PID Selectors in FastSim

N. Arnaud, L. Burmistrov, A.Perez, A. Stocchi

Outlook

- Pion and Kaon LH – based selectors
- PidCalib application
- Problems we found
- To do list

LABORATOIRE DE L'ACCÉLÉRATEUR LINÉAIRE

UNIVERSITÉ PARIS-SUD 11

Pion and Kaon LH – based selectors

- SVT - (dE/dx)
- DCH - (dE/dx)
- DRC - Cerenkov angle
  - number of photons
- Backward EMC - time
  (just added to FastSim)
- Forward PID system
  - RICH Cerenkov angle
    (currently not in FastSim)
  - or
  - Time Of Flight (TOF)
    (purr physical model in FastSim)

Each of these subsystems should have 6 parameters which used to construct Pi/K LH – based selectors
1) Measured value
2) Expected error
3-6) expected value for particle hypothesis

Construct pulls → $pull = (val_{meas} - val_{exp})/err$

Gaussian Likelihood value → $LH = e^{(-pull^2)}/norm$

Combine information from all subsystems

→ $LH_{tot} = LH_{DRC\_Cerenkov} \times LH_{DRC\_PhotNum} \times LH_{DCH} \times LH_{TOF} \times LH_{SVT} \times LH_{EMC\_bwd}$

Construction of the Likelihood ratios

→ $\mathbf{LHR_{KvsPi} = LH_K/(LH_K+LH_{Pi})}$

→ cut on Likelihood ratios to select tracks

PidCalib application

Before tuning cuts to get reasonable selectors all ingredients have to be checked, for this reason PidCalib application was developed.

PidCalib use root – tuple generated by Single Track generator

PidCalibApp.C		main program
	Inputs	1) path to the root-tuple
		2) Tree name
	Output	root-file with all histoframs
PidCalib.C PidCalib.h		main class which make loop over Tracks and fill histograms
PidCalibConst.h		All main constants like number of momentum and theta bin are defined here
HistContainer.C HistContainer.h		Container of all histograms
PidCalibMakefile		make file make -f PidCalibMakefile
BuildPulls.C		use Output root-file to produce .ps files with plots

Checking Pulls

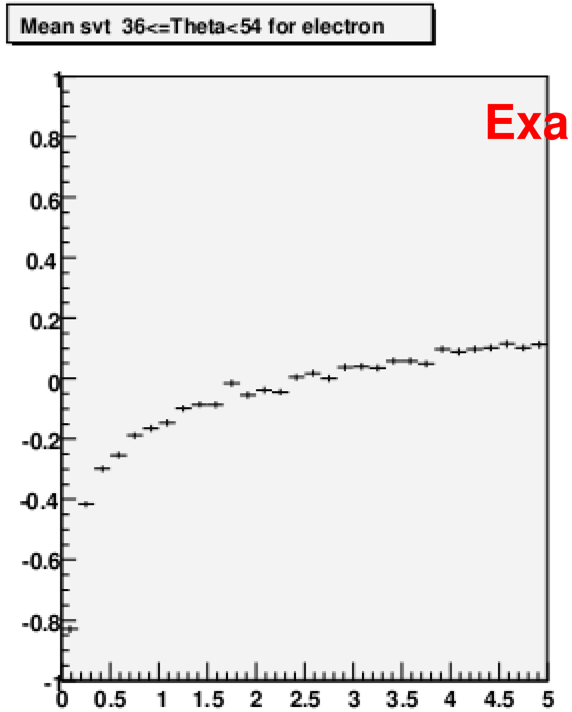
We concentrate on the BaBar configuration

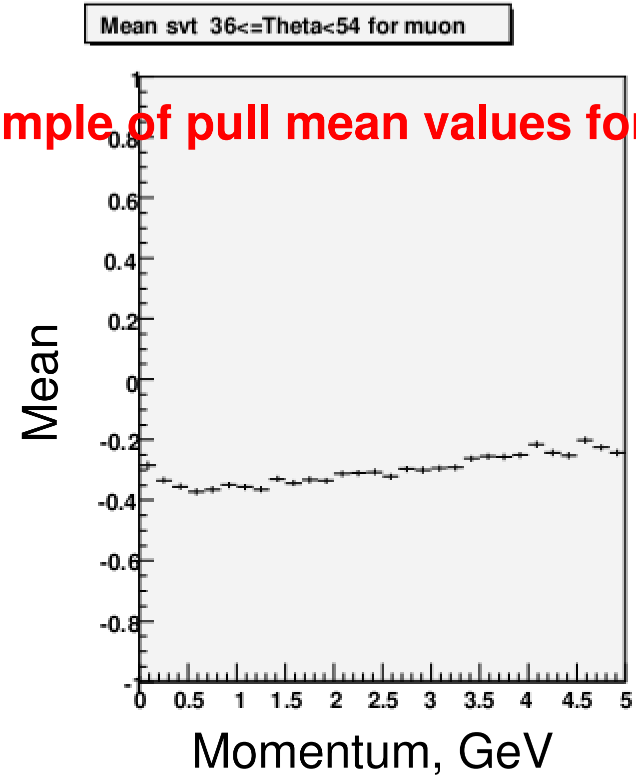
We have 30 momentum bins from 0.0 to 5.0 GeV
10 theta bins from 0.0 to 180.0°
5 particles
3 detectors

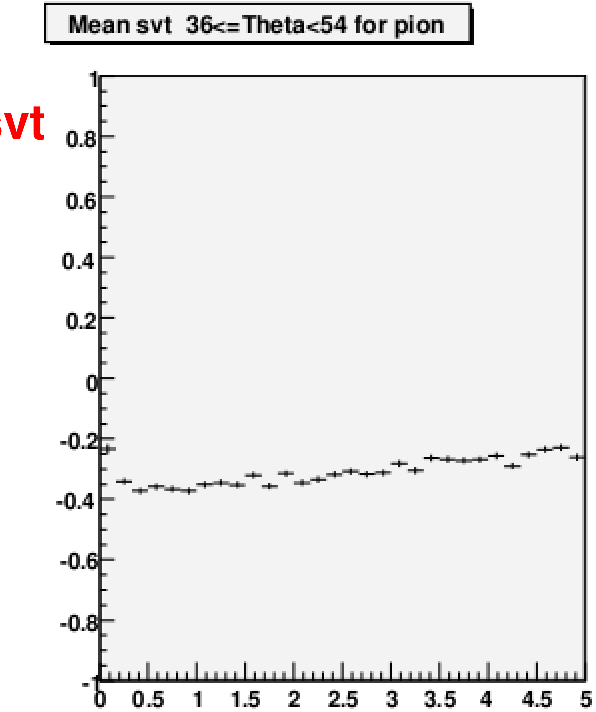
30 x 10 x 5 x 3 = 4500 histograms was filled up with pulls than fitted with Gaussian

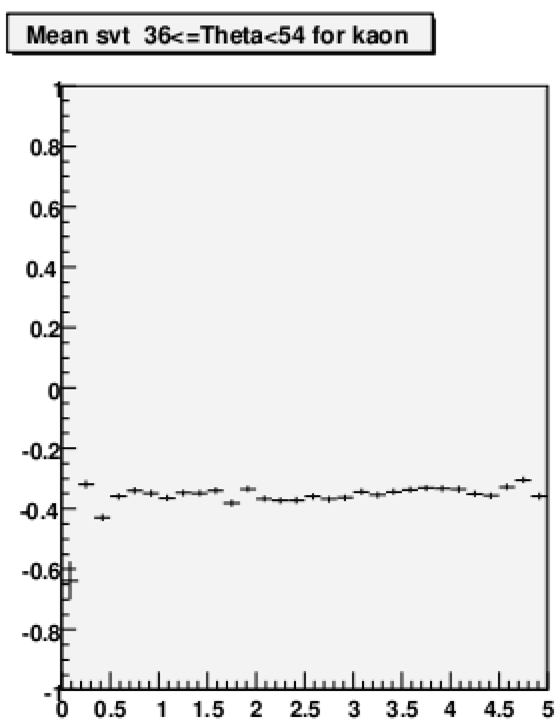
So Mean and RMS in different momentum and theta regions for given detector and particle was obtained

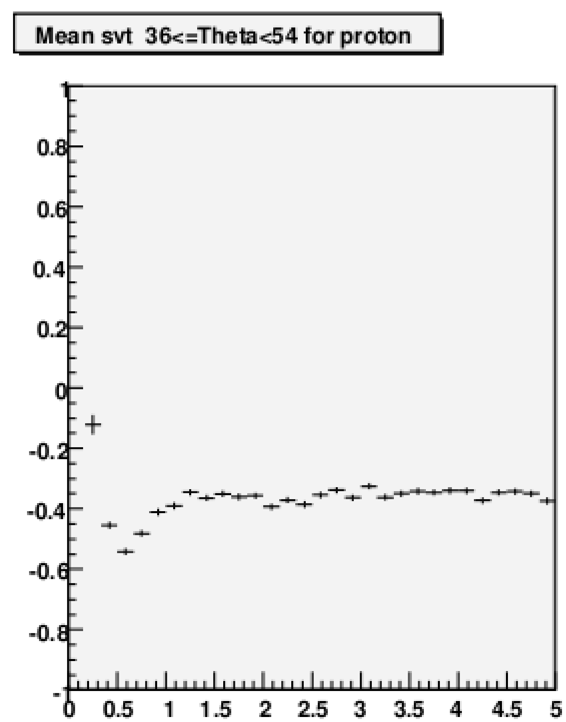
Example of pull mean values for svt









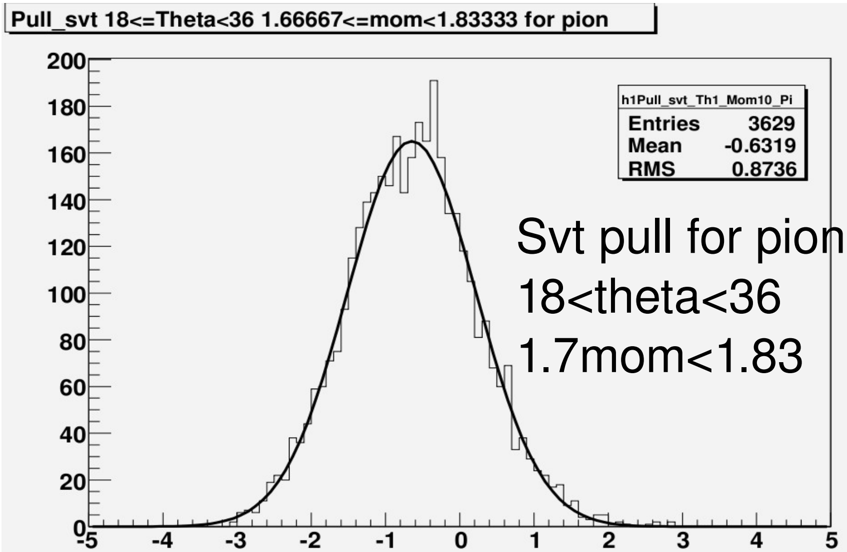


Mean of the pull vs momentum for SVT

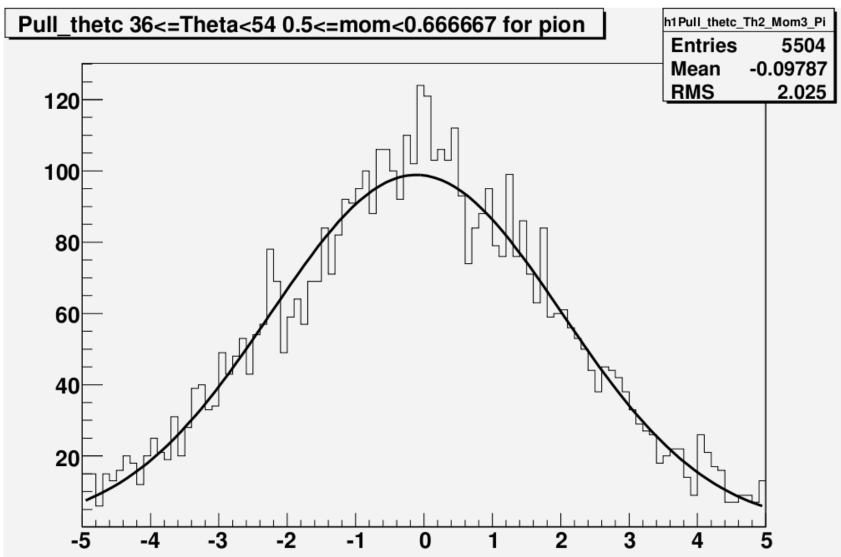
36^o<theta<54^o

BaBar configuration

Problems we found



SVT:
problems with mean and RMS
Mean is shifted by not negligible value which depend from theta region.
RMS is 0.8 (can it be due to truncation???)



DRC:
problems with mean and RMS
Mean is shifted in low momentum region
RMS is badly defined in all momentum and theta regions

DCH:
In general very good, but there is problems in low momentum region
This is problem which come from the fact that expected dE/dx is shifted (badly reconstructed momentum).

To do list

- Compare performances of the selectors in different theta regions with BaBar tables
- Retune selectors
- Add information in to the wiki page about PidCalib
- Add validation plots

Backup

Pull is root of the LH - based selector

Pull characterize by 3 main parameters

- Mean – should be equal to 0 if not (expected value is not well calculated)
- RMS – should be equal to 1 if not (expected value of error is not well defined)
- Shape – should be Gaussian since we you Gaussian Likelihoods

If mean of the pulls is shifted differently for two particle hypothesis we loose in separation power.
One possible reason of this shifting be explained in next way.
Expected value usually is the function of reconstructed momentum.
Than if error of momentum is the function of momentum this shift would be exist.

If RMS is > 1 expected error is less than it should be.
One possible reason is that usually error of expected value is not taken into account.