sPHENIX Track Reconstruction

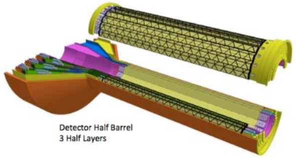
Christof Roland, MIT

TPC Workshop

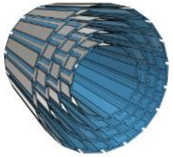
July 12, 2019

CERN

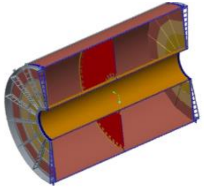
The sPHENIX tracking system



- MVTX
  - **3 layer vertex tracker based on Monolithic Active Pixel Sensors**
  - **Provides impact parameter resolution**

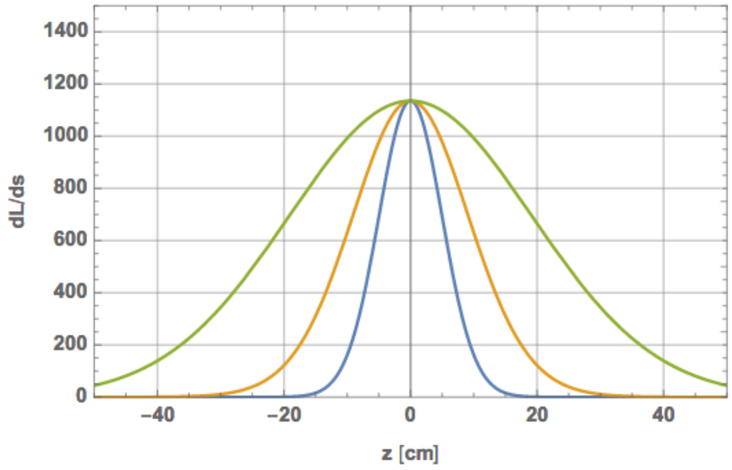


- INTT
  - **2 hermetic Layers of Silicon Strip detectors**
  - **Fast response time for pileup disambiguation**

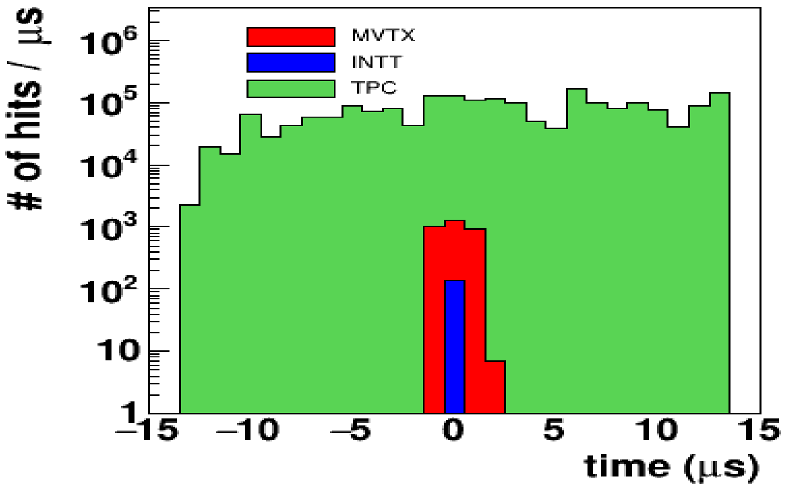


- TPC
  - **48 layer continuous readout TPC**
  - **Main tracking device**
  - **Provides momentum resolution**
- All detectors immersed in a 1.4T magnetic field

Data taking conditions



0, 1.5, 3 mrad crossing angles

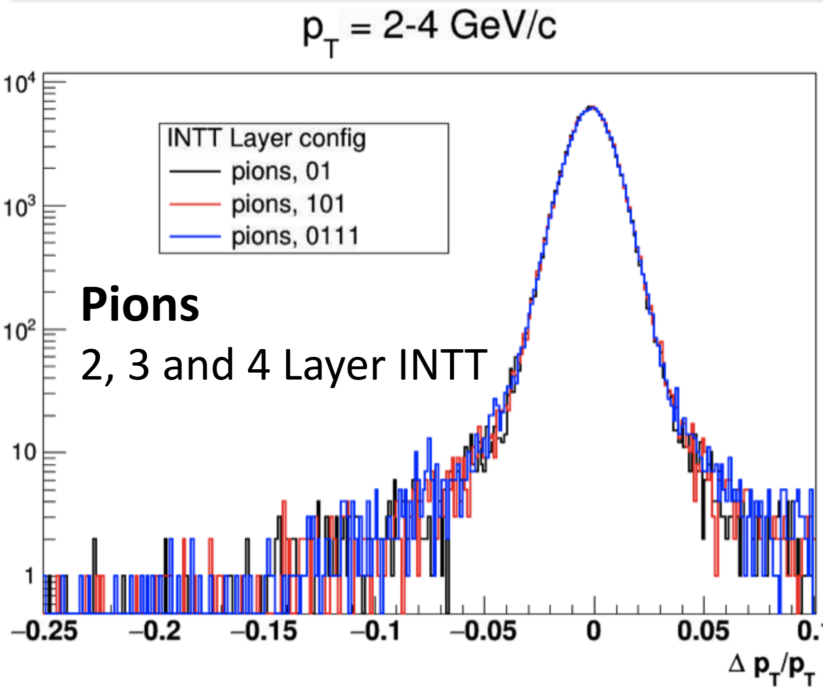


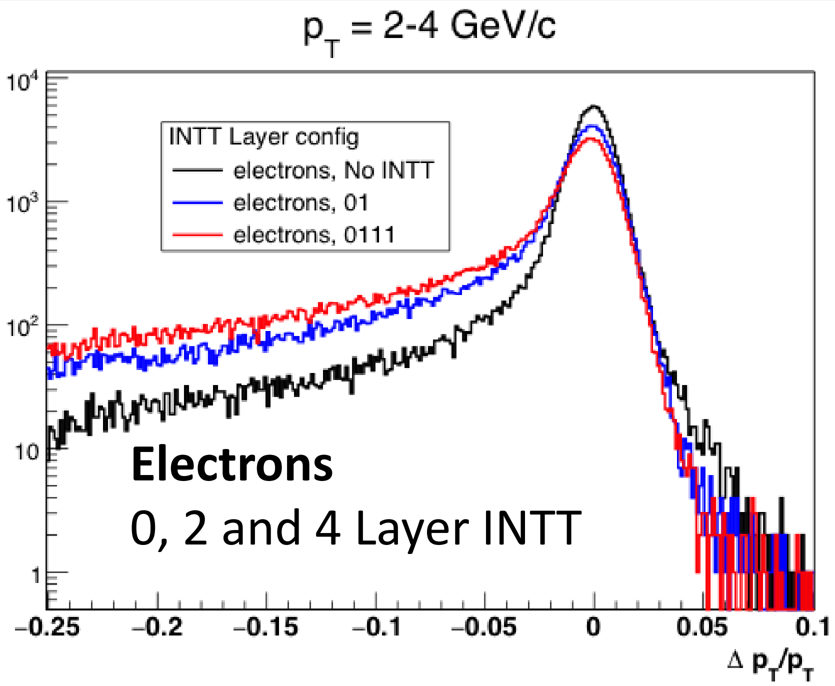
- RHIC will provide up to 200kHz interaction rate of Au+Au
  - **Large beam diamond, σ ~30cm at 0 mrad crossing angle**
- Useful fiducial volume due to detector layout constrained to |vz| < 10cm
  - **Collect 15kHz of min bias data**
    - **Substantial out of time pileup contribution in the TPC volume**

DETECTOR OPTIMIZATION

Track Parameter Estimation vs INTT Mass (nLayers)

sPHENIX

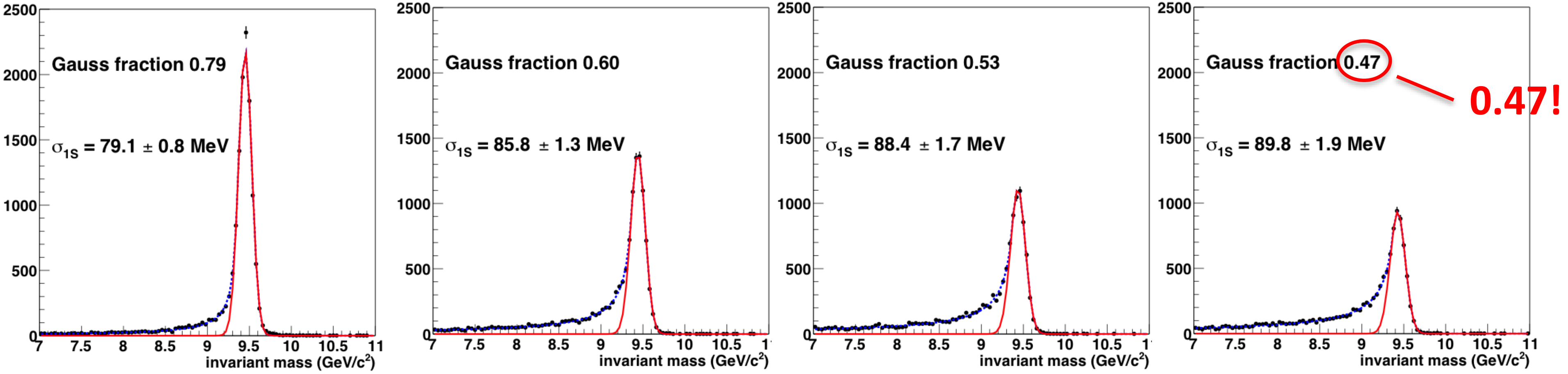




- The pion momentum resolution does not depend on the number of INTT layers
  - **Verified by independent study based on the LIC detector model**
- The electron reconstruction is severely affected by the material budget
  - **Contrains the capability to study quarkonia in the electron channel ☹**

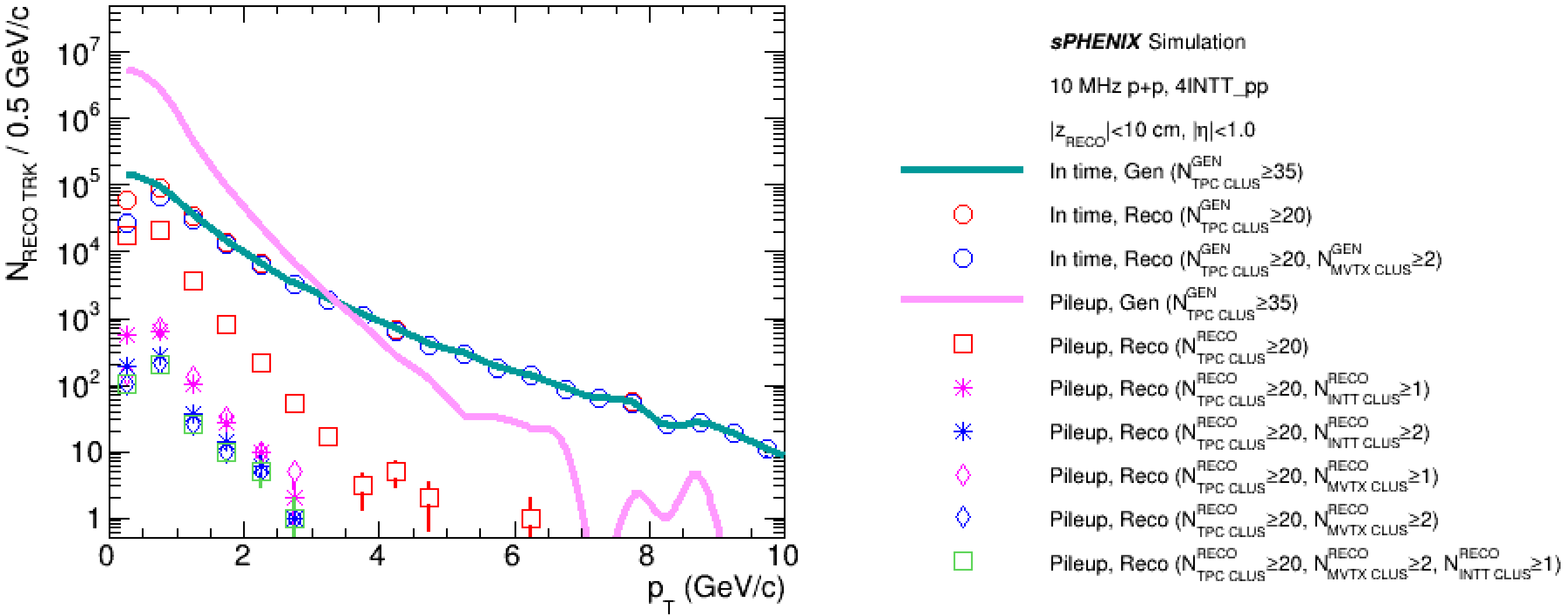
Impact on Upsilon Mass Spectra…

No INTT | 2 Layer INTT | 3 Layer INTT | 4 Layer INTT



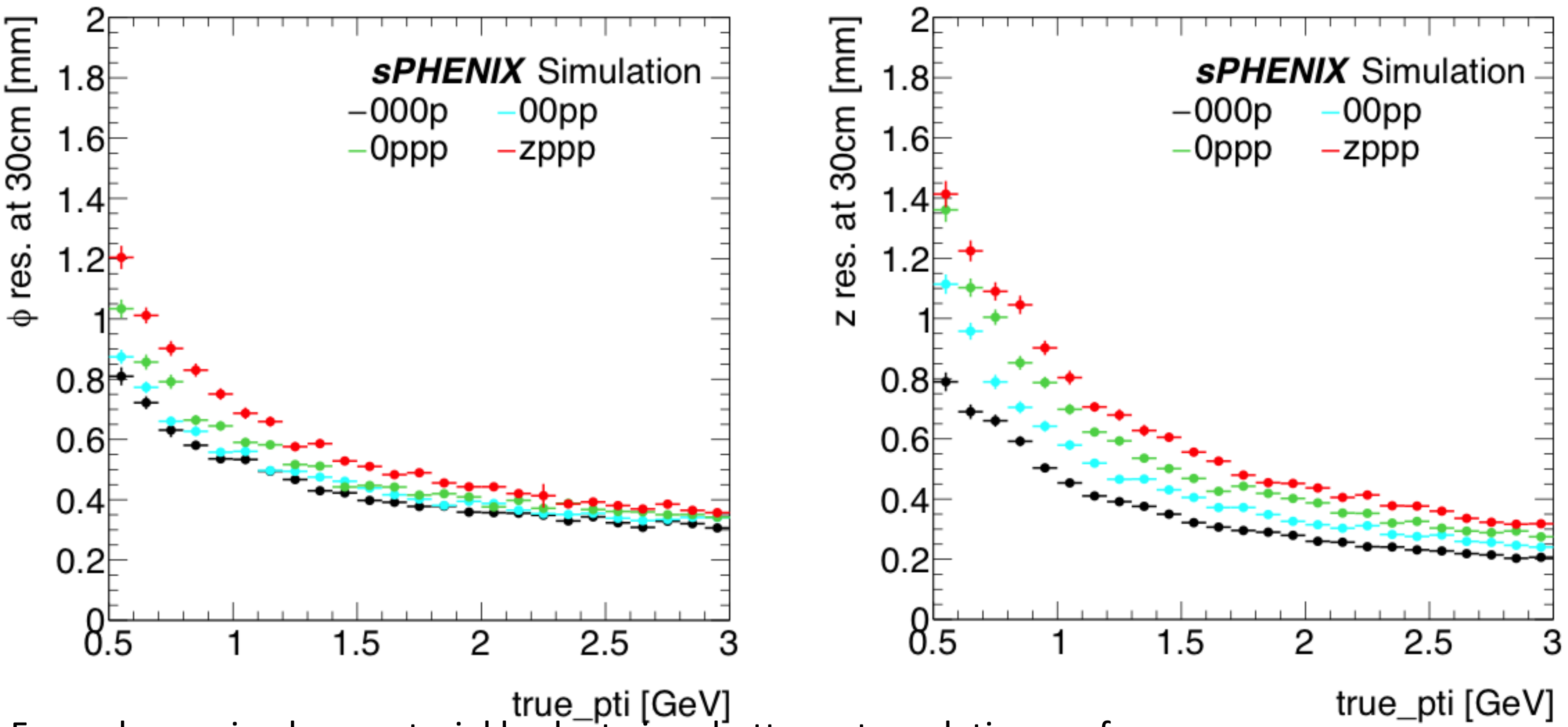
- Simulation done with the latest INTT and TPC simulation
  - **Need to reduce the material budget to the minimum necessary**

pp pile-up track contribution



- Requiring 2 MVTX hits and one INTT hit per reco track gives good rejection power against tracks originating from out of time pile up

Extrapolation from ITS to TPC

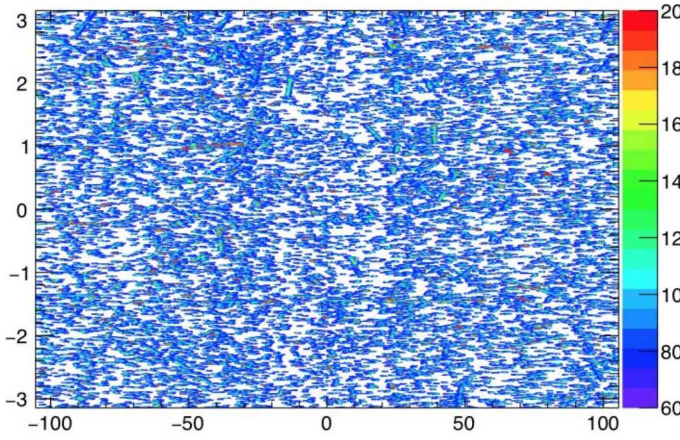


- Fewer layers, i.e. less material budget gives better extrapolation performance
  - **phi resolution similar with 1 and 2 layers.**
  - **Extrapolation precision significantly worse than the expected position accuracy of the TPC (~150μm)**

PATTERN RECOGNITION

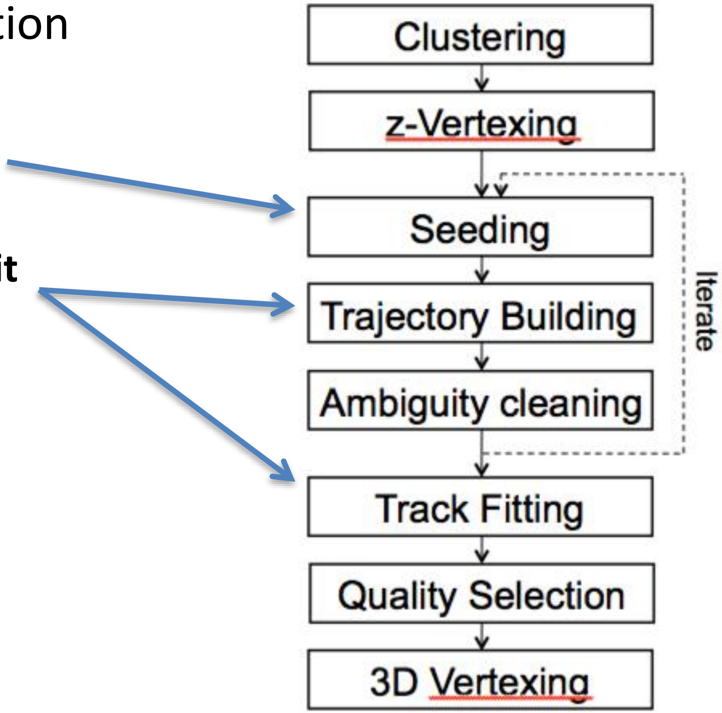
Pattern recognition challenges

- Optimal parameter estimation
  - **Momentum resolution**
  - **Impact parameter resolution**
- Robustness against very high occupancy
  - **High tracking efficiency**
  - **Little or no hit density dependence**
    - **Expect up to 30% detector occupancy in the TPC**
- Efficient CPU usage
  - **Goal is to reconstruct 100 Gevents per year**
  - **CPU usage dominated by track reconstruction**



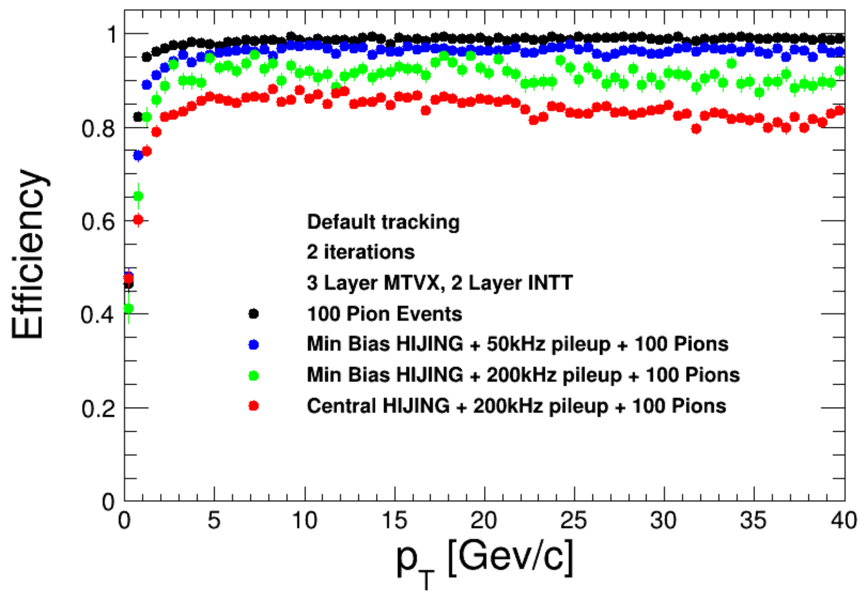
The algorithm

- Iterative Kalman Filter based track reconstruction package
  - **Hough transformation based seeding algorithm**
    - **Provides redundance against missing hits**
    - **Outside in approach**
  - **Track propagation and fitting based on the GenFit package**
    - **Open source software**
    - **Well tested through use in different experiments**
      - **E.g. PANDA, BELLE**
    - **Manpower efficient implementation**
  - **2 Iterations with hit removal and different seed constraints**
    - **4 hits out of 7 layers**
    - **6 hits out of 12 layers**



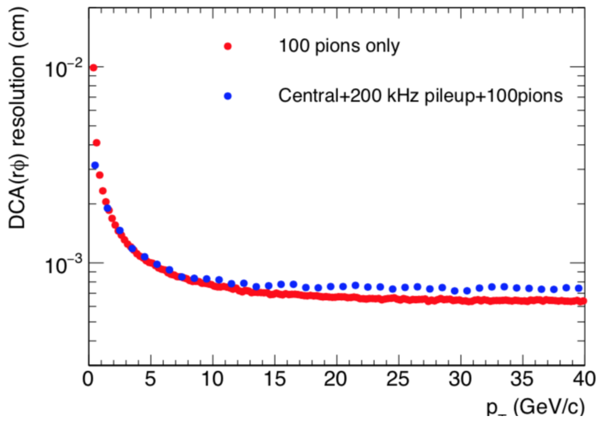
Performance: Tracking Efficiency

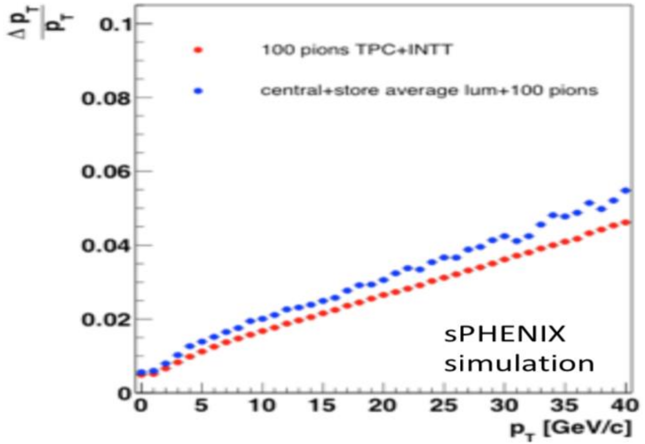
- 100 pions embedded in HIJING
  - **Various instantaneous luminosity scenarios**
- Workable efficiency in worst case occupancy
  - **Efficiency recoverable with**
    - **Better clustering algorithm**
    - **more iterations at the expense of CPU cycles**



Performance: Parameter Estimation

- Unbiased momentum and impact parameter resolution in low occupancy events
- Slight deterioration of performance at high occupancy due to cluster centroid determination of overlapping clusters
  - **To be fixed by more sophisticated clustering algorithm, e.g. neural networks**

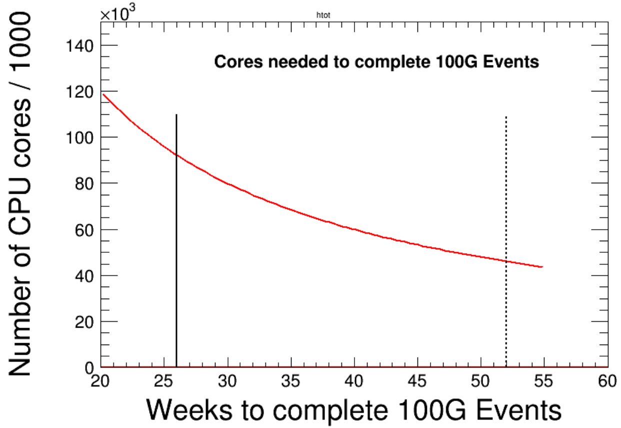




CPU resources for Tracking

- Prepare to reconstruct 96 billion events in Year-3
    - **96 000 000 000 ev / 3600 * 24 * 365 sec ~ 3000ev/sec**

- Resources needed
    - **Assuming 15 sec per event:**
        - **45k equivalent-cores to reconstruct the data**
        - **within the year they were taken**
            - **Currently 34kCores allocated to STAR+PHENIX**
        - **90k equivalent-cores for fixed latency reconstruction,**
            - **i.e reconstructing the data as they are taken modulo a calibration delay of 2-3 days**
    - **Set target for tracking to 5 sec***
        - **leave 10sec per event for calibration, Calo reconstruction, Particle Flow etc.**

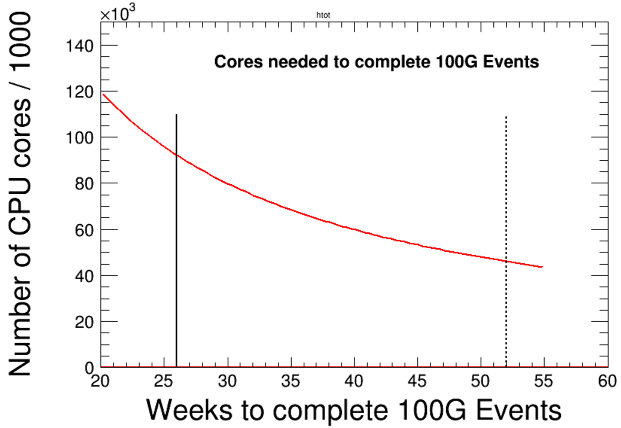


***Benchmark numbers discussed in the Computing Review, July 2018**

CPU resources for Tracking

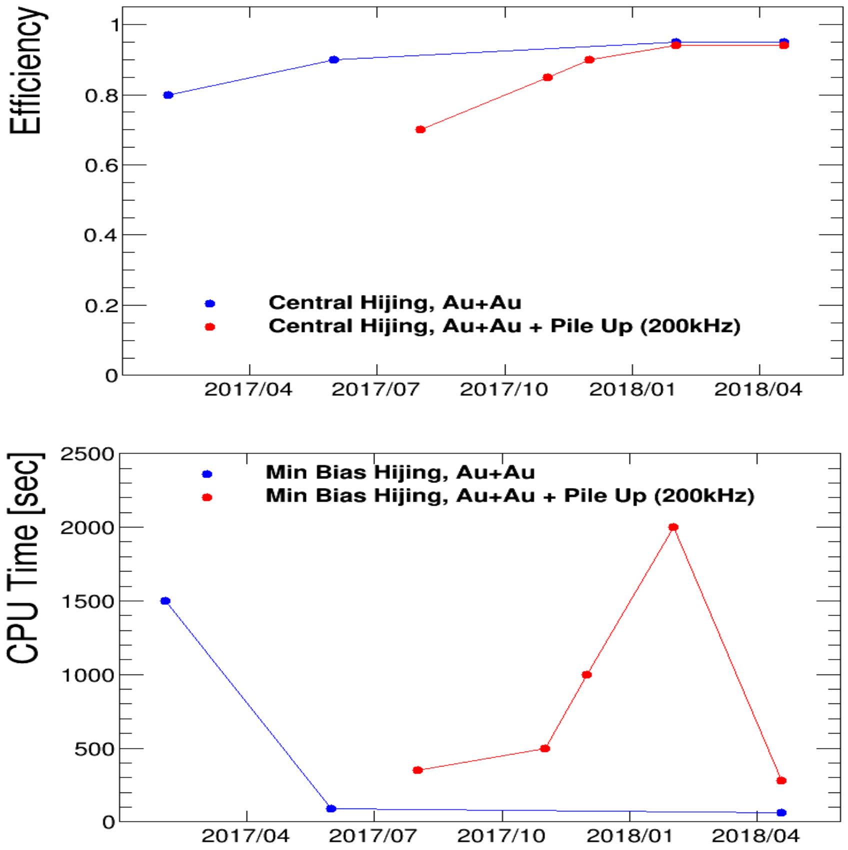
- Prepare to reconstruct 96 billion events in Year-3
    - **96 000 000 000 ev / 3600 * 24 * 365 sec ~ 3000ev/sec**

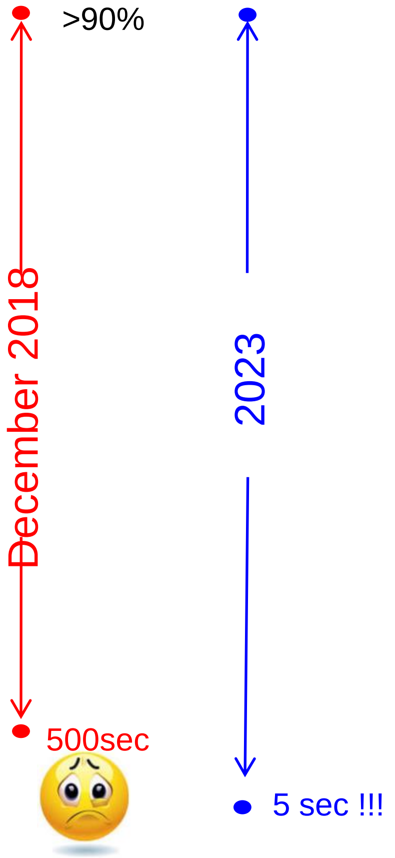
- Resources needed
    - **Assuming 15 sec per event:**
        - **45k equivalent-cores to reconstruct the data**
        - **within the year they were taken**
            - **Currently 34kCores allocated to STAR+PHENIX**
        - **90k equivalent-cores for fixed latency reconstruction,**
            - **i.e reconstructing the data as they are taken modulo a calibration delay of 2-3 days**
    - **Set target for tracking to 5 sec***
        - **leave 10sec per event for calibration, Calo reconstruction, Particle Flow etc.**



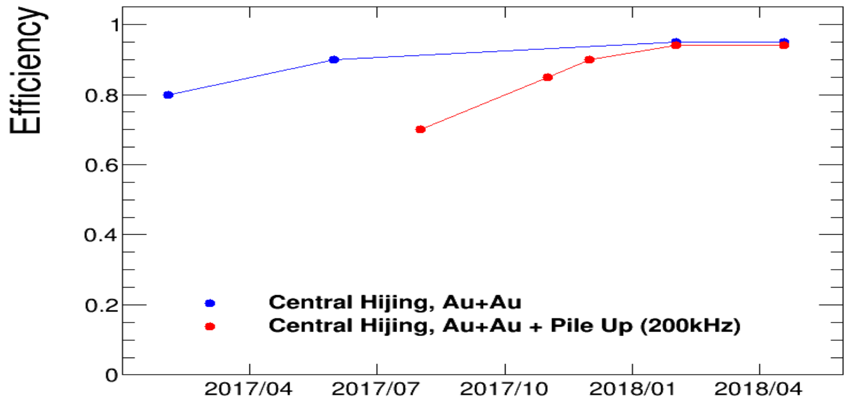
***Benchmark numbers discussed in the Computing Review, July 2018**

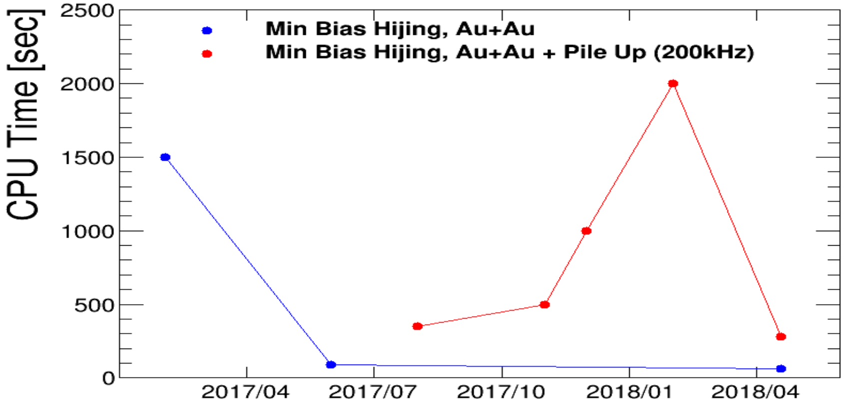
sPhenix Tracking Evolution vs time

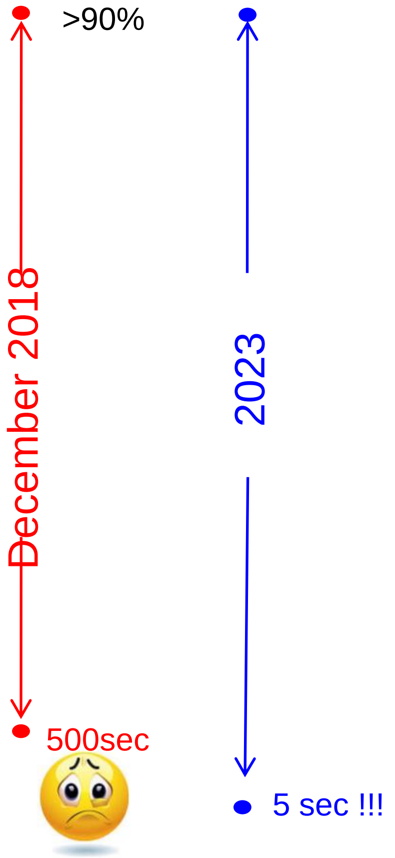




sPhenix Tracking Evolution vs time





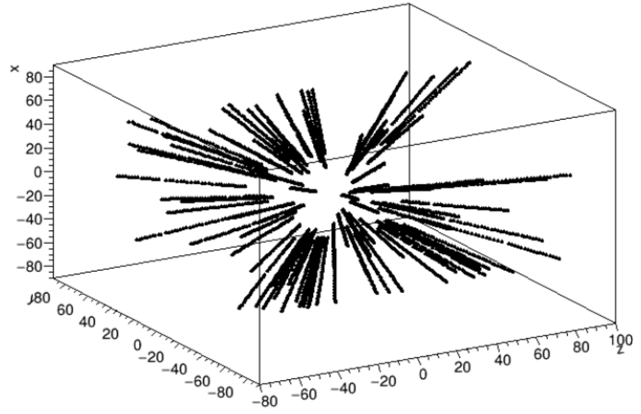


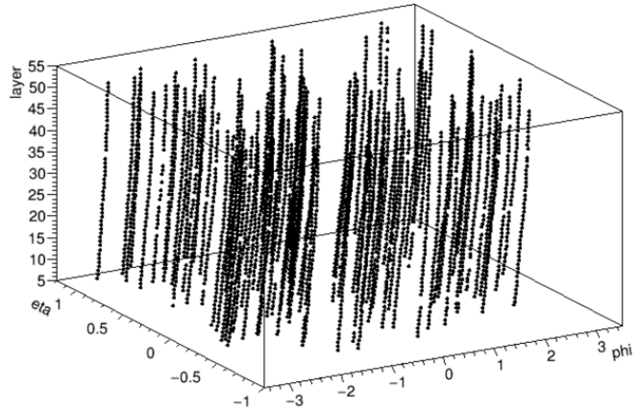
Need to speed up the track reconstruction by 2 orders of magnitude !!

Outlook on algorithms…

Ideas for new Seeding scheme

- Nearest neighbor search in TPC
  - **Use geometric indexing**
    - **E.g. R-trees from boost**
    - **Efficient access to hits near a given position (prediction)**
- Proof of principle study
  - **800ms to assemble track stubs in AuAu+100kHz pileup**
    - **Hough seeding was ~400sec**
- Optionally track the entire TPC
  - **For the short extrapolation distances in the TPC we can probably find computationally cheaper algorithms than the KF**
    - **Graph Tracking and/or Cellular automata**
    - **Ideas from the TrackML challenge**
- Update the Kalman Filter to match to the INTT+MVTX to state of the art code
  - **Replace GENFIT with ACTS?**
  - **GENFIT seed fit -> 40ms/track (~6hits on track)**
  - **ACTS Package**
    - **Track propagation – 0.5ms/track (53 Layers)**
    - **Full RKF fit – 1-2 ms/track**





Outlook

- We have ~3 years to bring our tracking code up to speed!
  - **Current algorithm provides very high tracking efficiency and good robustness in view of the high occupancy conditions**
    - **Too slow and consumes to much memory ☹**
- Target fixed latency reconstruction of recorded data -> 5sec/ev
  - **We have a good idea of the code performance we need to achieve**
  - **There is technology available to get us there**
  - **Progress currently limited by the small size of the Tracking Team**
    - **Special thanks to Tony, Haiwang, Darren and Sookhyun for their heroic effort!**
- Open to creative ideas for new algorithms
  - **Particle tracking is a very active field of research**
  - **Many exciting new technologies emerging**
    - **Machine learning solutions**
    - **GPU/FPGA hardware acceleration**
  - **Many working "external" packages ready for testing**
    - **E.g ACTS (Open source ATLAS Tracking)**
    - **Manpower efficient implementation!**

BACKUP

Is 5 sec per event a realistic goal?

Current sPhenix CPU Benchmark

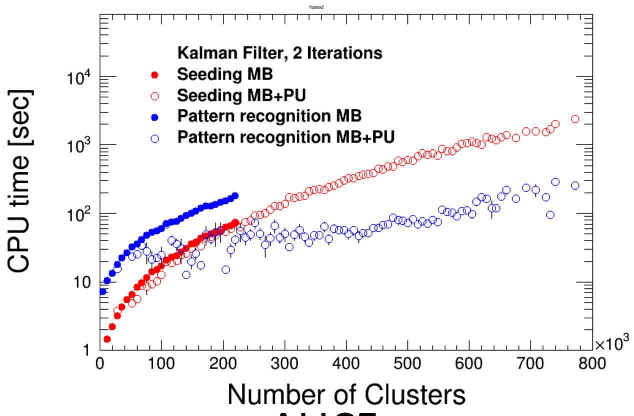
- Min bias + pileup (200kHz): 480sec
    - **Seeding: 420sec**
        - **scales with: $nhit^2$**
        - **Out-of-time pileup hits a big concern**
    - **Pattern Recognition: 60sec**

Examples from other experiments

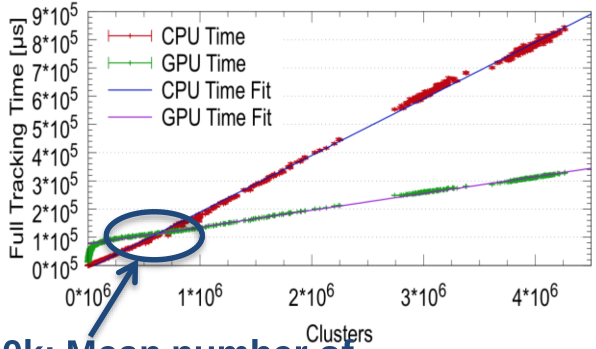
- HLT tracking of ALICE
    - **TPC Only -> sPhenix seeding step**
    - **~1sec per ALICE event on 1 CPU, 24 Threads**
        - **0.15 sec when scaled with sPhenix number of clusters on 24 threads**
        - **Min Bias + PU 420 sec -> 3.5 sec (one thread)**
- ATLAS tracking (ACTS Package)
    - **Track propagation – 0.5ms/track (53 Layers)**
    - **Full RKF fit – 1-2 ms/track**

5 sec per event should be feasible with state of the art technology

sPhenix



ALICE

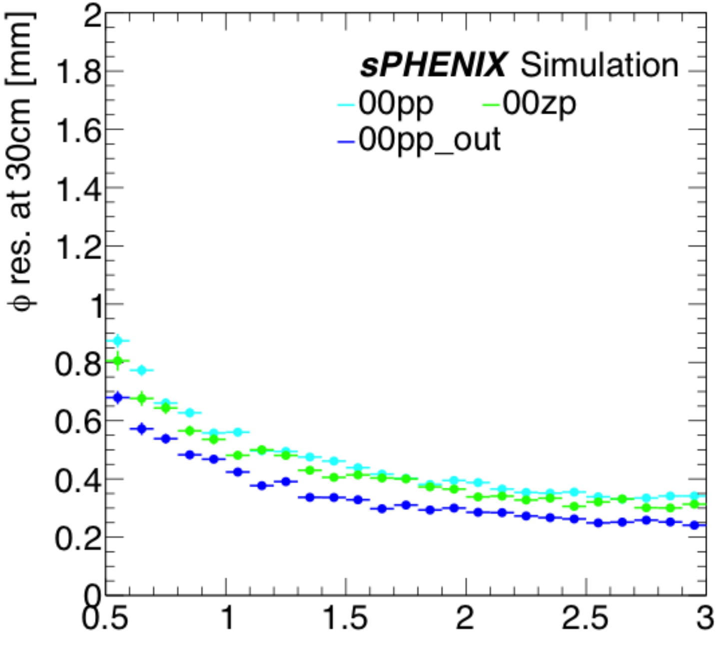


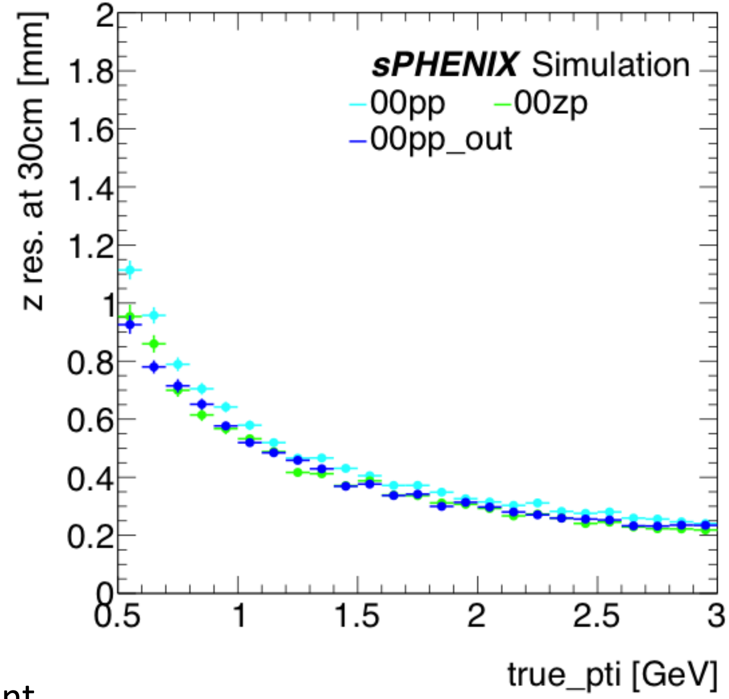
400k: Mean number of clusters in Min Bias + pileup

Extrapolation to the TPC

- Need 150um position accuracy in the TPC to achieve our momentum resolution goal
  - **Space charge distortions of O(3mm) expected**
  - **Need external calibration**
- Test extrapolation precision from MVTX + INTT
  - **Truth tracking to assemble the hits + Kalman Fit**
  - **Extrapolate to 30cm radius and compare extrapolation to truth hit position**

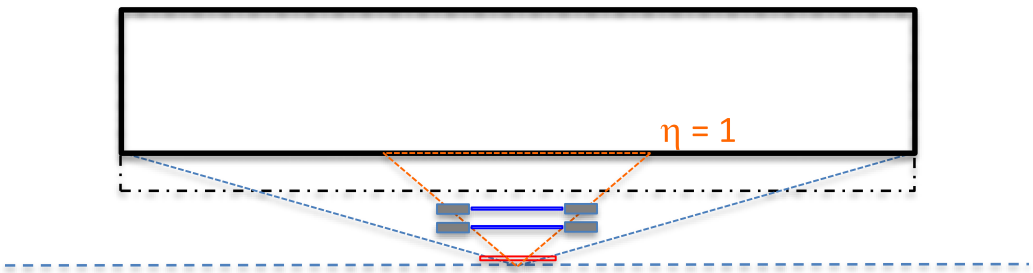
Comparing 2 layer configurations





- Moving layers to larger R gives modest improvement
- Adding a layer with Z resolution gives only a small improvement
  - **Two layers with phi resolution offers more redundance in case of dead channels/chips**

Placement of INTT layers



- Moving the INTT as close as possible to the TPC improves the extrapolation precision
  - **But cuts into eta acceptance and puts Electronics in the path of high eta tracks**
- Putting to outer INTT layer at ~11cm gives full eta coverage ($|\eta|<1$) for $|z_{VTX}|<10$cm
  - **Poorer extrapolation accuracy to the TPC**
- Decided to prefer eta coverage over extrapolation precision

Benchmarks from LHC experiments

- LHCb gets a reconstruction time of ~1ms per pp event + pileup (~500 tracks) from heavy use of neural networks, lookup tables and machine learning…

- CMS tracking software takes ~3ms per track with at nhit2 CPU dependence -> would translate to 1.5-5 sec for a sPhenix event

- ATLAS is about to release an open source tracking package – ACTS project
  - **A. Salzburger has provided preliminary estimate for the performance for sPhenix**

- Many proposals sent to NSF etc studying potential tracking acceleration using GPUs, FPGAs and ML in any linear combination thereof. Very promising approach.

- CONCLUSION
  - **Assume a conservative kalman filter approach with a fully optimised code implementation**
  - **Estimated target CPU performance:**
  - **MB events + pileup:**
    - **5 sec/event for tracking**
    - **5 sec/event for TPC clustering, calibration, 3D vertexing and other services**
    - **5 sec/event for calorimeter and particle flow reconstruction**

    => 15 sec/event total projected event reconstruction time

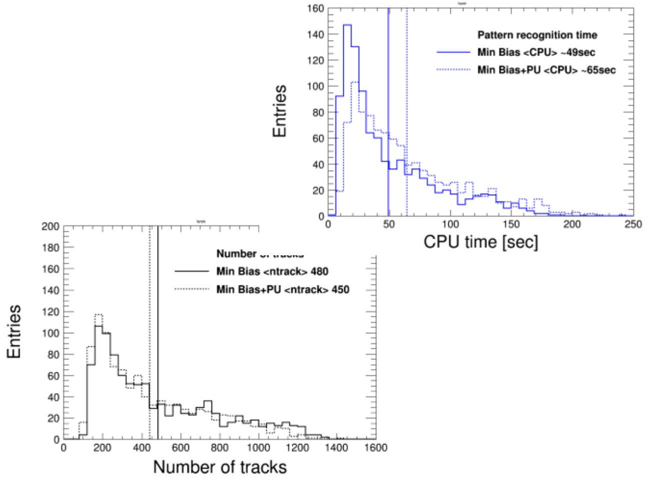
Outlook on CPU performance I

The current Kalman Track fit is very slow

- GenFit performance:
  - **seed fit -> 40ms/track (~6hits on track)**
  - **Internal Geometry handling not optimized**
- CMS track fit performance
  - **1ms per track (14.5 hits on track)**
- ATLAS tracking (ACTS Package)
  - **Track propagation – 0.5ms/track (53 Layers)**
  - **Full RKF fit – 1-2 ms**
- Projection:
  - **Min Bias 49 sec -> ~1.0 sec**
  - **Min Bias + PU 65 sec -> ~1.5 sec**

```
One central HIJING event
Number of Tracks: 1329
=============== Timers: ===============
Seeding time:                 64.7336 sec
Seed Iter1 time:                28.1227 sec
Seed Iter2 time:                36.6109 sec
     - Seeds Hough:           64.6106 sec
     - Seeds Final:           0.0484869 sec
     - Seeds Cleanup:           0.0484863 sec
Pattern recognition time:     164.646 sec
     - Track Translation time: 67.9439 sec
     - Cluster searching time: 1.27433 sec
         - Encoding time:           0.0821716 sec
         - Map iteration:           0.96942 sec
     - Propagation time:    95.8403 sec
     - Cleaning time:    0.178403 sec
     - Kalman updater time:     44.5779 sec
=======================================
```

GenFit dominated



Data Volume Estimate

- Compressed Raw Data event size
  - **Run 3,5 Au+Au: 2.3Mbyte**
- Final Analysis Objects/Model not yet defined
- Candidate models under consideration:
  - **ALICE HLT data format**
    - **Store tracks + hit information as residuals relative to tracks**
      - **Allows to reapply distortion corrections and refit tracks**
      - **Provides a compression factor of ~5, i.e ~500kbyte/event**
  - **CMS miniAOD like data format**
    - **Limited precision storage of track/particle parameters including covariance matrices**
    - **~25bytes per PF candidate after root compression**
    - **For 800 PF candidates (2xnTracks) -> 20kbyte/event**

Event summary – 9BG event

- Total Memory usage of this event -> 8.95GB
  - **Number of Hits:**
    - **5481770 size: ~40bytes -> ~200MB**
  - **Number of Clusters:**
    - **284729 size: 144 + n refs: 4249983 -> ~50MB**
    - **Stored 3 times (at least) Clusters, 3DHits (hough), Measurements (genfit) => 150MB**
  - **Number of Cells**
    - **TPC|INTT|MAPS:**
    - **9212401 | 12481 | 83604 cell size: 136 -> 1210MB**
  - **Known sources of memory consumption account for ~2GB**

- Need to be careful with the implementation of our storage objects
  - **Likely culprit for "dark" memory: inheritance from TObject + heavy usage of templated data structures (STL)**
  - **Get rid of the cells (planned for next release) and optimize our simulation, reconstruction and storage strategies to optimize memory**