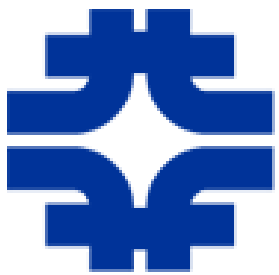
# Fast Pattern Recognition for CMS Track Finding

**Malina Kirn**

**12/14/2006**

**Progress Update, AMSC 663**

**Advisor: Dr. Nicholas Hadley**

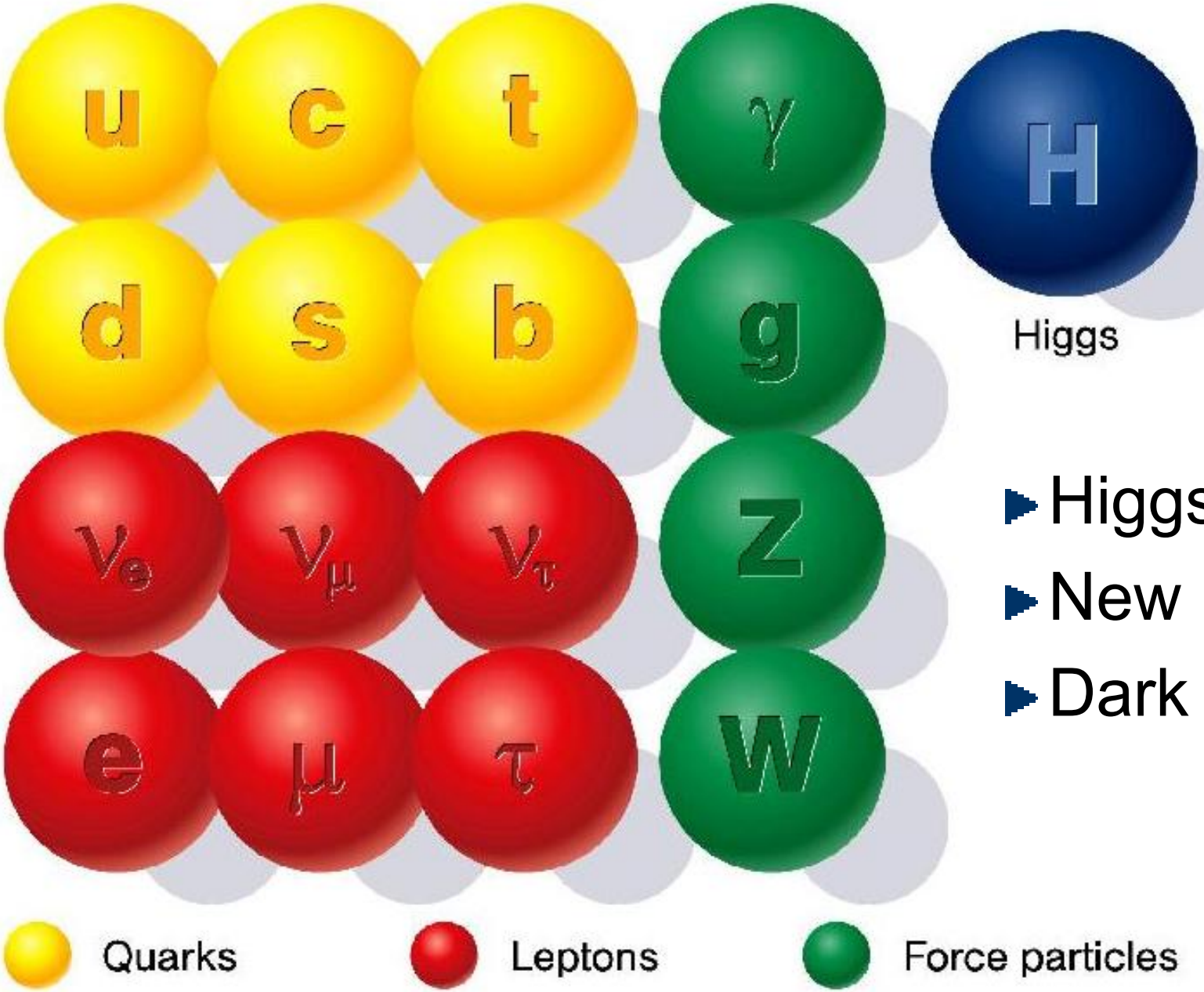




# Project Goal

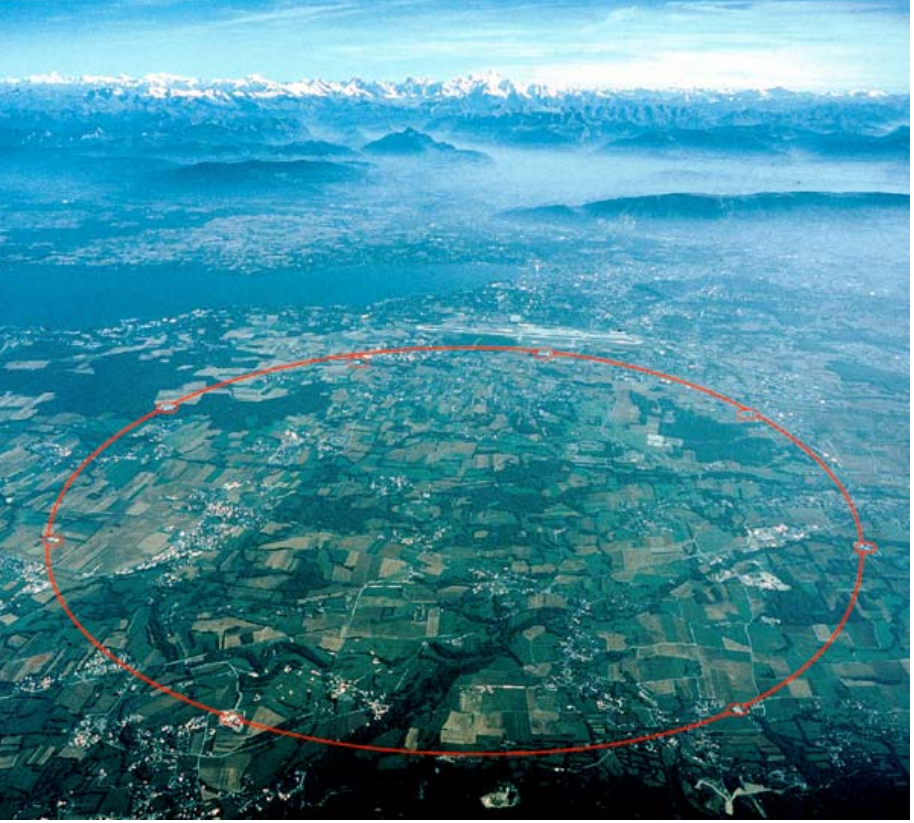
- Goal: Improve the running time of an existing track finding software package designed for the CMS experiment while preserving physics performance.
- Track finding algorithm:  Reconstruct helical tracks from 3D distribution of points detected by the tracker.
  - RoadSearch Track Finder
  - Combinatorial Track Finder (CTF or CKF)

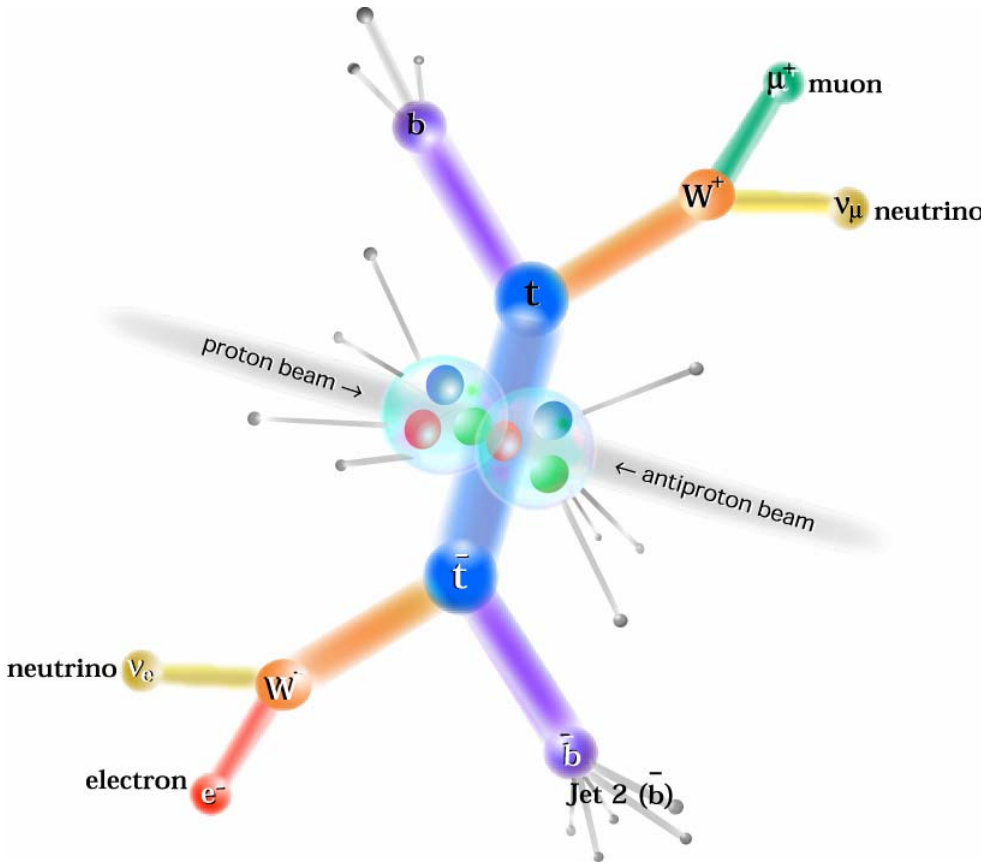
# Outline

- Review
  - Physics, Detector
- RoadSearch Algorithm
  - Review
  - Changes
- Milestones
  - Performance Report
  - RoadSearch Modification Proposal

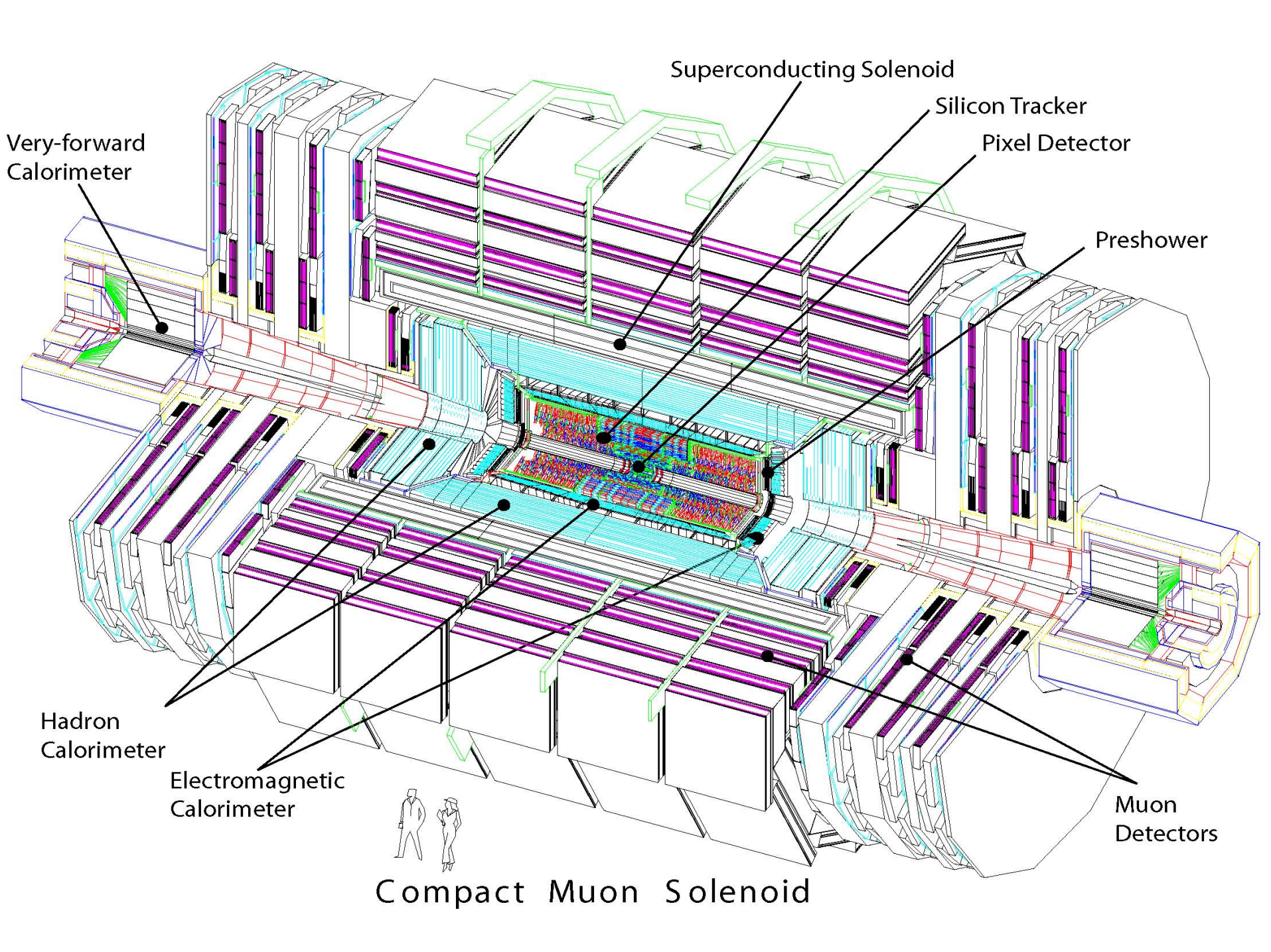


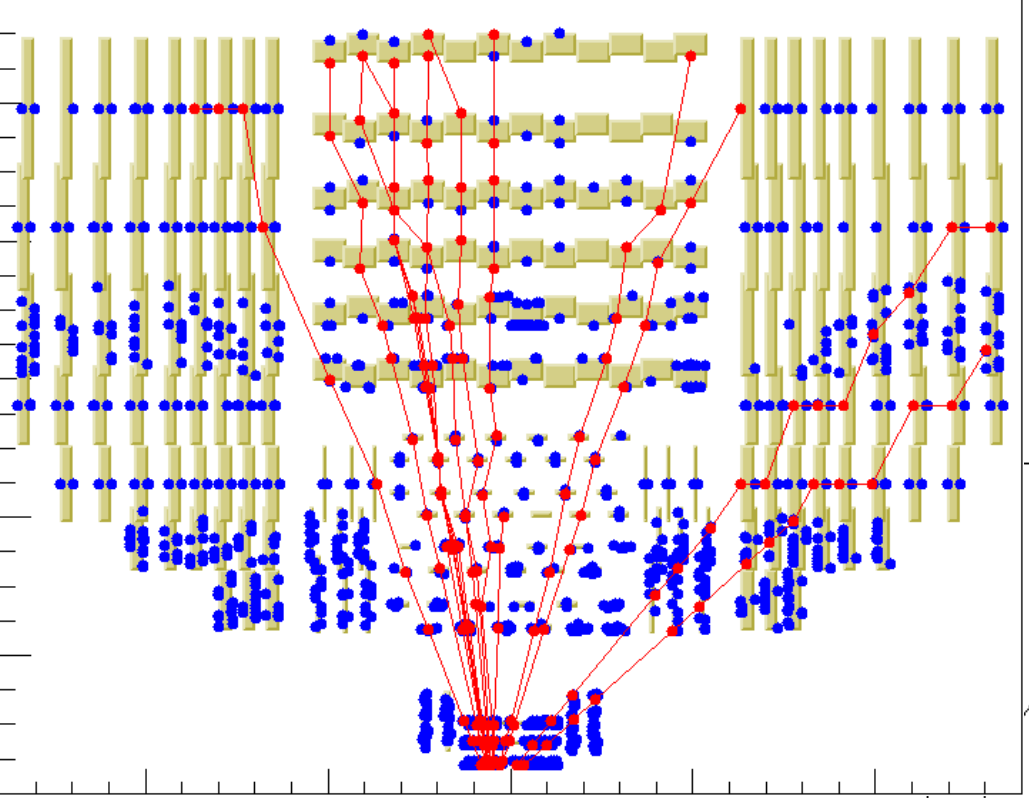

- Higgs boson
- New physics
- Dark matter



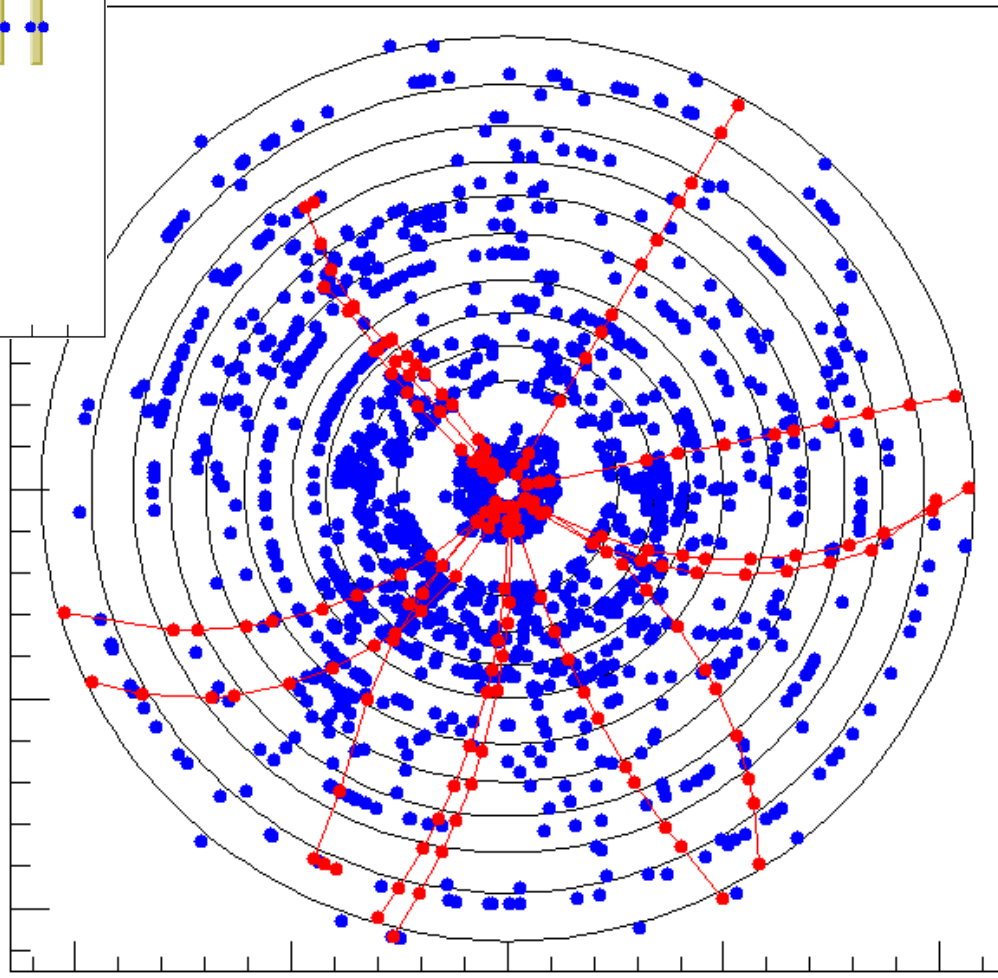



- Large Hadron Collider
  - Particle accelerator
  - p+p beams, 7 TeV
- Compact Muon Solenoid Detector


Superconducting Solenoid
Silicon Tracker
Pixel Detector
Very-forward
Calorimeter
Preshower
Hadron
Calorimeter
Electromagnetic
Calorimeter
Muon
Detectors
Compact Muon Solenoid



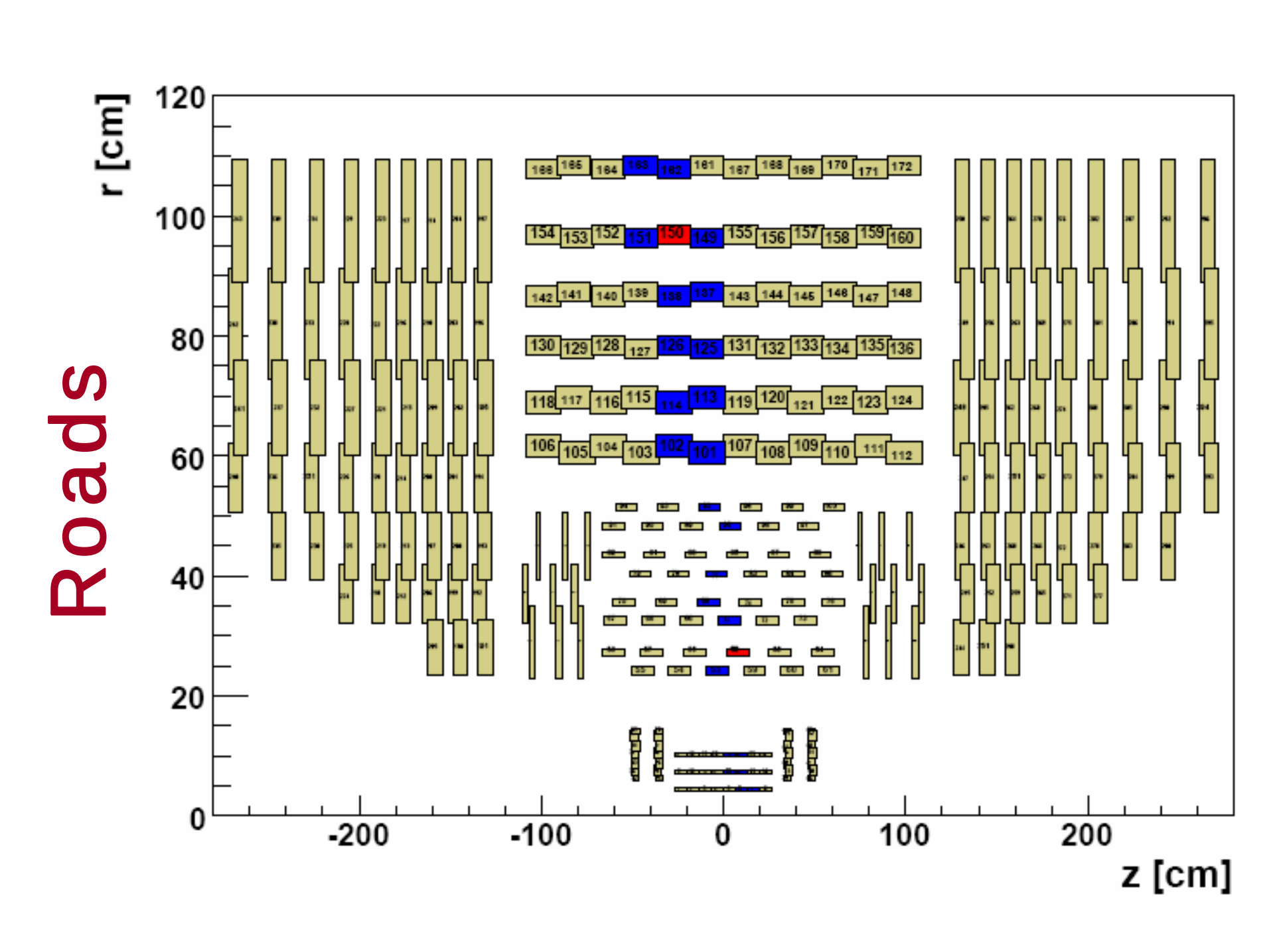
- Goal:
  Turn hits (dots) into tracks (helices)



- Final data rate of 150 Hz (events/sec)
- 1,000 – 10,000 hits/events

# RoadSearch Algorithm: Review

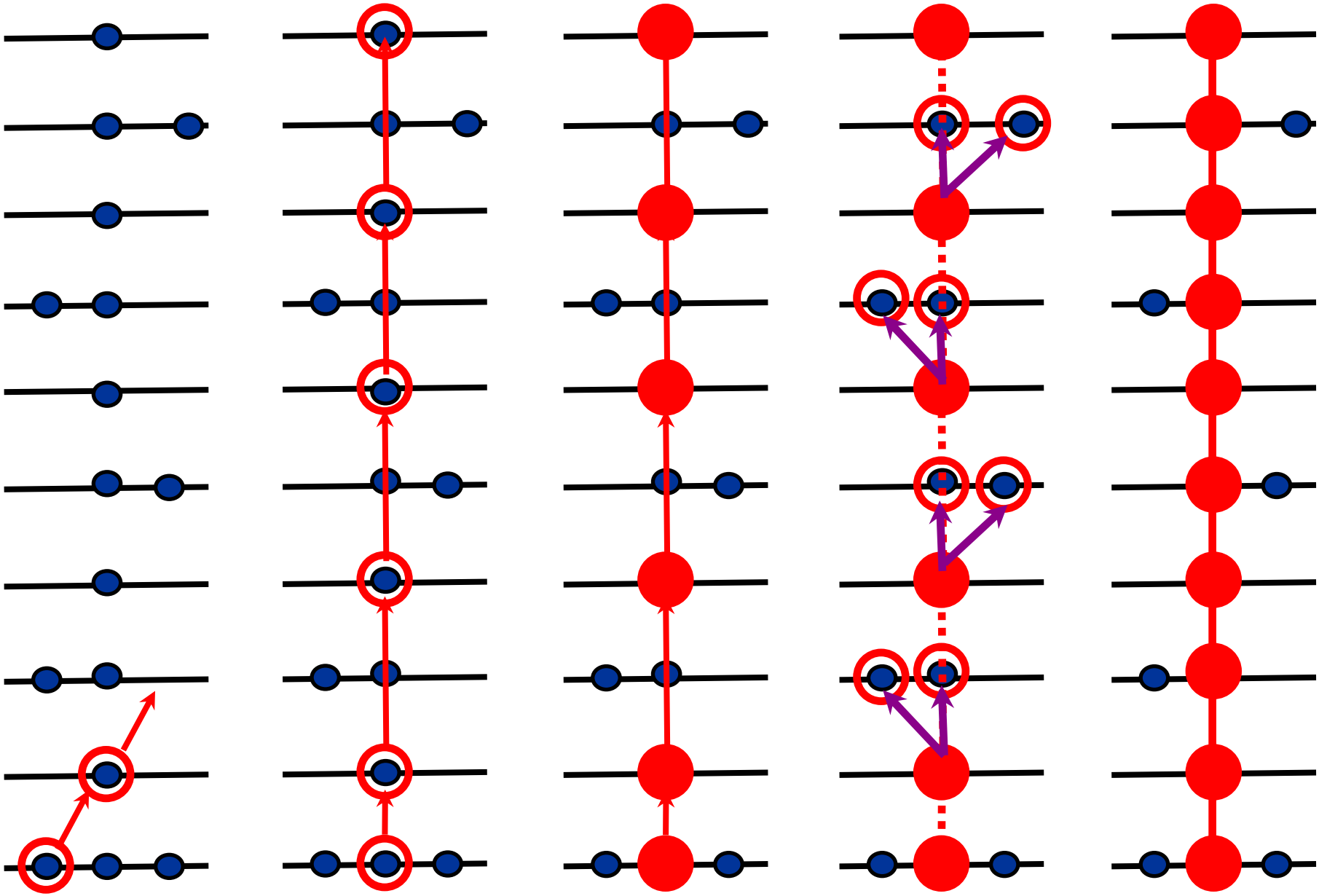
- Create seeds
  - Seed = endpoints of a track
- Create clouds
  - Clustering of hits around trajectory
  - Cloud cleaning now moved into cloud creation
- Create track candidates
- Fit final tracks


Roads
r [cm]
z [cm]
120
100
80
60
40
20
0
-200
-100
0
100
200

# RoadSearch Algorithm: Changes

- Cloud Cleaning
  - Process of merging similar clouds together
  - Moved into cloud creation step (loop over clouds only once)
  - Only consider neighboring clouds
- Technical Improvements
  - Parameters & data read from files once instead of multiple times
- Track Candidate Maker

# Track Candidate Maker



# Milestones

- Performance Report
  - Timing
  - Validation
- RoadSearch Modification Proposal
  - Hough transform
  - Coding tricks
  - Experimental determination of optimal location of cleaning(s)
  - Banana shaped clouds
  - Ordering of access to hit collection

# Performance Report: Timing

| CMSSW_1_2_0_pre5 | $\mu^-$, $p_T$=100 GeV | | $\mu^-$, $p_T$=10 GeV | | $\mu^-$, $p_T$=1 GeV | | $H^0 \to Z^0 Z^0 \to \mu^+\mu^-\mu^+\mu^-$ | | Minimum Bias | |
|---|---|---|---|---|---|---|---|---|---|---|
| events | 1000 | | 1000 | | 1000 | | 1000 | | 1000 | |
| | sec/event | % | sec/event | % | sec/event | % | sec/event | % | sec/event | % |
| roadSearchSeeds | 0.0504 | 14.0% | 0.0479 | 13.7% | 0.0541 | 14.5% | 0.0515 | 5.8% | 2.3334 | 13.8% |
| rawRoadSearchClouds | 0.0026 | 0.7% | 0.0026 | 0.7% | 0.0026 | 0.7% | 0.0085 | 1.0% | 5.5362 | 32.8% |
| rsTrackCandidates | 0.0672 | 18.7% | 0.0684 | 19.6% | 0.0620 | 16.7% | 0.2339 | 26.3% | 2.2837 | 13.5% |
| rsWithMaterialTracks | 0.0072 | 2.0% | 0.0071 | 2.0% | 0.0063 | 1.7% | 0.0231 | 2.6% | 0.0399 | 0.2% |
| **Net RoadSearch** | **0.1274** | **35.4%** | **0.1261** | **36.1%** | **0.1250** | **33.6%** | **0.3170** | **35.7%** | **10.1932** | **60.4%** |
| globalMixedSeeds | 0.0044 | 1.2% | 0.0039 | 1.1% | 0.0043 | 1.2% | 0.0101 | 1.1% | 0.1009 | 0.6% |
| ckfTrackCandidates | 0.1327 | 36.8% | 0.1405 | 40.2% | 0.1397 | 37.5% | 0.3858 | 43.4% | 1.4499 | 8.6% |
| ctfWithMaterialTracks | 0.0080 | 2.2% | 0.0079 | 2.2% | 0.0085 | 2.3% | 0.0253 | 2.8% | 0.0489 | 0.3% |
| **Net CTF** | **0.1451** | **40.3%** | **0.1523** | **43.6%** | **0.1526** | **41.0%** | **0.4212** | **47.4%** | **1.5997** | **9.5%** |

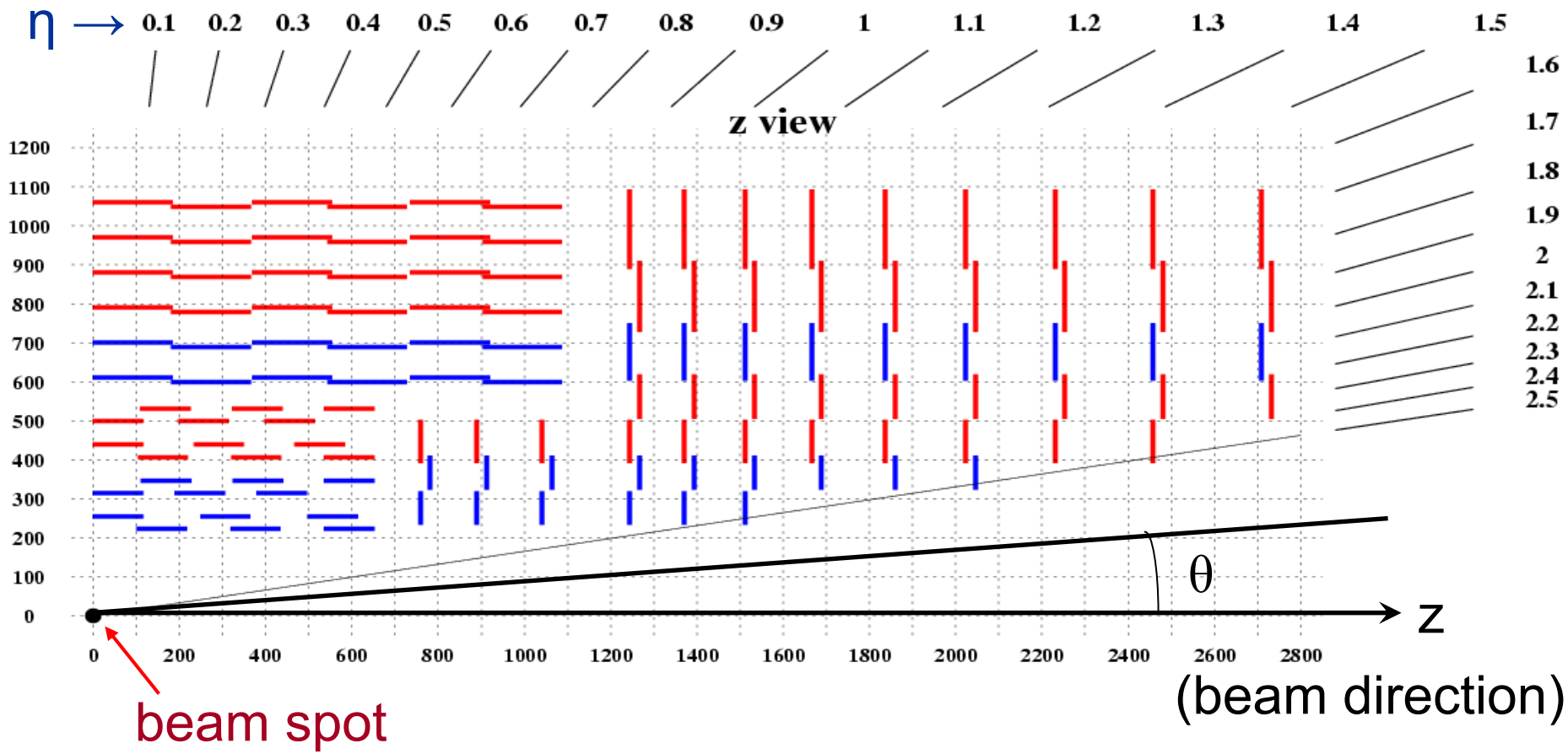
# Performance Report: Validation

- We use a “Track Associator”

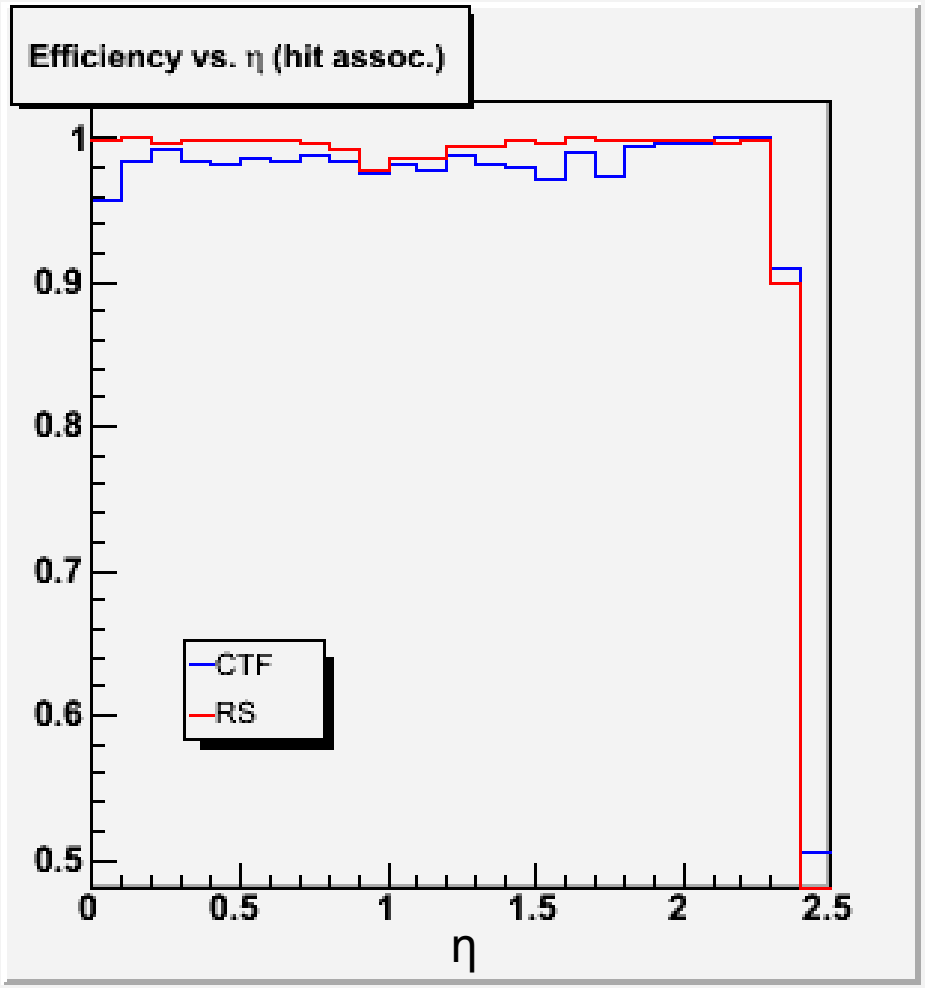

  - Goal is to compare reconstructed tracks to the simulated tracks.
  - Associate via hits or $\chi^2$
  - Efficiency = (# reco associated w/sim)/(# sim tracks)
  - Purity = (# sim associated w/reco)/(# reco tracks)
  - Plot vs. $\eta = -\ln[\tan(\theta/2)]$ and $p_T$
- 10,000 $H^0 \rightarrow Z^0\ Z^0 \rightarrow \mu^+\ \mu^-\ \mu^+\ \mu^-$ , Single $\mu$, varying $p_T$

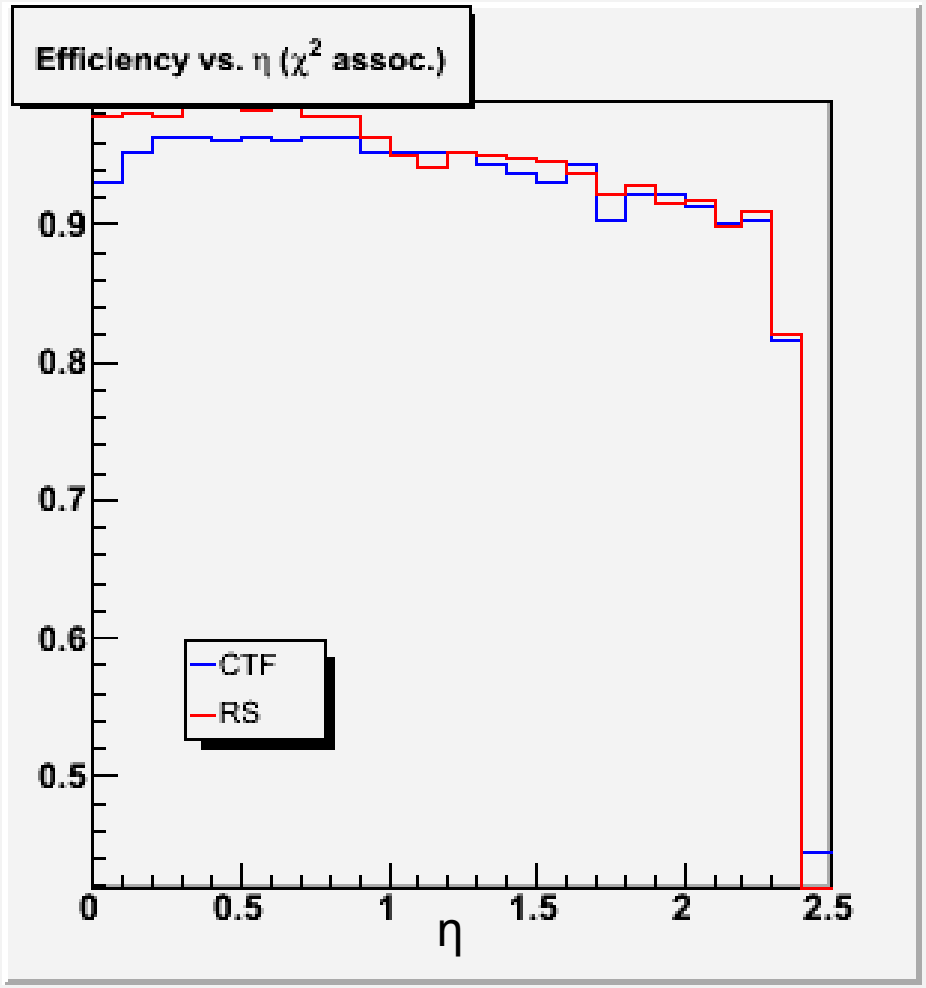
# Validation: Tracker Geometry

$\eta = -\ln[\tan(\theta/2)]$

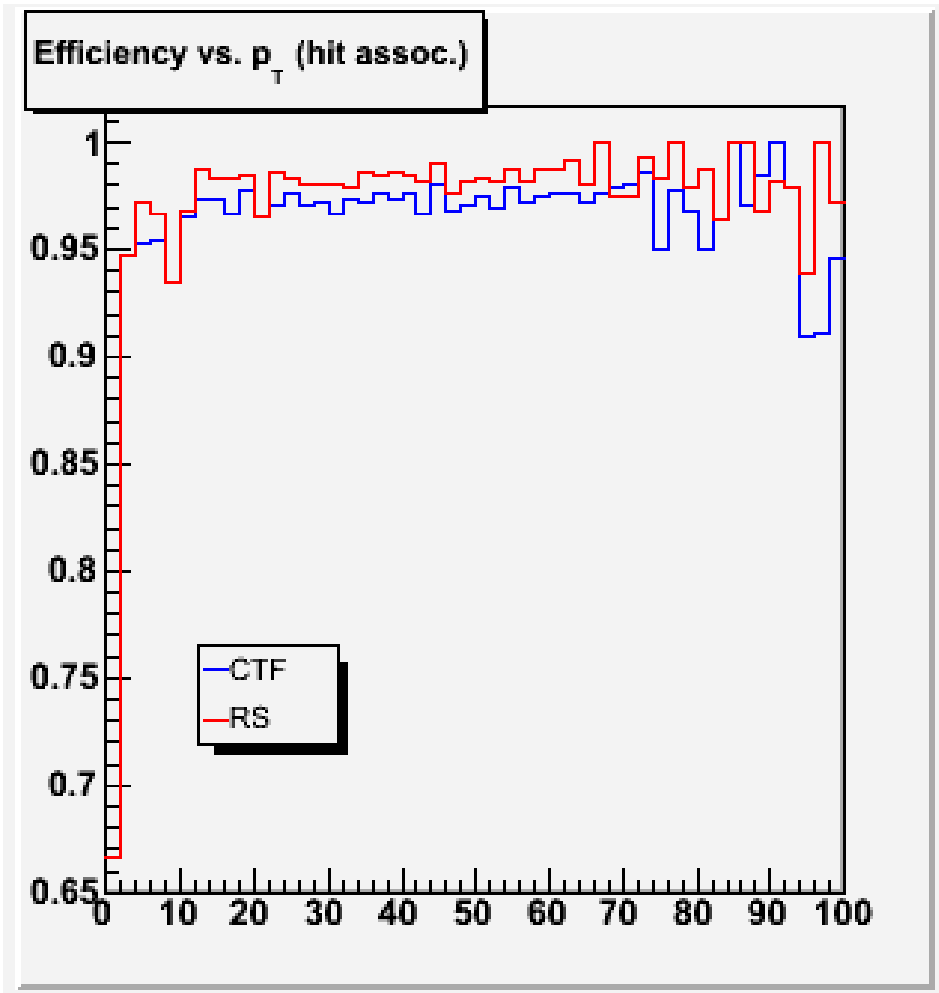


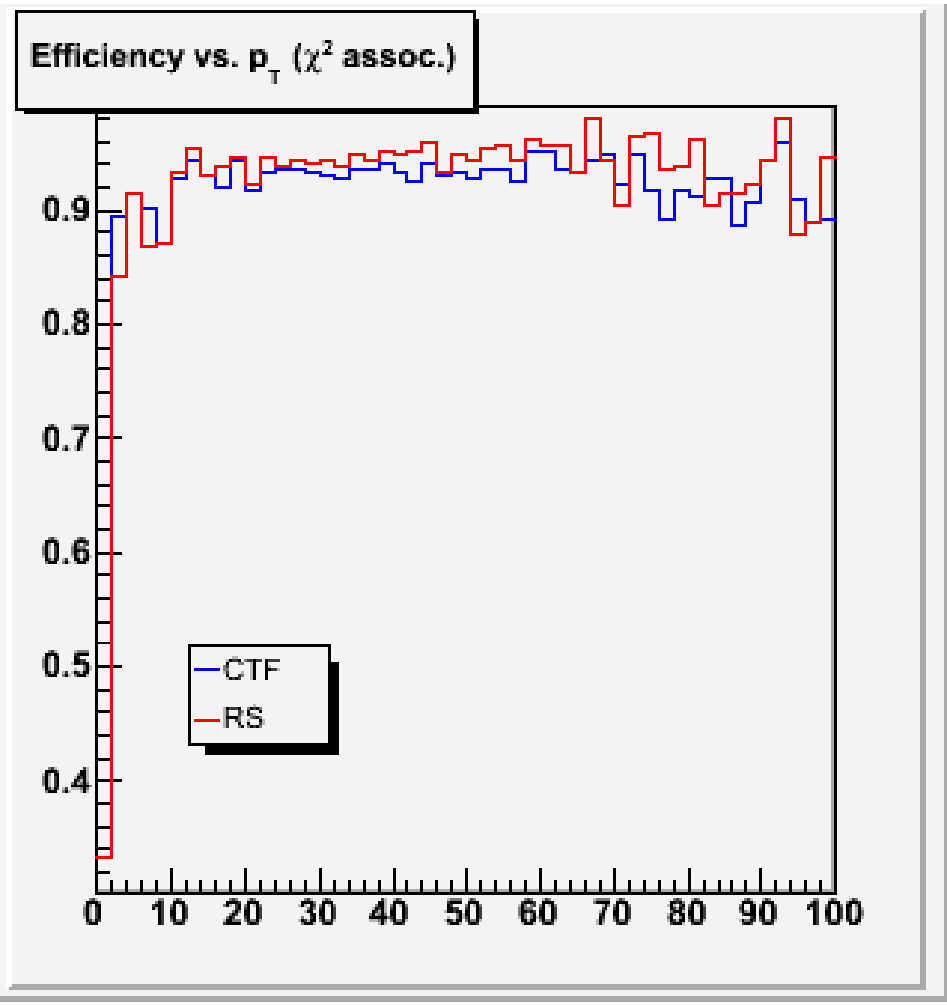
# Validation: $H^0 \rightarrow Z^0\ Z^0 \rightarrow \mu^+\ \mu^-\ \mu^+\ \mu^-$



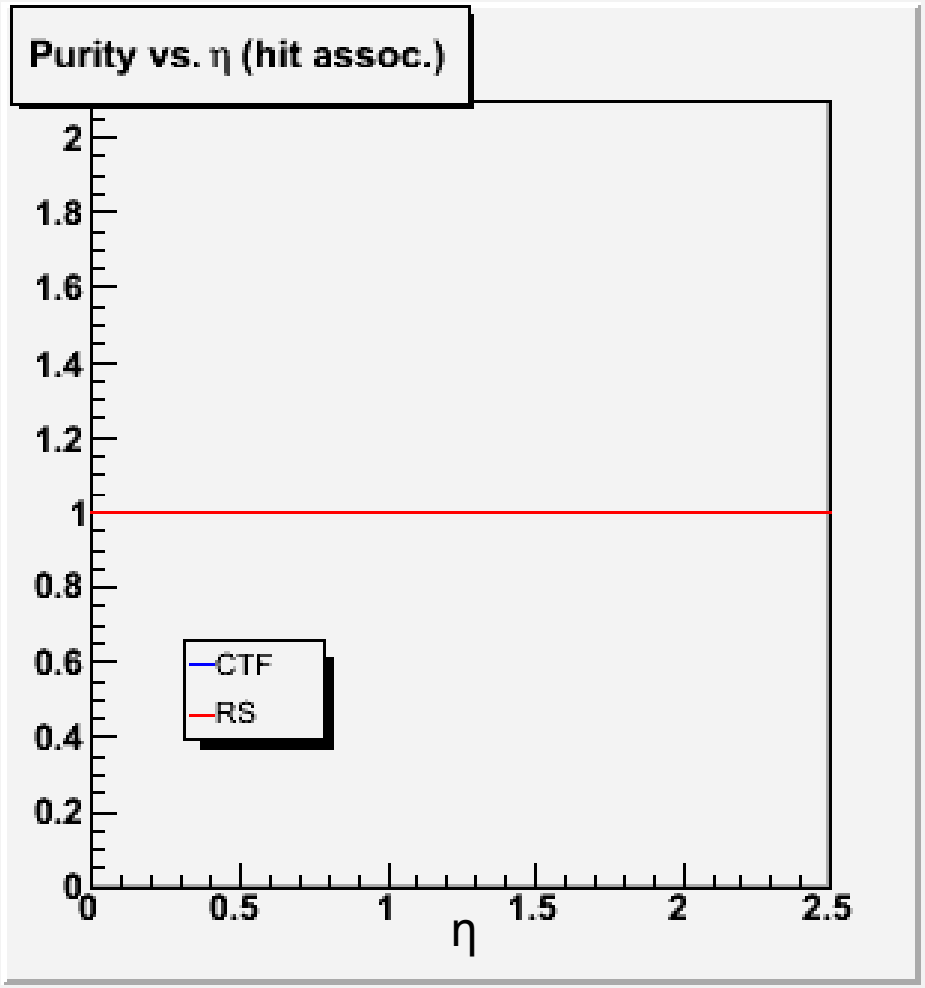



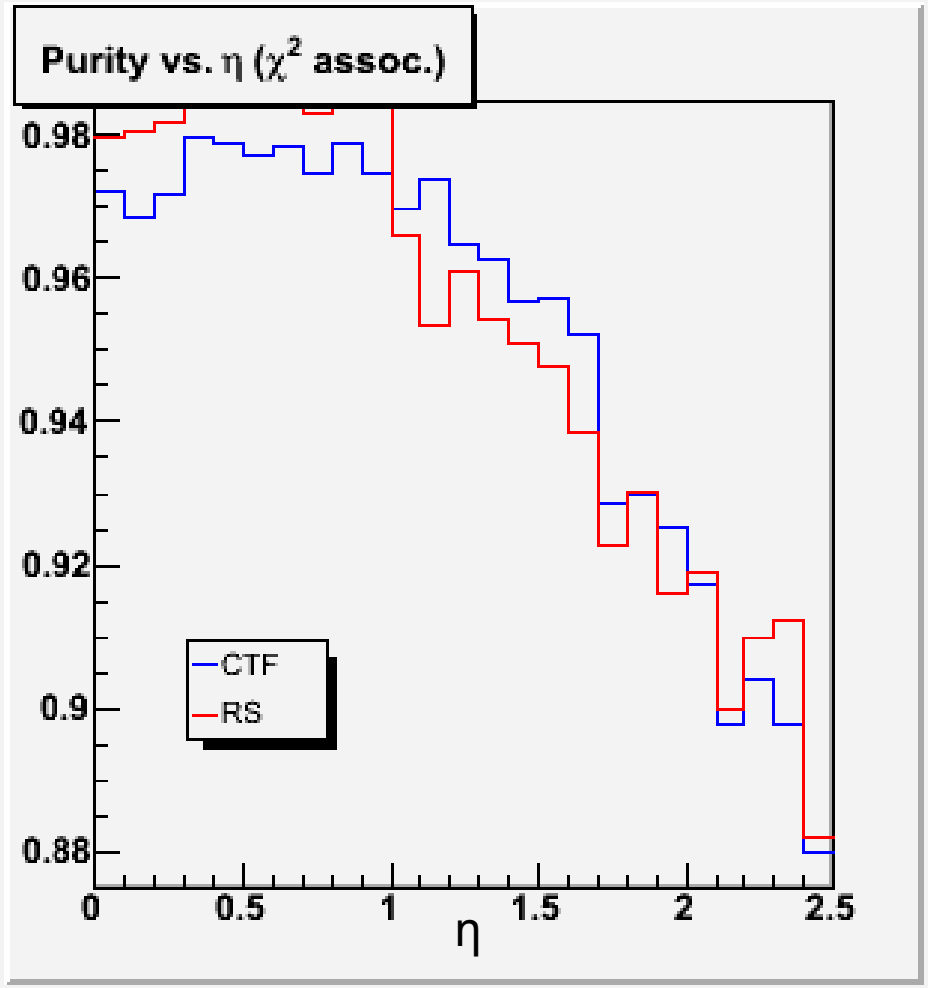
# Validation: $H^0 \rightarrow Z^0\ Z^0 \rightarrow \mu^+\ \mu^-\ \mu^+\ \mu^-$



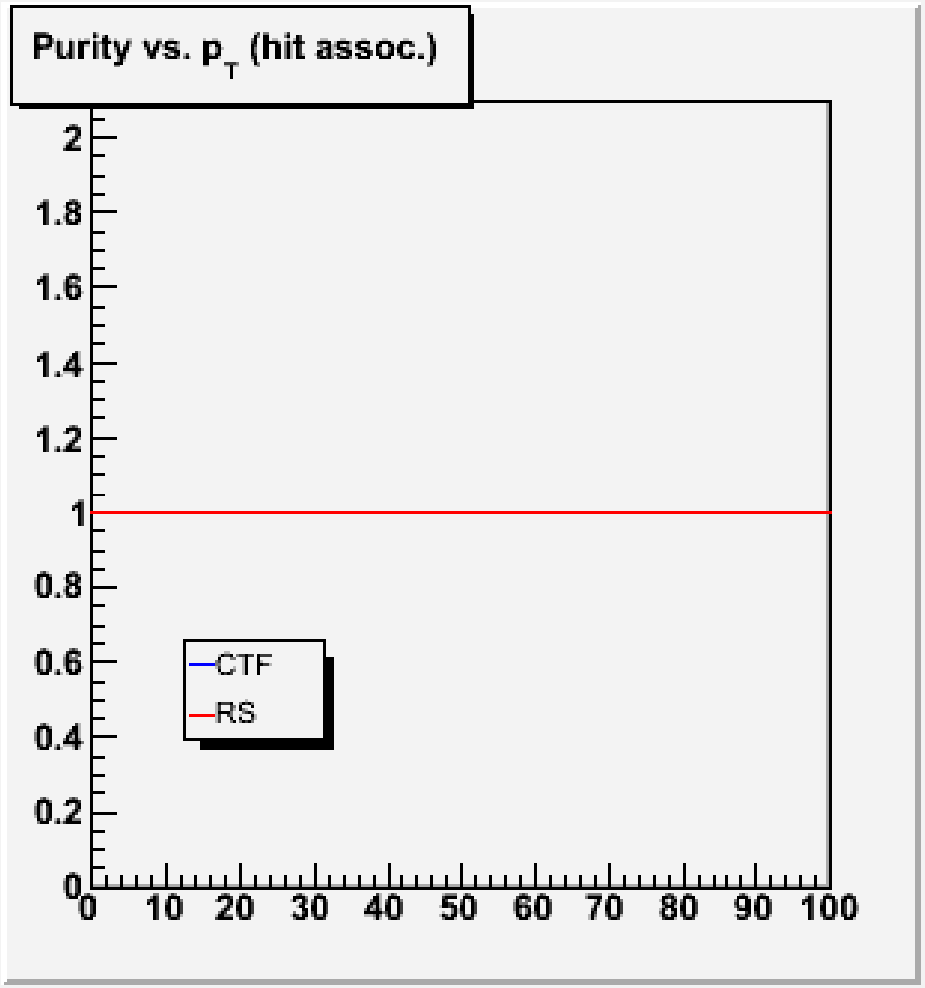



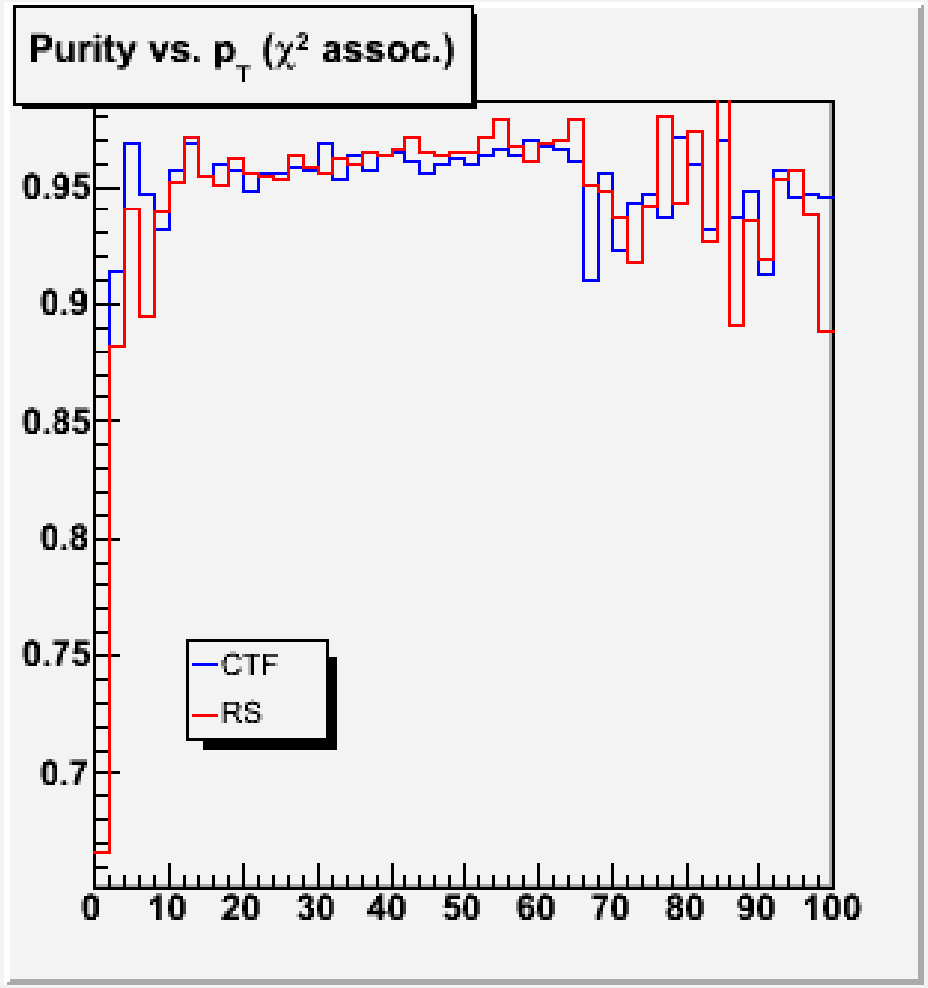
# Validation: $H^0 \rightarrow Z^0$ $Z^0 \rightarrow \mu^+ \mu^- \mu^+ \mu^-$



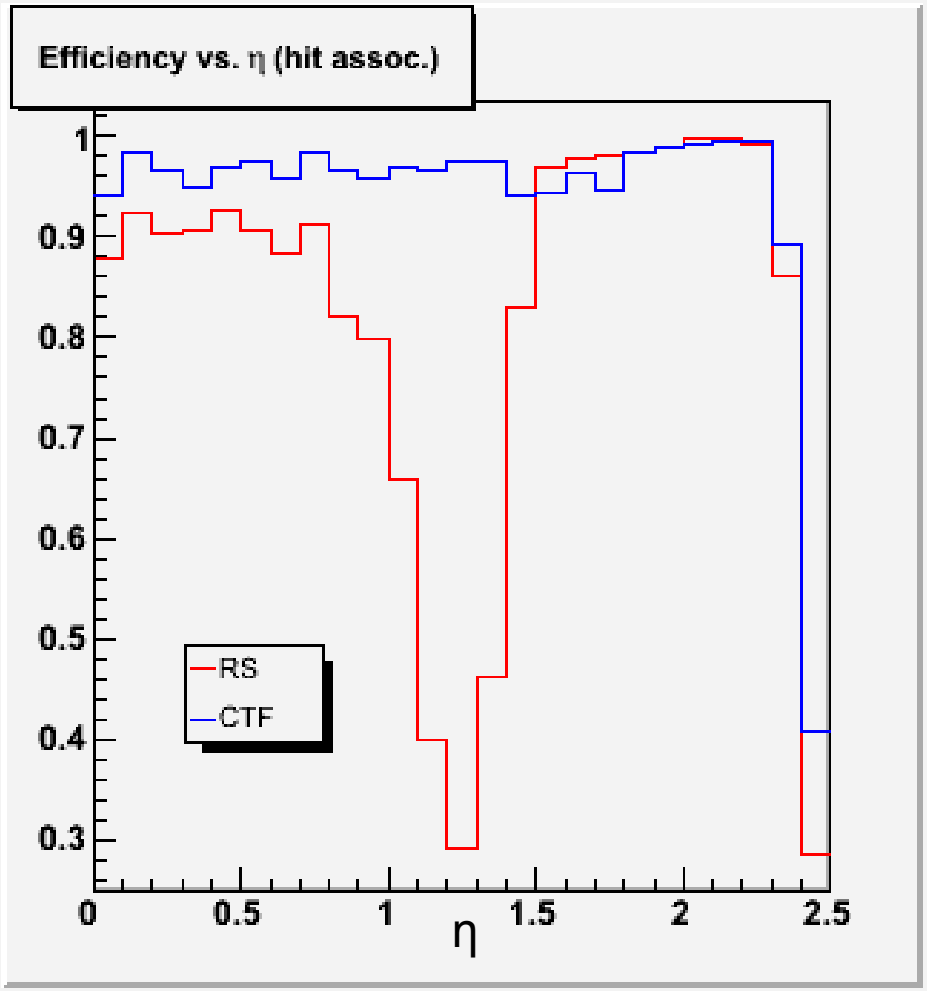



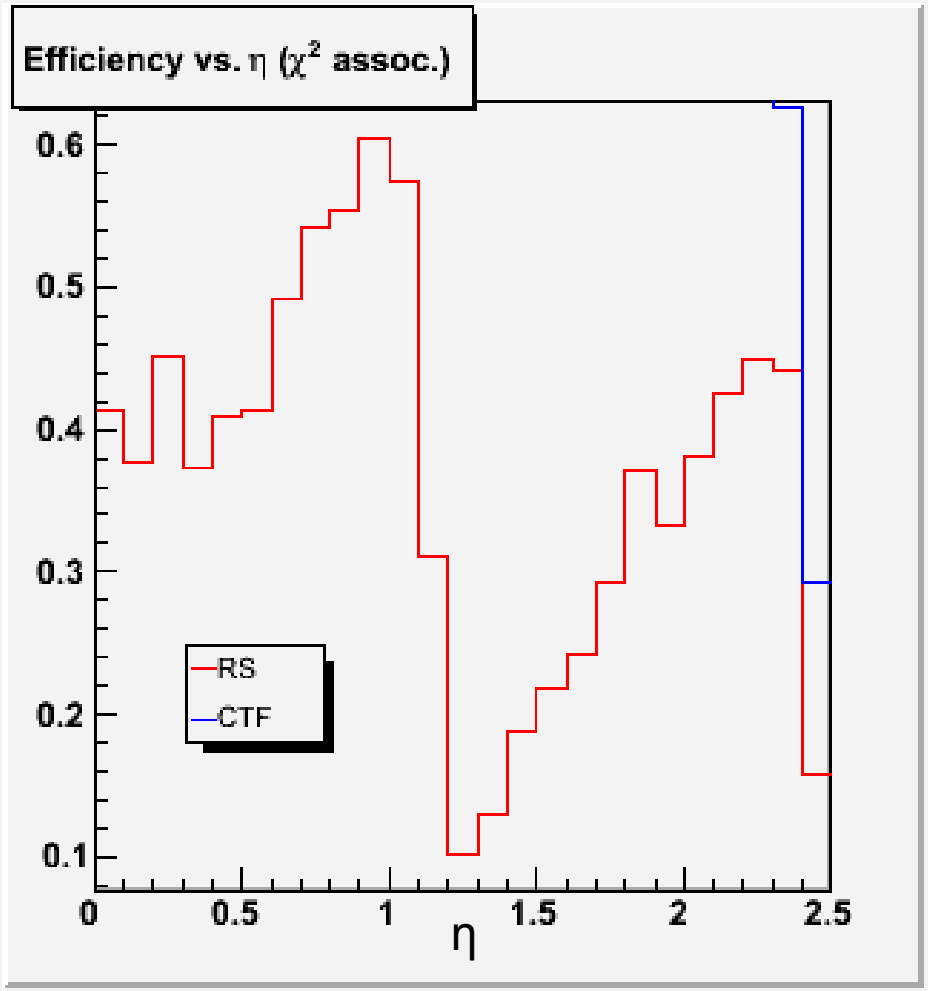
# Validation: $H^0 \rightarrow Z^0\ Z^0 \rightarrow \mu^+\ \mu^-\ \mu^+\ \mu^-$



# Validation: 50% $\mu^+$, $\mu^-$, $p_T$=1 GeV

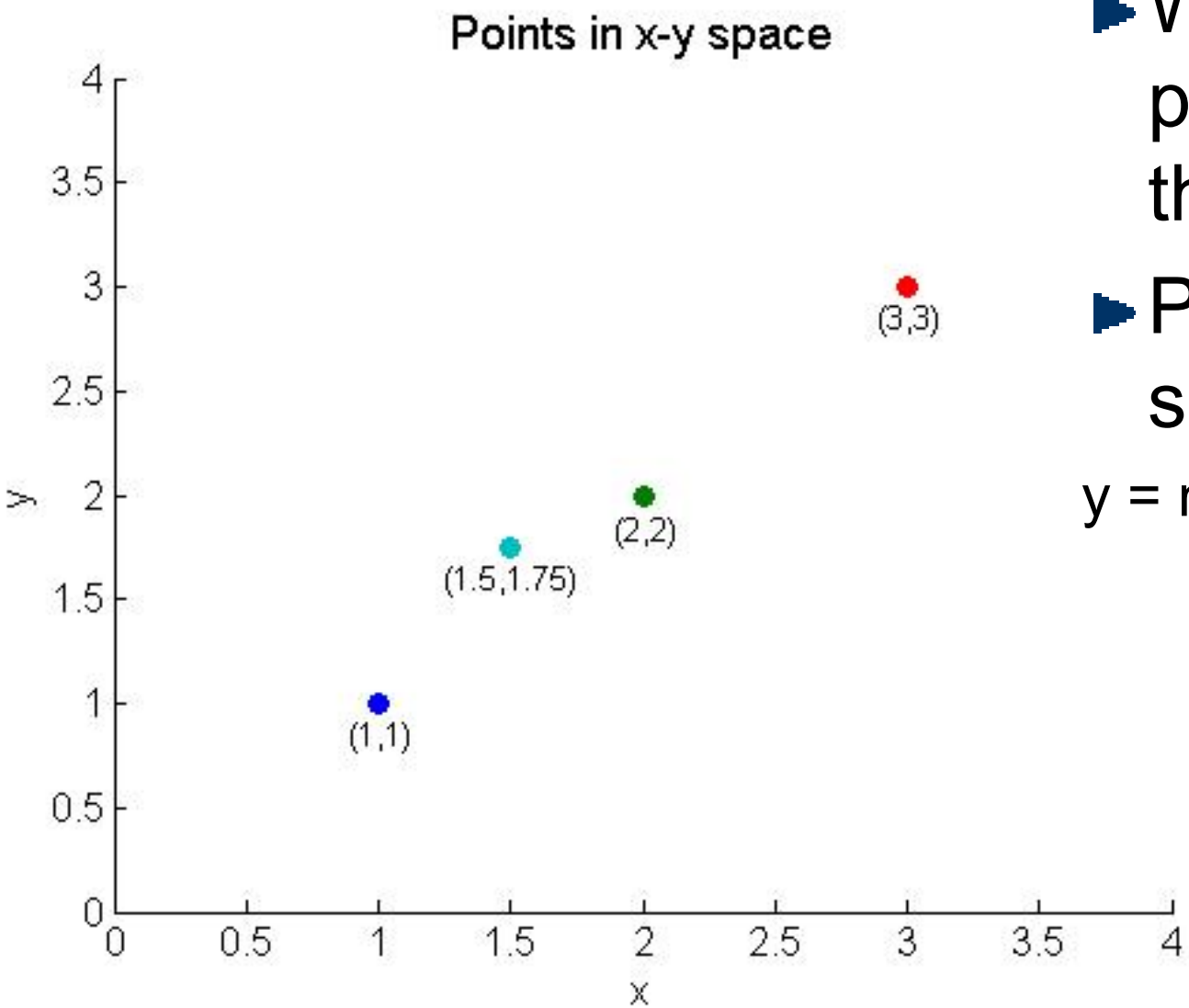


# RoadSearch Modification Proposal

- First:
  - Hough transform
- Then (in no particular order):
  - Coding tricks
  - Ordering of access to hit collection
  - Experimental determination of optimal location of and extent of cleaning(s)
  - Banana shaped clouds

# Proposal: Hough Transform




- What lines are present given these points?
- Plot in b-m space!

$y = mx+b \rightarrow b = -xm+y$

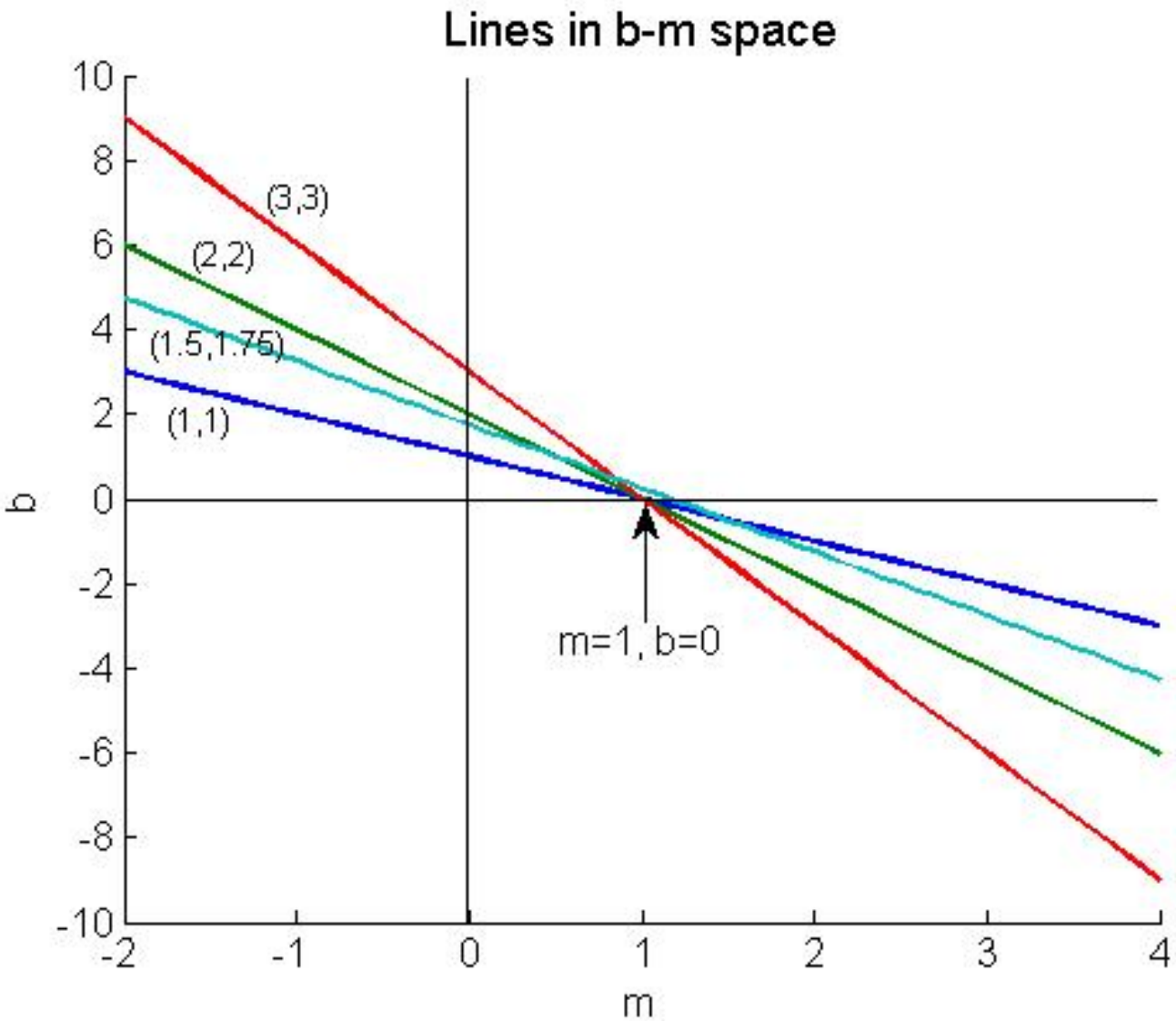
$b = -1m+1$

$b = -1.5m+1.75$

$b = -2m+2$

$b = -3m+3$

# Proposal: Hough Transform




- Harvest points of intersection in b-m space.
- O(n) !
- Can be modified for more complex objects & error
- Will test in seed finding, cloud making, and possibly track candidate creation

# Proposal: Coding Tricks

- Loop splitting

Poor performer:

```
sum = 0;
for (i=0; i<size; i++)
{ sum = sum + x[i]; }
```

Better:

```
split = size;
while (split != 1)
{
  split = split/2;
  for (i=0; i<split; i++)
  { x[i] = x[i] + x[i+split]; }
}
sum = x[i];
```

- Loop divisions

Poor performer:

```
for(i=0; i<n; i++)
{
  y[i] = x[i]/a;
}
```

Better:

```
b = 1/a;
for(i=0; i<n; i++)
{
  y[i] = b*x[i];
}
```

# Proposal: Order of Hit Access

- Cloud maker, possibly track candidate maker
- Current cloud maker creates clouds one at a time, where clouds share hits
  - Access clouds once and in order
  - Access hits multiple times, out of order
- Instead, loop through hits, modifying all the clouds as you go
  - Access clouds multiple times and out of order
  - Access hits once, in order

# Proposal: Order of Hit Access

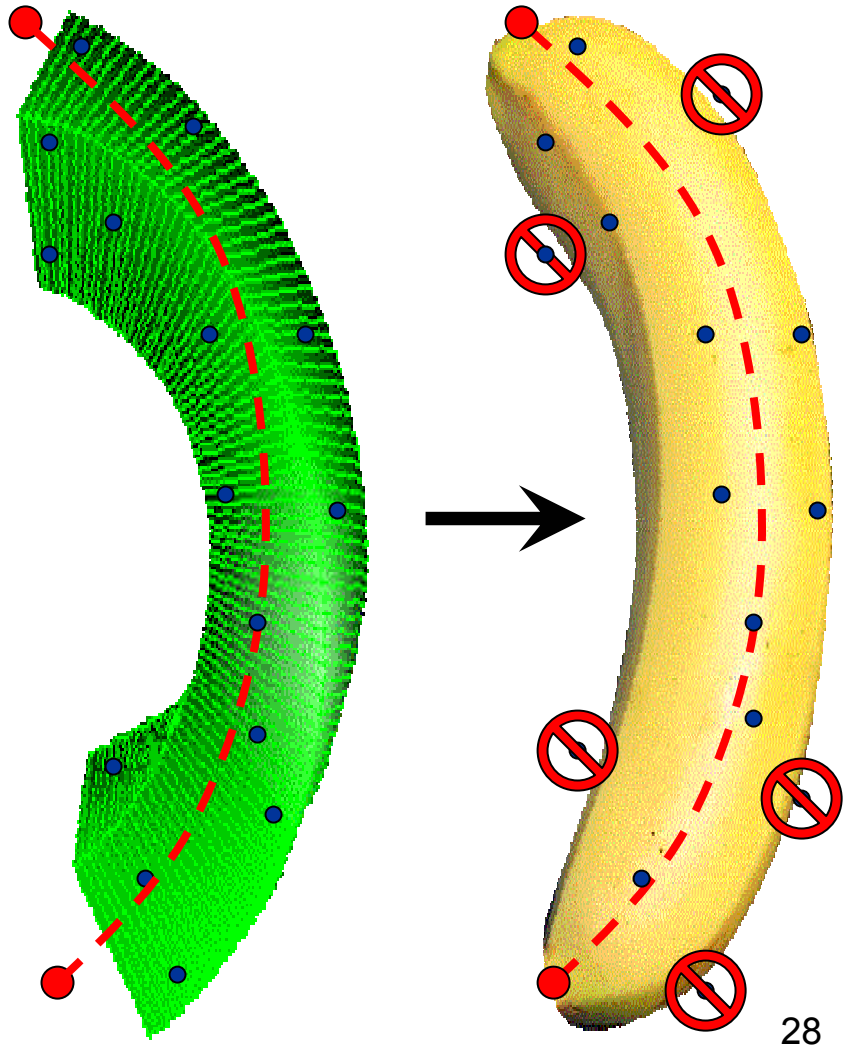
- Substantially increases complexity of code
  - Flow less ‘natural’
  - Requires complex intermediate objects w/pointers
- Is this worth trying?
  - What is the ratio:
    (# of detectors in clouds)/(total # detectors)?
  - If ratio is ‘small,’ yes, worth testing

# Proposal: Location(s) of Cleaning

- Cleaning = removal of duplicates
- Possible at all stages
  - Seeds
  - Clouds
  - Track Candidates
- Experimentally determine optimal location(s) for and extent of cleaning

# Proposal: Banana Shaped Clouds

- Cloud maker is currently undergoing change to ‘linguini’ shape.
- Since we have some level of confidence in our seeds, a ‘banana’ shape would be fair.
- Banana will have more parameters and is a more complex object.
- Is the reduced number of hits in cloud worth the increase in complexity of the cloud maker?



# Summary

- Goal: Improve the running time of an existing track finding software package designed for the CMS experiment while preserving physics performance.
- Milestones
  - Performance Report
    - Now have baseline
    - Others have substantially increased physics performance and decreased running time in last semester
  - RoadSearch Modification Proposal
    - Have a solid ‘plan of attack’ for next semester

# Acknowledgements


- Kevin Burkett (aka “The Bad Cop”)
  - Guidance, new track candidate maker
- Giuseppe Cerati
  - Track Validator
- Bill Dorland
  - Providing resources & contacts
- Ramani Duraiswami
  - Idea of Hough Transform
- Oliver Gutsche (aka “The Good Cop”)
  - Guidance, timing
- Nicholas Hadley
  - Guidance, physics, Pythia
- Pete Stewart
  - Lecture notes & book draft providing ‘coding tricks’
- Alan Sussman
  - Idea of ordering of data access