BEZIER TRACKER V4

BEN JONES

Seed Finding and Bezier Tracking in LArSoft - Technical Manual

Ben Jones, MIT

November 14, 2013

Abstract

This note describes the Seed Finding and Bezier Tracking TPC reconstruction algorithms in LArSoft. I give a conceptual overview of seeds and bezier tracks, provide instructions for how to use these objects for reconstruction and analysis applications, and then give a detailed technical description of the algorithms and data products involved in the SeedFinding and BezierTracking process.

Contents

1 Introduction **2**

2 Conceptual Overview **2**
2.1 Seeds . . . 2
2.2 Bezier Tracks . . . 2

3 Seed Finding **5**
3.1 Using Seeds in LArSoft . . . 5
3.1.1 Producing seeds with SeedFinderModule . . . 5
3.1.2 Calling SeedFinderAlgorithm . . . 5
3.2 The recob::Seed Data Product . . . 6
3.3 Seeds in the Event Display . . . 8
3.4 Seed Finding Performance . . . 8
3.5 Technical Description of the SeedFinderAlgorithm . . . 8
3.5.1 Public Interfaces . . . 8
3.5.2 Parameters of the SeedFinderAlgorithm . . . 8
3.5.3 Cluster overlap checking . . . 8
3.5.4 Internal book keeping tables . . . 11
3.5.5 The Seed finding loop . . . 12
3.5.6 Determining seed centers and directions . . . 12
3.5.7 Consolidating and extending seeds . . . 13
3.5.8 Final attempt for unseeded combinations . . . 13

4 Bezier Tracking **13**
4.1 Using Bezier Tracks in LArSoft . . . 13
4.1.1 Producing Tracks with BezierTrackerModule . . . 13
4.1.2 Retrieving BezierTracks from the Event . . . 13
4.2 The trkf::BezierTrack Analysis Object . . . 14
4.2.1 Internal Structure and Organization of the Bezier Track . . . 14
4.2.2 Bezier Interpolation and the BezierCurveHelper . . . 15
4.3 Bezier Tracks in the Event Display . . . 15
4.4 Bezier Tracking Performance . . . 16
4.5 Technical Description of the BezierTrackerAlgorithm . . . 16
4.5.1 Parameters of the BezierTrackerAlgorithm . . . 16
4.5.2 Producing the organized hit collection . . . 17
4.5.3 Making Bezier Tracks . . . 17
4.5.4 Overlap Filtering and Track Joining . . . 18
4.5.5 Bezier Vertexing . . . 19
4.5.6 Bezier Calorimetry . . . 19

5 Conclusions **20**

STATUS OF SEEDING AND BEZIER TRACKING

Recently I showed updated seed finder. Improvements in seed finding in fact caused problems for bezier tracking in last MCC

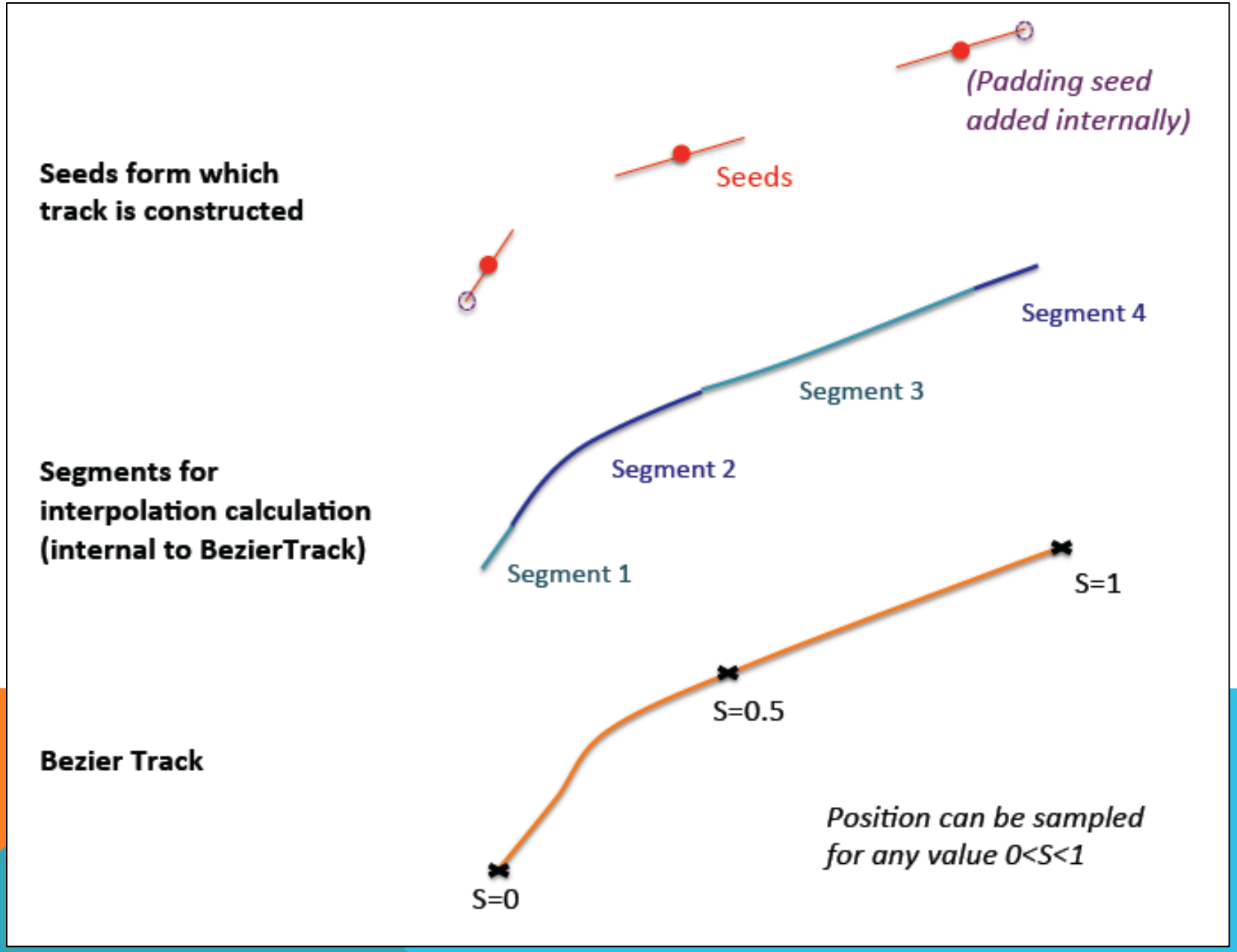
The new Bezier algorithm is much more robust against these effects.

As with the seed finding, the major improvements have involved moving down to hit level information wherever possible.

In terms of functionality, this may be the last “major iteration” of BezierTracker.

There remains room for improving the speed of the algorithm, if we like the way the results look after MCC2.N

WHAT DO WE NEED FOR A BEZIER TRACK?

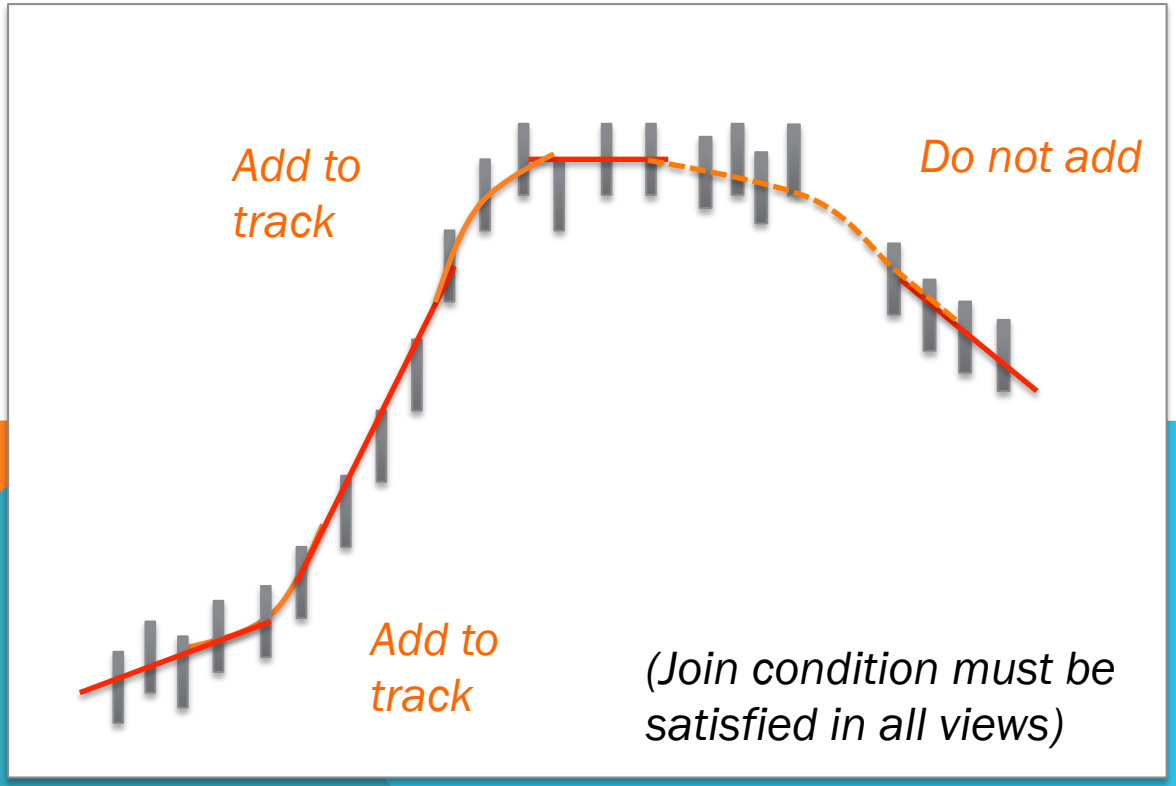


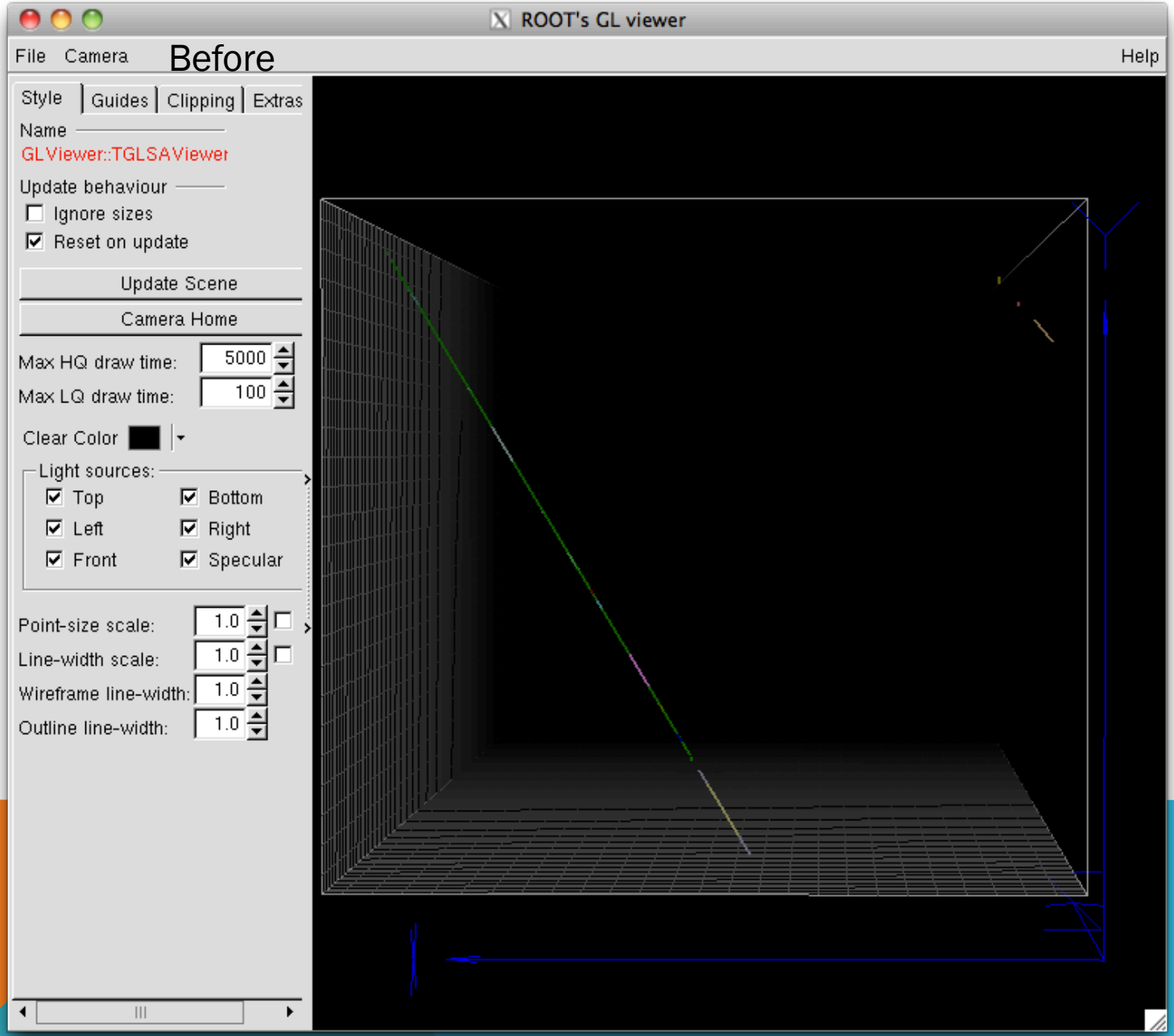
WHAT IS NEW?

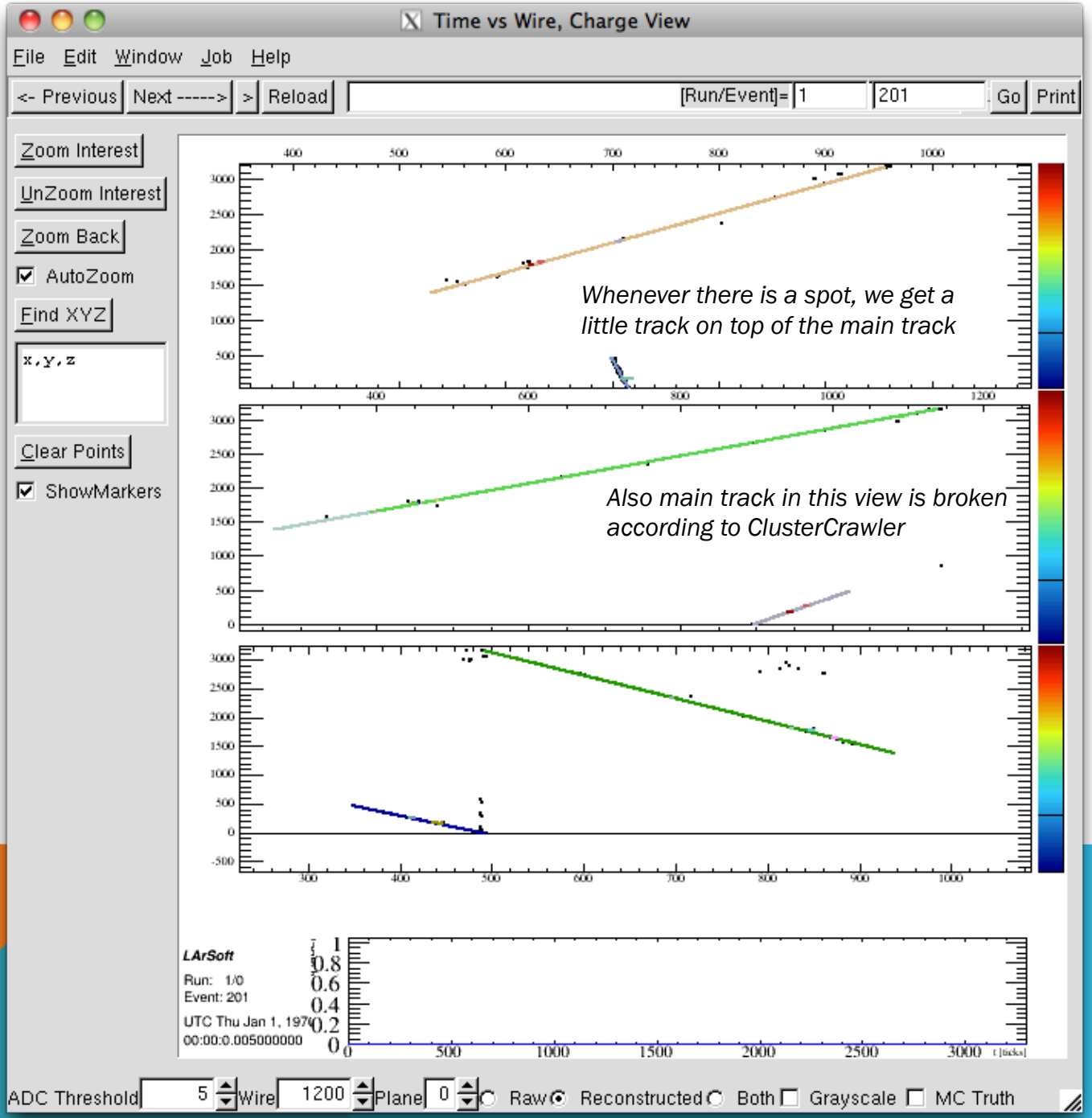
To first order, the main change is that extrapolations from seed to seed now proceed via comparisons to hits in each view

This means we can robustly go longer distances without leaving the track activity.

Booking keeping to do this quickly is the hard part.

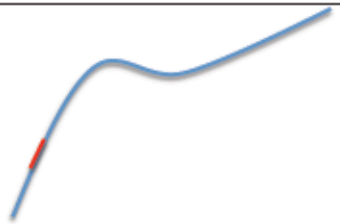
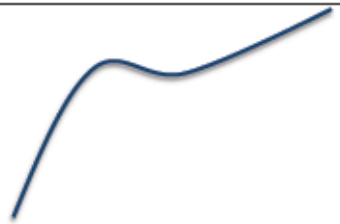
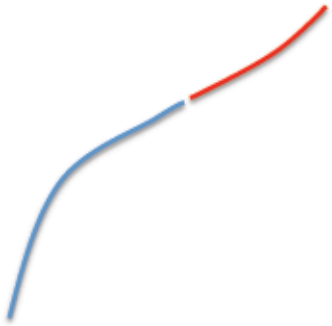



ROOT's GL viewer
File
Camera
Before
Help
Style
Guides
Clipping
Extras
Name
GLViewer::TGLSAViewer
Update behaviour
Ignore sizes
Reset on update
Update Scene
Camera Home
Max HQ draw time:
5000
Max LQ draw time:
100
Clear Color
Light sources:
Top
Bottom
Left
Right
Front
Specular
Point-size scale:
1.0
Line-width scale:
1.0
Wireframe line-width:
1.0
Outline line-width:
1.0


Time vs Wire, Charge View
File Edit Window Job Help
<- Previous
Next ----->
>
Reload
[Run/Event]=
1
201
Go
Print
Zoom Interest
UnZoom Interest
Zoom Back
AutoZoom
Find XYZ
x,y,z
Clear Points
ShowMarkers
Whenever there is a spot, we get a little track on top of the main track
Also main track in this view is broken according to ClusterCrawler
LArSoft
Run: 1/0
Event: 201
UTC Thu Jan 1, 1970
00:00:0.005000000
ADC Threshold
5
Wire
1200
Plane
0
Raw
Reconstructed
Both
Grayscale
MC Truth

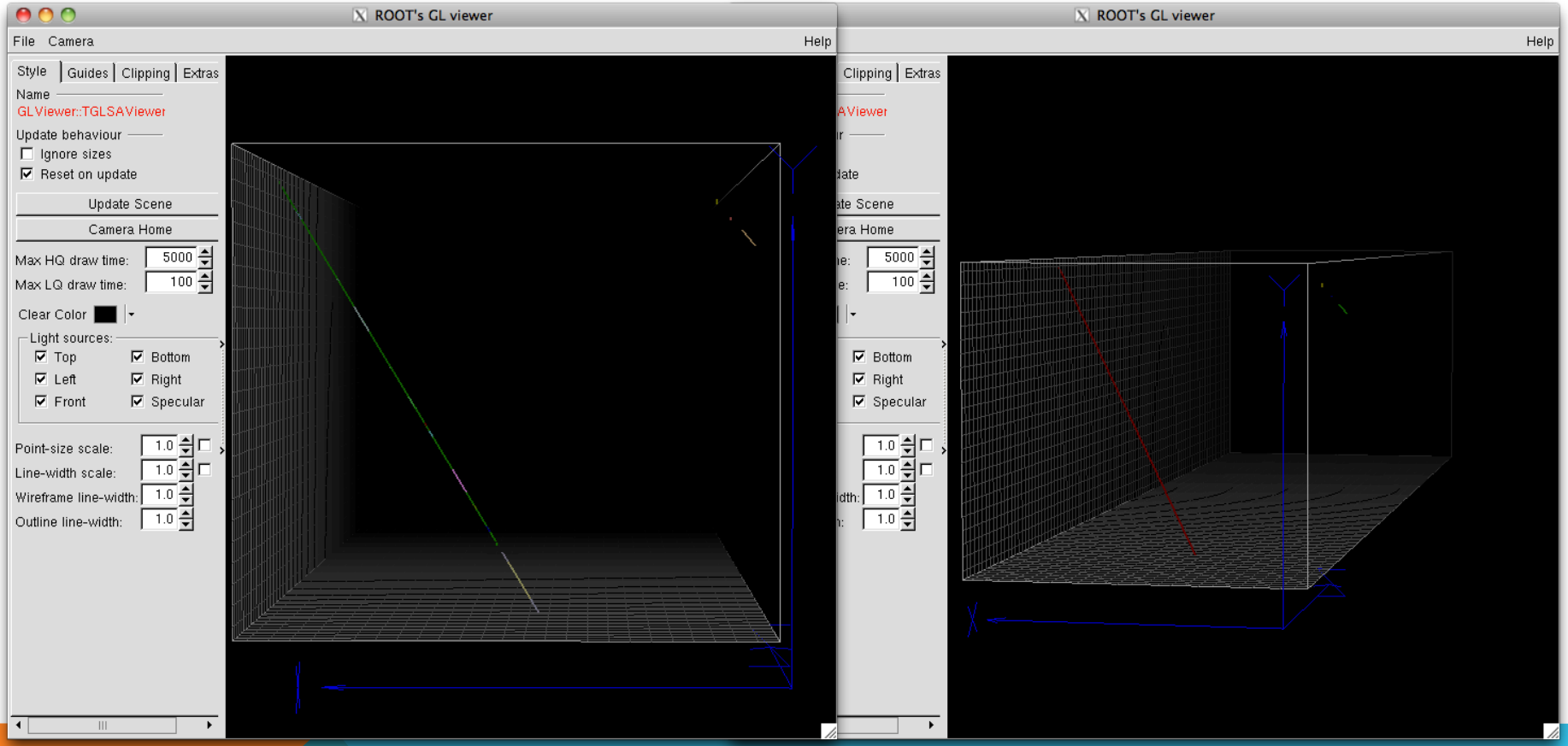
NEW JOINING METHODS

Improvements in accuracy of tracking from the hit information allow us to use much more precise filtering / joining methods

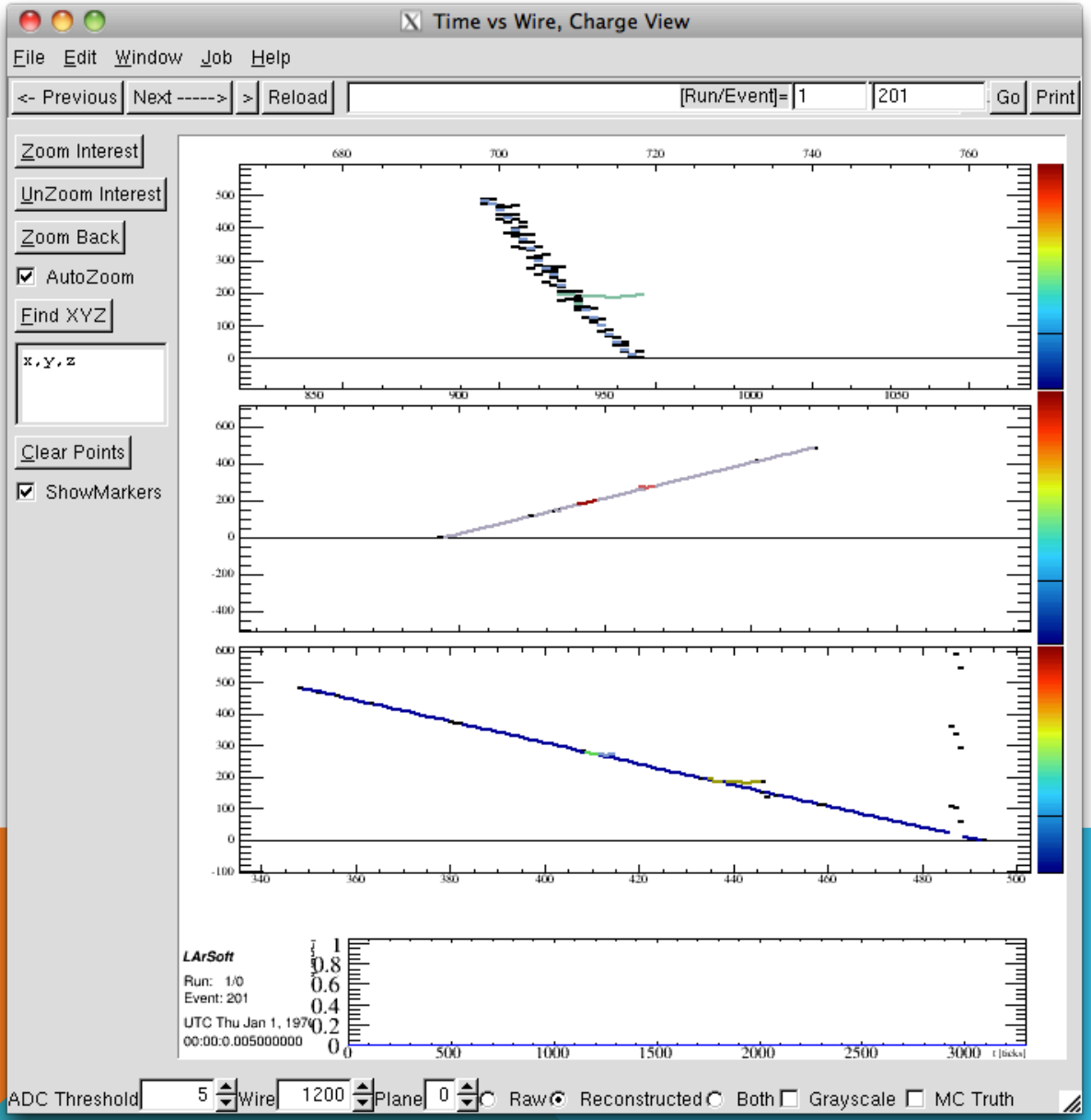
Method	Before	After
FilterOverlapTracks		
MakeOverlapJoins		
MakeDirectJoins		

Before

After



The one that got away:



(with CCCluster)

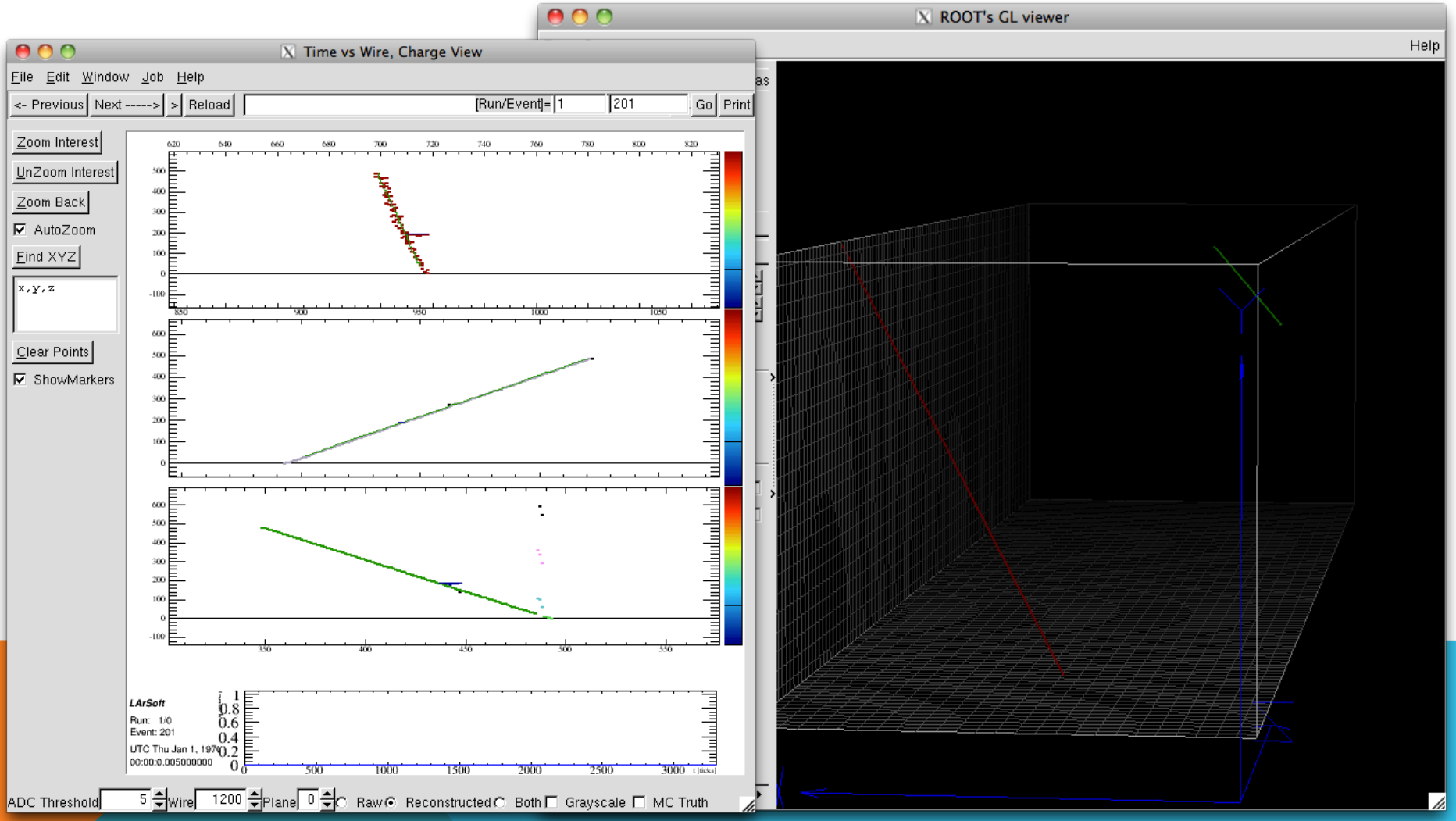
Multiple clusters not a problem

Rather the black spots in this blue line…

DBCluster is more robust against this. But in that case multiple “fake” hits on wide tracks make tracking very difficult.

Choices, choices!!

With FuzzyCluster:



Presently shortest section of the technote…

4.4 Bezier Tracking Performance

This section will be filled in following MCC 2.2

MCC Request:

Run BezierTracker twice, using both FuzzyCluster and CCCluster as inputs, for comparison.

Note:

This version of bezier tracker is slower, but still comparable to other reco algs (100s/ cosmic overlay evt), if we like how the output looks, I will endeavour to optimize for next MCC.