Bhabha Tracking Efficiencies

Martin Sobotzik

12.04.2019

Johannes Gutenberg Universität Mainz

Motivation

- I would like to estimate the tracking efficiency on phase 2 data
- I use Bhabha events because if one track is reconstructed then the other particle should also produce a track

$$\epsilon = \frac{\text{Number of Bhabha events with exactly 2 tracks}}{\text{Number of Bhabha events with 1 or more tracks}}$$

- This idea comes from some plots presented by Sam Cunliffe in previous tracking and ECL meetings.

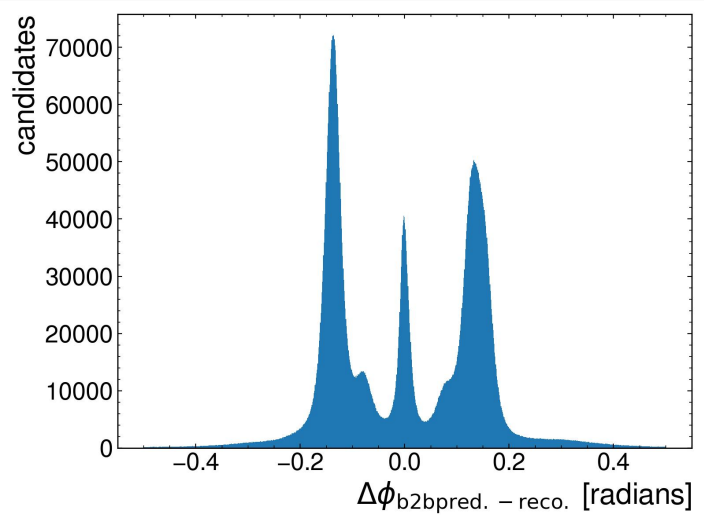
Getting Started

- Cuts Sam used:
  - gamma:probe '(E > 0.1)'
  - gamma:tag '(clusterE > 3.0)'
  - vpho:cand 'reconstructed from gamma:probe and gamma:tag'
  - $0.296706 < \theta < 2.61799 \rightarrow$ It has to hit the ECL
  - nCleanedTracks[abs(dz) < 2.0 and abs(dr) < 0.5 and nCDCHits > 0 and pt > 0.15] < 1 $\rightarrow$ bad quality hits
- Cuts I use:
  - $M(\text{vpho}) > 8.0\,\text{GeV} \rightarrow$ For the vpho to have a mass of at least $8\,\text{GeV}$, gamma:tag and gamma:probe must have at least an energy of more than $3\,\text{GeV}$

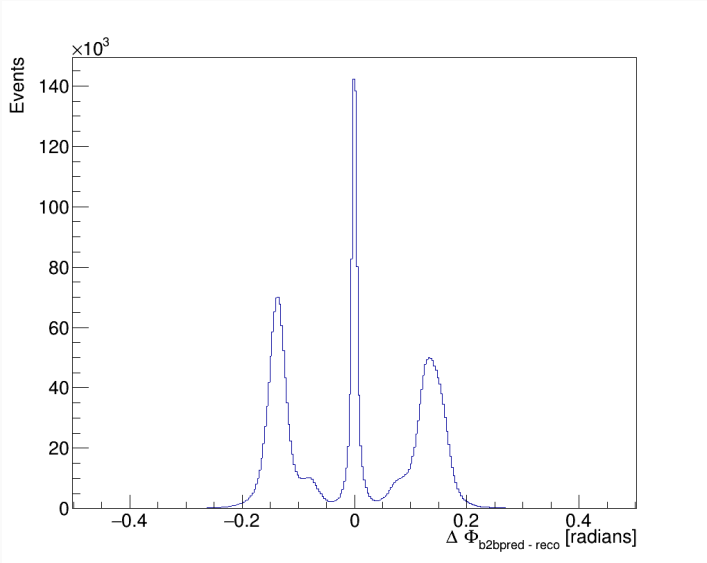
Reproducing Plots

- For data I using prod6 (this is because as a starting point I tried to reproduce Sam's plots)
- I am using the following data:
  /hsm/belle2/bdata/Data/release-02-01-00/DB00000438
  /prod00000006/e0003/4S/r02*/all/mdst.sub00/*.root

Sam: All data files



My



Original idea

- Treat every hit in the ECL as a photon
- After reconstruction, check if there is a track associated with the cluster

```
fillParticleList('gamma:probe','clusterE > 0.0001 ',path=mypath)
fillParticleList('gamma:tag','clusterE > 0.0001',path=mypath)
reconstructDecay('vpho:cand -> gamma:probe gamma:tag',mass_cut,path=mypath)
```

- As a first check I ran over MC11 $\mathrm{ee} \to \mathrm{ee}$ samples with the same steering file
- Far too few vpho were reconstructed and the daughters had no tracks associated
- Explanation: The γ list is only filled by particles without a track
- So the reconstruction from two γ lists is exactly what we don't need

How to proceed

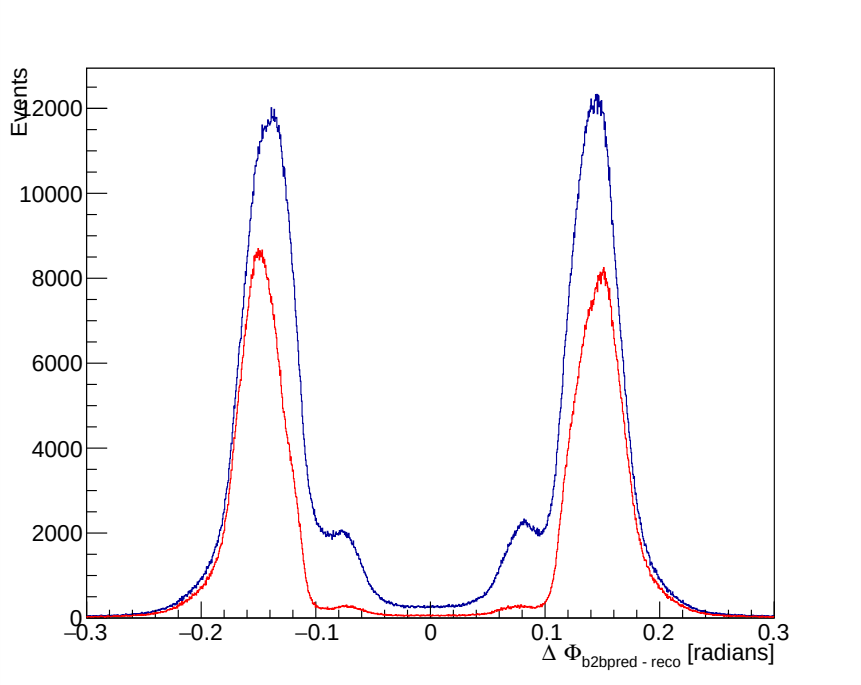
- If there is a cluster in the ECL, we check if that cluster has a track associated. If it has than it is an $\mathrm{e}^+/\mathrm{e}^-$. If it has not then it is a γ
- So the list we want is a list filled with every event in the ECL

```
fillParticleList('gamma:probe','clusterE > 0.01 ',path=mypath)
fillParticleList('e+:all','clusterE > 0.01', path=mypath)

reconstructDecay('vpho:elec0 -> e+:all','',path=mypath)
reconstructDecay('vpho:gamma0 -> gamma:probe','', path=mypath)

copyLists(outputListName='vpho:mypho_un',inputListNames=['vpho:elec0','vpho:gamma0'],path=mypath)
rankByHighest('vpho:mypho_un','daughter(0,clusterE)',path=mypath)
cutAndCopyList('vpho:mypho','vpho:mypho_un','',path=mypath)

reconstructDecay('vpho:bhabha -> vpho:mypho vpho:mypho',mass_cut,path=mypath)
```

- To add these two lists a *pseudo* vpho has to be introduced
- The vpho we want can then be reconstructed

Running on MC

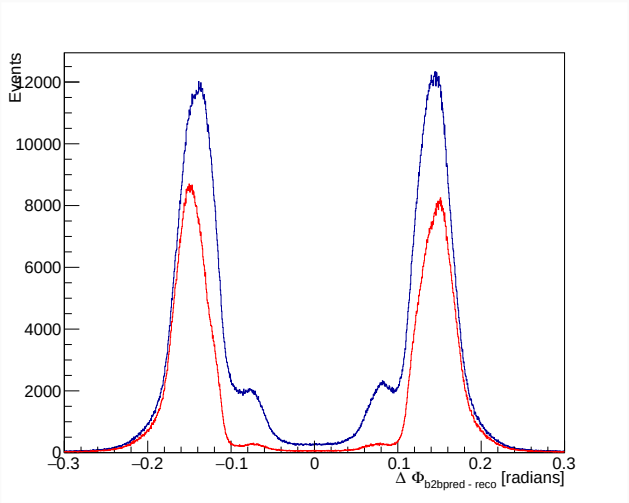
- Running on MC11 $\mathrm{ee} \rightarrow \mathrm{ee}$ with new steering file



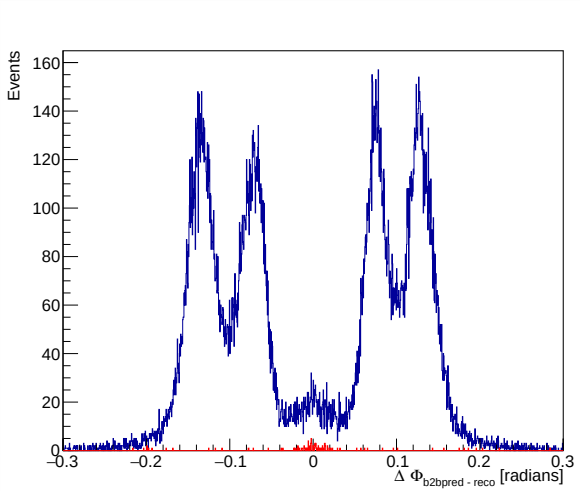
- Pure Bhabha sample $\rightarrow$ No middle peak caused by $\gamma\gamma$ events
- Difference between reconstructed and MCTruthMatched is caused by inefficiency

Running on MC

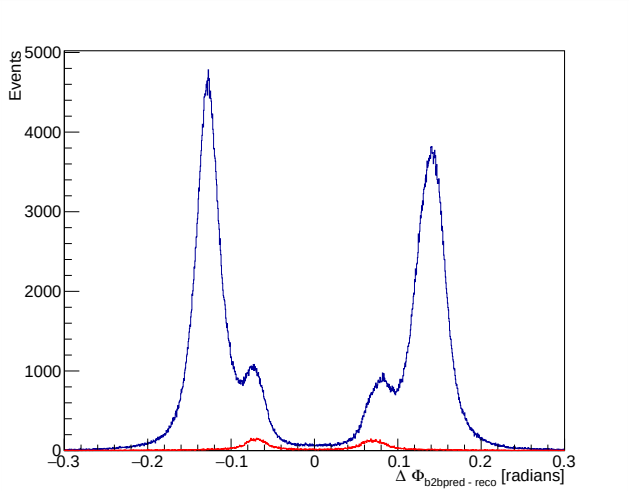
all



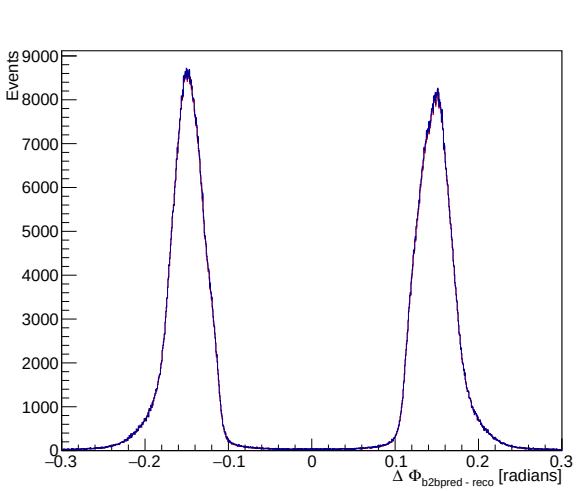
- MC11 $\mathrm{ee} \rightarrow \mathrm{ee}$ sample
- Reconstructed blue
- MCTruthMatched red



$\gamma\gamma$



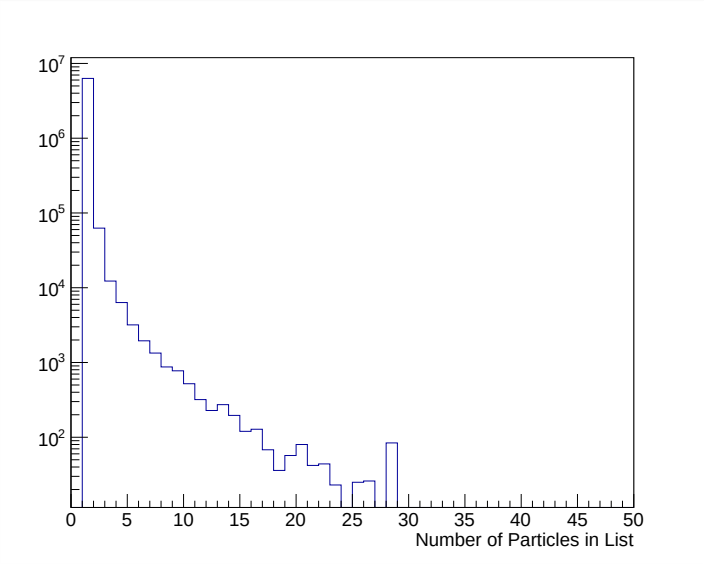
$\gamma\mathrm{e}$

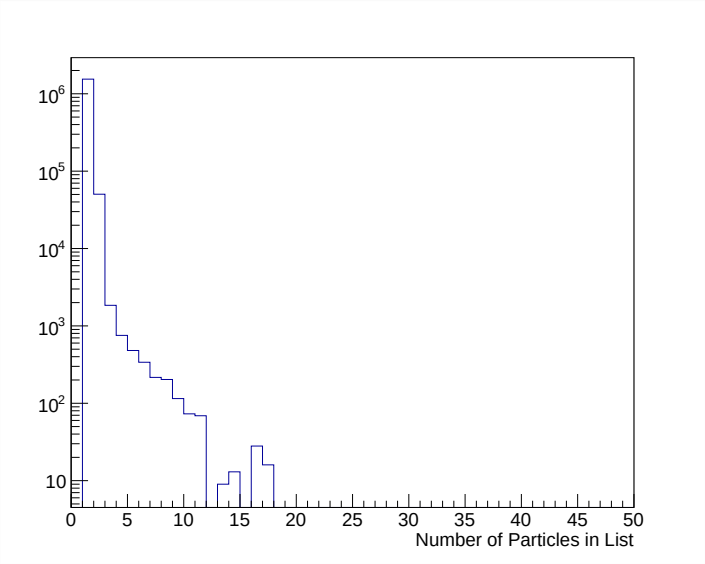


ee

Number of candidates

- Number of reconstructed vpho per event is oftentimes bigger than 1





Next steps

- Select best vpho
- Cut on $\Delta\Phi_{\text{b2bpred -reco}}$ peak and calculate a first efficiency