Track reconstruction of cosmic ray real data with the CMS tracker

Piergiulio Lenzi

Dipartimento di Fisica & INFN Firenze

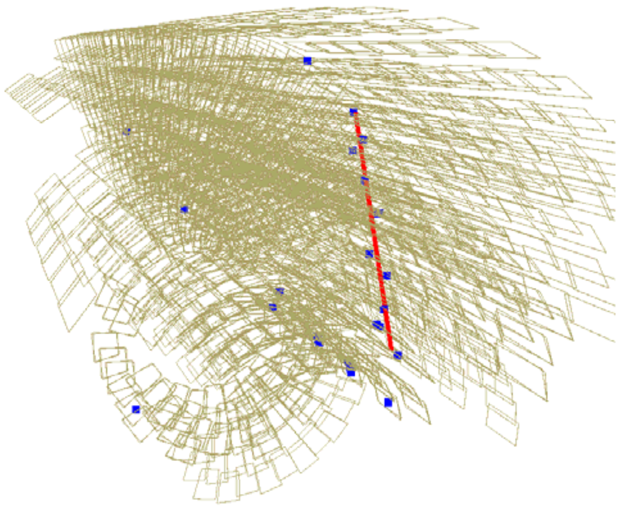
on behalf of the

Silicon Strip Tracker Collaboration

Firenze, RD07 Conference, 27th June 2007

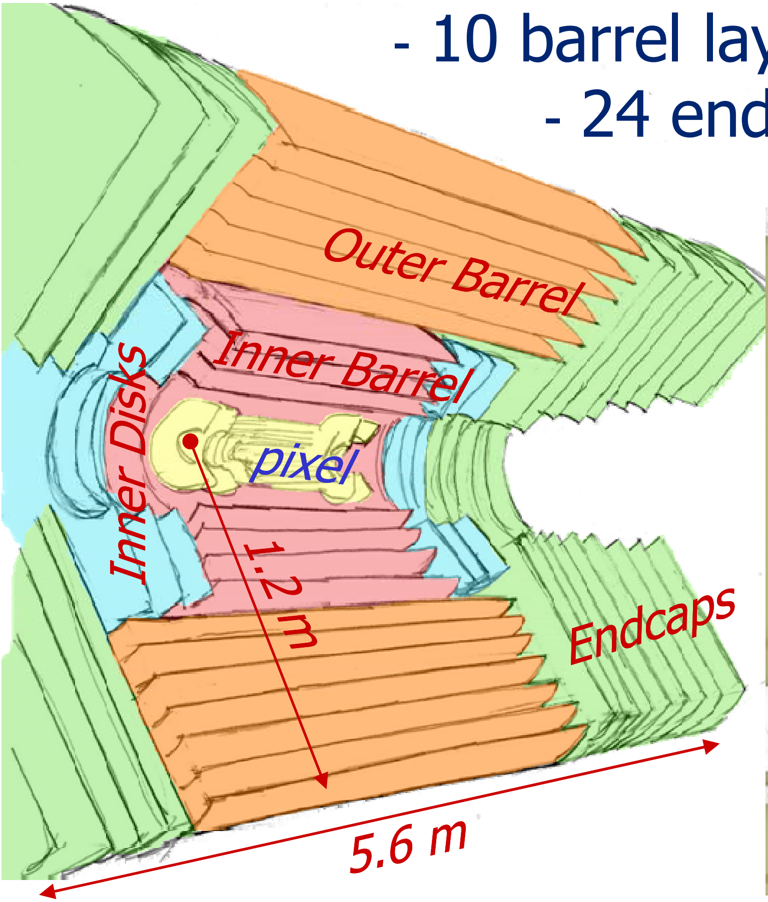
Outline

- The CMS Silicon Strip Tracker
- Overview of the final stages of the construction
- Cosmic Ray data taking
- Track reconstruction with the Combinatorial Track Finder Algorithm
- Track related analyses



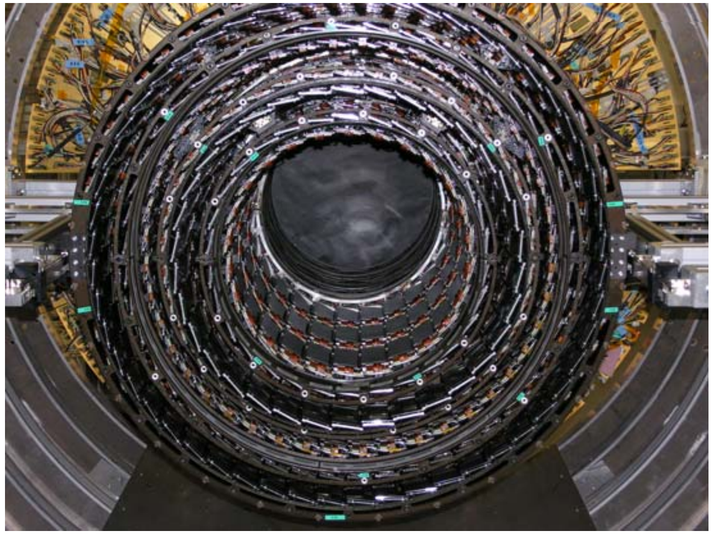
The Silicon Strip Tracker

- ~15000 detector modules
  - 10 barrel layers
    - 24 endcap disks

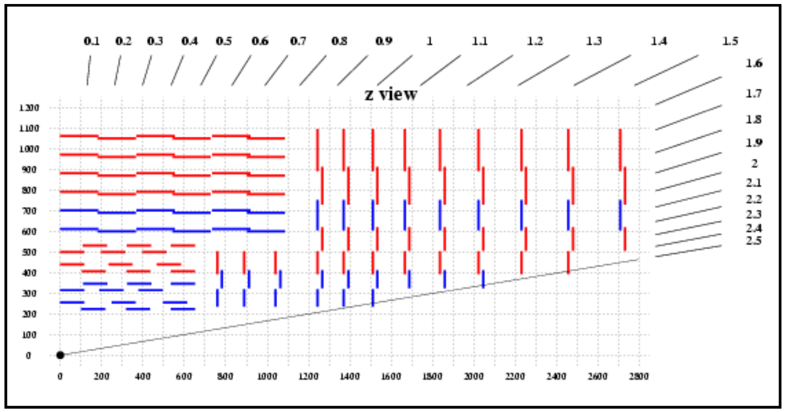


~ 9.6 million readout channels

A detector module



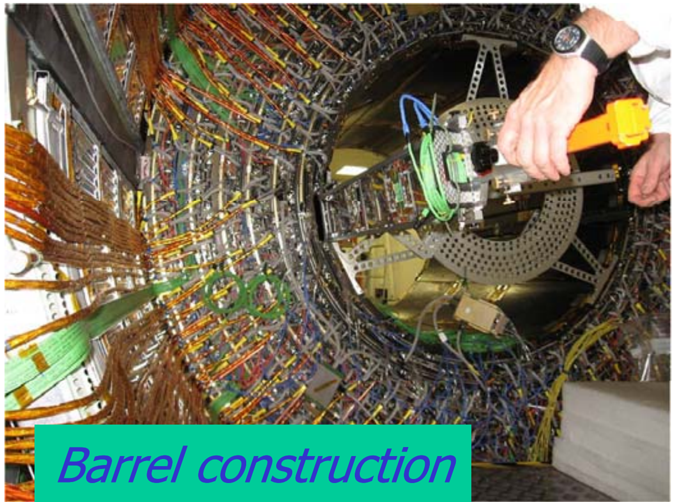
The inner barrel (TIB) before insertion



In red single sided detectors, in blue double sided ones

Final Integration stage

- The Silicon Strip Tracker is currently located at CERN building 186, in the Tracker Integration Facility (TIF)



Barrel construction

Barrel part fully integrated

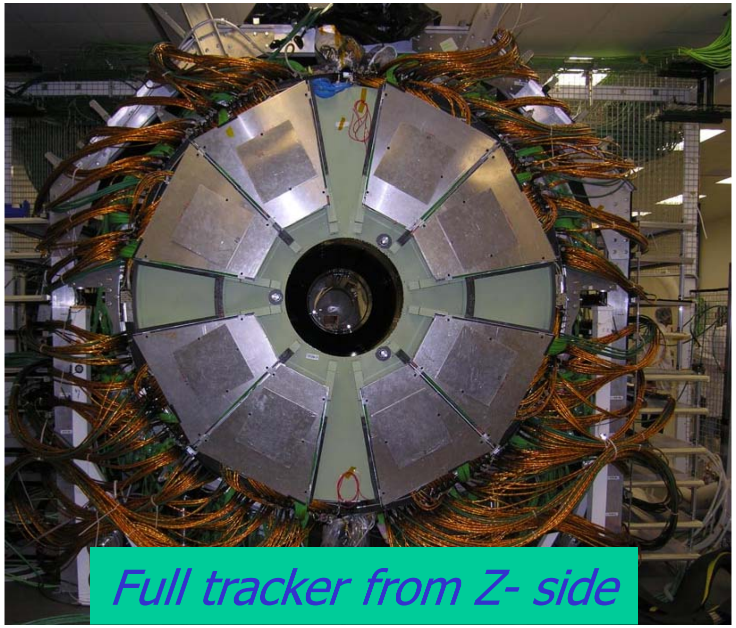
Endcap insertion

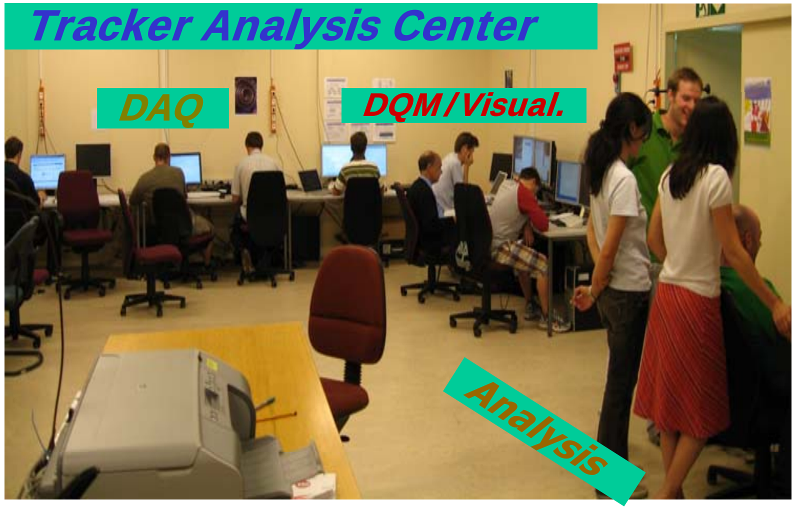
The Quality of the CMS Tracker Sub-Detectors is Excellent:

- Dead or Noisy Strips < 0.3%
- Signal/Noise > 25 in Peak Readout Mode
- Same Quality maintained throughout the Integration of the Sub-Detectors into the CMS Tracker

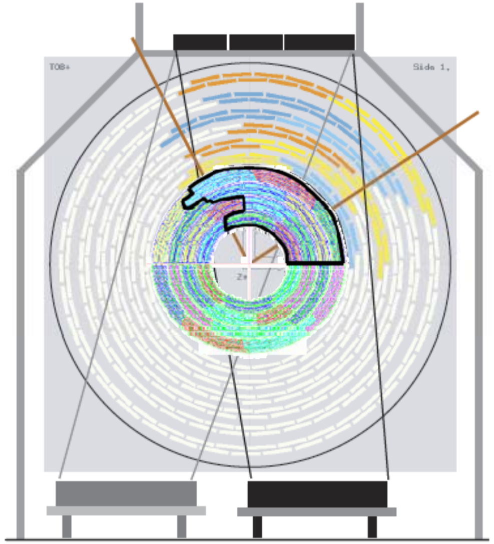
Motivations of the test

- The test has been carried out to deeply investigate the tracker behavior in final-like conditions
  - From the hardware point of view
    - More or less 14% of the tracker operated as a whole
    - Long runs
    - Final DAQ and Detector Control System
  - From the reconstruction point of view
    - Data quality monitoring
    - Track reconstruction
    - Offline studies of the detector behavior using track information
    - Alignment

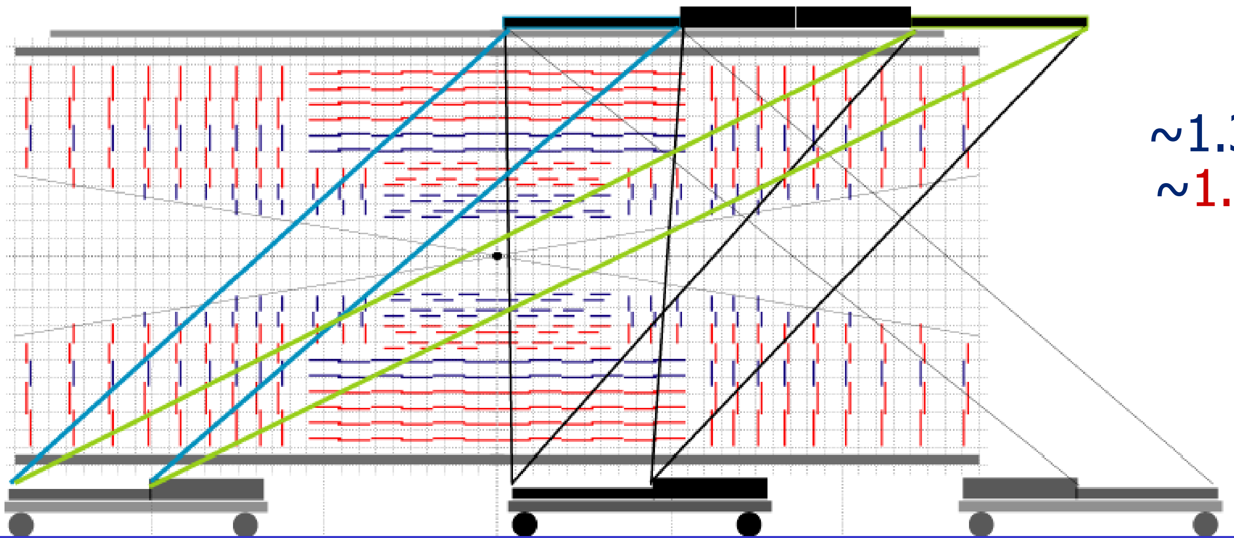




Cosmic Ray data taking



- A cosmic ray trigger has been setup using plastic scintillators
- Scintillators placed
  - on top of Tracker Support Tube
    - 4 scintillators (100x30 cm²)
    - 2 scintillators (100x40 cm²)
  - bottom of TST : 1 scintillator (80x75 cm^2)
- 5 cm lead put in place on bottom scintillator: (Stopping power ~180MeV)
- Trigger Rate
  - on TOB/TIB: ~3.8 Hz (pos. 1) ~1.1 Hz (pos. 3)
  - on TEC+: ~1.5 Hz (pos. 2 ; Angle ~45°)



~1.3 million channels read
~1.8 times the CDF tracker

The reconstruction chain

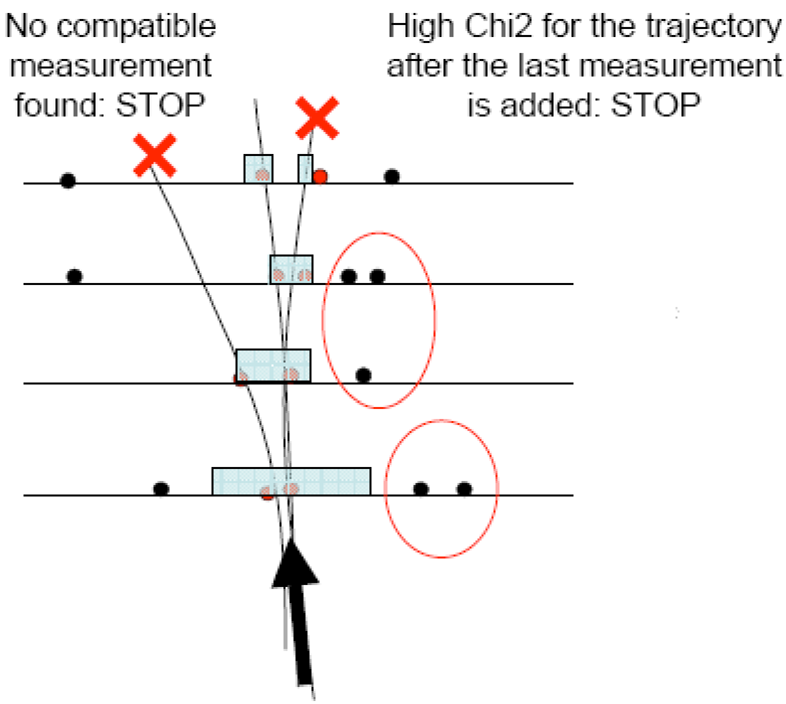
- Reconstruction has been carried out in the official CMS analysis framework (CMSSW)
- It schematically follows these stages:
  - Unpacking of raw data collected from the ADCs
  - Cluster reconstruction
  - Estimation of local hit position (RecHit)
  - Track reconstruction
    - Seeding
    - Pattern recognition
    - Track fitting
- The track reconstruction has been done with 3 different algorithms
  - Combinatorial Track Finder (default CMS)
  - Road Search (default CMS)
  - Cosmic Track Finder
- I will talk about the Combinatorial Track Finder

Track Seeding

- A seed is the starting trajectory segment
- In the standard tracking the seed is built out of a couple of hits in the inner layers plus the vertex constraint OR out of a hit triplet in the inner layers
- *Cosmic Ray peculiarity: the seed has to be built from both inner and outer layers without any vertex constraint*
  - We developed a new seed provider to handle non standard conditions
  - It was written in a sufficiently general way to handle also other situations, such as beam halo muons' tracking and V0 reconstruction

Pattern Recognition

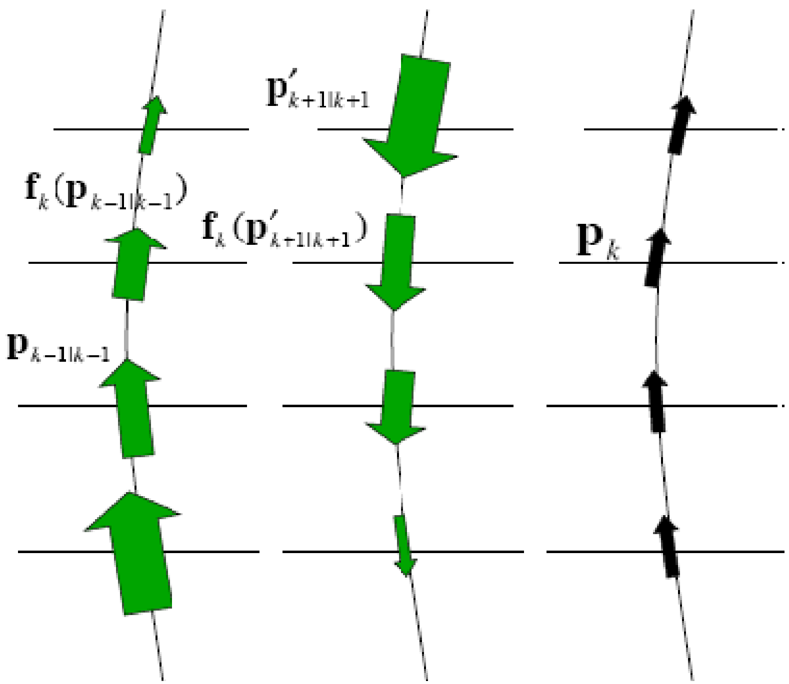
- It constits in the recognition of the hits belonging to a track out of all the hits in the event
- It produces Track Candidates



- *Cosmic Ray peculiarity:*
  - *The tracker is hermetic and offers optimal superposition only for tracks coming from the origin*
  - *For the cosmics it has holes as well as zones of high detector superposition*
  - *Cosmic tracks cross the entire tracker from side to side. We had to make the pattern recognition sufficiently general to do that*

Track Fitting

- The collection of hits belonging to each Track Candidate is passed to the Kalman Filter fitter
- Two stages
  - *Fitting*
  - *Smoothing*



- During the Fitting stage the state vector on each surface is calculated using all the previous hits
- Only the last state employs the full track information
- All the hits are fitted again starting from the last hit
- The forward and backward fit informations are combined to give on each surface an estimate of the state vector employing the full information available
- *Nothing different is needed for the cosmics*

Simulation: efficiency

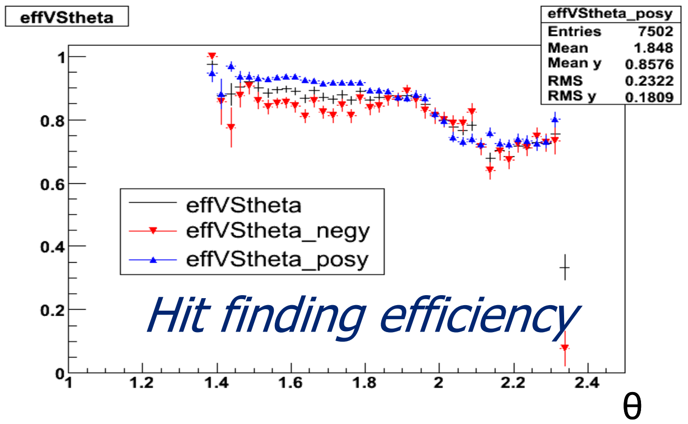
- A cosmic ray simulation has been setup, implementing the cosmic ray spectrum down to 200 MeV
- The CMS cosmic ray generator has been used together with a filter to mimic trigger configuration
- An overall tracking efficiency of (98.7±0.1)% has been measured on the generated sample
- Some inefficiency detected mainly in the forward region

Seed ε (%)	Track ε (%)
99.1±0.1	98.7±0.1 (99.6±0.1)

Seeding and tracking efficiences, in () the algorithmic efficiency

Simulation: efficiency

- The trajectory builder currently used in the standard CTF tracking takes one hit per layer
- For the cosmics we use a different trajectory builder which can take more than one hit per layer (overlaps), and that will become soon the default one for the CTF
- Cosmics are very useful for this, due to the high superposition
- The hit finding efficiency is a bit lower, but it does not spoil the tracking efficiency (at least one hit per layer is taken)

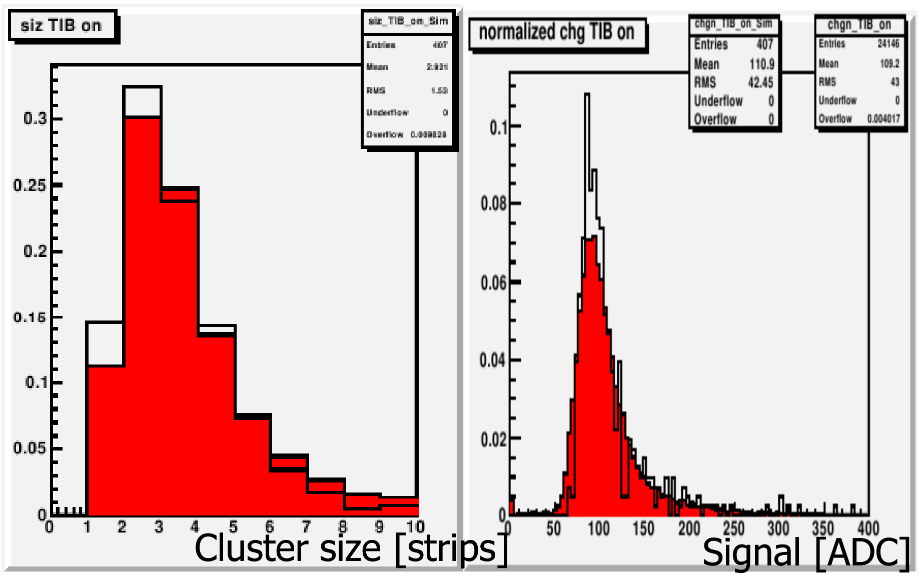


- *Hit finding efficiency as a function of track θ:*
  - *Blue: upper part of the tracker*
  - *Red: lower part of the tracker*
  - *Black: total*
- *At the moment unable to recover overlaps in the seeding region. This feature will be added soon*

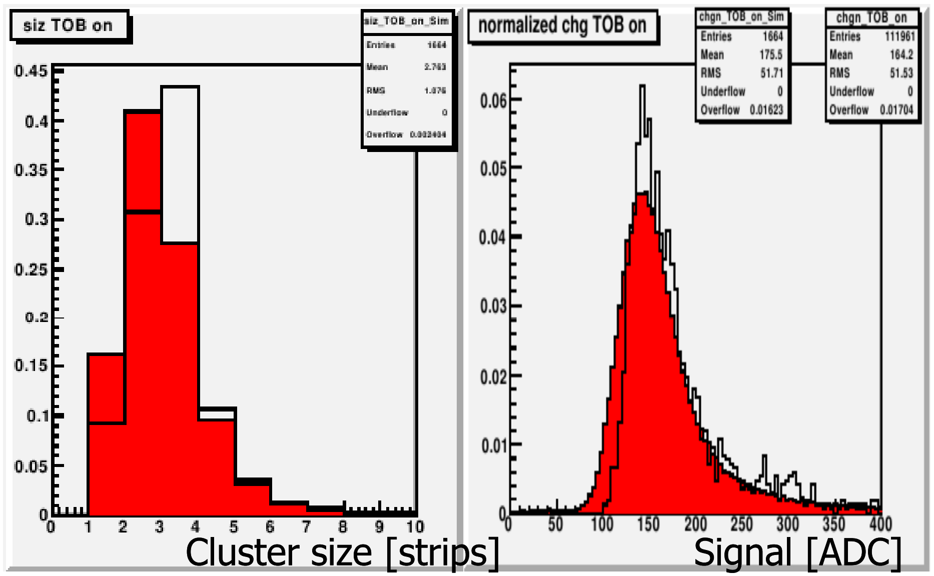
Data-MC Comparison

- The excercise has been very useful to verify the detector simulation
- Here you can see a couple of examples:
  - Cluster charge for clusters on track (Red→data, black→MC)
  - Cluster size for clusters on track (Red→data, black→MC)
- Good agreement; important for simulation tuning

Inner Barrel

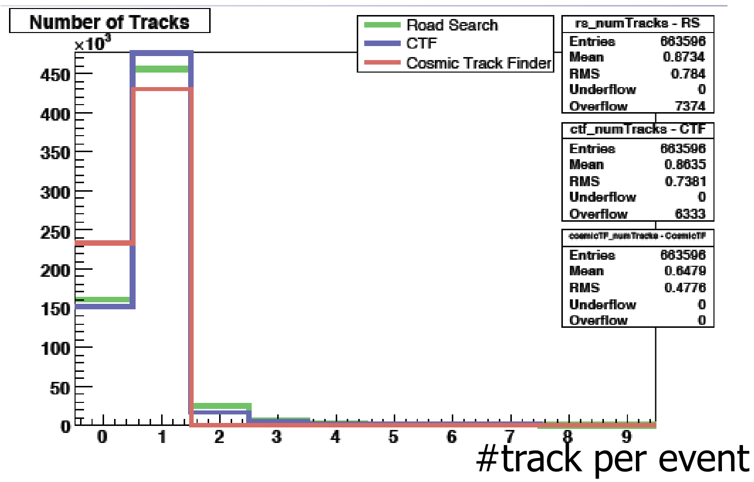


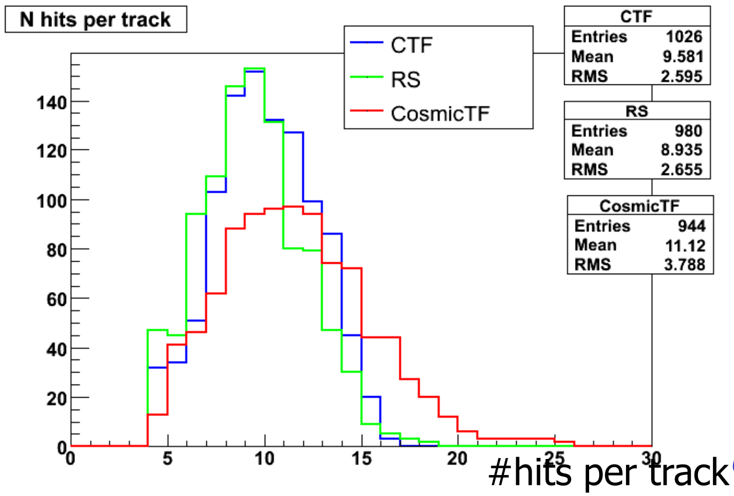
Outer Barrel



Data: some track related quantities

- About 3.5 Million events recorded; data taking still ongoing, we will take data till the first half of July
- The number of events with at least one reconstructed track is roughly 0.6 X number of triggers



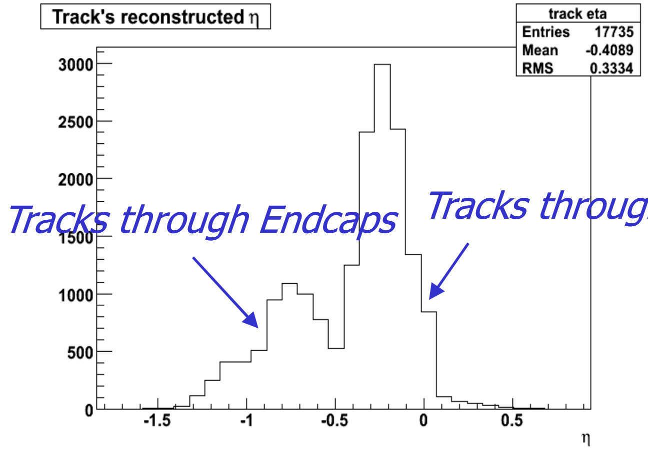


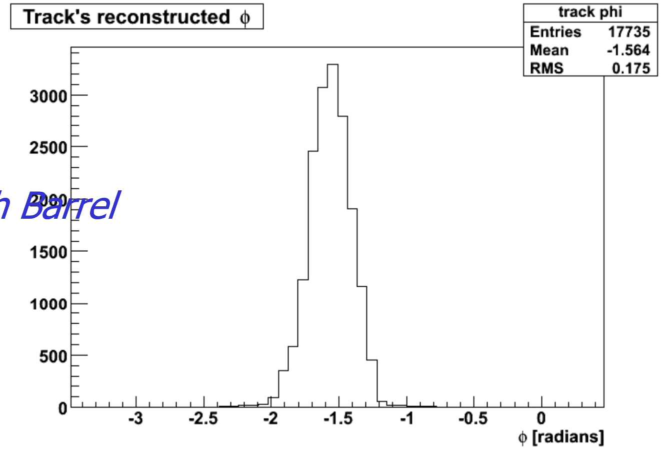
C. Noeding

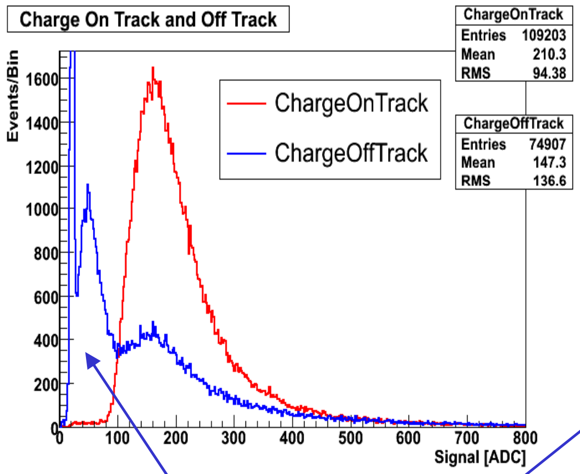
Number of reconstructed tracks per event and Number of hits per track in a typical run. The 3 algorithms give similar results (CTF is in blue), considering that the χ^2 cut is different. In the next reprocessing we will use the same cuts for all the algorithms

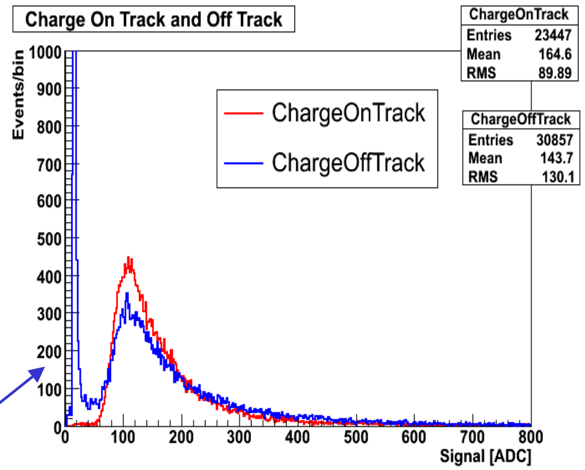
Some track related quantities

- η and φ distribution, consistent with what is expected from scintillators' position









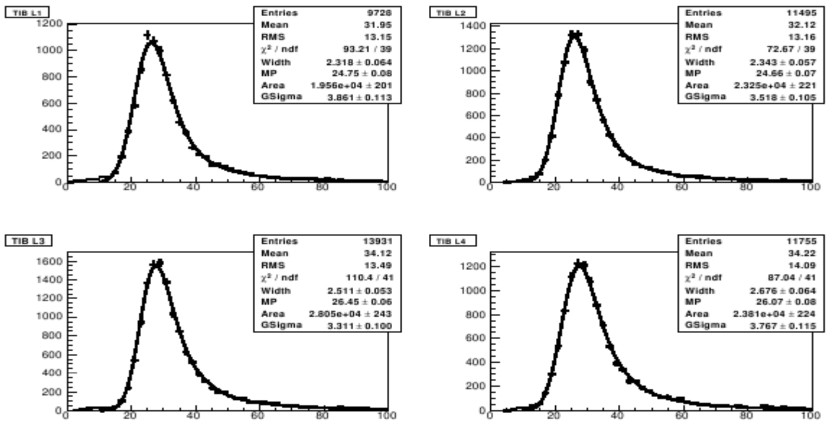
Good noisy hits rejection

- On track and Off track cluster charge distributions
  - Signal clusters not associated to the tracks are due to
    - Misalignment
    - Low energy tracks with high multiple scattering. We don't have any momentum estimate

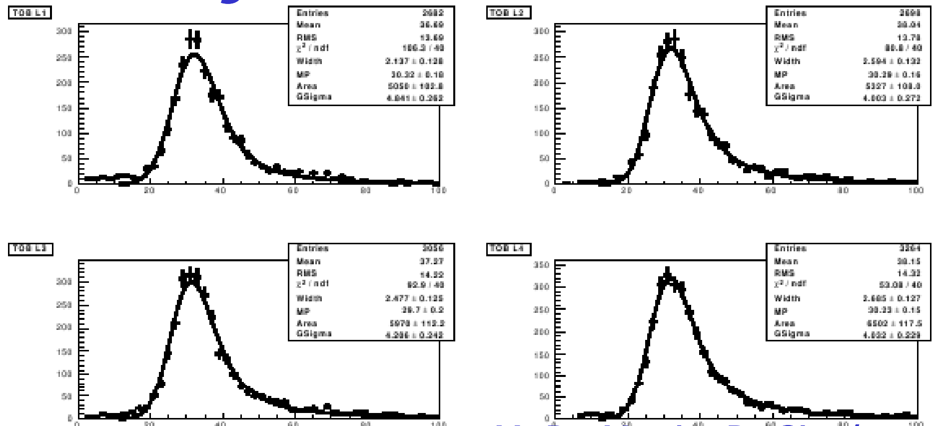
Signal to Noise Ratio

- The track information has been very useful also to investigate detector level quantities
- Signal to noise ratios corrected for track angle:
  - Inner Barrel→~25
  - Outer Barrel→~38
- Consistent with the expectation
- It confirms that the quality of the performances has been maintained on the large scale

Signal to Noise for Inner Barrel

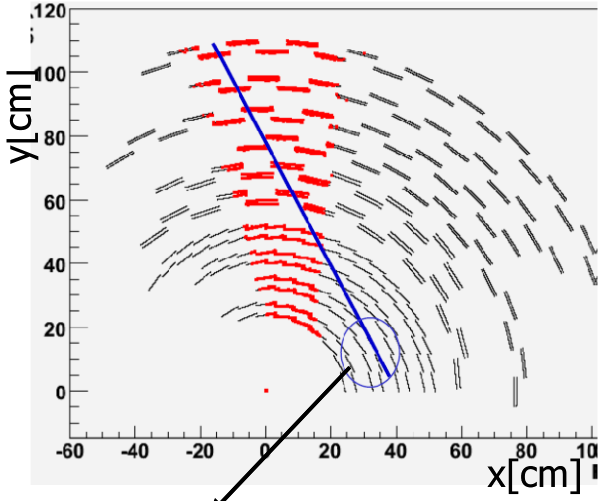


Signal to Noise for Outer Barrel



M. De Mattia, D. Giordano

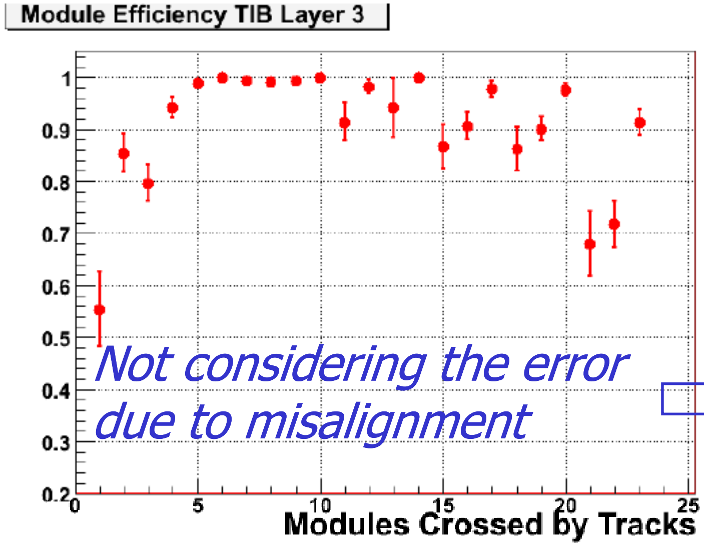
Hit finding efficiency on the data



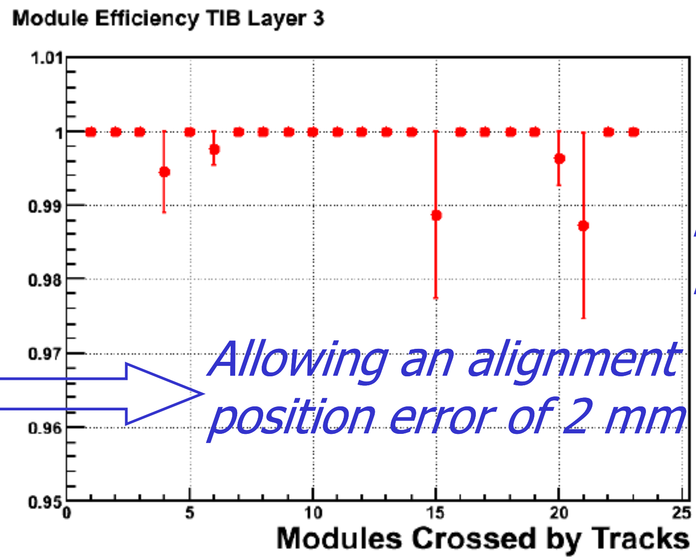
Not selected

- Selected a fiducial region for high quality tracks (no shallow incidence, good scintillator coverage…)
- The hit finding efficiency is defined as :

$$\frac{\text{Number of hits found in the predicted position}}{\text{Number of predicted hit positions}}$$



Not considering the error due to misalignment



Allowing an alignment position error of 2 mm

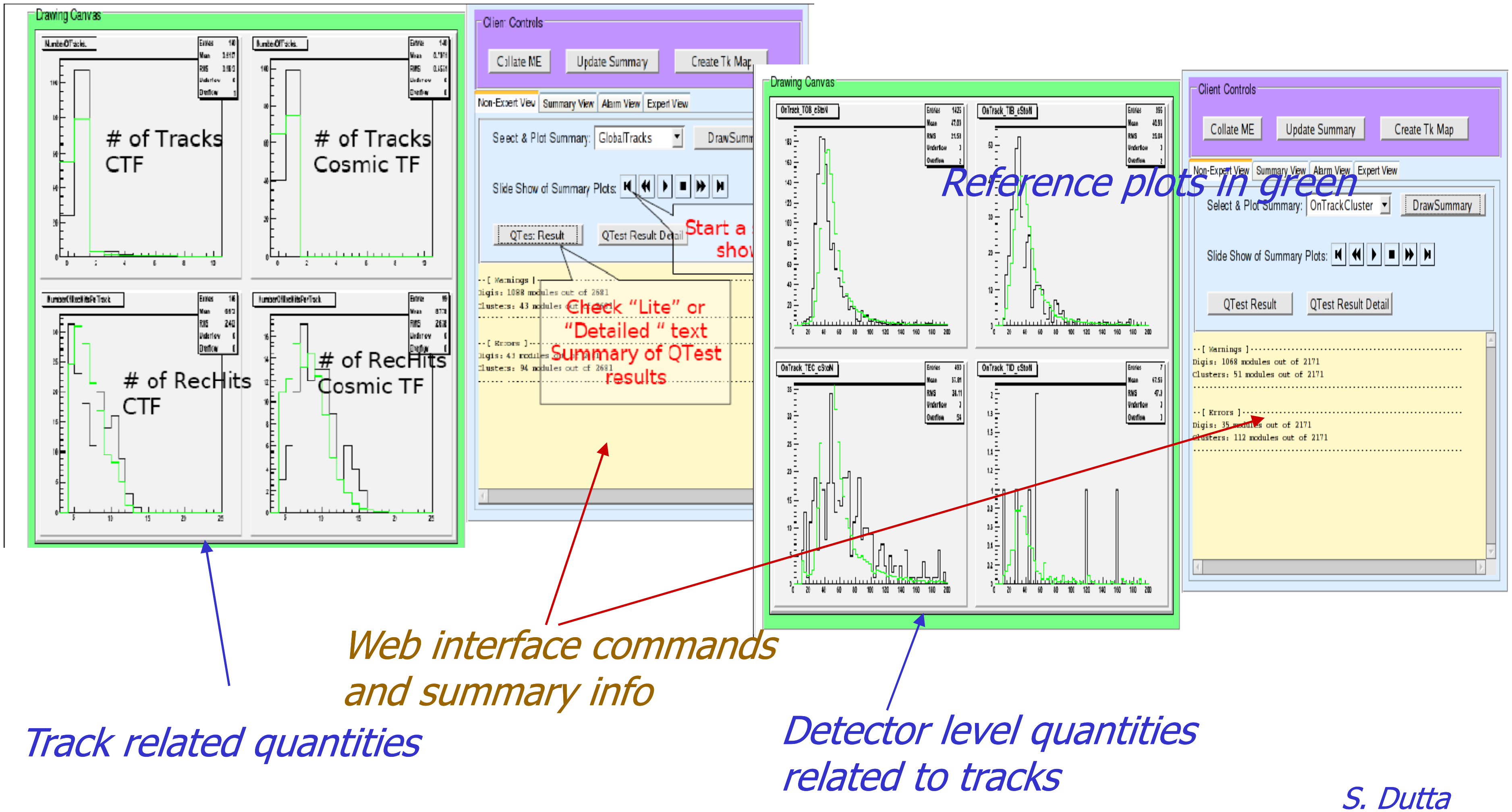
Taking the misalignment effect into account we recover efficiency

D. Benedetti

Online track reconstruction

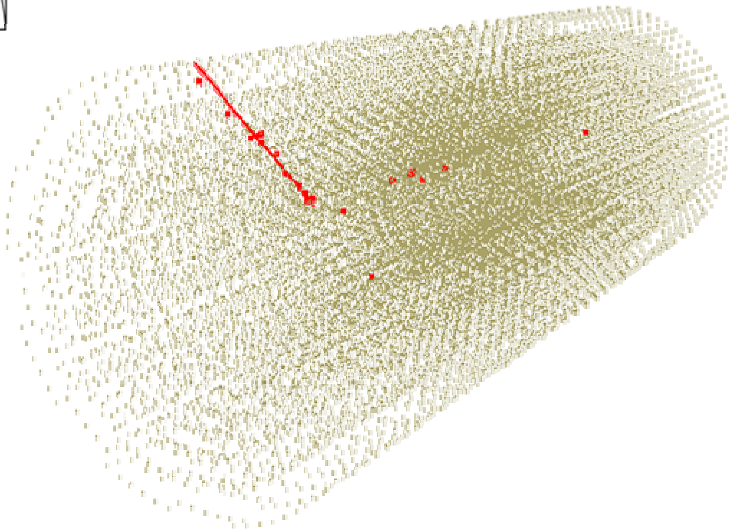
- Track reconstruction has been run online for the first time
- Shifters could run online Data Quality Monitoring (DQM) to check the status of data taking
- Plots are available through a web interface
- The DQM allowed the monitoring of
  - local quantities
    - Cluster charge and noise
    - Charge On track and Off Track
    - Residuals
  - Global quantities
    - Track parameters (eta, phi….)
    - Number of associated hits

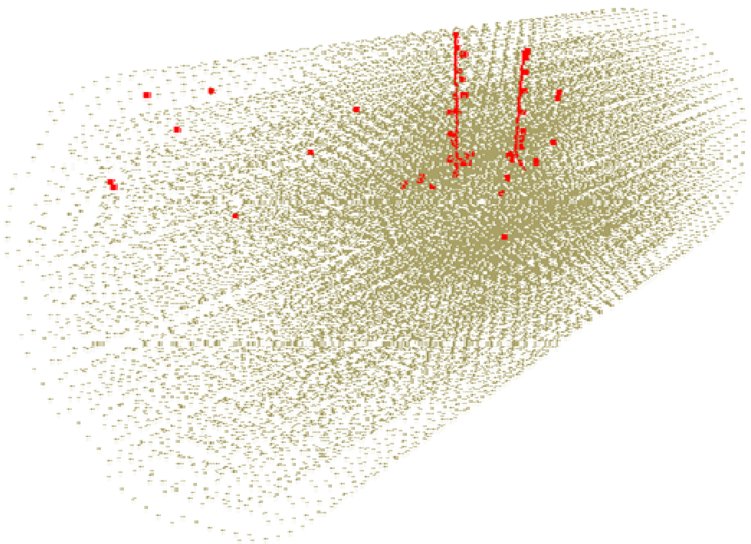
Online track reconstruction

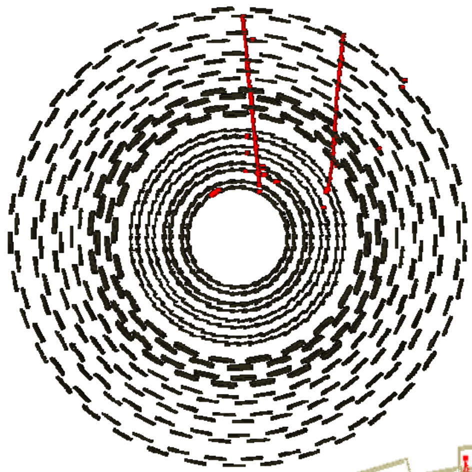


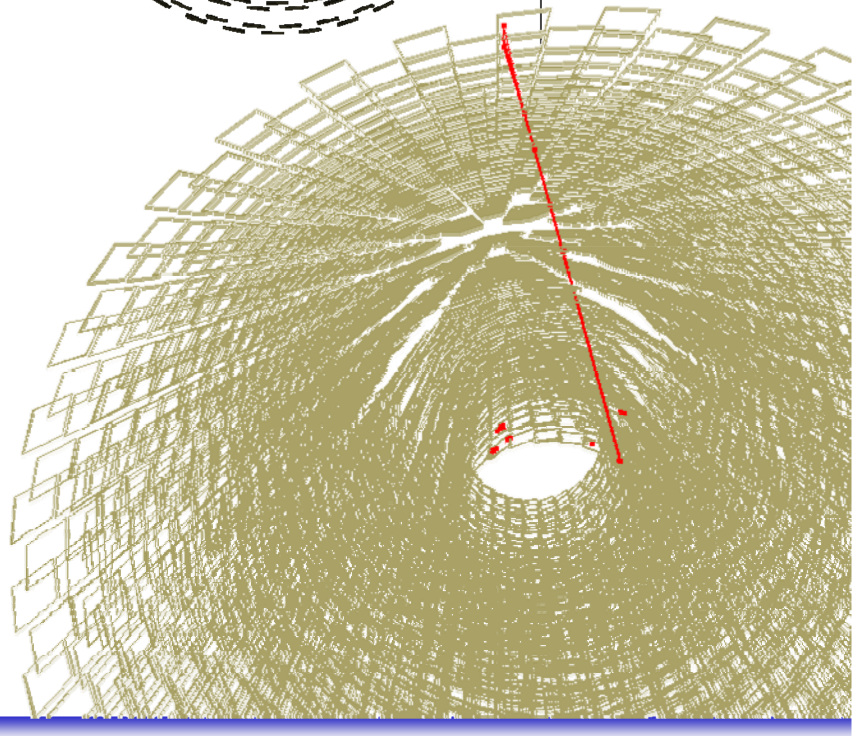
S. Dutta

Some event displays









Foreseen analysis

- The cosmic ray data will be used to test the alignment algorithms
- Cosmic ray tracking will be very usefull also for the tracker commissioning in the CMS cavern and will give an important contribution for the final detector alignment
- dE/dx studies are ongoing

Conclusions

- In the past few months, for the first time, a big part of the CMS tracker has been operated as a whole, and the first real data from cosmics have been recorded
- The CMS Combinatorial Track Finder Algorithm has been adapted to handle the cosmic ray case
- For the first time it has been run on real data, both online and offline
- Various track related analyses have been developed, thus providing useful tools for the understanding of the detector behavior
- Cosmic ray tracking will be very useful also for the P5 commissioning and alignment

Backup

The Kalman Filter

- The CTF is based on the Kalman Filter algorithm
  - The Kalman Filter is a “dynamic” Least Squares method
  - It follows three stages:
  - *Prediction*
    - On each detector surface k the track is described by a state vector $\vec{p}_k$ and is linked to the previous state by an evolution equation $\vec{f}$ plus a scattering contribution $\vec{w}$

$$\vec{p}_{k|k-1} = \vec{f}(\vec{p}_{k-1}) + \vec{w}_{k-1}$$

    - The vector of measurement $\vec{m}$ on the surface is linked to the state vector by

$$\vec{m}_k = \vec{h}(\vec{p}_k) + \vec{\varepsilon}_k$$

    - In linear approximation the preceeding equations are

$$\vec{p}_{k|k-1} = F_k \vec{p}_{k-1} + \vec{W}_{k-1} \qquad \vec{m}_k = H_k \vec{p}_{k-1} + \vec{\varepsilon}_{k-1}$$

    - The covariance matrix of $\vec{p}_k$ is calculated with linear error propagation

The Kalman Filter

- *Update*
  - The information concerning the hit on the k layer is used (V_k is the covariance matix of the measurement, while C_k is the covariance matrix of the predicted state)

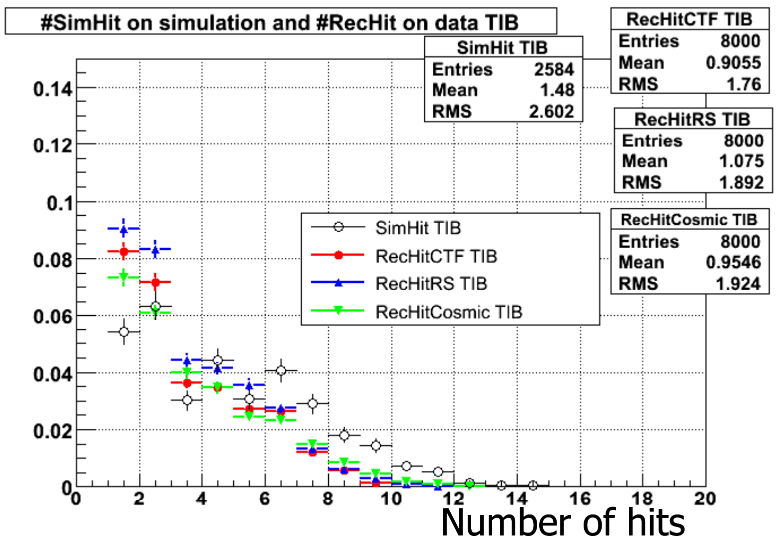
$$\vec{p}_{k|k} = \vec{p}_{k|k-1} + K_k(\vec{m}_k - H_k\vec{p}_{k|k-1}) \qquad K_k = (C_k^{-1} + H_k^T V_k^{-1} H_k)^{-1} H_k^T V_k^{-1}$$

- *Smoothing*
  - After the update, all the states contain the information form all the preceeding hits plus the one on the current surface
  - In order to employ the full information from all the track hits for all the states, the prediction and update are also done in the reverse way, and the two results are combined

Data-MC comparison

- Number of simulated hits belonging to the simulated track and number of hits associated to the reconstructed tracks in data are compared.
- In the plots all the 3 algorithms are reported (CTF in Red)
- Good agreement among the 3 track finders; some discrepancy with the MC due to misalignment

Inner Barrel



Outer Barrel