

Update on HoughLineFinder and Introduction to Kalman Filter Tracking Techniques

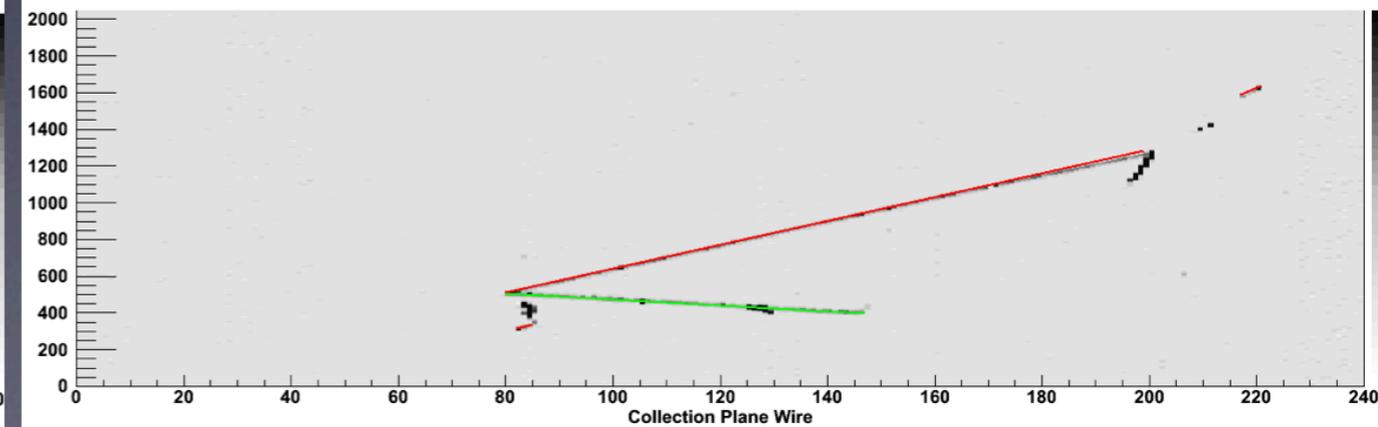
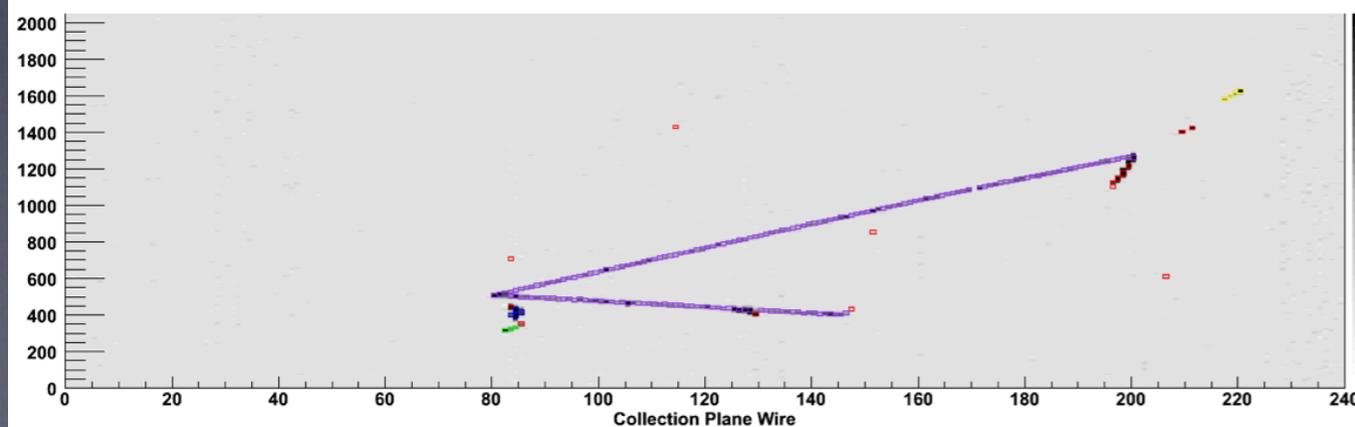
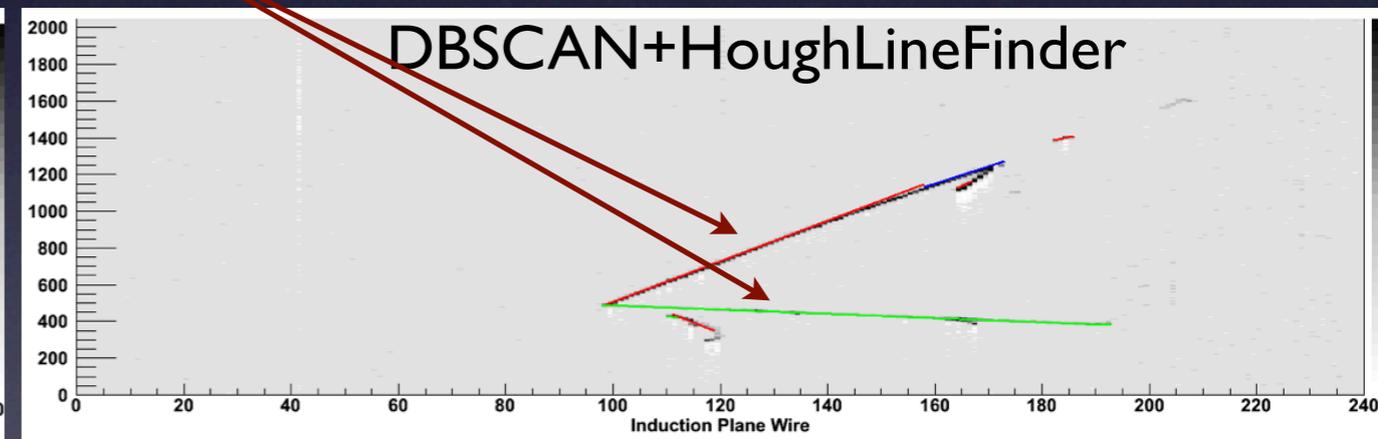
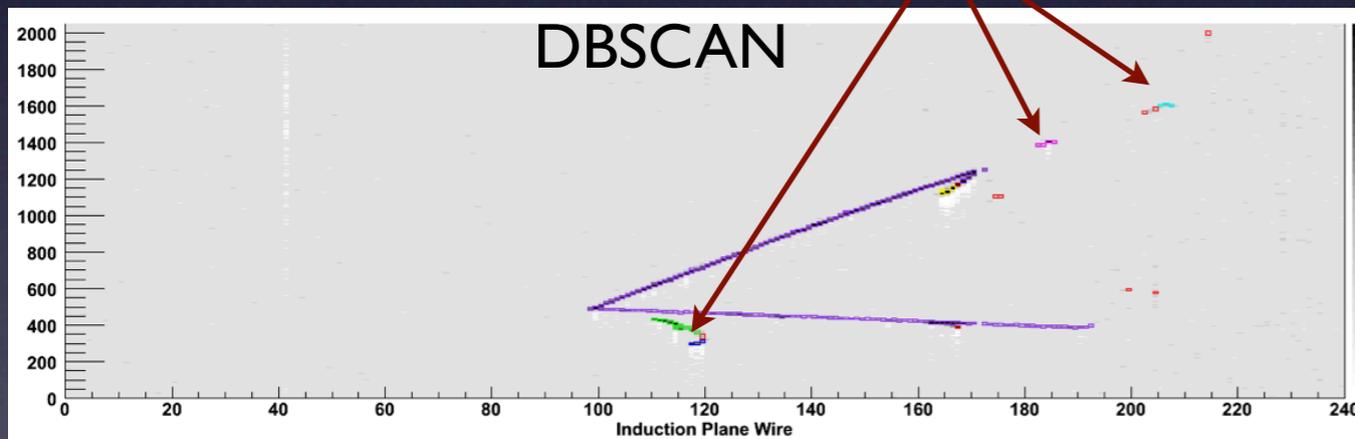
Joshua Spitz
4/1/2010

HoughLineFinder

- Brian P has made numerous improvements to HoughLineFinder recently.
 - Most notably, the speed of the algorithm has improved by about an order of magnitude by making the Hough Accumulator a vector of maps rather than a vector of vectors.
- With Kinga's help, I have added a `HoughLineFinder::Ana` method for filling histograms with relevant information about the line finding results and MC comparison. I will commit this code soon.

Thoughts on Cluster/Track Finding

- Cluster finding (DBSCAN and HoughLineFinder) is at a fairly advanced level in LArSoft.
- As it is now, cluster finding is a two step process:
 - 1. Run DBSCAN on all hits (from deconvolution, FFT hit finding) in a plane.
 - 2. Run HoughLineFinder on the clusters found with DBSCAN.
 - In terms of cluster finding, HoughLineFinder merely breaks up the clusters found with DBSCAN into “sub-clusters”. That is, a DBSCAN cluster that is not associated with a HoughLineFinder line is still considered a cluster. So, there are “not-line-like” clusters and “line-like clusters.”



Thoughts on Cluster/Track Finding

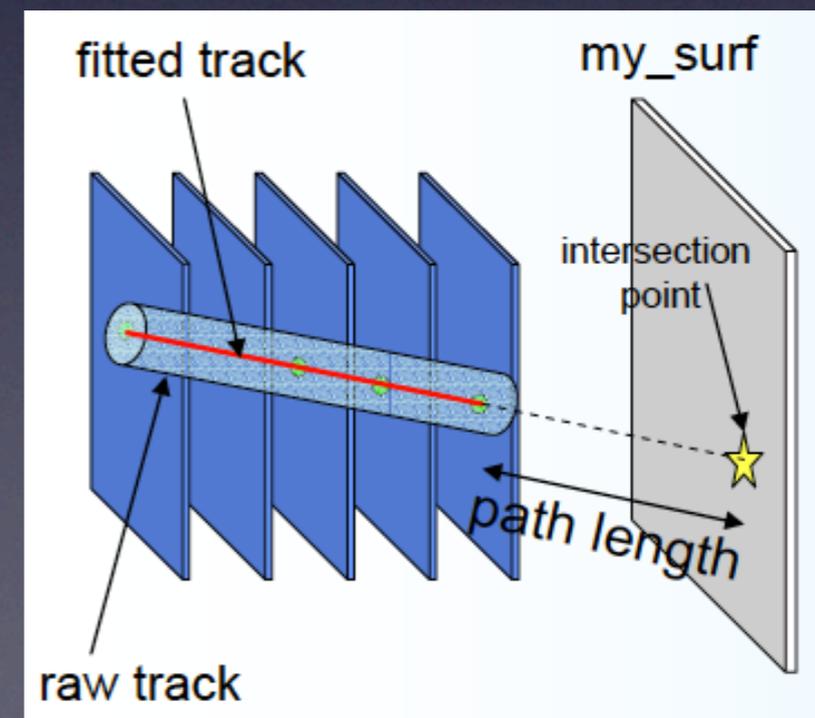
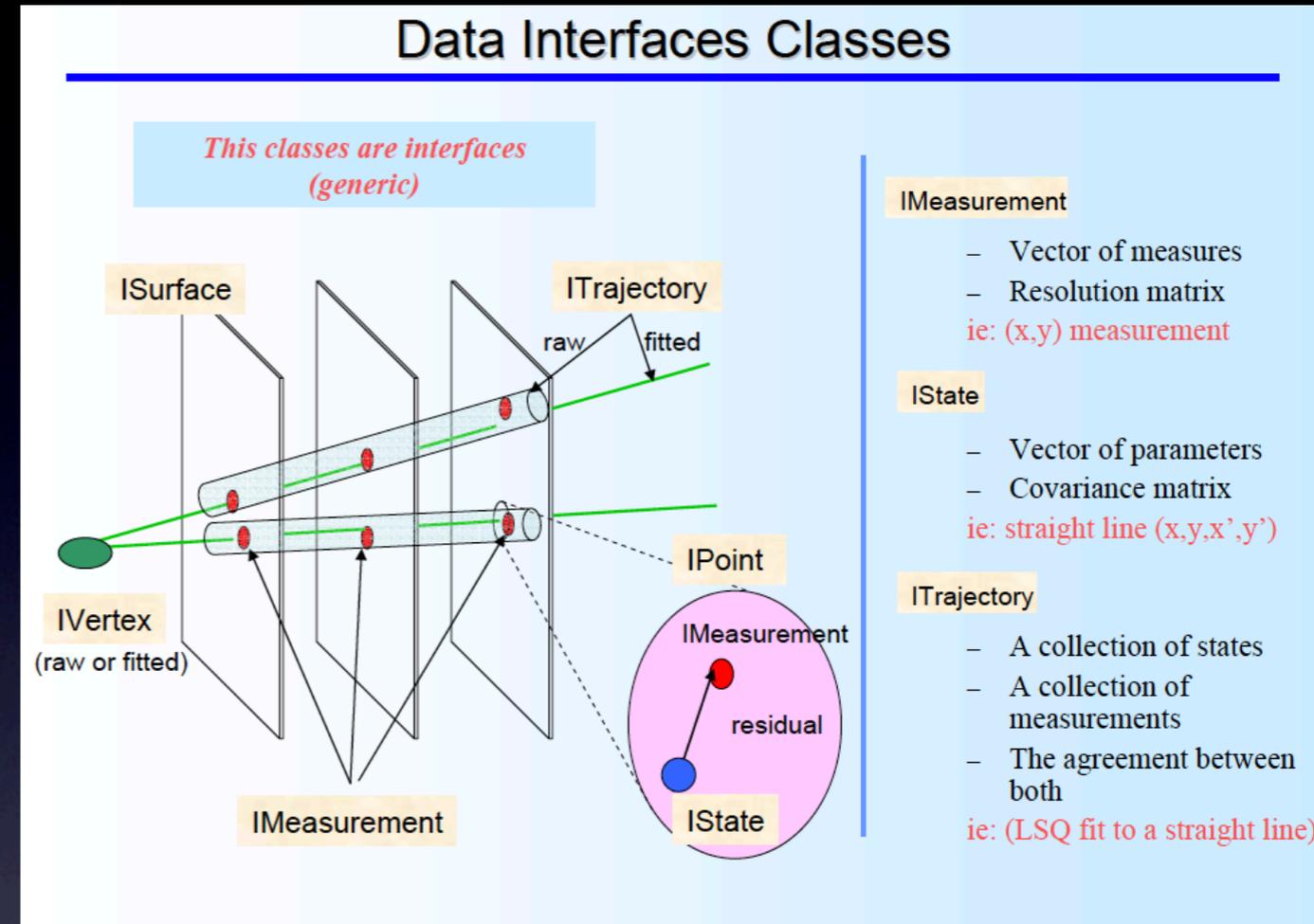
- A tracking algorithm takes hits from a cluster and attempts to fit trajectories (fitting parameters can vary along a trajectory) to the hits and eventually attempts to join clusters/trajectories to a vertex.
- HoughLineFinder can be considered a cluster finding AND tracking technique (even though it is in the ClusterFinder package) as it returns information about the trajectory of line-like tracks.
- We are at the point where we should start thinking about advanced tracking, i.e multiple scattering, vertex finding, shower reconstruction, etc. The Kalman filter is usually employed in HEP for these purposes.

RecPack

- It seems necessary to employ external code for implementing advanced reconstruction techniques in LArSoft.
- RecPack is an HEP-specific program for reconstructing trajectories and vertices using a Kalman Filter.
- It has been used by SciBAR and LHCb, among others.

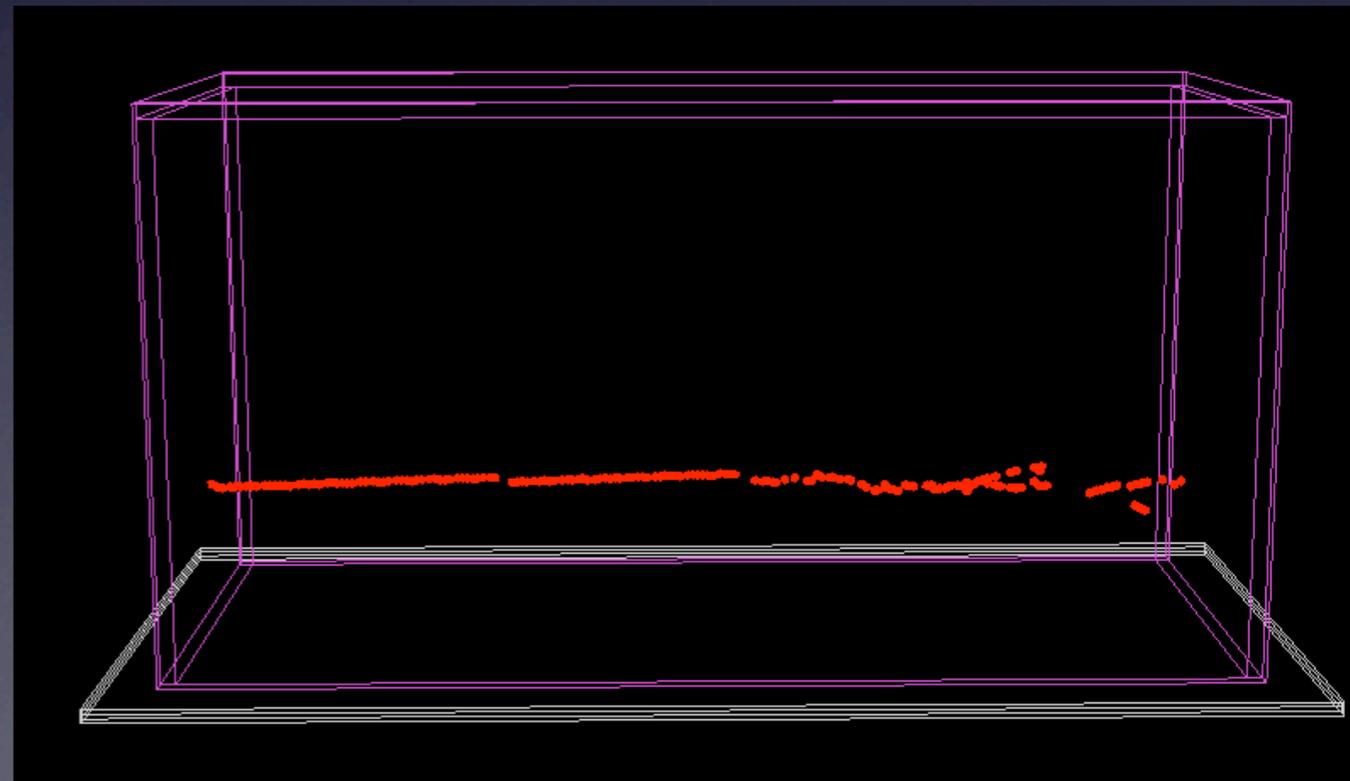
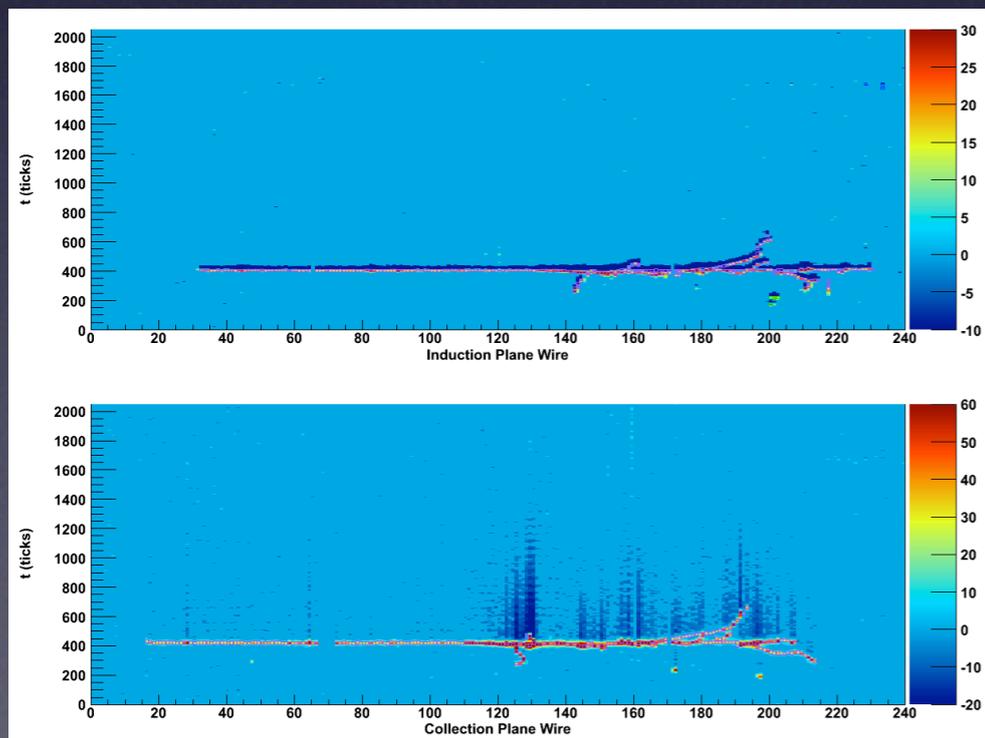
RecPack services

- Geometry
 - Definition of experimental setup.
- Model
 - Models for random noise, multiple scattering, energy loss, propagation, projection, intersection with surfaces, etc.
- Fitting
 - Given a seed state and measurements/empty states, this service transforms the information into a Kalman-filter-fitted trajectory.
- Navigation/Propagation
 - Propagation to a surface, propagation to a given length, path length computation.
- Matching
 - Finds the probability of two objects (trajectory-trajectory, measurement-trajectory, trajectory-vertex,) being related to each other (e.g. χ^2/ndf)
- Simulation
 - Validation tool.



Track fitting

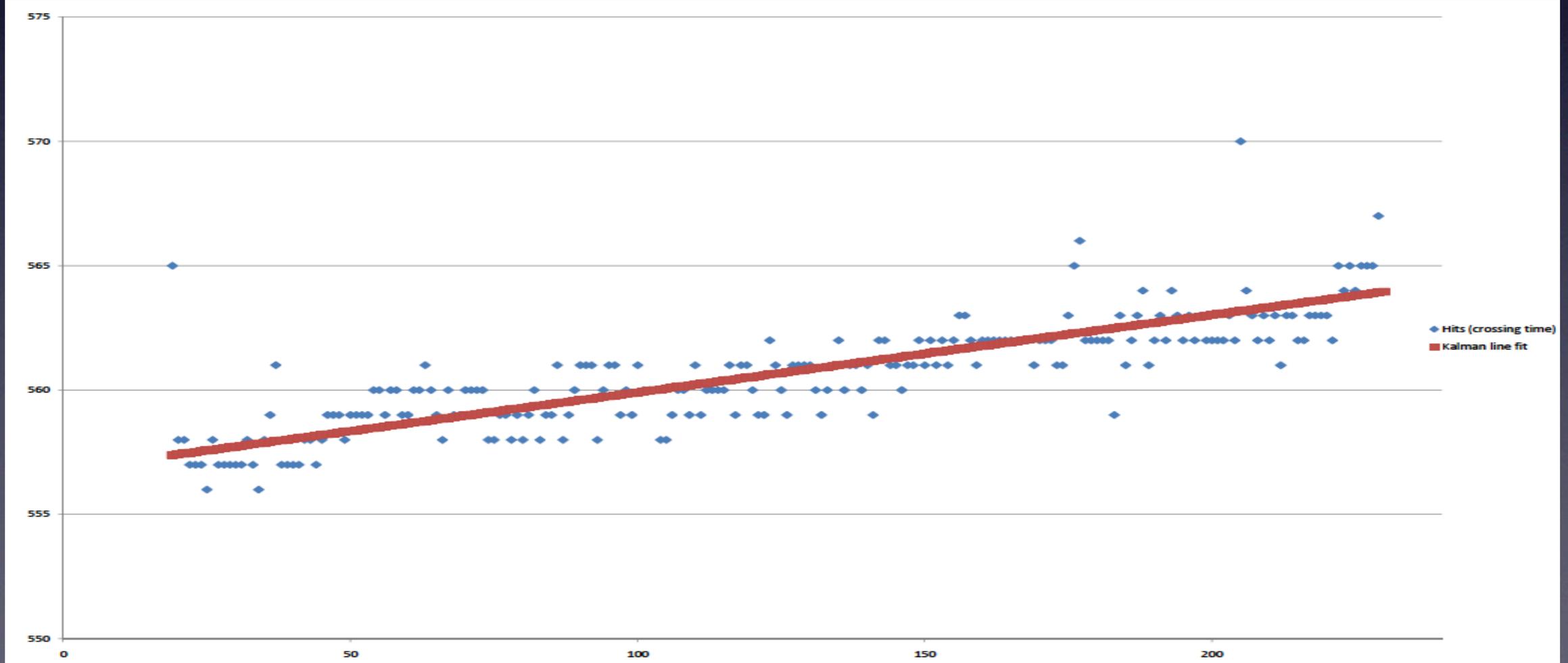
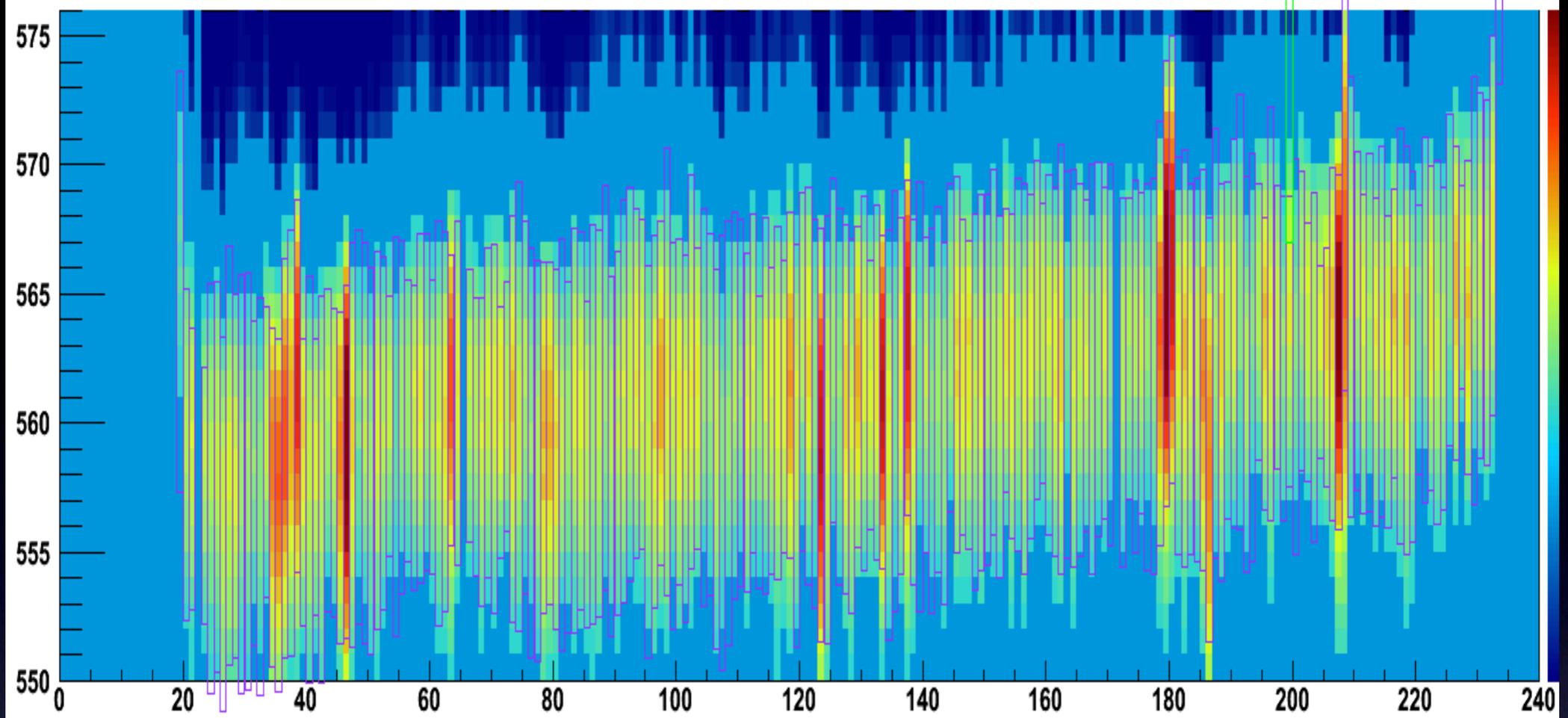
- The Kalman track fitting routine should take 3D hits as an input.
- It would be best if these hits were associated with 3D clusters. 3D clusters allow the Kalman filter various “seeds”, or suggested starting points, each associated with a different cluster.
- Any systematic error associated with matching 2D hits can be handled well with a Kalman filter.



3D hits (from Maddalena/Ornella)

RecPack code work

- RecPack was a pain to install on a public machine (flxi09). It took me a little while but I worked out all of the dependencies and such and have it up and running in my working area.
- The software is somewhat supported but suffers from lack of documentation.
- I don't think it will be too difficult to implement a RecPack-LArSoft interface.
- As a first step, I tried to reconstruct a 2D ArgoNeuT muon track with the RecPack Kalman filter (outside of LArSoft).



Conclusions

- The HoughLineFinder class is pretty much complete aside from parameter tuning (using MC). It can be used as a cluster finder and as a track fitter.
- The machinery is in place to do (wire,time) MC comparisons to the HoughLineFinder output. I am looking forward to the 3D track matching software in order to compare to MC in (x,y,z).
- I have been looking into some more advanced reconstruction techniques through the RecPack package. The software will be useful in reconstructing/fitting non-trivial tracks and vertices.
- My main goal in the near-term is to get the RecPack software interface with LArSoft going so other people can start using it.