



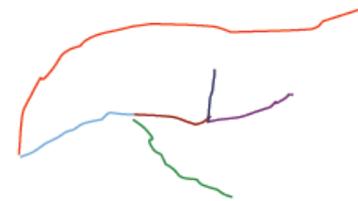
LArSoft 3D Seed Finding

Ben Jones

LArSoft Tracking Subgroup
Meeting



3D Tracking

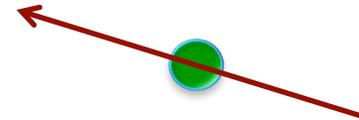


3D

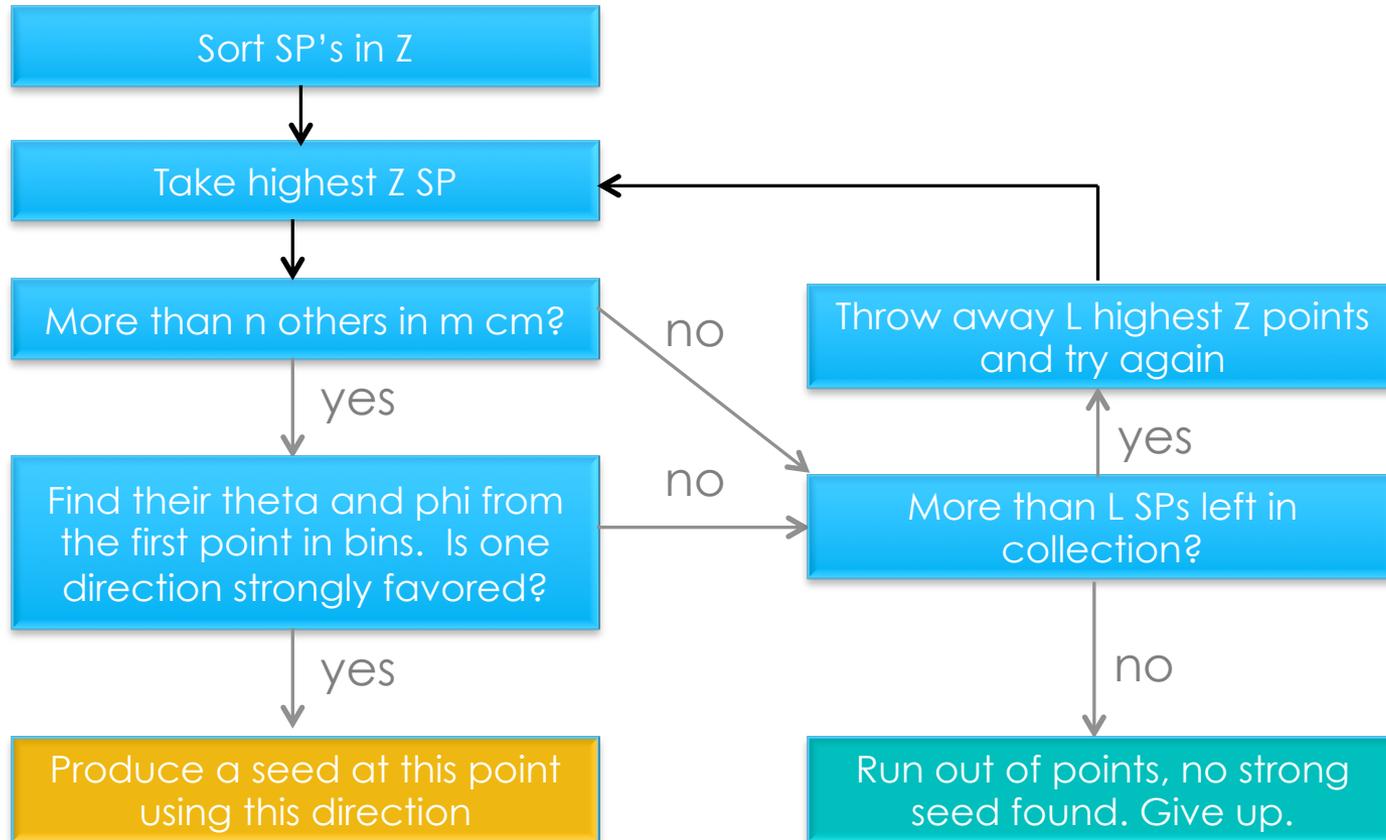
- A couple of LArSoft meetings ago I outlined a plan to develop a tracking algorithm based on finding 3D features in 2D hits by seeding and stepping
- Herb had been thinking along similar lines, and has since provided a detailed plan for the “stepping” part – generalized Kalman Filtering
- This is much more advanced than my proposal, and in my opinion, it is the best way to proceed at this juncture.
- The seeding tools already developed for my algorithms can, with some work, be used with Kalman steppers.
- I update on them in this talk.

Seed base class

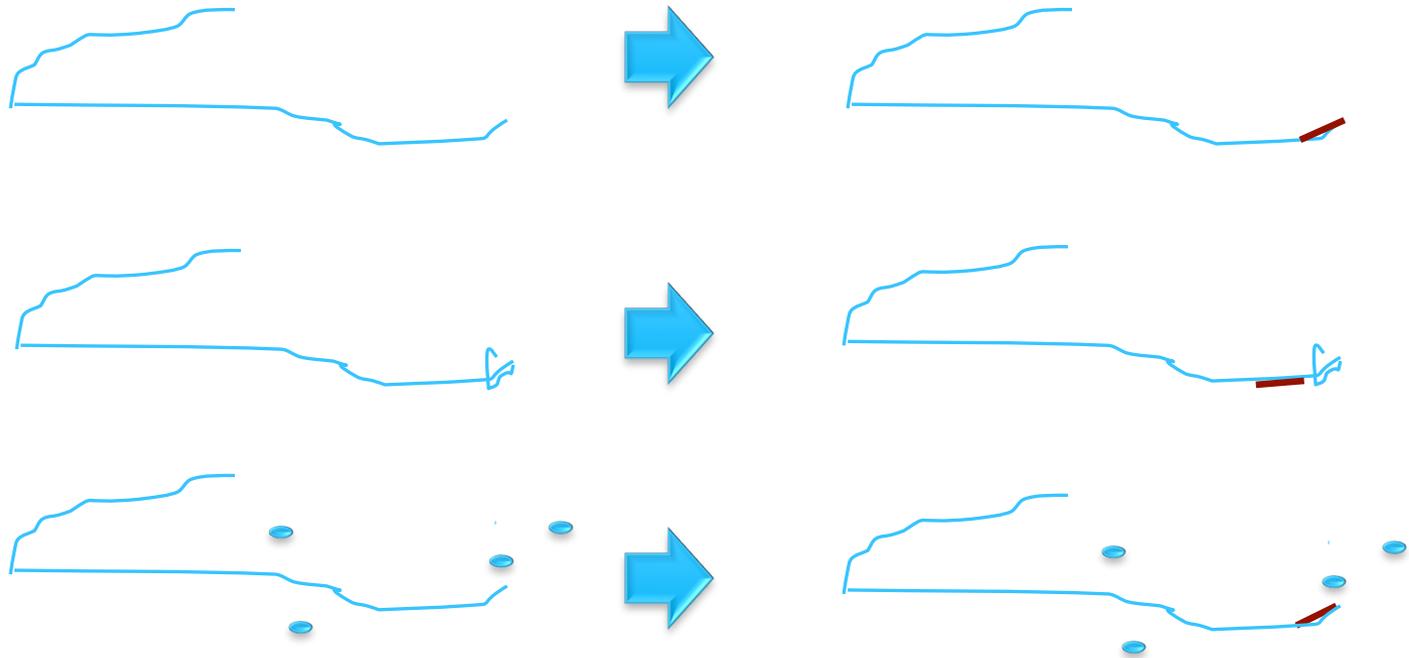
- Keep it general, and simple.
- Design decision made to not use existing objects. Seeds should not have any link to hits, and should not be confused with reconstructed objects.
- A seed is a short 3D segment, generated from averaged properties of several spacepoints.
- The seed base class was added to RecoBase, and it contains:
 - Position
 - Position Error
 - Direction
 - Direction Error
 - Validity flag



2. Produce Seeds

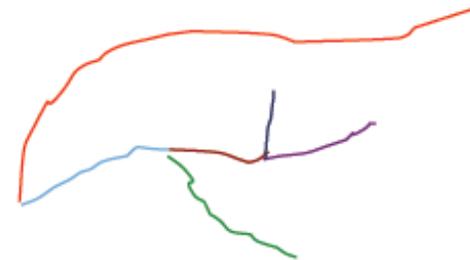


With these rules...



SeedFinderService

- SeedFinderService added to TrackFinder package.
- Current implementation accepts a `vector<vector<spacepoint> >`.
- Finds exactly one seed for each `vector<spacepoint>`.
- Exactly one because my original algorithm was going to step through all hits from this cluster combination, discard any it collected and try to generate a new seed from leftovers.
- Can this procedure work alongside kalman filtering? How to interface services in this case?



3D

SeedFinder

- Corresponding module added to TrackFinder, performs the following task:
 - Generate spacepoints from cluster combinations
 - Feed to SeedFinderService
 - Store seeds in event
 - (Optional) create track objects for evd

Changes to algorithm implementation since last talk:

- Seed elevated to a base class
- Seed finding code all tidied up and in repository
- Default fhicl files added
- Seeds now stored with both position and direction errors
- Angle of hit from seed point is calculated from center of mass of seed points, not just from highest Z.
- Option to produce corresponding tracks to view in evd (probably not ultimate solution)

Evd for Seeds

- Last time Brian suggested that instead of coding evd to see seeds, make tracks instead.
- However, track constructor seems to take both clusters and spacepoints.
- Giving the track just 2 spacepoints and empty cluster (corresponding to seed definition) doesn't seem to draw anything – does it only draw track hits and not track spacepoints on the 2d displays?
- Maybe I need to go write new evd code after all. Would like to avoid if possible, though.

Truth Seeds

- TruthSeeds module to generate seeds based on true track ends. Currently a different service, based on experience with truth / reco spacepoint confusion.
- Not ready to commit yet.
- Once TruthSeeds works, there will also be a module to compare true and reco seeds.

Timescale for Delivery

- I am on LArSoft about 20% at the moment. This should be 50%, I hope it will be soon.
- My estimated timescales:
 - Compiling + running first draft : done
 - Truth seed finding, validated and running : 2 weeks
 - Reliable seed finding from spacepoints with roughly the right parameters : 1 month