```

```
fsigbg->SetParLimits(3,0.1,3.);
fsigbg->SetParLimits(5,0.01,1.);

int maxtries=5;


// fit all the histos with voigtian
for (i=0;i<nbins;i++)
{
    c4->cd(i+1);
    fbg->SetParameters(100,x[0],-30,0.5);
    fbg->SetParLimits(1,x[i]-thdev,x[i]+thdev);
    hbg[i]->Fit("fbg","","",mlow,x[i]);

    l1.DrawLine(sigwinlow,0,sigwinlow,hbg[i]->GetMaximum()*0.5);
    l1.DrawLine(sigwinup,0,sigwinup,hbg[i]->GetMaximum()*0.5);
    c4->Update();

    c2->cd(i+1);
    int tries=maxtries;
    fsigbg->SetParLimits(4,x[i]-thdev,x[i]+thdev);
    fsigbg->SetParLimits(7,x[i]-thdev,x[i]+thdev);
    fsigbg->FixParameter(6,fbg->GetParameter(0));
    fsigbg->FixParameter(7,fbg->GetParameter(1));
    fsigbg->FixParameter(8,fbg->GetParameter(2));
    fsigbg->FixParameter(9,0.5);
    fsigbg->FixParameter(5,0.1);

    fsigbg->SetParameters(10,thresh,1,1,x[i],0.1,fbg->GetParameter(0),fbg->GetParameter(1),fbg->GetParameter(2),0.5);
    mass[i]->Fit("fsigbg","q");
    while (--tries && fsigbg->GetParError(0)/fsigbg->GetParameter(0)>1.)
    {
        mass[i]->Fit("fsigbg","qm");
    }
    cout <<maxtries-tries<<endl;

    mass[i]->GetListOfFunctions()->Add(new TF1(*fbg));
    mass[i]->Draw();


    l1.DrawLine(thresh,0,thresh,mass[i]->GetMaximum()*0.25);
    l2.DrawLine(sigwinlow,0,sigwinlow,mass[i]->GetMaximum()*0.5);
    l2.DrawLine(sigwinup,0,sigwinup,mass[i]->GetMaximum()*0.5);

    fsig->SetParameters(fsigbg->GetParameters());
    //fv.SetParameter(4,0);
    double n=fsig->Integral(sigwinlow,sigwinup)/mass[i]->GetBinWidth(1);  // find number of signals in peak by integral
    int bkg=fbg->Integral(sigwinlow,sigwinup)/hbg[i]->GetBinWidth(1);
    nsum+=n;
    //ey[i]=fabs(fsigbg->GetParError(0)/fsigbg->GetParameter(0)*fabs(n));
    ey[i]=sqrt(n+bkg);

    //if (n<0) n=0;
    y[i]=n;

    // estimate error of integral; I chose the same relative error like the one of the amplitude parameter
    // most likely this is not correct
    cout <<i<<": S = "<<y[i]<<" +- "<<ey[i]<<"   B = "<<bkg<<"   ---> B/S = "<<bkg/n<<endl;

    c2->Update();
}

cout <<"Nsum = "<<nsum<<"   generated: "<<gensum<<endl;

c3->cd(2);

fexi->SetParLimits(0,mdsj-1.2,mdsj+1.2);
fexi->SetParLimits(1,0.005,4.6);
fexi->SetParameters(mdsj+0.2*rand.Rndm()-0.1,Gamma+0.2*rand.Rndm()-0.1,100.0);
```

```
    //fexi->SetParameters(mdsj,Gamma);
    //fexi->SetParameter(2,1.5/1000*N);
    //fexi->SetParameter(3,0);


    double *yf=new double[nbins];
    double *eyf=new double[nbins];
    double *xf=new double[nbins];
    double *exf=new double[nbins];

    int fbins=0;

    for (i=0;i<nbins;i++)
    {
        if (ey[i]/y[i]<3)
        {
            xf[fbins]=x[i];
            yf[fbins]=y[i];
            exf[fbins]=ex[i];
            eyf[fbins]=ey[i];
            fbins++;

        }
    }


    // finally creat graph with measured values and fit the exitation function to it
    //TGraphErrors *gr=new TGraphErrors(nbins,x,y,ex,ey);
    TGraphErrors *gr=new TGraphErrors(fbins,xf,yf,exf,eyf);
    gr->GetXaxis()->SetTitle("#sqrt{s} [MeV]");
    gr->GetXaxis()->SetTitleColor(1);
    gr->GetXaxis()->SetTitleOffset(1.4);
    gr->GetXaxis()->SetNdivisions(505);
    gr->GetYaxis()->SetTitleFont(42);

    gr->GetYaxis()->SetTitle("reco'd signals");

    gr->Draw("AP");
    gr->Fit("fexi");
    gr->Fit("fexi","m");
    gr->Fit("fexi","m");
    //gengr->Draw("same");
    //gr->Draw("same");
    c3->cd();
}
```