Tracks Reconstruction Performance

Karol Krizka
on behalf of many people

October 12, 2022

International MUON Collider Collaboration

UNIVERSITY OF BIRMINGHAM

MCC Meeting

Our Onion Detector

Details

heavily based on CLIC detector

hadronic calorimeter

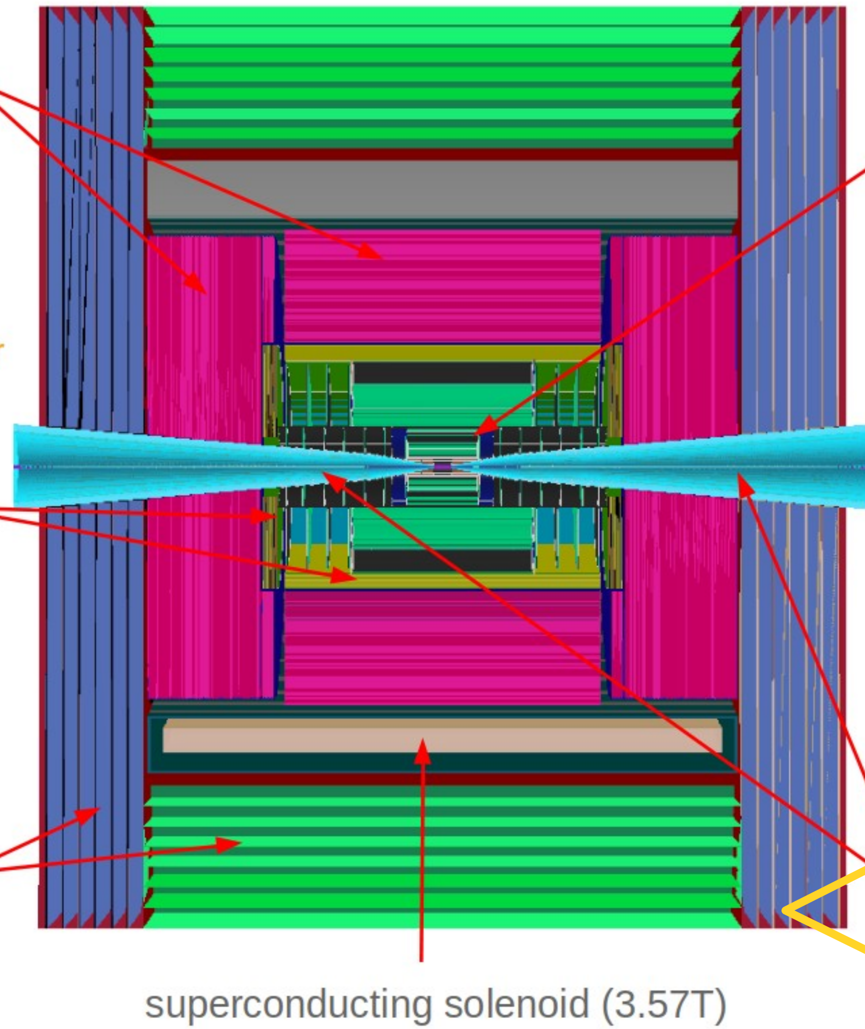
- 60 layers of 19-mm steel absorber + plastic scintillating tiles;
- 30x30 mm^2 cell size;
- 7.5 λ_I.

electromagnetic calorimeter

- 40 layers of 1.9-mm W absorber + silicon pad sensors;
- 5x5 mm^2 cell granularity;
- 22 X_0 + 1 λ_I.

muon detectors

- 7-barrel, 6-endcap RPC layers interleaved in the magnet's iron yoke;
- 30x30 mm^2 cell size.



tracking system

- Vertex Detector:
  - double-sensor layers (4 barrel cylinders and 4+4 endcap disks);
  - 25x25 μm^2 pixel Si sensors.
- Inner Tracker:
  - 3 barrel layers and 7+7 endcap disks;
  - 50 μm x 1 mm macro-pixel Si sensors.
- Outer Tracker:
  - 3 barrel layers and 4+4 endcap disks;
  - 50 μm x 10 mm micro-strip Si sensors.

shielding nozzles

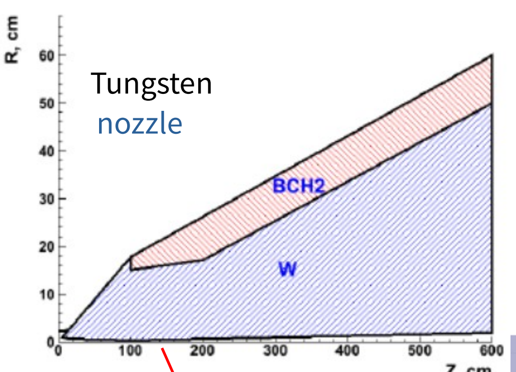
- Tungsten cones + borated polyethylene cladding.

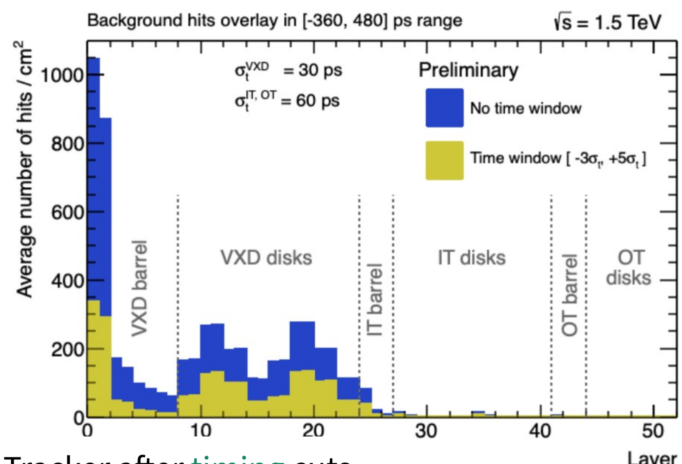
Beam Induced Background

arxiv:2105.09116

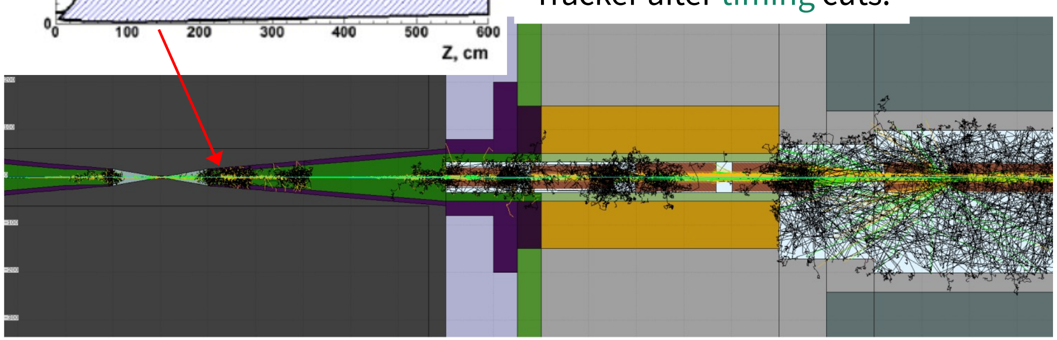
- **BIB = muon beam decay and strike the detector**
- **Several main mitigation**
  - 10° tungsten nozzle to shield from beam decay products
  - Precision timing information from detectors

Wed morning MDI session

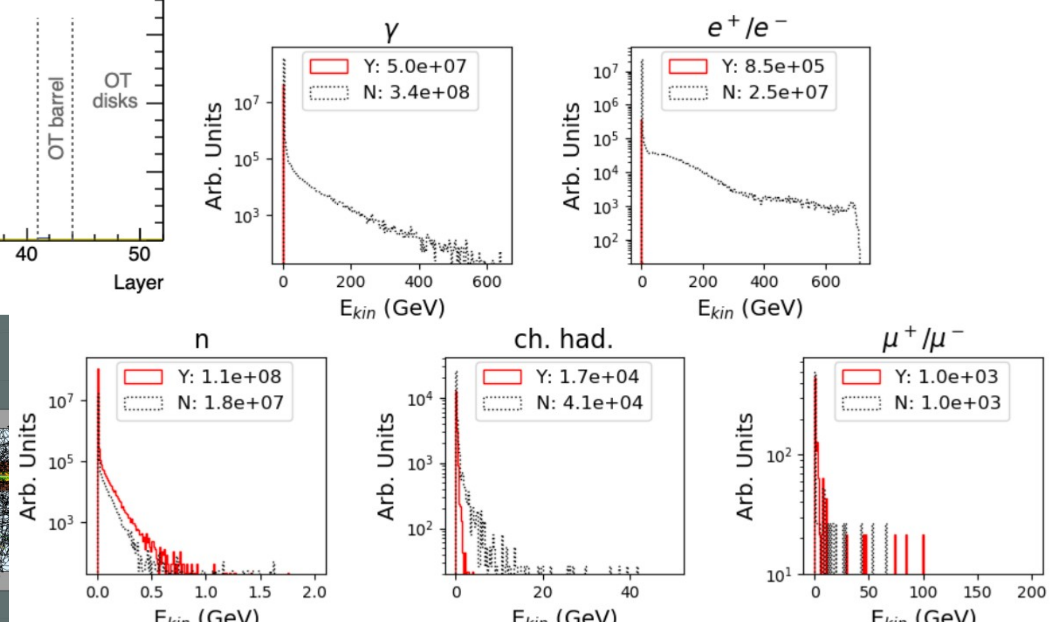




Tracker after timing cuts.

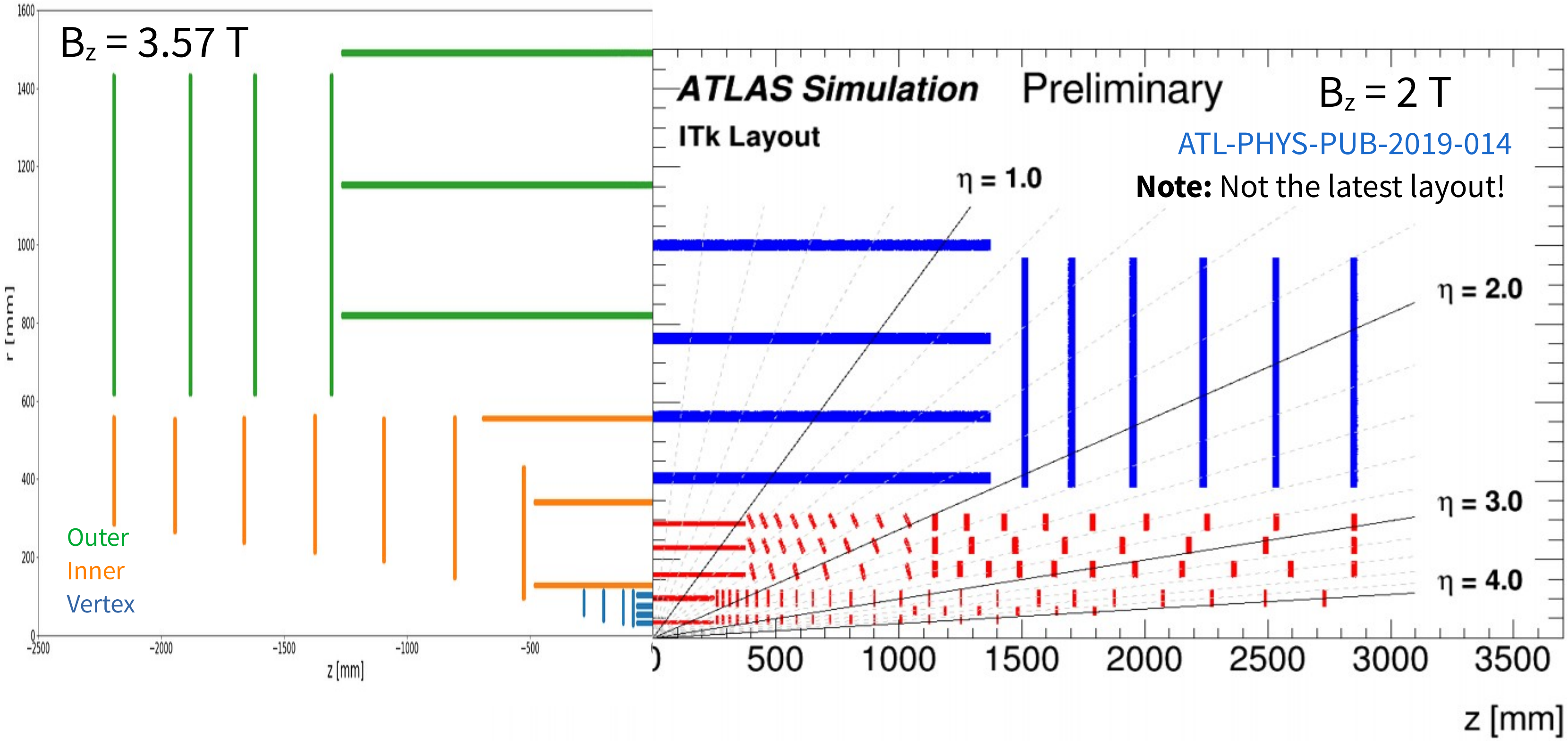


FLUKA simulation of BIB before reaching the detector.



Particle energy spectra with (Y) and without (N) nozzle.

The Scale of BIB

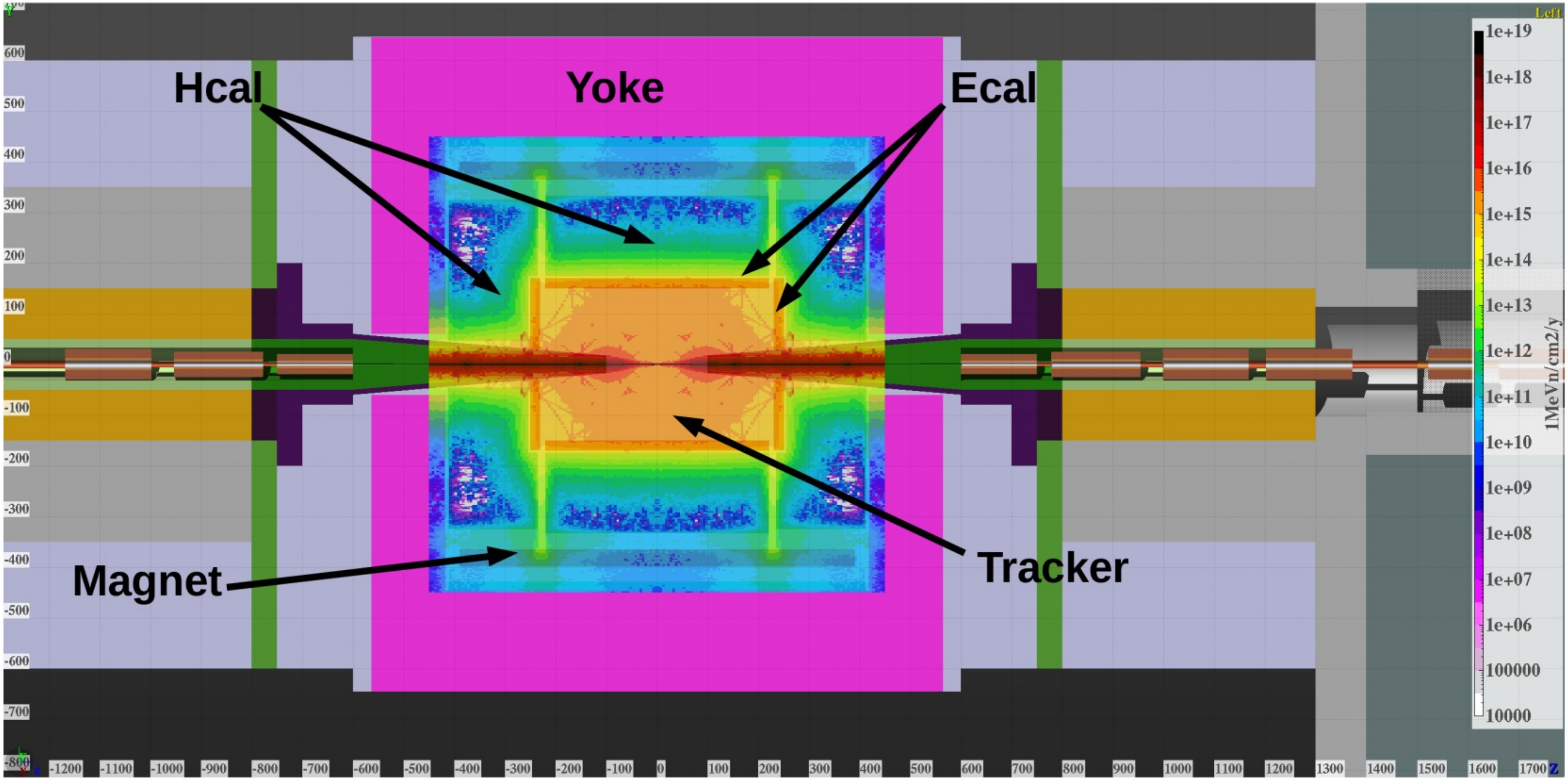


Hit density
after timing cuts
10x HL-LHC

	ITk Hit Density [mm^{-2}]	MCC Equiv. Hit Density [mm^{-2}]
Pix Lay 0	0.643	3.68
Pix Lay 1	0.022	0.51
Str Lay 1	0.003	0.03

ITk Pixels TDR, ITk Strips TDR

Radiation Damage From BIB

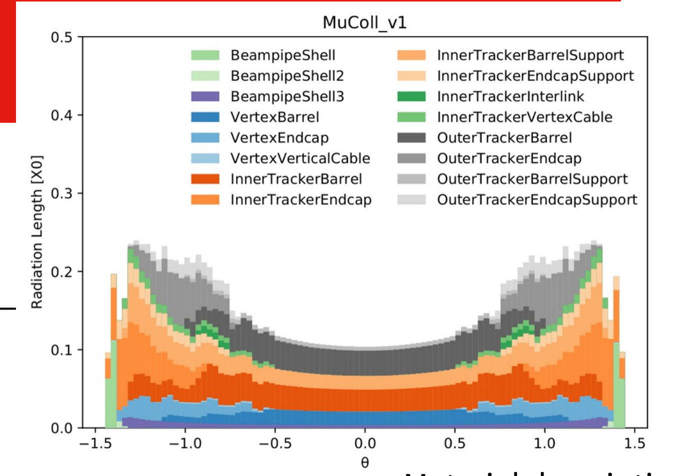


Dose comparable to full HL-LHC luminosity.

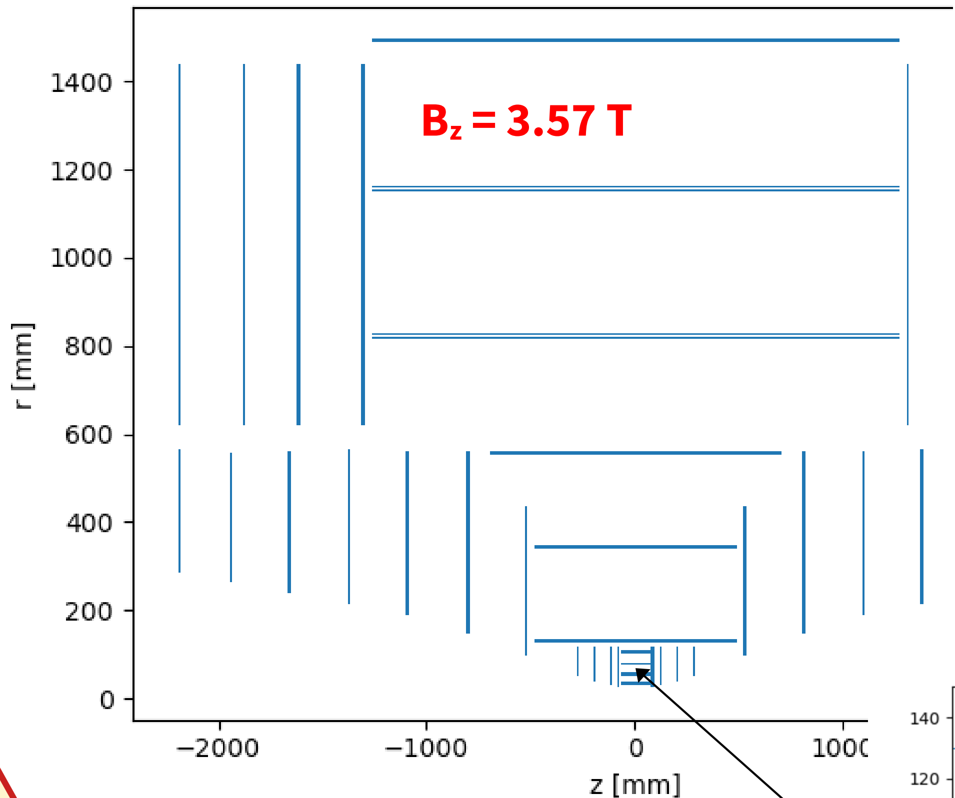
source

All-Silicon Tracking Detector

Details



Material description



Outer Tracker (OT)

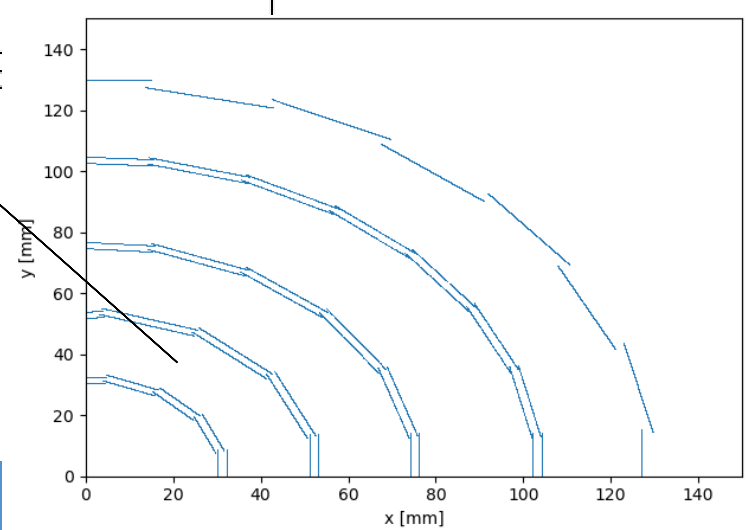
- micro-strips
- 50 µm x 10 mm
- σ_t = 60 ps

Inner Tracker (IT)

- macro-pixels
- 50 µm x 1 mm
- σ_t = 60 ps

Vertex Detector (VXD)

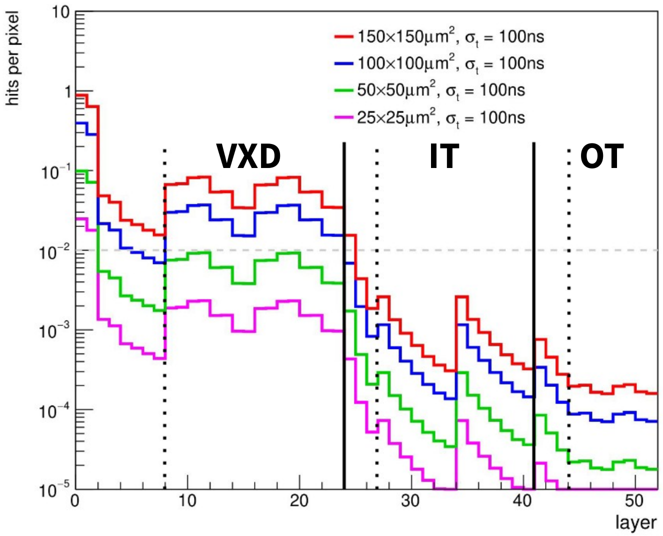
- pixels
- 25 µm x 25 µm
- σ_t = 30 ps
- double layers

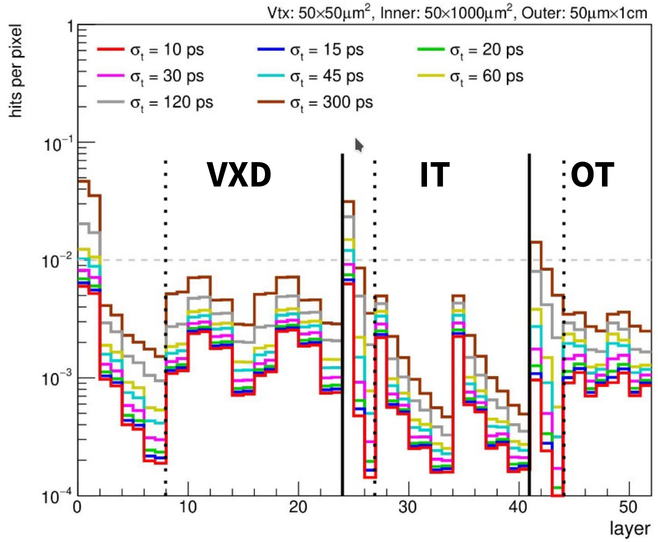


4D tracking critical

Pixel Size and Timing

Details





- **Goal is <1 % occupancy per pixel.**
  - Pixel size optimized to achieve this
  - Precision timing also plays important role
    - Needed for on-detector filtering (for readout)
- **Need to be careful about slow particles**
- **Resolutions are approximated in simulation using Gaussian smearing**

Current Assumptions

	Cell Size	Sensor Thickness	Time Resolution	Spatial Resolution
VXD	25 μm x 25 μm	50 μm	30 ps	5 μm x 5 μm
IT	50 μm x 1 mm	100 μm	60 ps	7 μm x 90 μm
OT	50 μm x 10 mm	100 μm	60 ps	7 μm x 90 μm

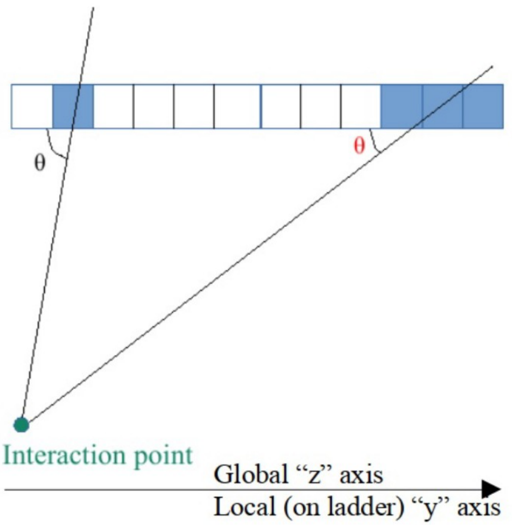
No difference between barrel and endcap.

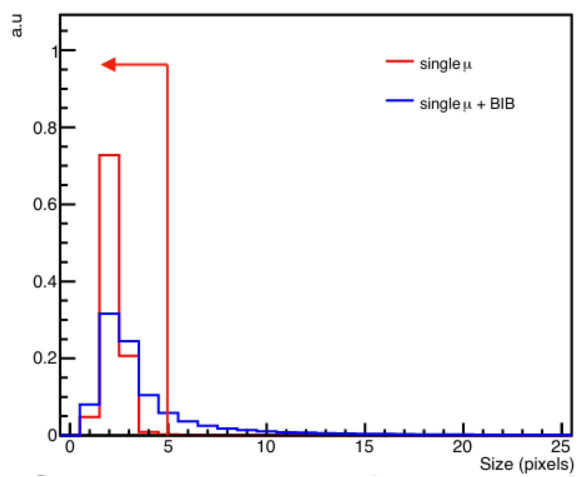
Advantages of Realistic Digitization

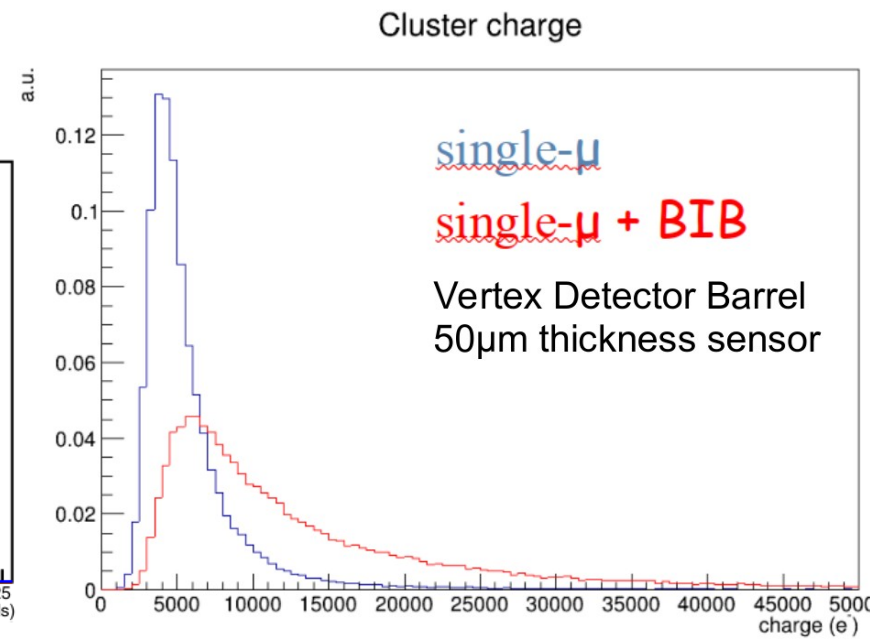
Details

Work In Progress: Currently not part of common workflow

- Provides a more accurate description of hit clusters
- Provides a handle on BIB rejection





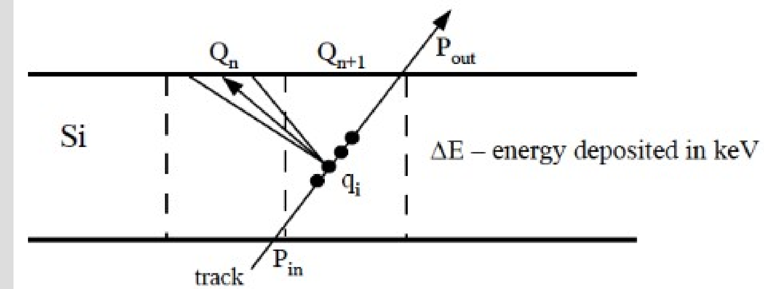


Requirement	Cut efficiency	Loose	Tight
Size-y cut vs. θ only	Single-μ	99.8 %	99.6 %
	Single-μ and BIB	55.2 %	43.7 %
Adding pixel size-x < 4	Single-μ	99.3 %	99.1 %
	Single-μ and BIB	37.4 %	30.7 %

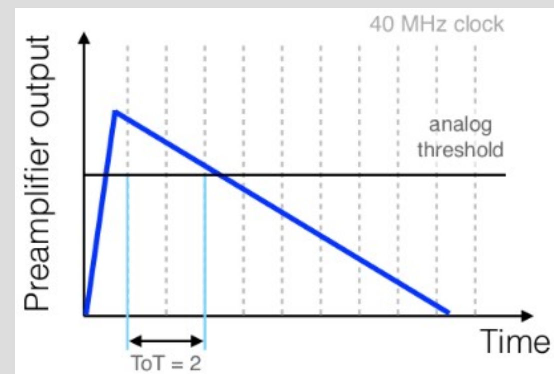
WIP Realistic Digitization

- **Two models for vertex modules**
  - Trivial (collect charge in pixel)
  - RD53A (complete simulation, ref)
- **Hoshen-Kopelman for clustering**
  - Eval alternatives as future development
- **Performance tested with full BIB**
  - Trivial: 100 s / evt
  - RD53A: 5000 s / evt

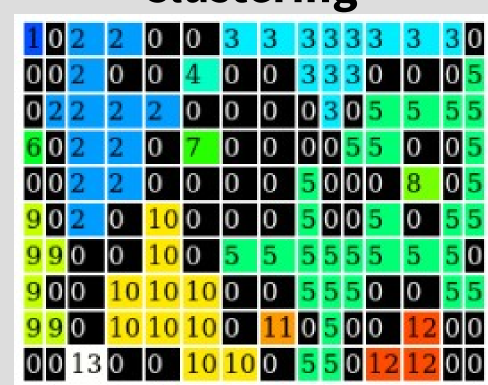
Charge Particle Deposits Details



Sensor Pixelization/Digitization



Clustering



Track Reconstruction Algo #1

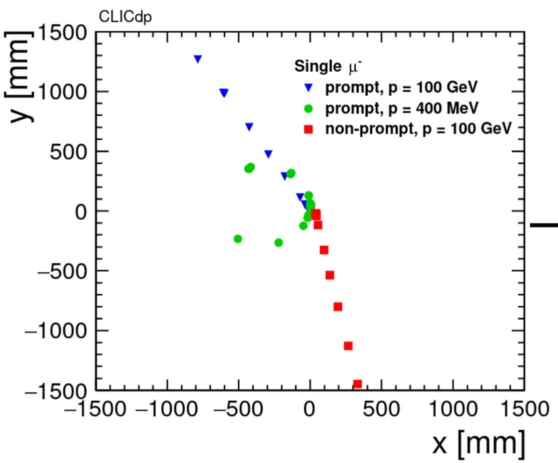
Details

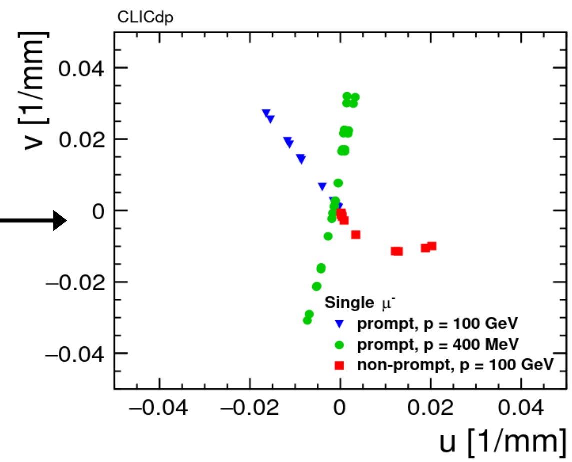
Global Hit Selection
ie: timing or double layers

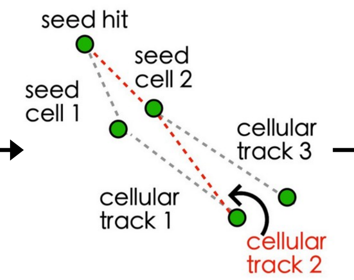
Conformal Transform
circular tracks → straight lines

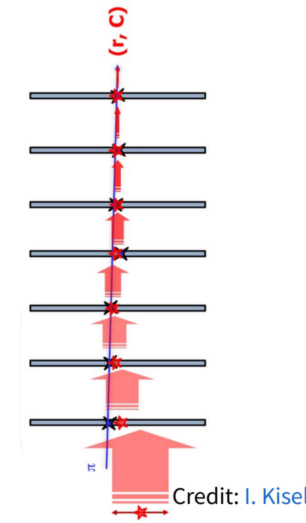
Cellular Automaton
straight "lines" → tracks

Kalman filter
Track fit









Remove BIB hits

Pattern Recognition

Track Fit

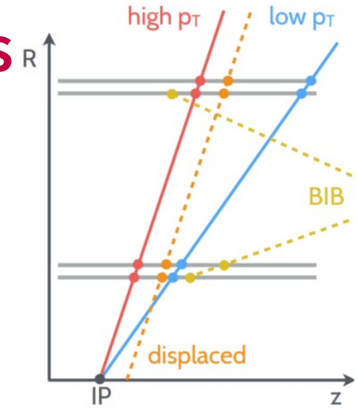
Algorithm + code inherited from CLIC software.

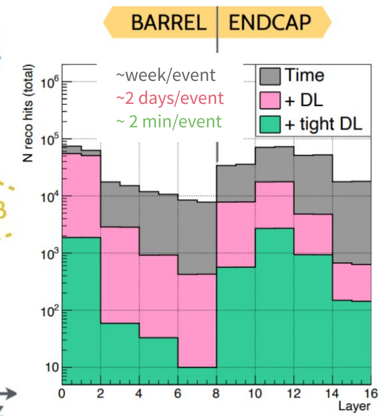
aka optimized for clean e^+e^- environment

Tracking Performance #1

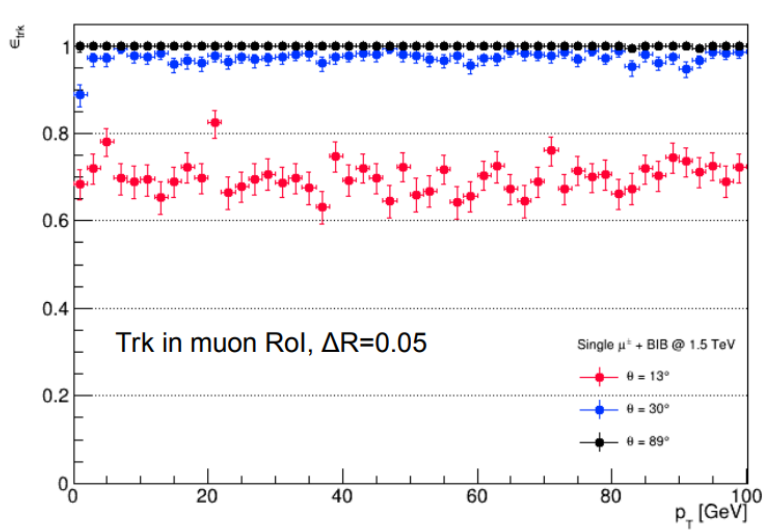
Details

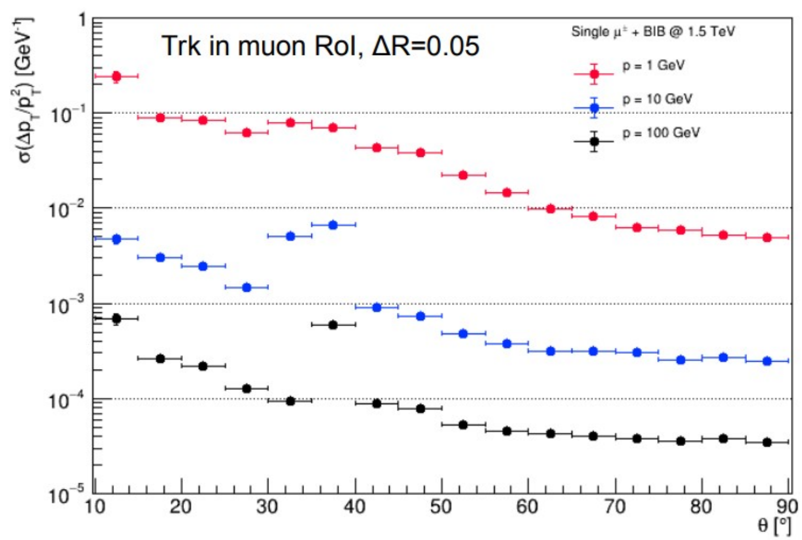
- **Employ hit multiplicity reduction strategies**
  - Region of Interest seeded tracking
  - Directional information from double layers
- **Require tight filtering for practical tracking**





- **Good track reconstruction once algorithm completes**

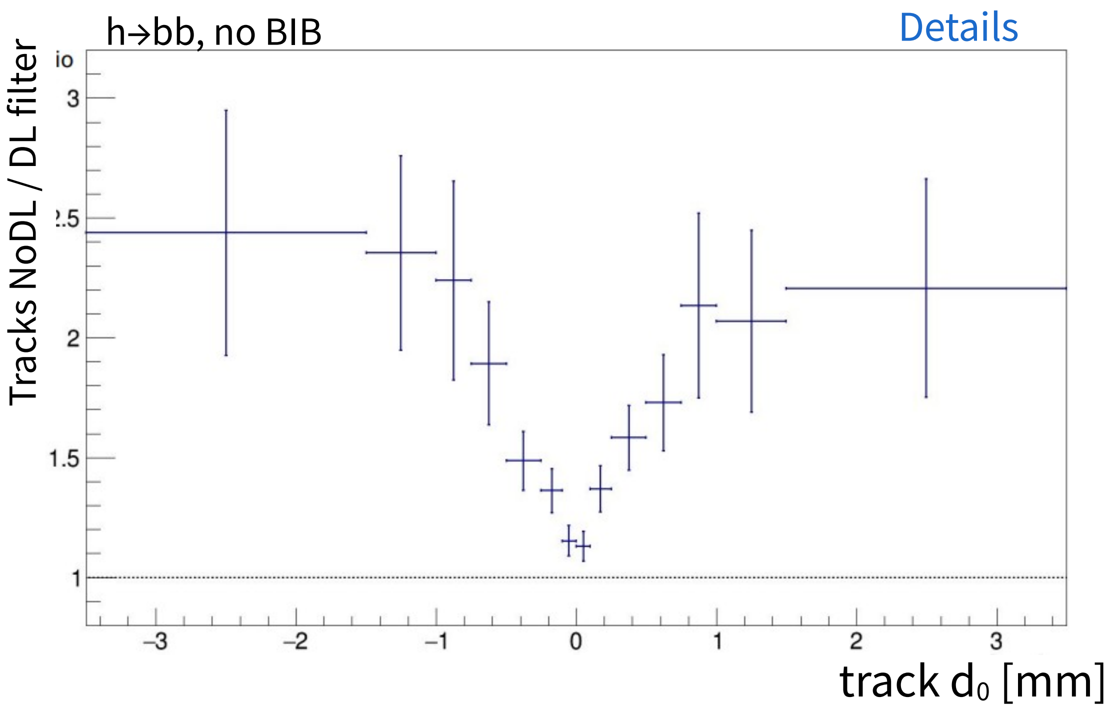


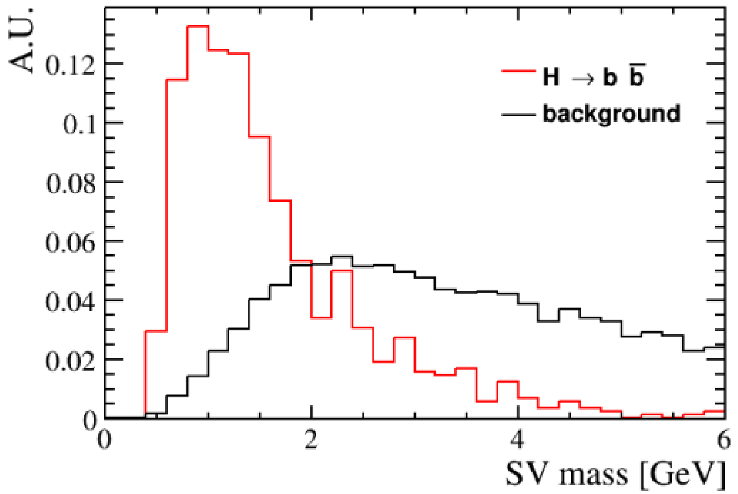


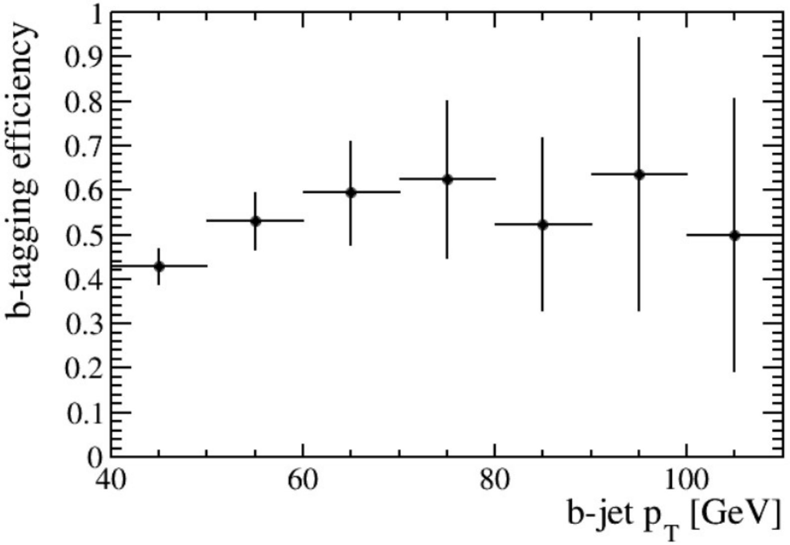
Flavour Tagging

Details

- **Secondary vertex reconstruction possible with BIB**
  - Caveat: using a very loose hit filter
- **Work ongoing on multivariate tagger**
- **Double layer filtering → possible bias**







A Common Tracking Software

- **ACTS is a standalone library for tracking algorithms**
- **Dedicated team working on advancing tracking algorithms**
  - Tracking is hard!
- **Allows us explore alternate algorithms**
  - Triplet-based seeding optimized for high multiplicity environments
  - Ongoing work to incorporate ML-based algorithms
- **Code optimization come for free**
  - Good software is even harder than tracking!
  - Also explores modern computing architectures (ie: GPU’s)

https://github.com/acts-project/acts

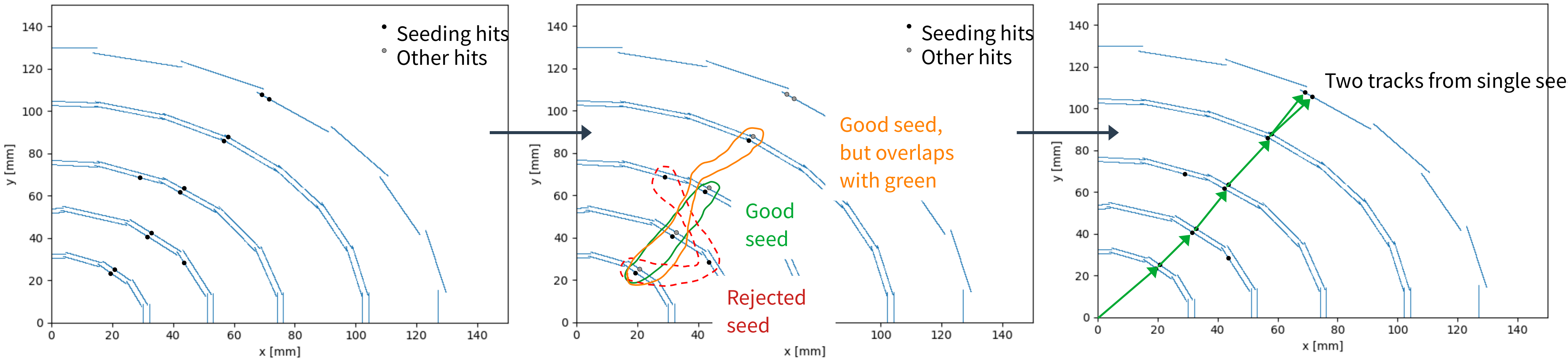
Fit Library	Kalman Filter Execution Time
ACTS	0.5 ms / track
iLCsoft	100 ms / track

Triplet Seeded CKF

Global Hit Selection
ie: timing, *

Seed Finding
Initial parameters for CKF

Combinatorial Kalman filter
Track fit



* Currently not leveraging double layers.

Remove BIB hits

Pattern Recognition

Track Fit

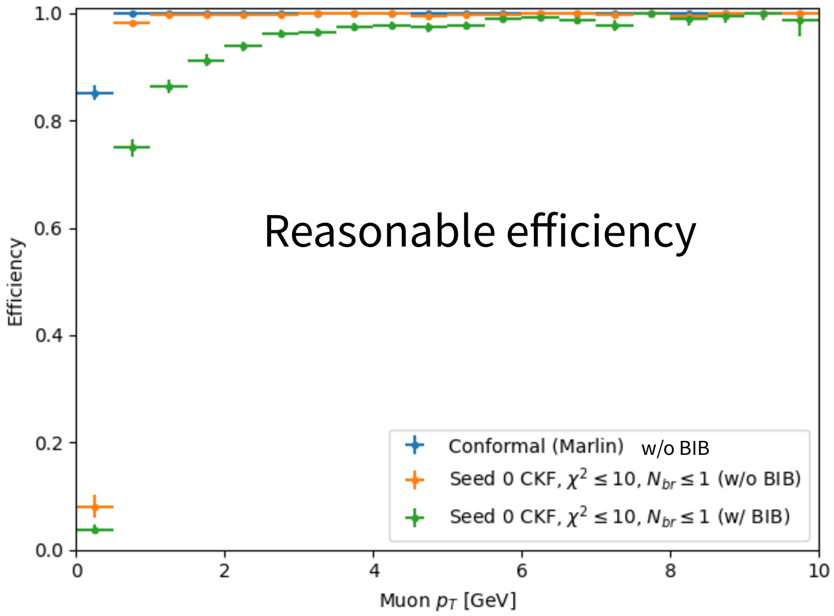
Similar algorithm used by ATLAS.

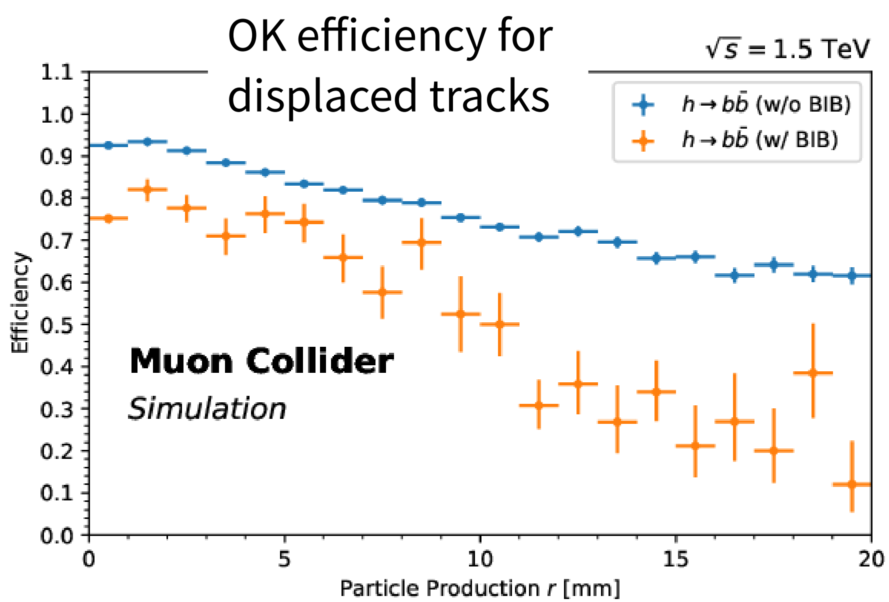
aka optimized for high hit multiplicity

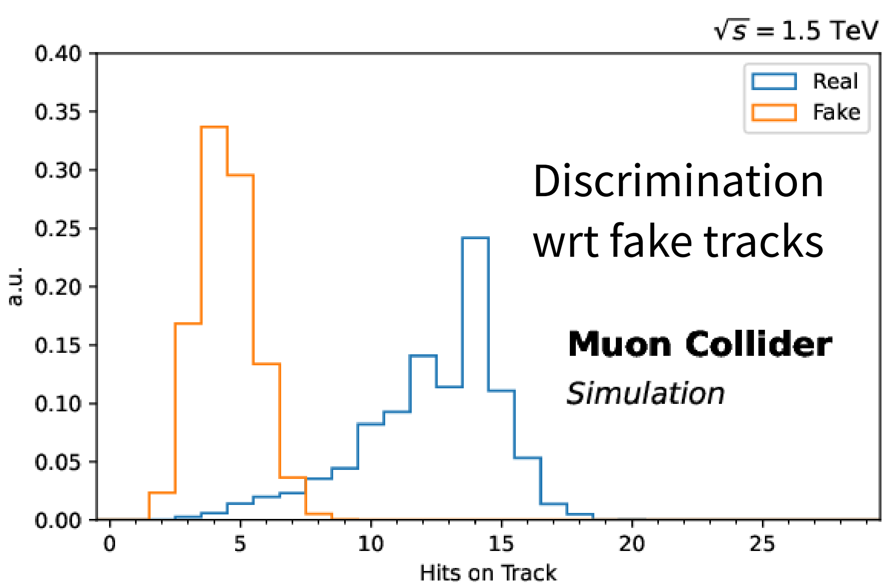
ACTS Track Finding

Details

- **Seeded CKF runs in ~4 min / event.**
- **Parameters need to be optimized.**
  - Seeding: *very narrow collision region*
  - CKF: No branching allowed





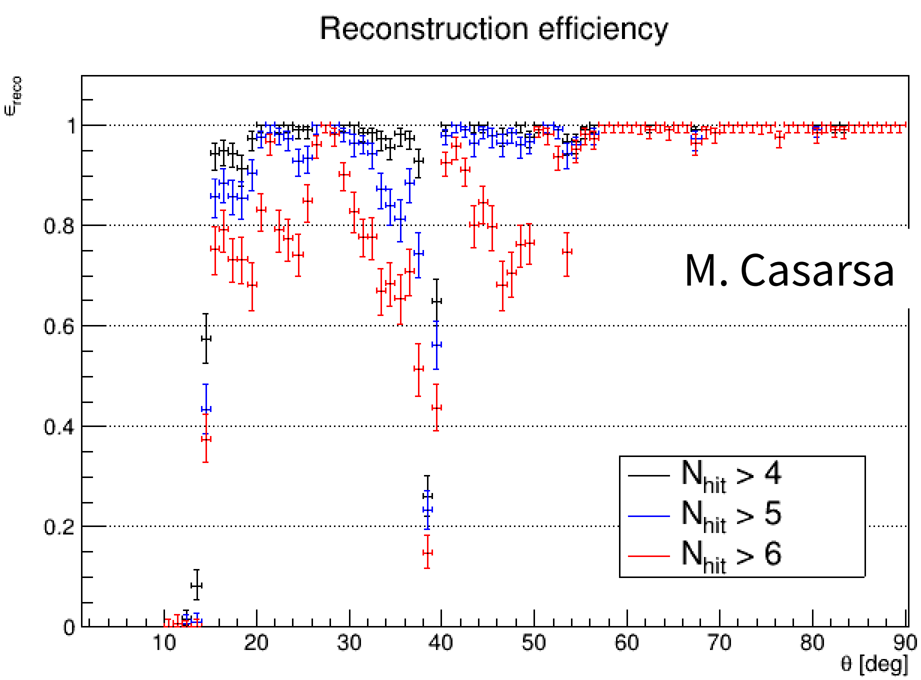


Electrons and Photons

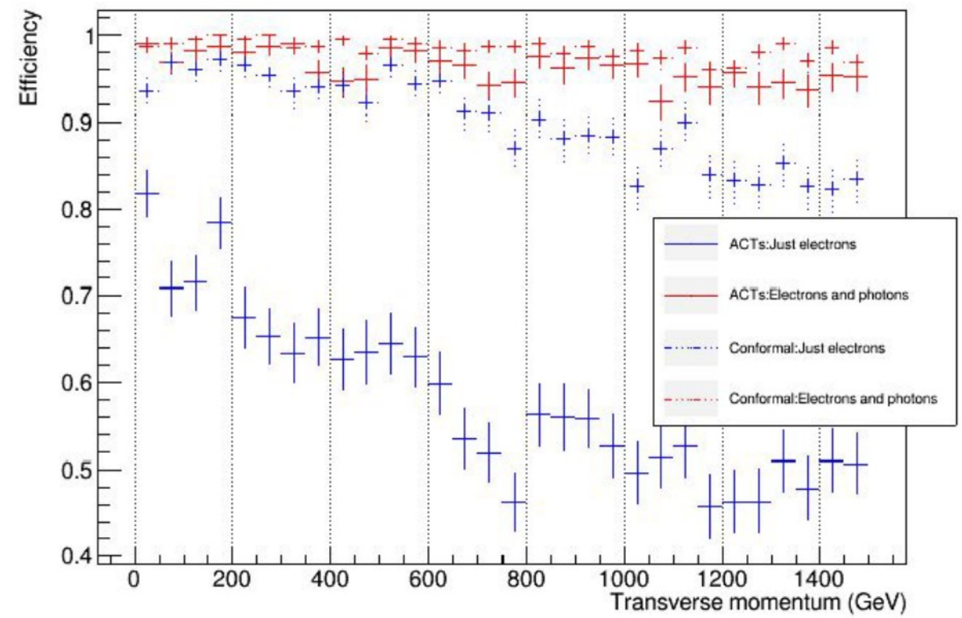
Details

Still need to study impact of ACTS tracking on object identification.

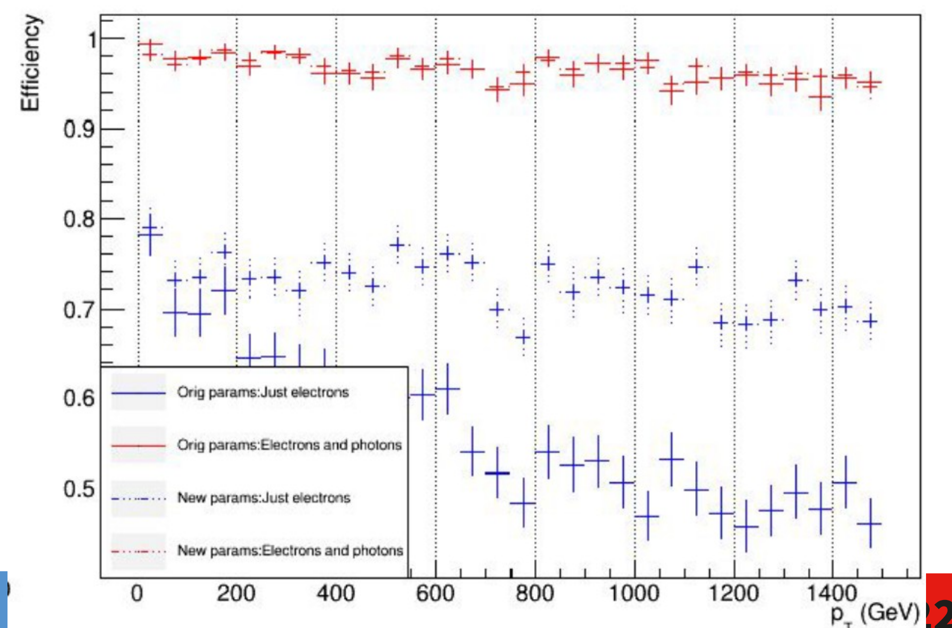
- Electrons reconstructed as photons.
- Sculpting from fake reduction cuts



Electron Reconstruction w/o BIB



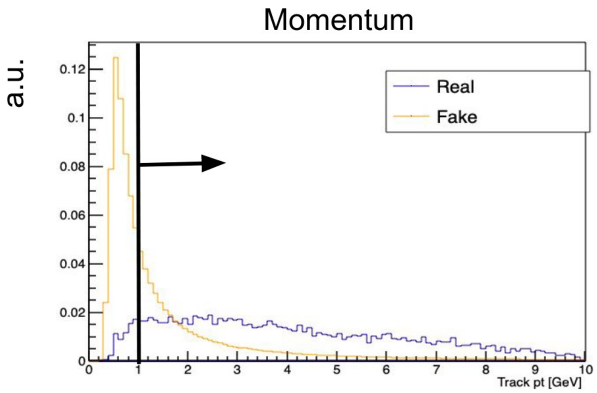
ACTS “Looser” Parameters

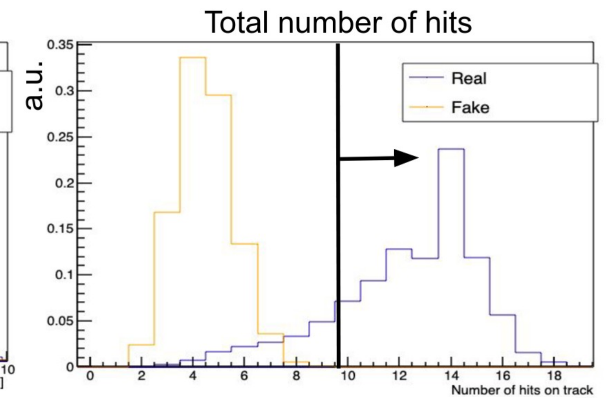


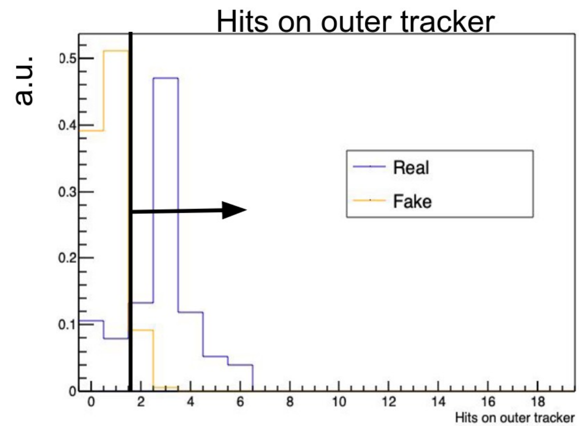
Rejecting Fakes

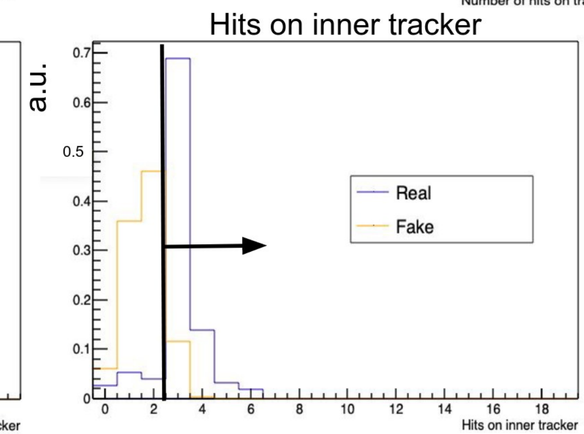
Details

- 100k fake tracks / event
- reduce to < 1 fake / event
- Still missing a few handles
  - χ^2, N_{holes}, timing
- Implemented as an (unreleased) processor

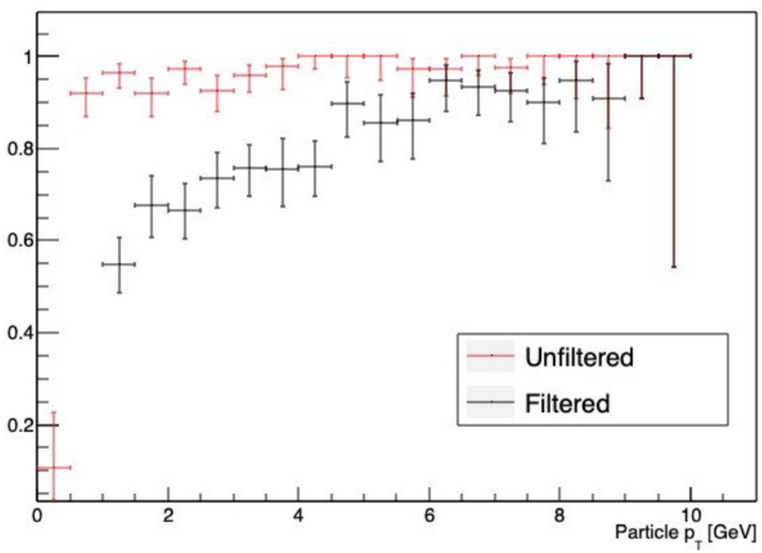


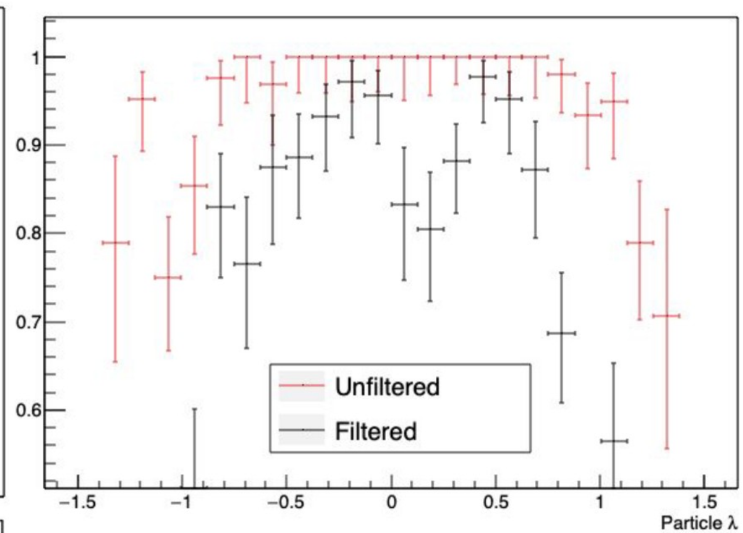


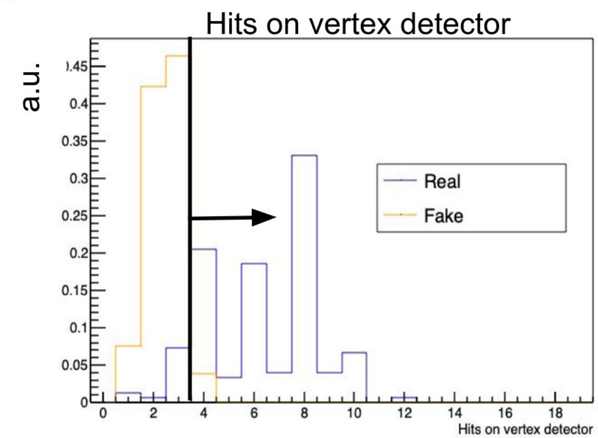




Efficiencies



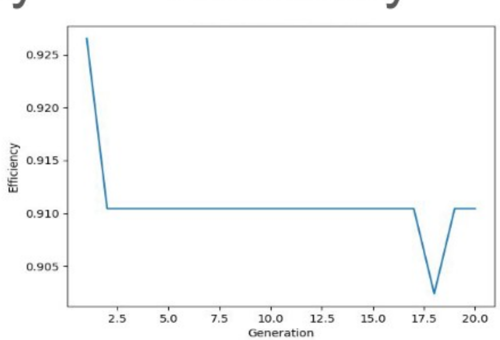
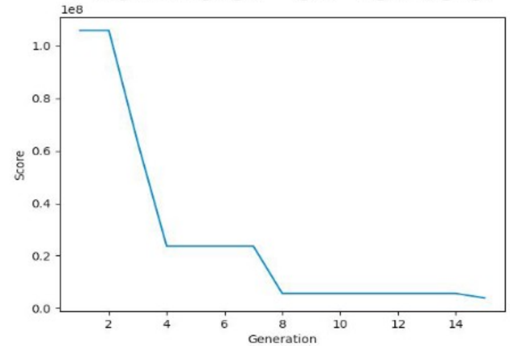
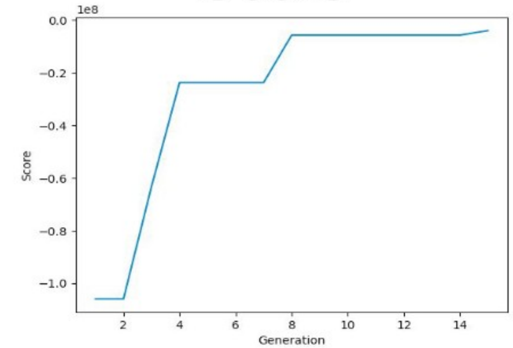




Rejecting Fakes: Optimization

Details

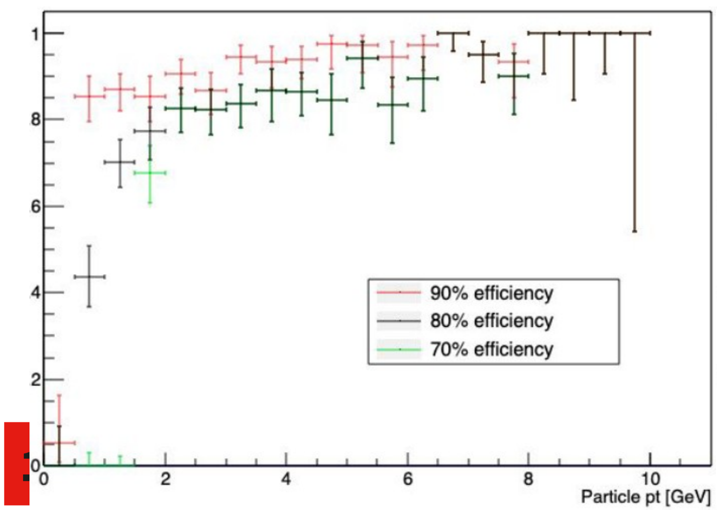
- TrackFilter optimized using evolutionary algorithms

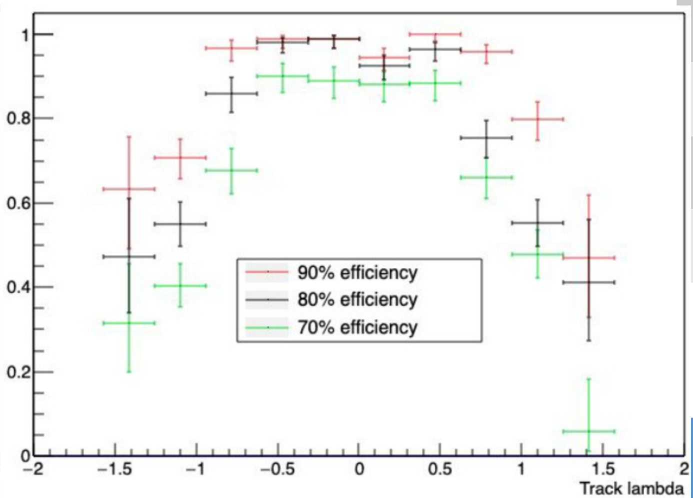
Target efficiency	Efficiency	Number of fakes	Score
90%			

- Studied a few fixed efficiency working points
- For <80% eff, start removing low p_T tracks

Eff WP	Fakes / event
90%	3900
80%	0.13
70%	0.06
64%*	0.08

* value by hand





Track Reconstruction Flow

MuonColliderSoft/ProductionConfig PR#9

DDPlanarDigiProcessor

- Hit filtering based on time

↓

ACTSSeededCKFTrackingProc

- Track finding and reconstruction.
- ~100k tracks in output

↓

TrackFilter

- Cuts on hits/holes/χ^2
- Independent of tracking processor
- *New! Not released yet!*

↓

RefitFinal

- Needed to get track state at calo

(TrackFilter ⇢) **Vertex Constrained Fit**

- Special collection for p_T>*X*
- TODO

TrackPerf: Package for Common Tracking Plots

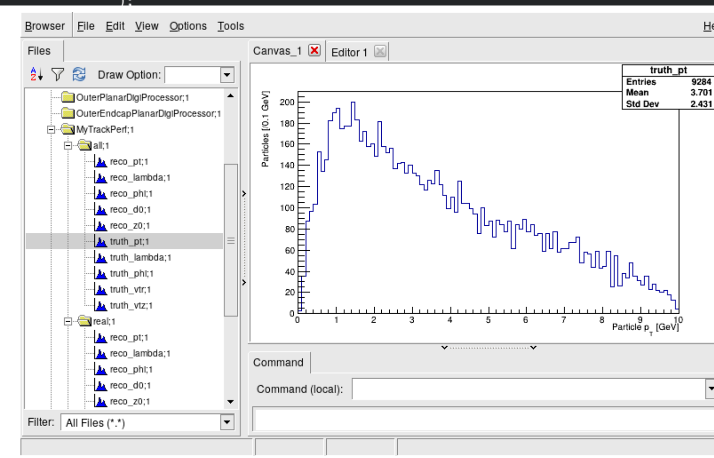
- **Common way to compare the different tracking approaches**
  - Started a new package TrackPerf (unreleased)
- **Functionality**
  - Input: EVENT::Track collection
  - Output: all the histogram you would want
    - Parameters of truth particles matched/not-matched/all
    - Parameters of tracks matched/not-matched/all
    - Resolution plots of all parameters
  - Configurable selection on truth particles
    - Default: charged, decay in tracker, left tracker
    - Option to filter for particles from b-meson decay (TODO)
  - ROOT Ttree for custom studies (TODO)

```
registerProcessorParameter("MatchProb",
                           "Minimum matching probabilty to be considered a good track-mc match.",
                           _matchProb,
                           _matchProb);

registerInputCollection( LCIO::MCPARTICLE,
                         "MCParticleCollection" ,
                         "Name of the MCParticle collection"  ,
                         _mcpColName,
                         _mcpColName
                         );

registerInputCollection( LCIO::TRACK,
                         "TrackCollection" ,
                         "Name of the Track collection" ,
                         _trkColName,
                         _trkColName
                         );

registerInputCollection( LCIO::LCRELATION,
                         "MCTrackRelationCollection" ,
                         "Name of LCRelation collection with track to MC matching",
                         _trkMatchColName,
                         _trkMatchColName
                         );
```



Missing plotting scripts!

Conclusions

- **Tracking is most advanced part of object reconstruction**
- **ACTS allows practical track reconstruction with full BIB in**
  - Demonstrates we can meet the challenge with *current computing resources*
- **Plenty of room for improvement (fake rejection, optimization)**
  - Sharing of common code, samples and reference results important!

Next big step:

- Reduction of BIB hits via realistic digitization
- Understand impact on object reconstruction and identification

BACKUP

Truth Tracking

Pattern Recognition

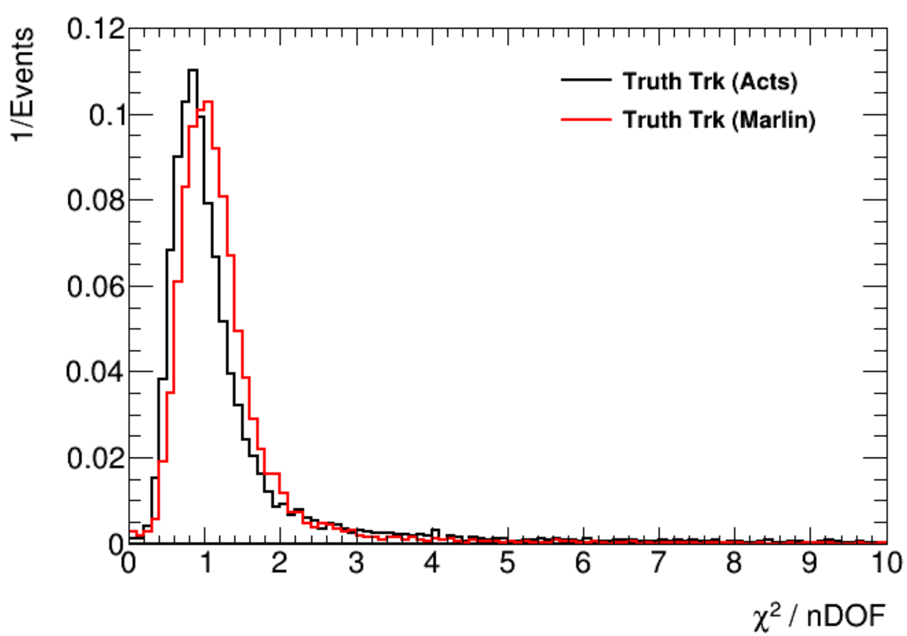
- Use hits associated to MC particle (100% efficiency)
- Same code for Marlin and ACTS

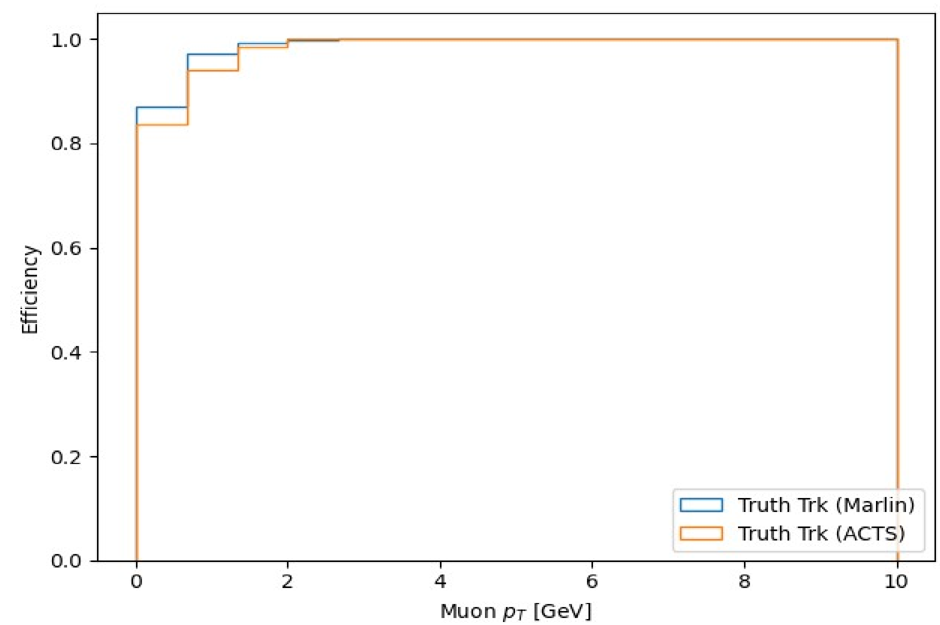
Track Fit

- Kalman Filter, but ACTS vs Marlin implementation

Same inputs, same algorithm, but different programmer.

Fit Library	Execution Time
ACTS	0.5 ms / evt
iLCsoft	100 ms / evt





Evolutionary Algorithm

N. Bruhwiler

Optimization with evolutionary algorithm

Chatain et al., *Evolutionary Algorithms for Tracking Algorithm Parameter Optimization* for EPJ Web of Conferences, 23 August 2021