BESIII Oxford Group Meeting

Martin Tat

Oxford LHCb

2nd December 2021

Introduction

- Strong-phase analysis of $D^0 \to K^+K^-\pi^+\pi^-$
- Measure c_i, s_i (and K_i) using double-tags
- Strategy:
  1. Determine single tag yields for normalization
  2. Determine $KK\pi\pi$ vs flavour tag yields to obtain K_i
  3. Determine $KK\pi\pi$ vs CP and self-conjugate tag yields
  4. Maximum likelihood fit to obtain c_i and s_i
- Focused a lot on covering *all* tag modes previously...
- Current plan is to first obtain K_i and make sure everything works
- Then run toys with yields extrapolated to $20\,\text{fb}^{-1}$ in fit to c_i and s_i

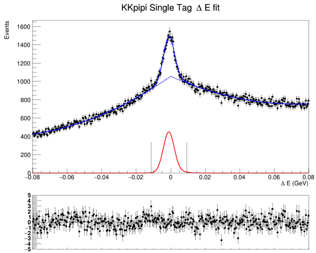
Previous work (before May 2021)

- Fitted single tag ΔE distributions
  - Double Gaussian + 2nd order polynomial
  - Apply cut at $\pm 3\sigma$
- Fitted single tag m_{BC} distributions
  - Signal shape from MC, Argus for combinatorial background, Gaussian for peaking backgrounds
  - Fit quality very poor in flavour tag modes because of higher statistics
- Double tag yield using sideband subtraction (same as $K_S KK$ strong-phase analysis)

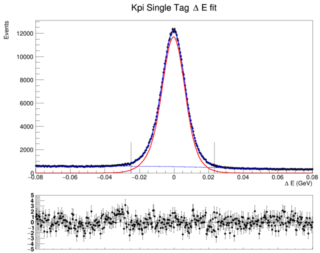
New ΔE fits

- Signal: Double Gaussian
- Background: Chebychev polynomials of arbitrary order for two- and three-body modes
- For $KK\pi\pi$ and $K\pi\pi\pi$ use two independent polynomials on each side of $\Delta E = 0$

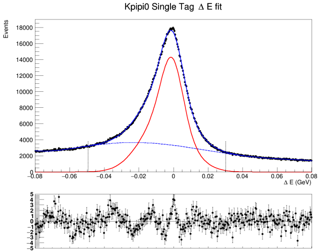
New ΔE fits



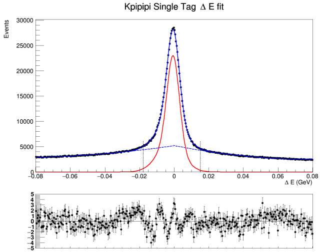
(a) $KK\pi\pi$



(b) $K\pi$



(c) $K\pi\pi^0$

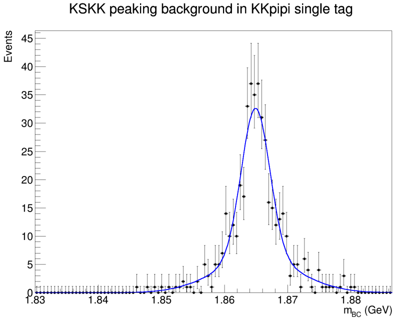


(d) $K\pi\pi\pi$

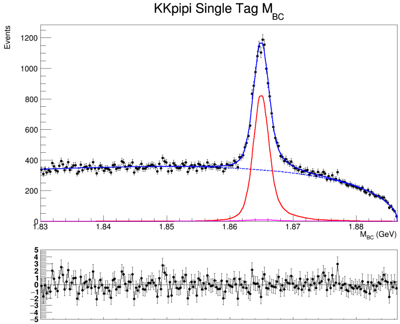
New single tag m_{BC} fits

- Signal: Signal MC shape (after truth matching)
- Resolution: Convolve with double Gaussian
- Combinatorial background: Argus shape
- Can add arbitrary shapes for peaking backgrounds
- New strategy for peaking backgrounds:
  1. Identify backgrounds with inclusive MC
  2. Generate signal MC and fit shape to this sample
  3. Calculate yield using relative efficiencies and branching fractions

New single tag m_{BC} fits



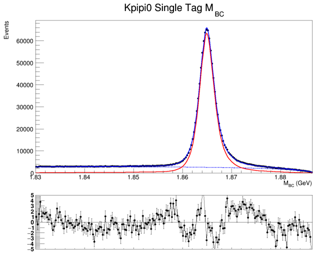
(a) $K_S KK \to KK\pi\pi$



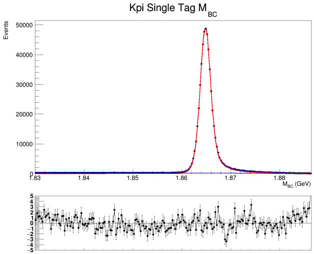
(b) $KK\pi\pi$ m_{BC} fit

Yield: $10\,821 \pm 198$

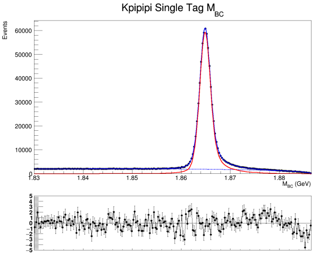
New single tag $m_{\rm BC}$ fits



(a) $K\pi\pi^0$ $m_{\rm BC}$ fit



(b) $K\pi$ $m_{\rm BC}$ fit



(c) $K\pi\pi\pi$ $m_{\rm BC}$ fit

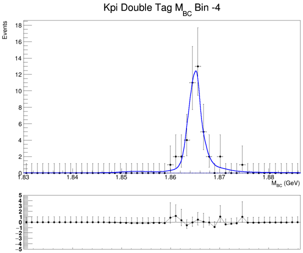
Double tag yield simultaneous fit ($KK\pi\pi$ vs $K\pi$)

- Should have enough events to perform fit of $KK\pi\pi$ m_{BC}
  - Same strategy as $\pi\pi\pi\pi$ strong-phase analysis
  - Use 2×4 bins for now
  - Scrap sideband subtraction

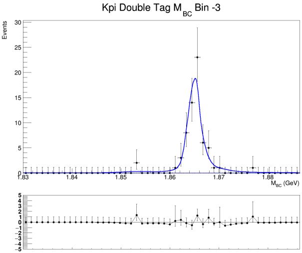
- Fit shapes:
  - Signal shape: Signal MC shape (after truth matching)
  - Resolution: Single Gaussian
  - Combinatorial background: Argus shape

- Fit strategy:
  1. For each double tag mode, perform simultaneous fit of all bins
  2. Float signal and combinatoria yield in each bin
  3. Gaussian shape and Argus slope are floated
  4. Then fix all shapes and any combinatorial background less than 0.5
  5. Fit a second time to obtain accurate signal yields in each bin

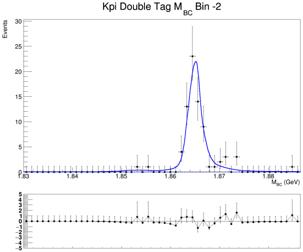
Double tag yield simultaneous fit ($KK\pi\pi$ vs $K\pi$)



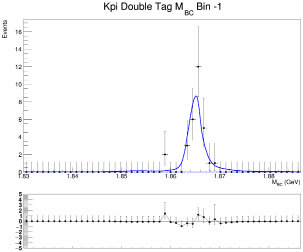
(a) Bin −4



(b) Bin −3

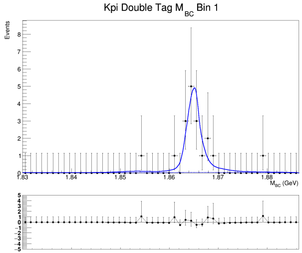


(c) Bin −2

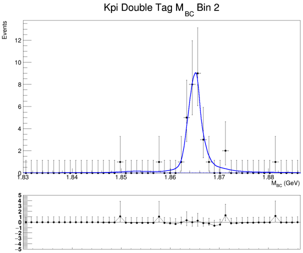


(d) Bin −1

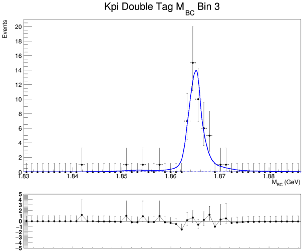
Double tag yield simultaneous fit ($KK\pi\pi$ vs $K\pi$)



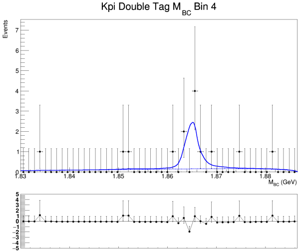
(a) Bin 1



(b) Bin 2



(c) Bin 3



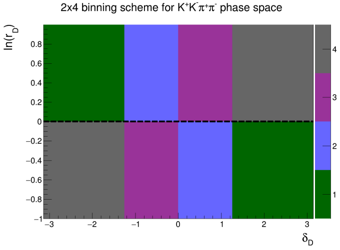
(d) Bin 4

Efficiency corrections

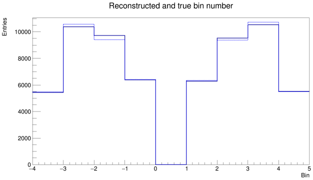
- Single tag efficiencies trivially corrected by dividing the yields
- For double tags, construct an efficiency matrix to account for both efficiency and bin migration
- For events reconstructed in bin i and generated in bin j:

$$\epsilon_{ij} = \frac{N_{ij}^{\text{reconstructed}}}{N_j^{\text{generated}}}$$

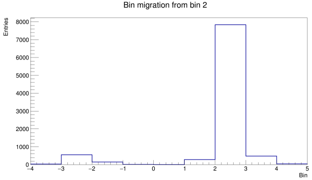
Efficiency corrections



(a) Binning scheme



(b) Reconstructed bin vs true bin



(c) Bin migration from bin 2

Bin migration much higher than expected (10%-15%)

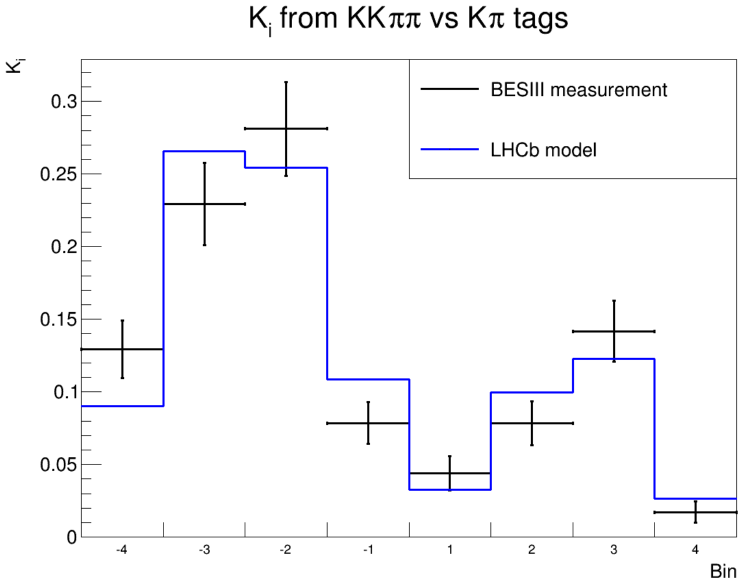
But net migration is relatively small

DCS corrections

Calculate using c_i, s_i, K_i from LHCb model

Bin	Correction
-4	1.0091 ± 0.0017
-3	0.9538 ± 0.0007
-2	0.9669 ± 0.0014
-1	1.0166 ± 0.0016
$+1$	1.0599 ± 0.0228
$+2$	0.7833 ± 0.0027
$+3$	0.8263 ± 0.0060
$+4$	1.1850 ± 0.0229

K_i results for $KK\pi\pi$ vs $K\pi$



Conclusion and next steps

- Single tag m_{BC} show acceptable fit quality now
- Double tag simultaneous fit works great!
- K_i from $KK\pi\pi$ vs $K\pi$ tag looks encouraging, but not perfect
- Next steps:
  - Include $K\pi\pi^0$, $K\pi\pi\pi$ and $Ke\nu$ tags in K_i determination
  - Calculate peaking backgrounds in each bin with LHCb model
  - Develop likelihood fitter for c_i/s_i and run some toys
  - Start looking at CP tag fits to data (but yields will be very low!)
  - Remove $K_S\phi$, include K_SKK and K_LKK as self-conjugate tags