Kalman Filter Overhaul

C Hunt
K Long, J Pasternak

22nd June 2015. CM42

The Kalman Filter

What is it?

An algorithm from signal processing theory, designed to deduce a system's current state ($\mathbf{x}$) from continuous or discrete external measurements.

If we consider a helical (straight) track as our system, the state is 5 (4) component phase space vector and the measurements correspond to the tracker planes
Our "signal" is now a continuous function of z, rather than the conventional time. So we can use a Kalman Filter!

The Basics

To fully define the system we need two things:

1. **The Propagator** uniquely defines the linear transition from the current position to some other arbitrary position.
   A Straight Track or a Helical Track!

2. **The Measurement** uniquely defines the linear transformation involved in making a single measurement of the system state.
   The fiber channels!

$$\mathbf{x}_k = \mathbf{F}_k \mathbf{x}_{k-1}$$
$$\mathbf{m}_k = \mathbf{M}_k \mathbf{x}_k$$

where $\mathbf{F}$ is the propagator matrix and $\mathbf{M}$ is the measurement matrix.

The Basics

With a deterministic linear propagator and a deterministic linear measurement, the Kalman Filter is statistically identicle to a linear least squares fit without knowledge of measurement errors.

But we can do better!

We may also include meaurement and process noises such the fit is now *the* optimal linear fitter taking into account all correlations and errors.

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \epsilon_k$$
$$\mathbf{m}_k = \mathbf{M}_k\mathbf{x}_k + \omega_k$$

where ϵ is the processs noise and ω is the measurement noise.

Kalman in MICE

Propagator

A helix derived from the the position and momentum components with deterministic energy loss, or a straight line propagator derived from the position and gradient components.

The process noise is derived from the Highland Formula for Multiple Coulomb Scattering.

Measurement

Reconstruct α the distance of the SciFi Cluster from the center of the plane, perpendicular to the direction of the fibers.

The measurement noise is the RMS of a uniform top hat signal with a width equal to the fiber channel width - $w/\sqrt{12}$

The Original Implementation

(Appologies for the lack of plots.)

Attempts to fully characterise the fit showed up some fundamental issues:

- If we turned off the process noise, Kalman did not agree with Pattern Recognition
- If we stopped the Geant4 Multiple Couloumb Scattering and Energy loss Kalman was not perfect
- If we gave Kalman a perfect seed it should not really have changed
- It was not extensible for future upgrades (field stepping, other implementations, etc).

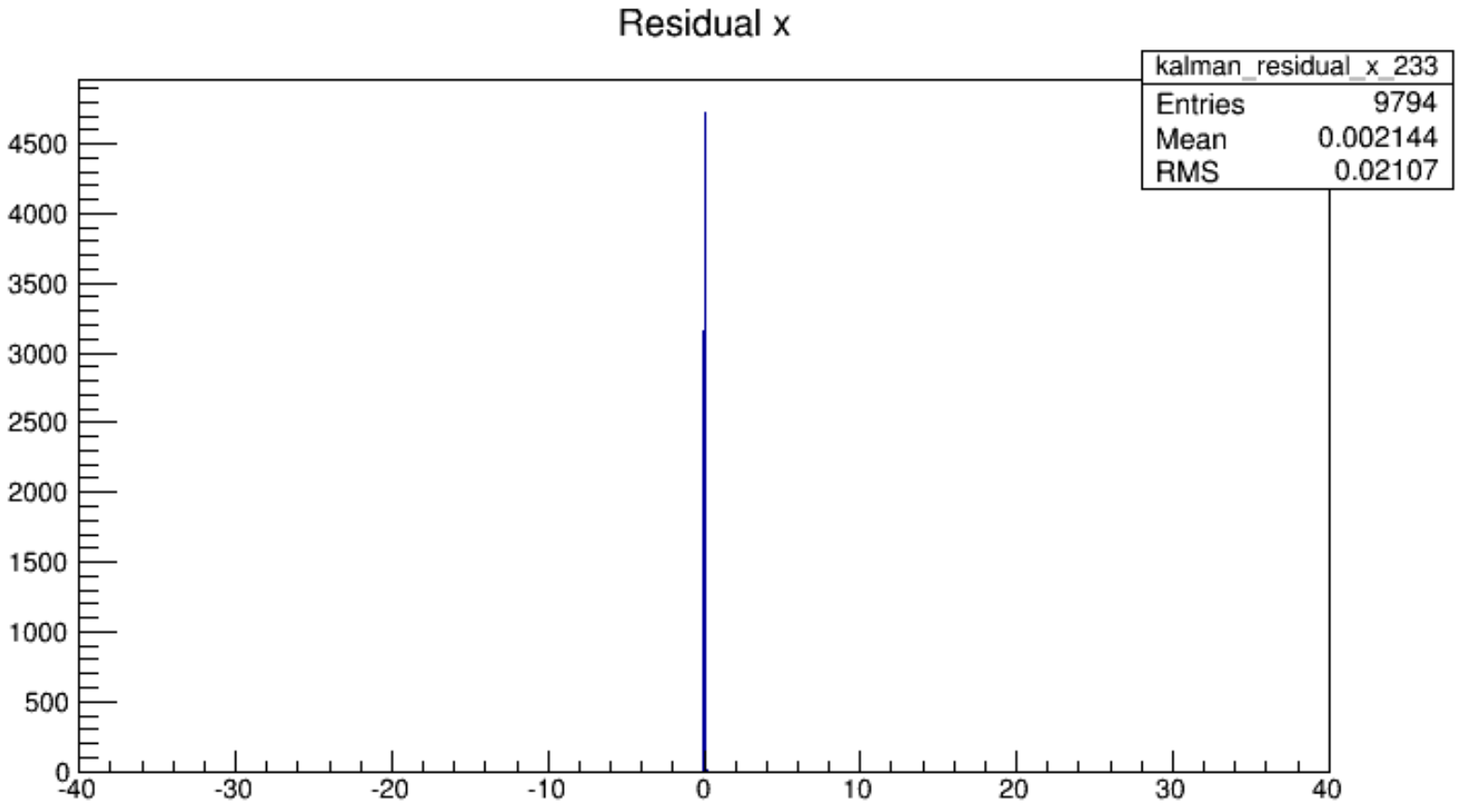
The fit did however work. So we need to be careful as the process noise gives the fit a lot of flexibility to hide some fundamental issues!

What We Did

1. Kept the complex algorithms - MCS, Eloss, coordinate transformations.
2. Restructured the core of the code - Kalman is now a standalone abstract algorithm that can be used by anyone!
3. Wrote the code to convert Clusters $\rightarrow$ State Vectors $\rightarrow$ SciFi Tracks.
4. Wrote specific Straight Track and Helical Track propagators, and the SciFi Measurement Class.
5. Wrote a Spacepoint trackfit to test the algorithm.
6. Fixed bugs and added functionality to the SciFi Recon and SciFi Geometry classes.

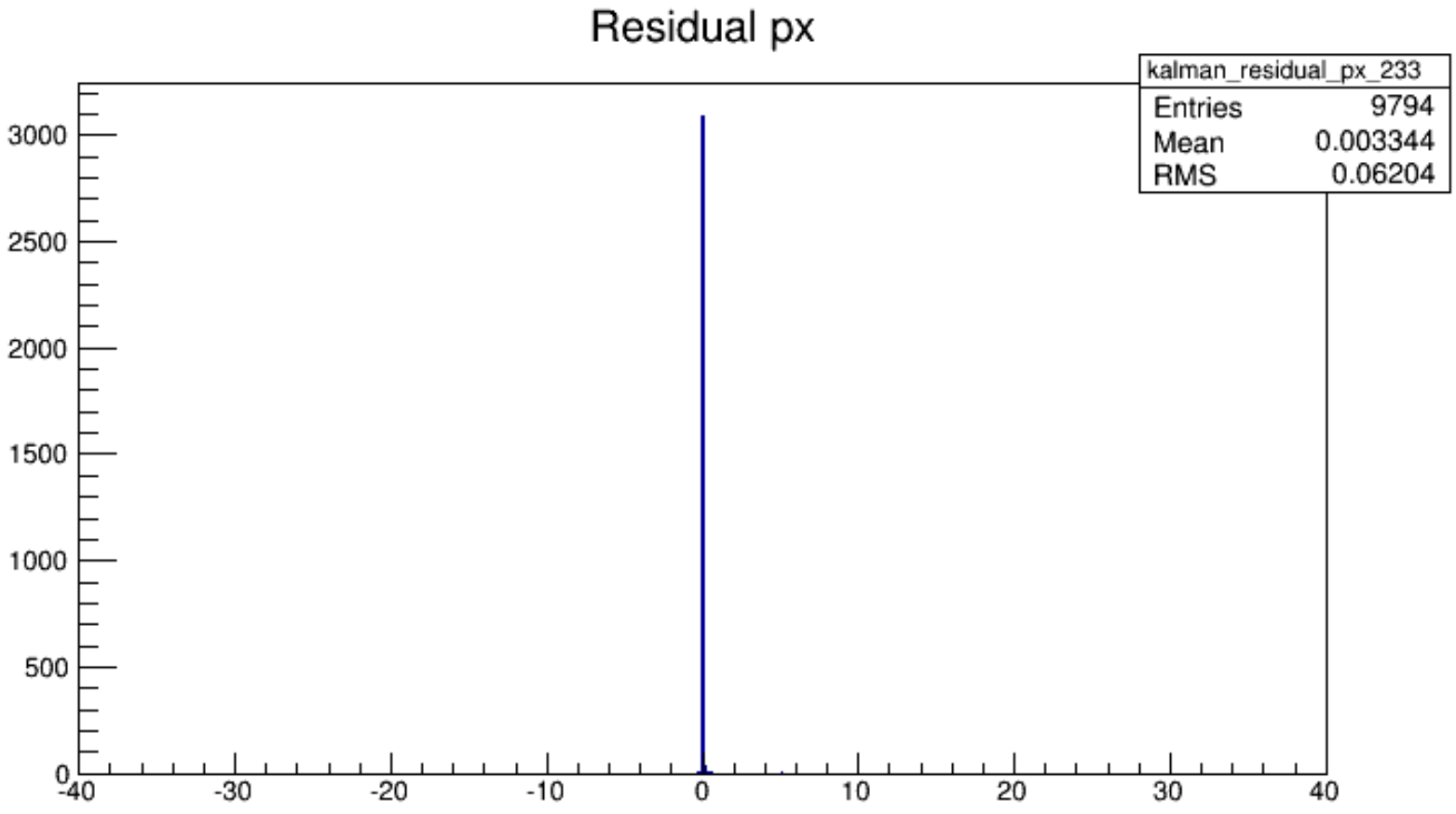
. . . and of course - extensive testing!

Perfect Kalman



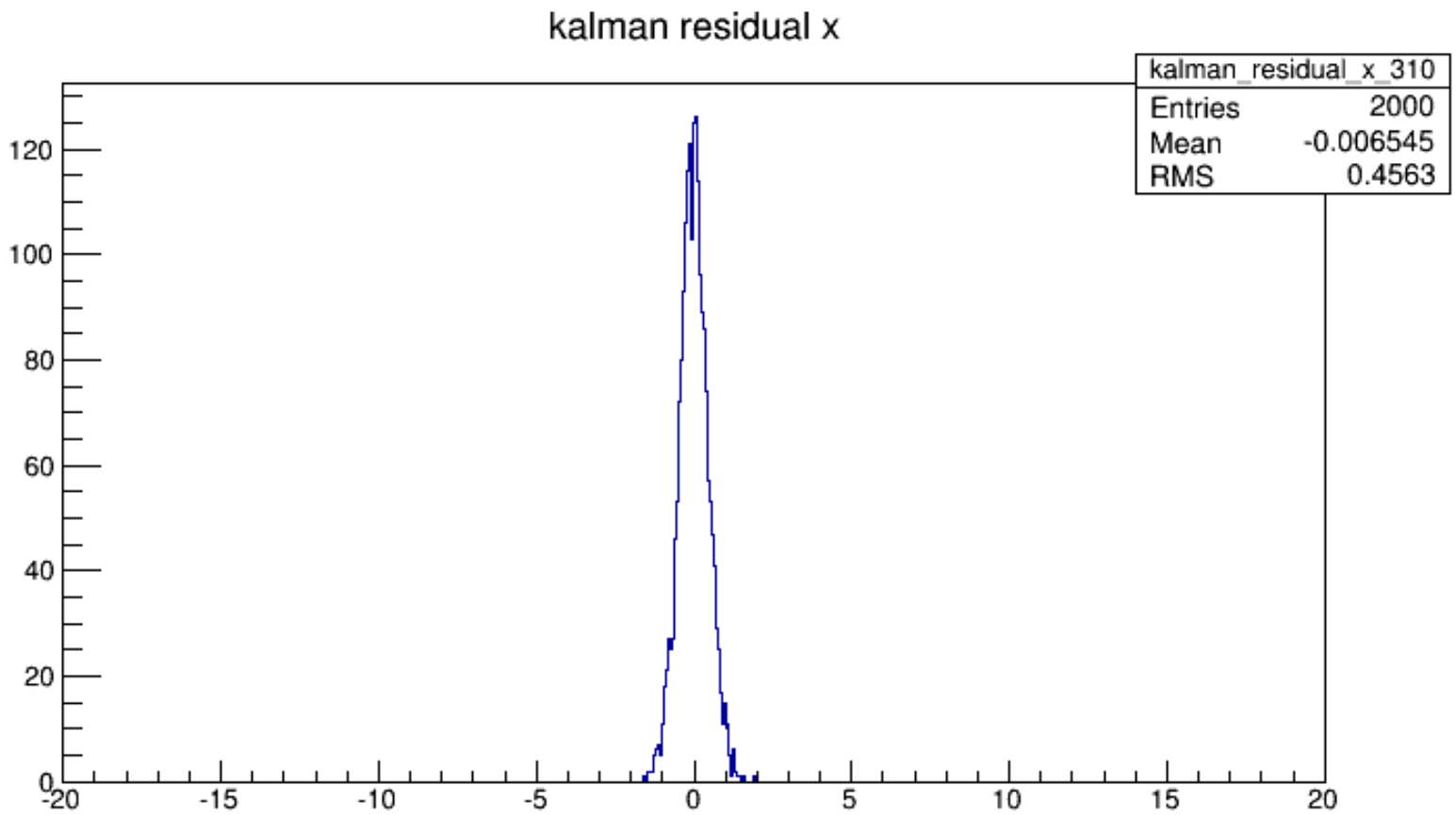
Perfect tracks in a perfect tracker - x Residuals [mm]

Perfect Kalman



Perfect tracks in a perfect tracker - p_x Residuals [MeV/c]

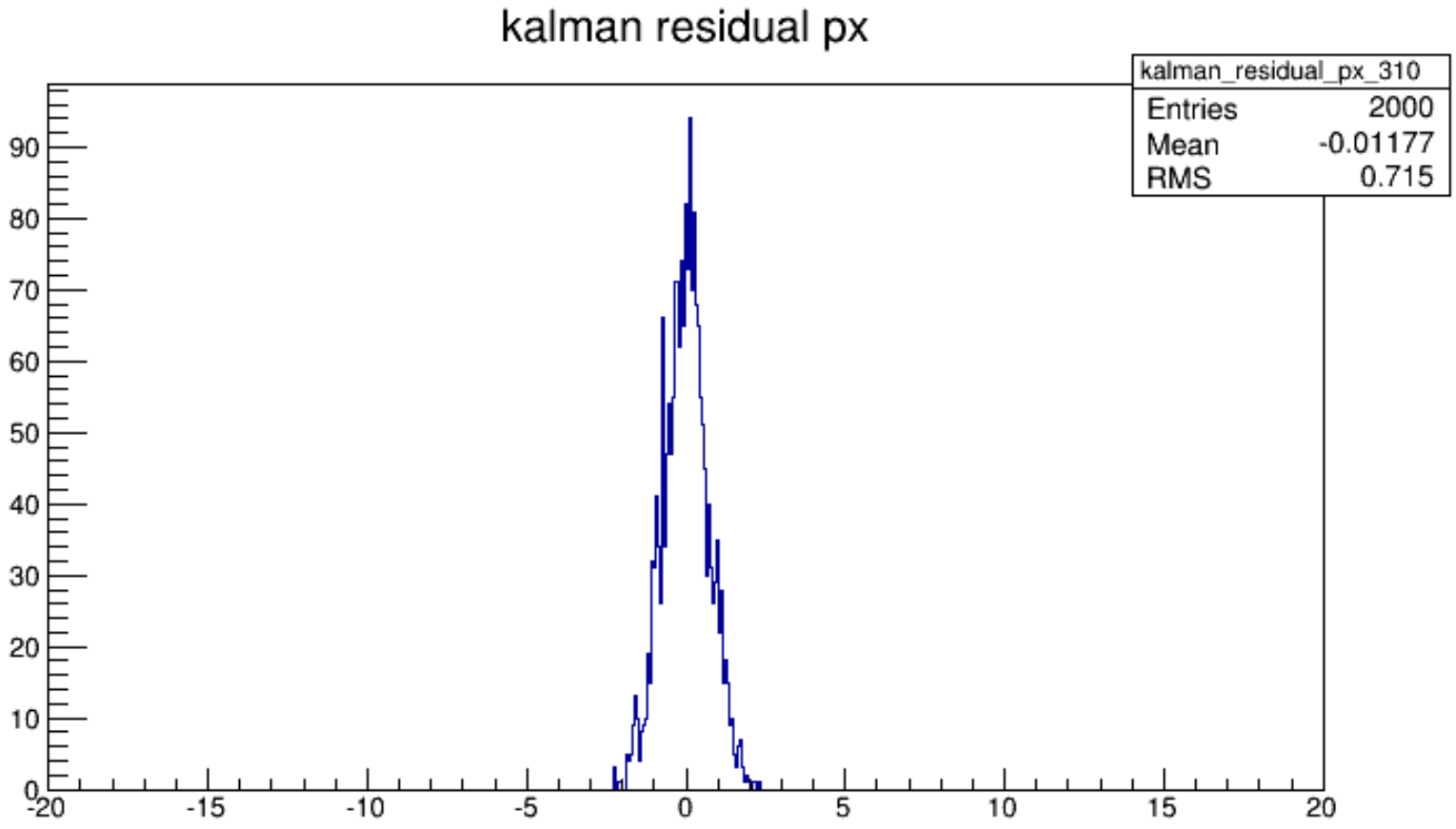
Half-Way



Full Tracker, No Energy Loss in Recon, No MCS in Recon, Geant4 with Mean Energy Loss only.

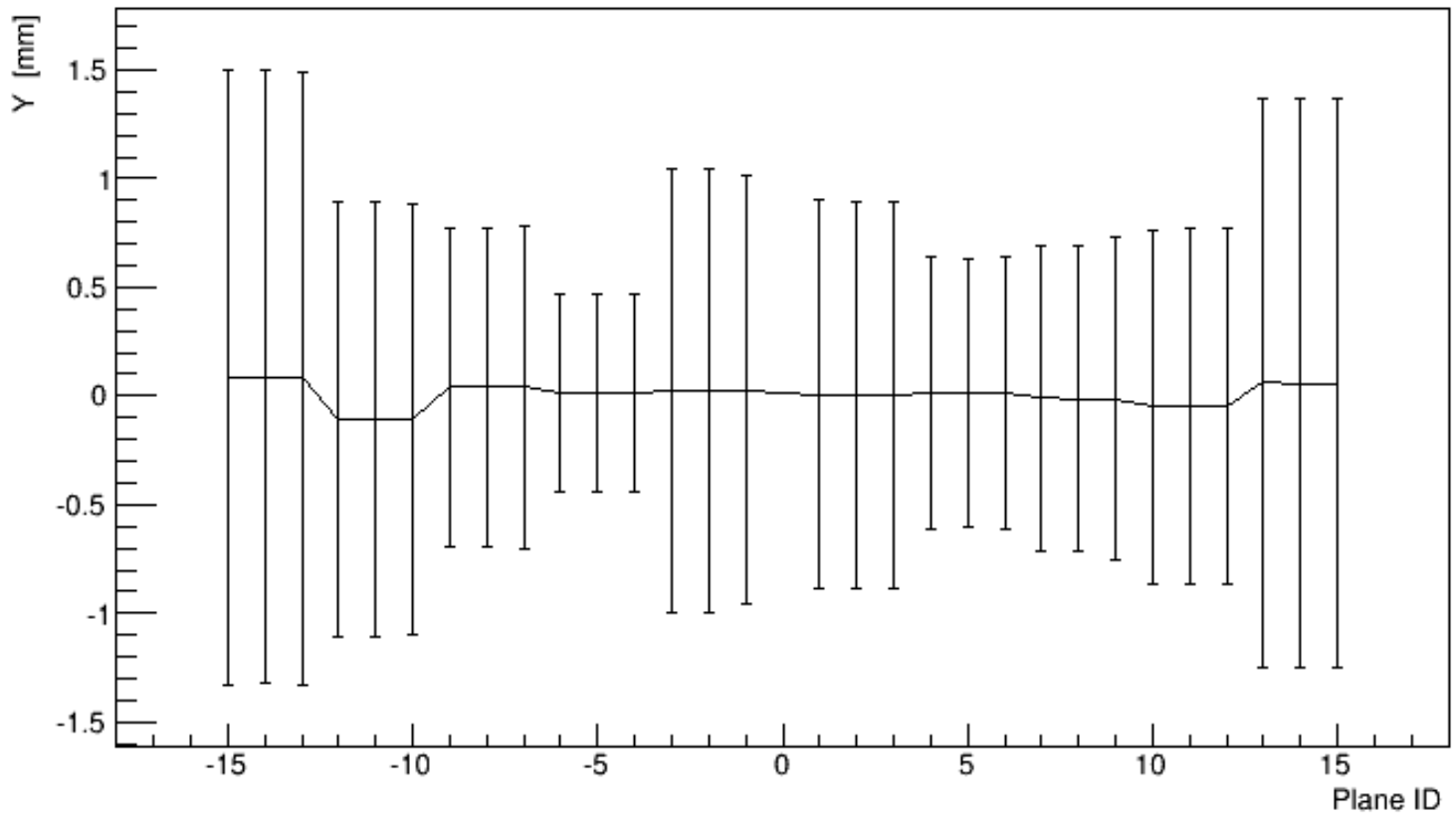
x Residual [mm]

Half-Way



Full Tracker, No Energy Loss in Recon, No MCS in Recon, Geant4 with Mean Energy Loss only.
p_x Residual [MeV/c]

Full Implementation



Full Tracker Sim, Eloss and MCS included in Fit. Full Geant4 Sim.
y Resolution [mm]

What's Left?

Still a few loose ends to tie up for the helix fit:

- E-Loss is slightly wrong - needs fixing.
- Charge determination still has a small bug
- Statistics need verifing properly - Chi-Squared Values are a little *too* good!
- Effect of non-uniform fields on fit - *In Progess!*
- Investigate other propagtors: Field stepper, 2nd order filtering, different models...

Current code is awaiting the next release. Bug fixes to follow in the comming week(s).

What's Next?

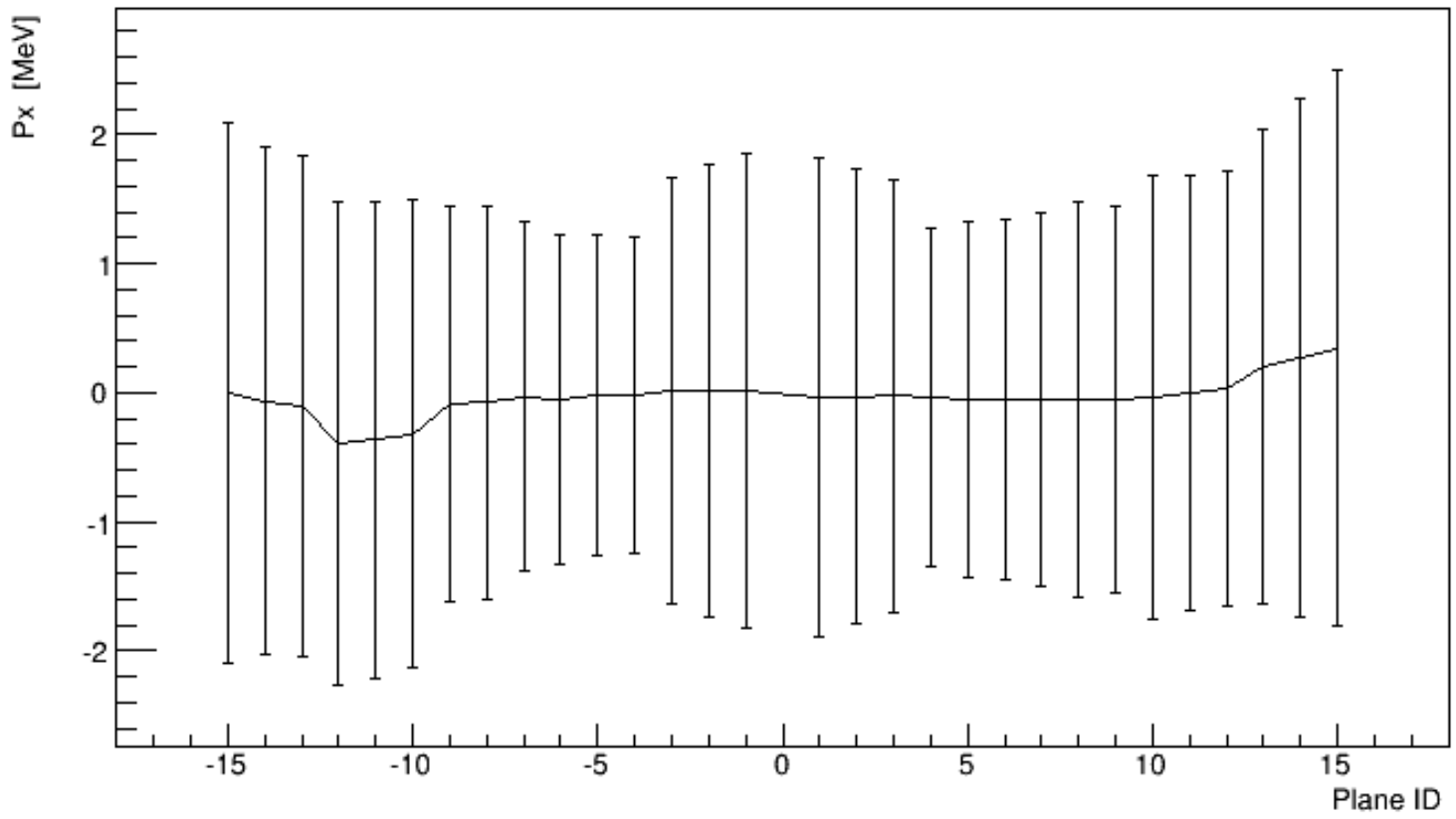
Critical to MICE Comissioning, the straight track fit is nearly perfect!

Currently being used in a quick tracker alignment study.

Watch This Space!

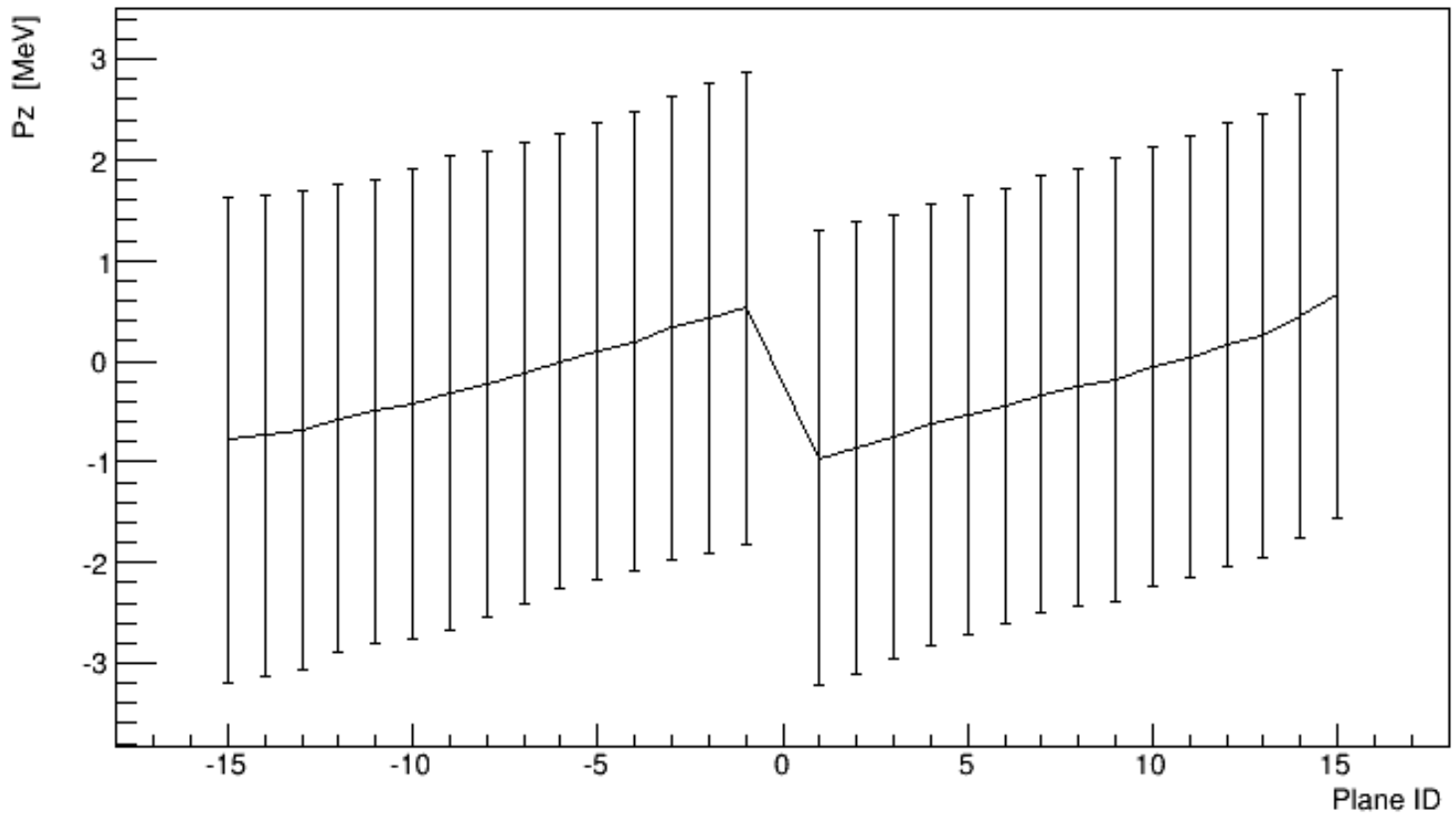
This slide intentionally left blank.

Full Implementation



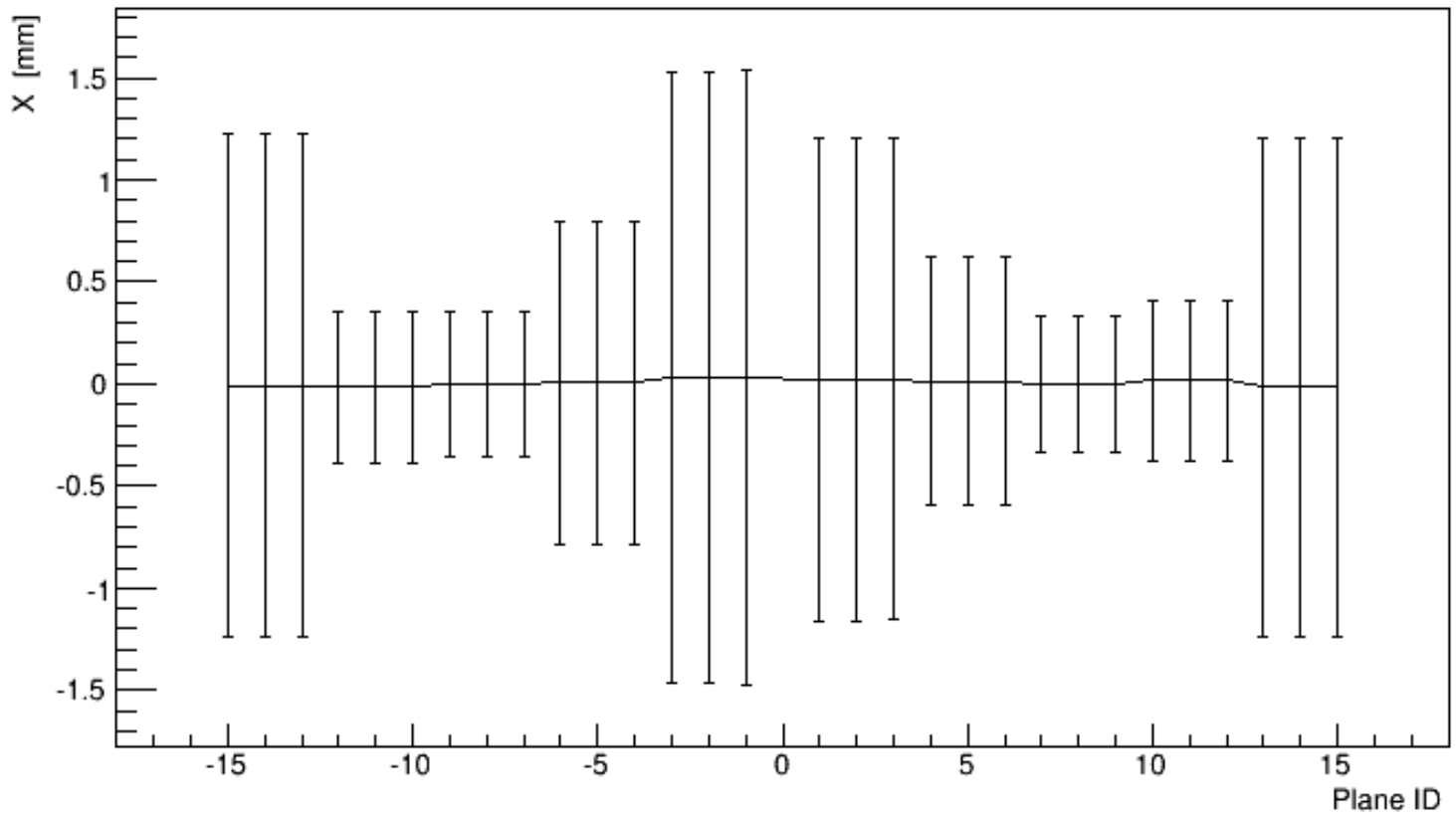
Full Tracker Sim, Eloss and MCS included in Fit. Full Geant4 Sim.
p_x Resolution [MeV/c]

Full Implementation



Full Tracker Sim, Eloss and MCS included in Fit. Full Geant4 Sim.
p_z Resolution [MeV/c]

Full Implementation (Straights)



Full Tracker Sim, Eloss and MCS included in Fit. Full Geant4 Sim.

x Resolution [mm]