The Alternative Demultiplexer

AlgAltDemux

Mark Thomson

University of Cambridge

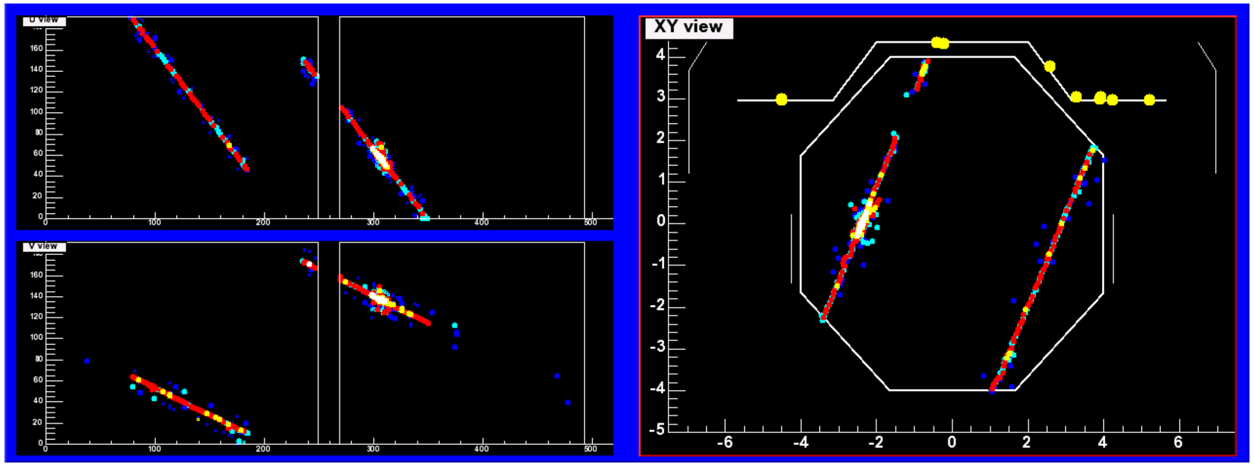
Outline:

❶ Introduction/Motivation

❷ Method

❸ Performance

❹ Conclusions/Future Work



❶ Introduction/Motivation/Initial Goals

Motivation:

★ Reconstruction in MINOS is non-trivial !

★ Quality of DeMuxing/Tracking $\Rightarrow$ impact physics sensitivity

particularly important for Atmospheric ν analysis

★ AltDeMux developed primarily for the Atmospheric Neutrino analysis

★ Aim for fast reconstruction (FAST means $<$ DigitListModule)

★ Provide an alternative to the Standard Package (DeMux)

allows possibility of rejecting DeMux failures by ANDing

★ Try to keep Algorithm ‘independent’ of Standard package

adopt a significantly different approach

❷ Method

Outline of Method:

① Identify 'GOLDEN' Demuxed hits on a plane-by-plane basis.
GOLDEN hits are 'unambiguous' EAST-WEST strip associations
used as seeds for the DeMuxing

② GOLDEN hits MAY BE used for Event Identification
used to steer demuxing stategy (not yet fully implemented)

③ The GOLDEN hits define TARGET regions for remaining planes.

④ Use target regions to search for further unambiguous GOLDEN hits

⑤ GOLDEN hits used to define new TARGET locations

⑥ place remaining hits close to nearest TARGET

⑦ Validate solution depending on Event Type

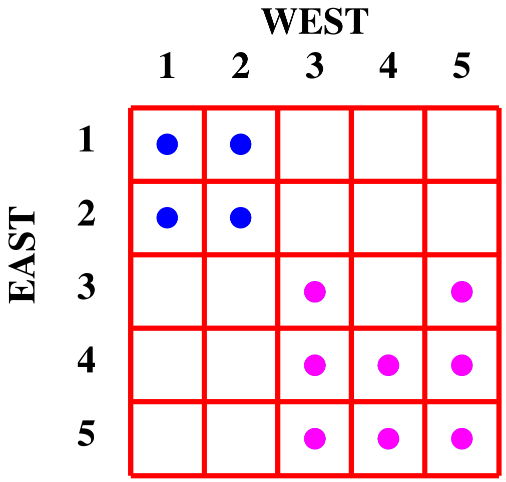
★ Now includes fairly sophisticated Crosstalk removal/tagging

Algorithm developed using Monte Carlo (Cosmics, Atmospherics neutrinos)

Method : Definition of Golden Hits

☆ **Iterate over planes**

☆ **Form matrix of all allowed East-West associations**



e.g. **5 East hits, 5 West Hits**

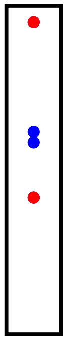
- **Group into unconnected groups.**
- **Look for unambiguous associations**

<u>Unambiguous Associations</u>

e.g. $2 \times 2 \Rightarrow$

2 Possible associations:

$\{$ **(E1,W1), (E2,W2)** $\}$ **or** $\{$ **(E1,W2), (E2,W1)** $\}$



Hits DeMuxed as *GOLDEN* if one, <u>and only one</u>, combination places hits ‘close’ together

A Few Comments:

① This works very well for planes with few combinations, *e.g.* $\mathbf{2 \times 2}$, $\mathbf{3 \times 3}$

② Planes with high hit multiplicities often remain ambiguous:
large showers
steep tracks

③ Therefore, try to reduce combinatorics

④ Breaking into unconnected groups helps,
$\mathbf{5 \times 5 \longrightarrow (3 \times 3)}$ + $\mathbf{(2 \times 2)}$

Clarify situation using:

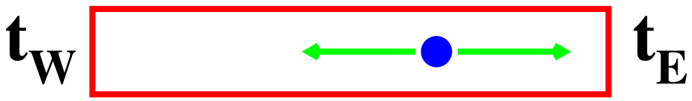
★ Timing information (recent IMPROVEMENTS here)

★ + Charge information

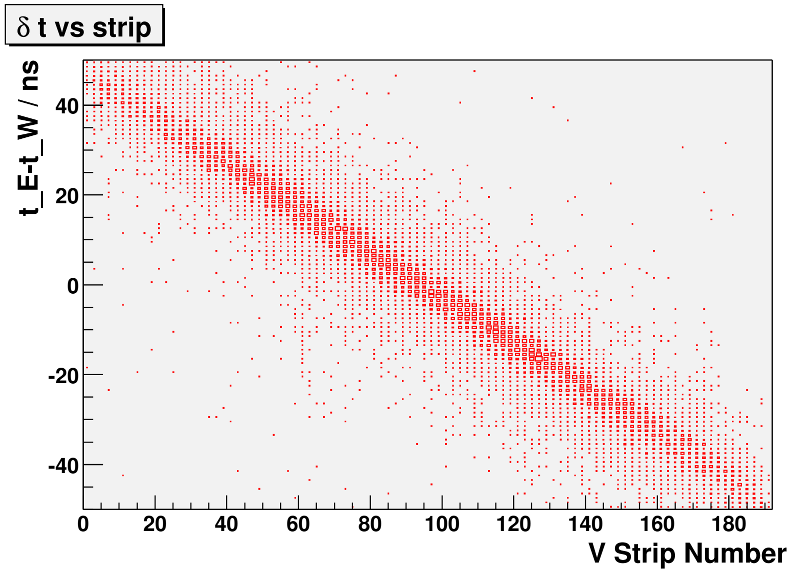
e.g. Timing - "3D DeMuxing"

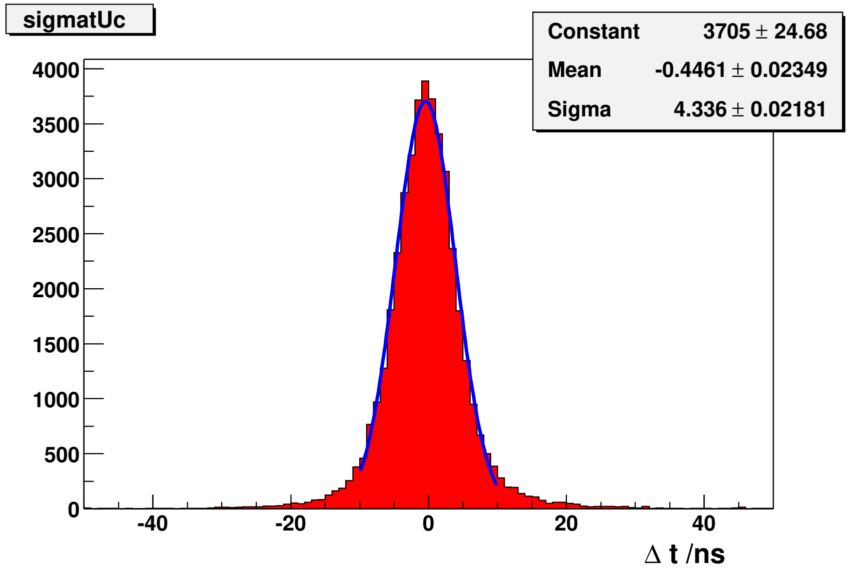
Use *U* view times to localize event in *V*

e.g. U Strip



$$t_W - t_E \rightarrow V$$





$\sigma_{\Delta t}$ = 4.3 ns (*i.e.* σ_t = 3.0 ns)

$$V_{strip} = 96 + \frac{\Delta t(ns)}{0.505}$$

$\sigma_{V_{strip}}$ = 8.6

σ_V = 36 cm

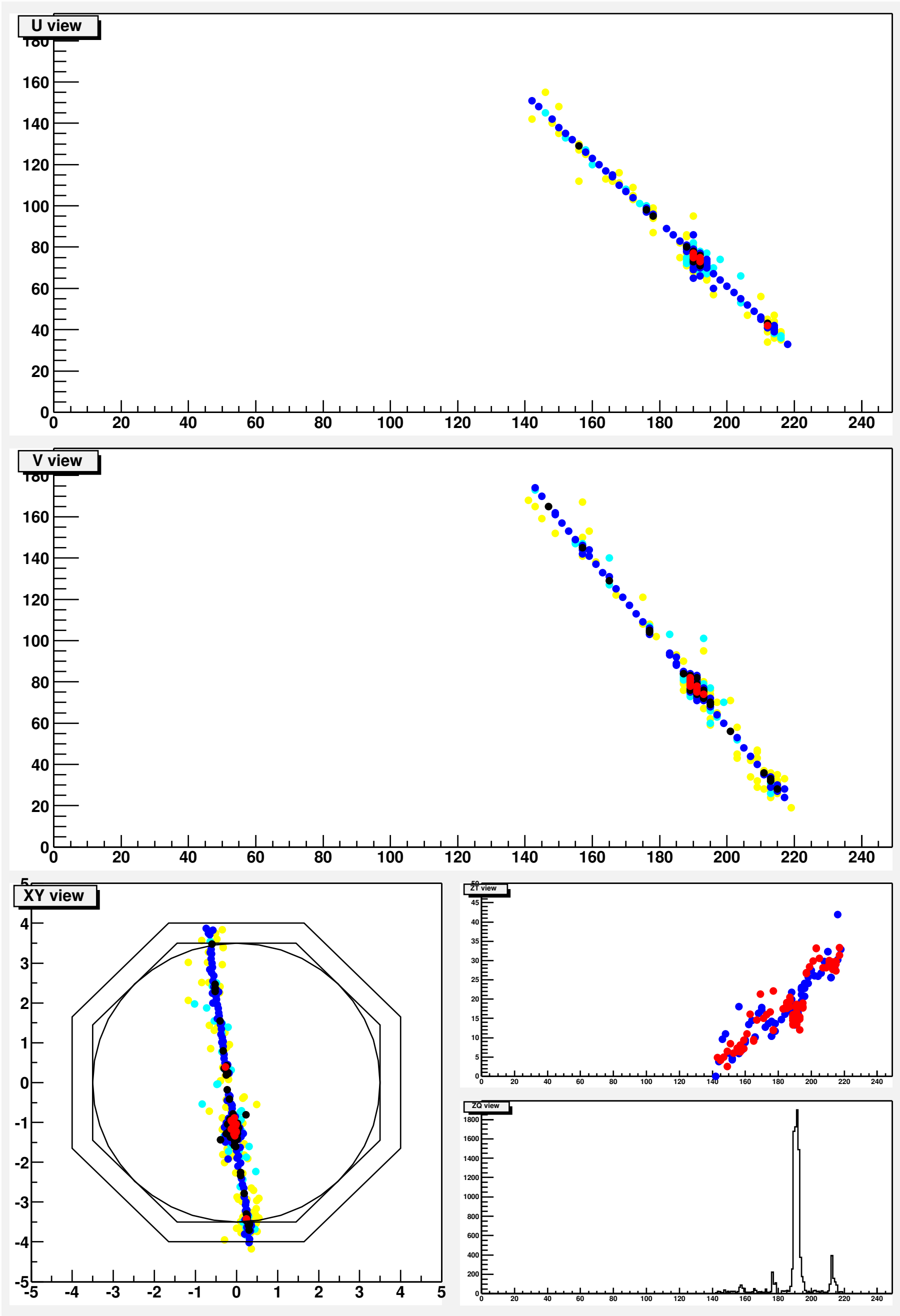
actually somewhat better now $\sim$ 30 cm

Timing - Example Event

★ A fairly complex event - large Bremsstrahlung

Event Display (SM1 only):

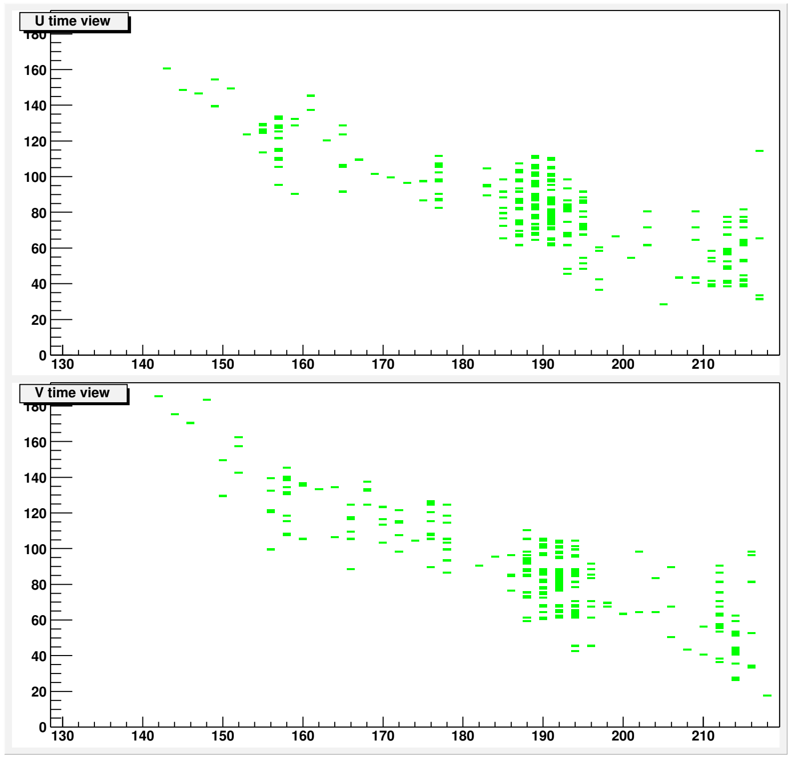
- ★ U-Plane View
- ★ V-Plane View
- ★ x-y View (no reconstruction)
- ★ time vs plane number
- ★ charge vs plane number



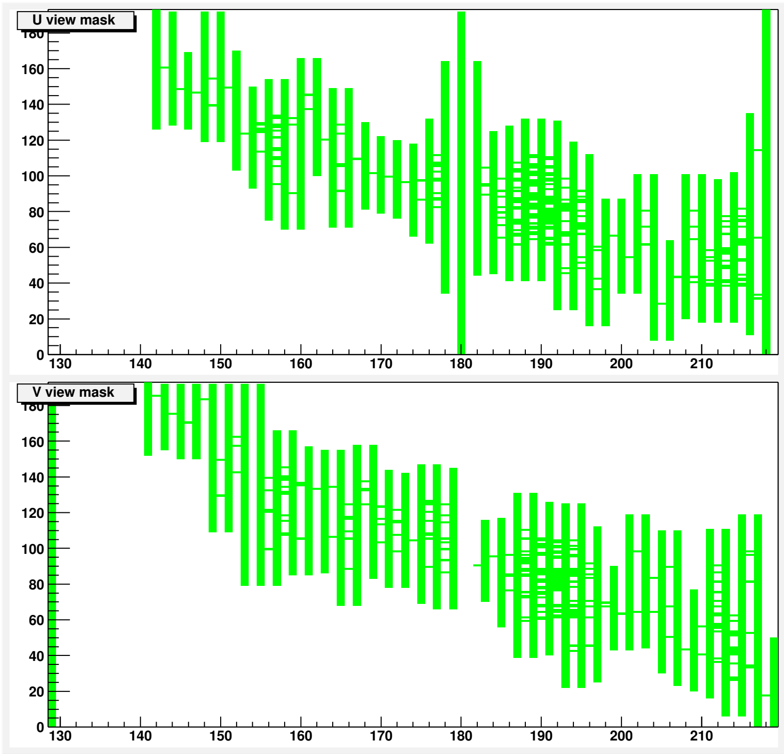
★ **Define mask of "acceptable strips" using timing info in other view.**

- ★ **Many planes have > 1 East hit and > 1 West hit**
- ★ **Multiple combinations ➡ U/V positions are not determined unambiguously by timing - <u>different</u> fibre lengths.**

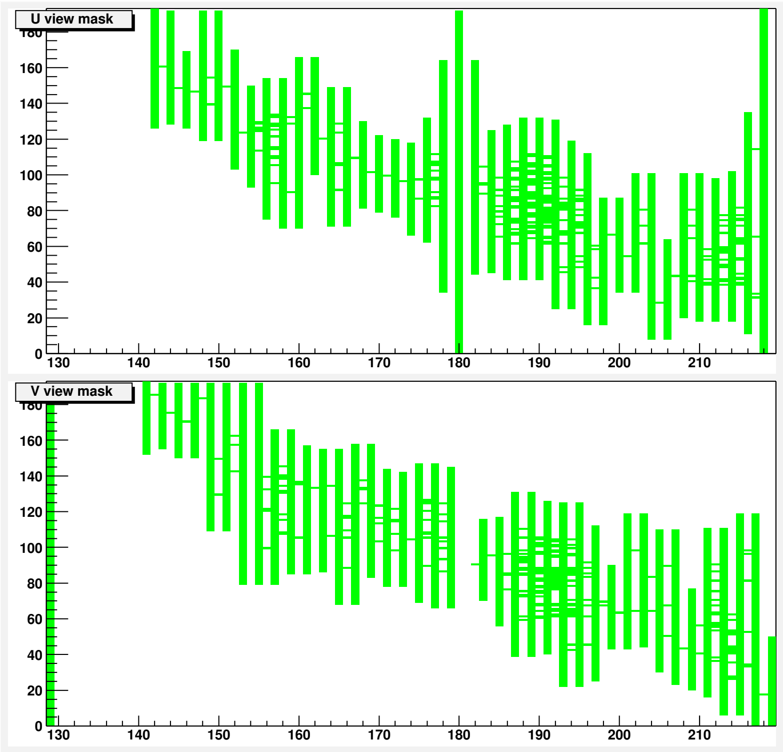
★ **BUT difference in fibre lengths 'small' compared to 16 m !**

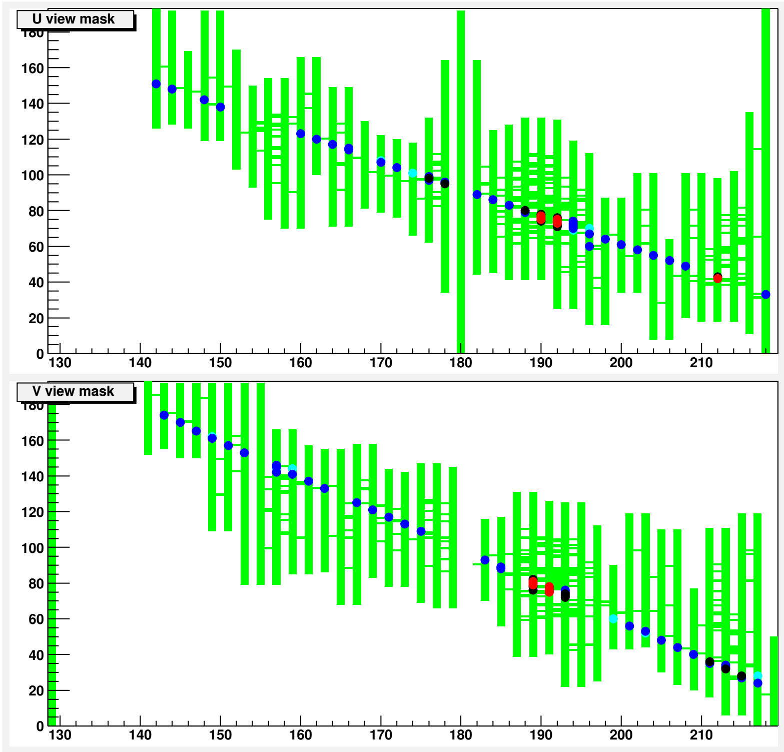


Form "Mask"



★ **The timing mask constrains locations of golden hits**





GOLDEN hits used as SEEDs to DeMux rest of event......

★ **<u>Note</u> now also used for Event ID**

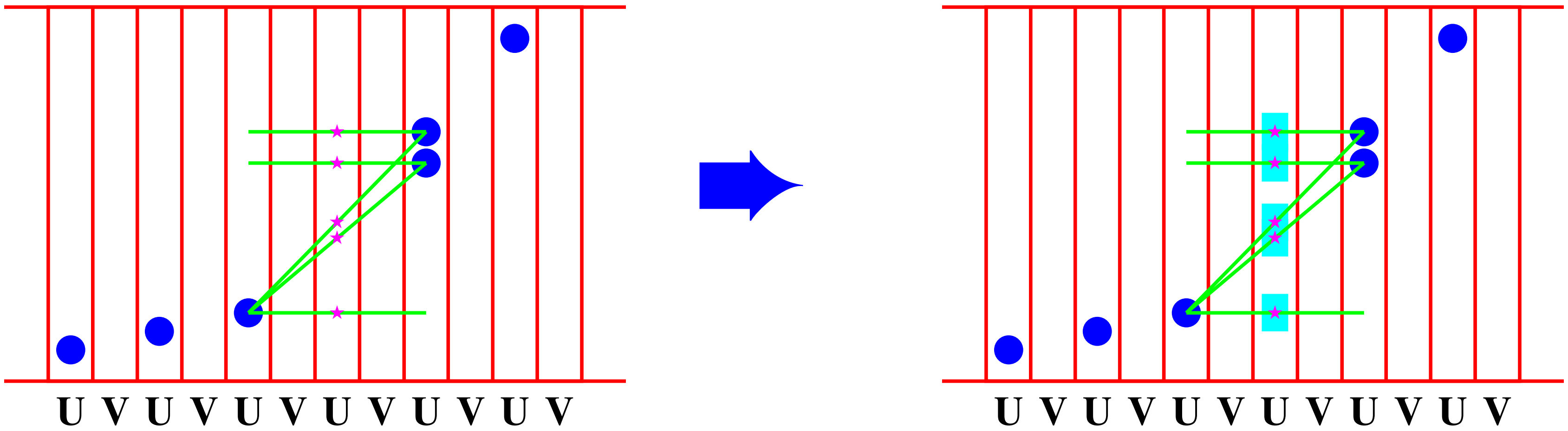
***e.g.* here from golden hits - looks like a single muon**

SEEDs and TARGETs

SEEDs used to identify "SUITABLE" positions (TARGETS) to DeMux hits in other planes.

A number of possible tactics used:

- **PROJECT_FORWARDS**
- **EXTRAPOLATE_FORWARDS**
- **EXTRAPOLATE_BACKWARDS**
- **PROJECT_BACKWARDS**
- **INTERPOLATE_FORWARDS**
- **INTERPOLATE_BACKWARDS**



NEW feature: Event identification allows favoured directions to be defined. Interpolate/Extrapolate using these directions.

Event Identification

GOLDEN hits used to make first attempt at Event ID

★ **On basis of GOLDEN hits - can make a good guess at event type, single cosmic, multiple muon, etc.**

★ **Use this information to steer subsequent demuxing.**
  - different tactics to extrapolate/interpolate between golden hits

★ **Identify MULTIPLE MUON using on U.o.M. Hough transform idea applied to GOLDEN hits**

★ **Identify SINGLE STRAIGHT MUONS if all(most) GOLDEN hits consistent with linear fit.**

★ **(At a later stage) Also identify CONTAINED EVENT candidates**

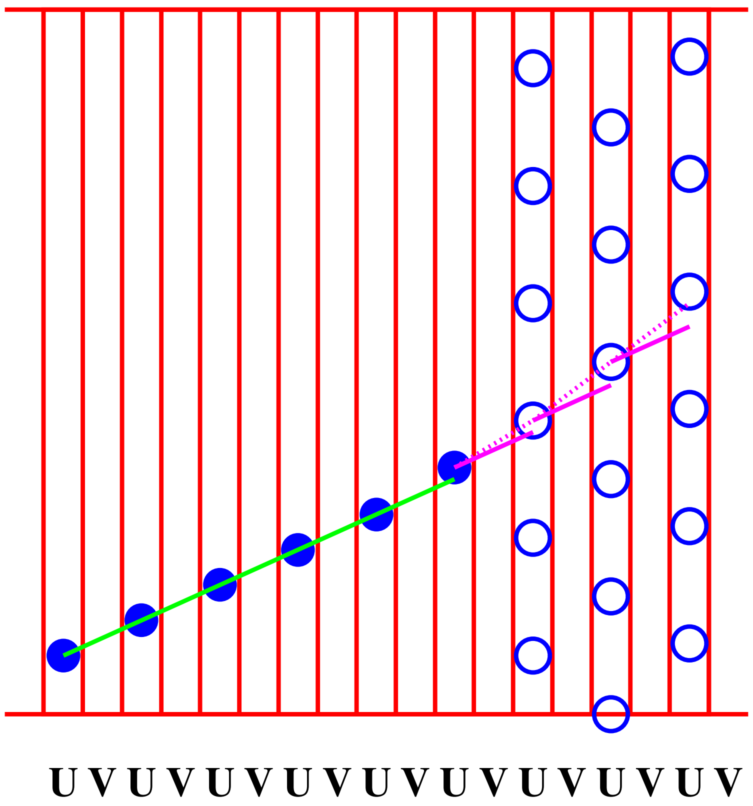
★ **Subsequent DeMuxing strategy depends on event ID**

★ **Incorrect event ID - doesn't usually result in a bad DeMux solution**

e.g. Single Muon

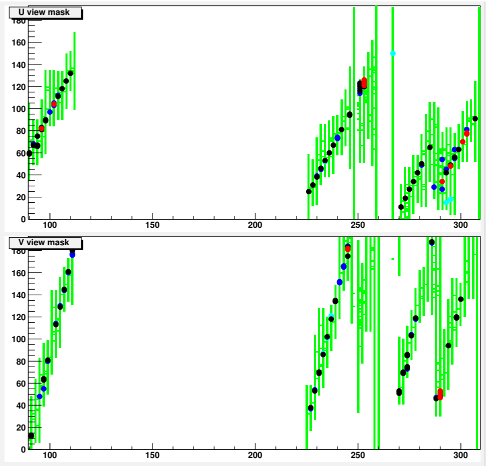
GOLDEN hits in both views give acceptable linear fits:

- ★ **Tag event as probable single muon**
- ★ **DO NOT force subsequent hits onto line**
- ★ **Just modify way targets are defined.**
- ★ **Use linear extrapolation between hits**
- ★ **OR extrapolation from last hit using fitted slope**

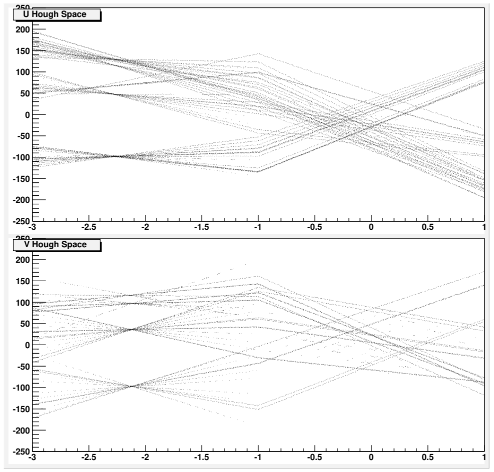


e.g. Multiple Muon Identification

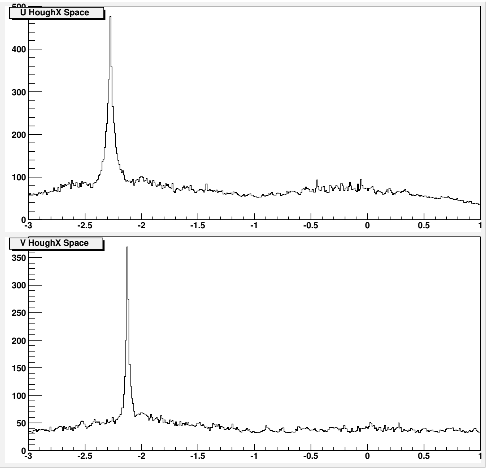
GOLDEN HITS



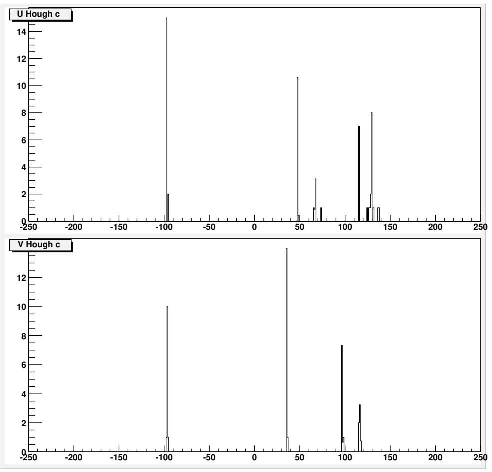
Hough Transform of GOLDEN HITS



Projection onto slope axis

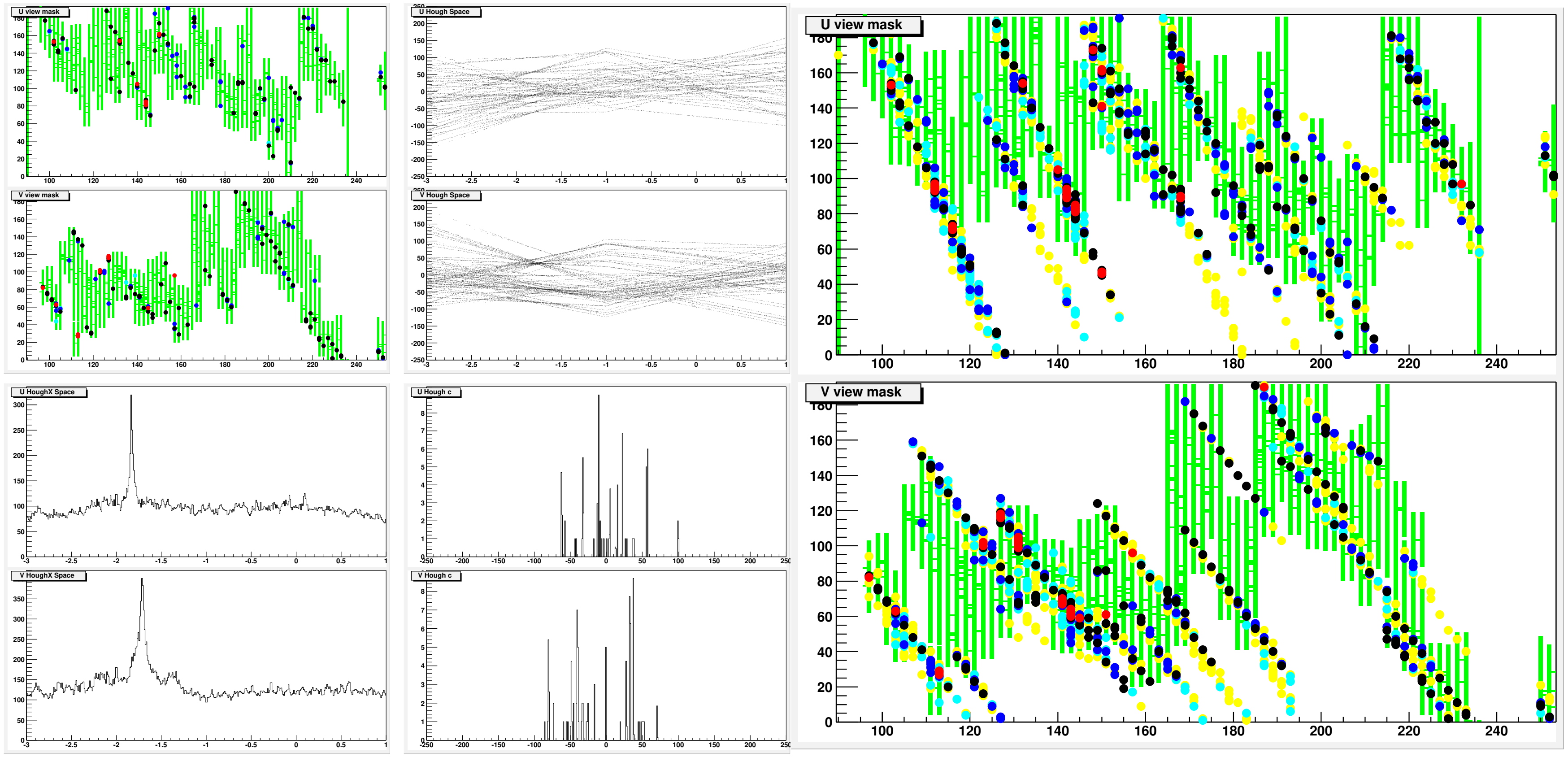


Intercept slice at favoured slope



A Complicated Example - 16(?) muons

In identified Multiple Muon events, Hough slope is used as basis to define TARGETS - improved reconstruction of multiples



★ **Multiple Muon reconstruction pretty good - but not finalised**

Overview of code

AlgAltDeMuxDevel::RunAlg() current CVS version

```
MakePixelMap(cdlh); // make map of PMT pixels for cross-talk removal

MakeAltListMap(cdlh);// fill basic data structure (NOT using Navigator class in AlgAltDeMux)

StripNoise(); // strip out any out-of-time hits

StripCrossTalk(); // remove cross-talk

GetFibreLengths(); // get the fibre lengths for this event and dump into an array - small but
significant improvement in speed

// Have now constructed basic structures - now proceed to DeMux event

MakeTimingMask(false); // First make a mask using timing information

MakeTimingMask(true); // Iterate on timing mask - bootstrap U and V view timing information

DemuxPass(0); // Now first demux pass - build a mask using loose cuts

DemuxPass(1); // Start to demux event for real - find GOLDEN HITS

Hough(); //Perform a Hough transform on Golden hits and identifiy Multiples

DemuxPass(2); // demux pass - look for new GOLDEN HITS

DemuxPass(3); // looser demux - still looking for double ended hits

DemuxPass(4); // demux anything that's left

AddBackCrossTalk(); // add back originally tagged as cross-talk

DemuxPass(5); // Demux hits originally tagged as cross-talk
```

❸ Performance

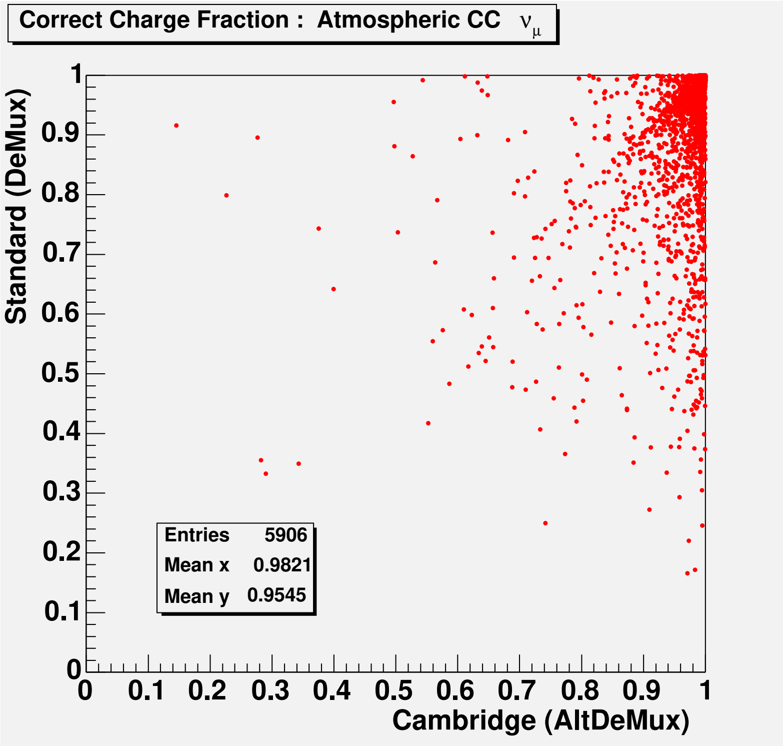
Compare AltDeMux with Standard Code (DeMux).

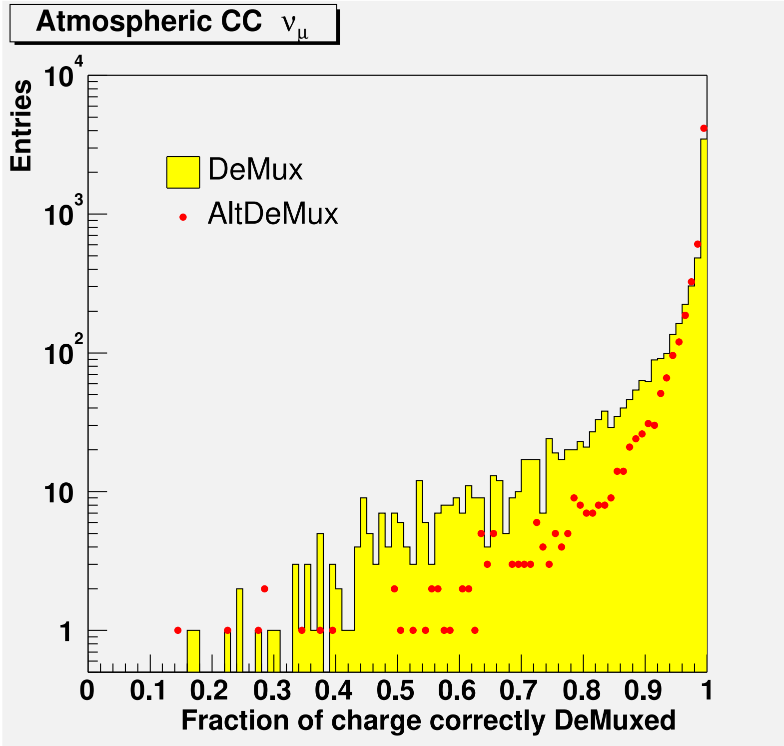
- ★ **Monte Carlo Events with ≥ 6 planes**
- ★ **DetSim - include Cross-talk, etc. (thanks to Nathaniel T. and Mike K.)**
- ★ **For Standard Code use:**
  - **AlgDeMuxBeam for beam MC**
  - **AlgDeMuxBeam for everything else**
- ★ **Comparisons for all events, *i.e.* ignore flags from DeMux and AltDeMux**

Look at fraction of charge DeMuxed in correct Strip, f_Q.

★ used current private version of AltDeMux - will put in CVS only when fully tested

ATMOSPHERIC ν_μ CC Events



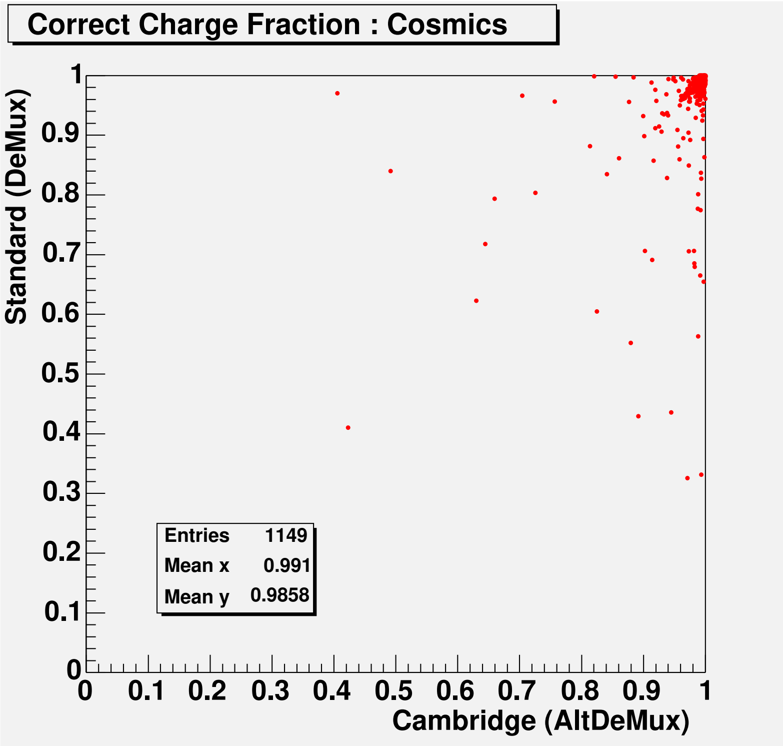


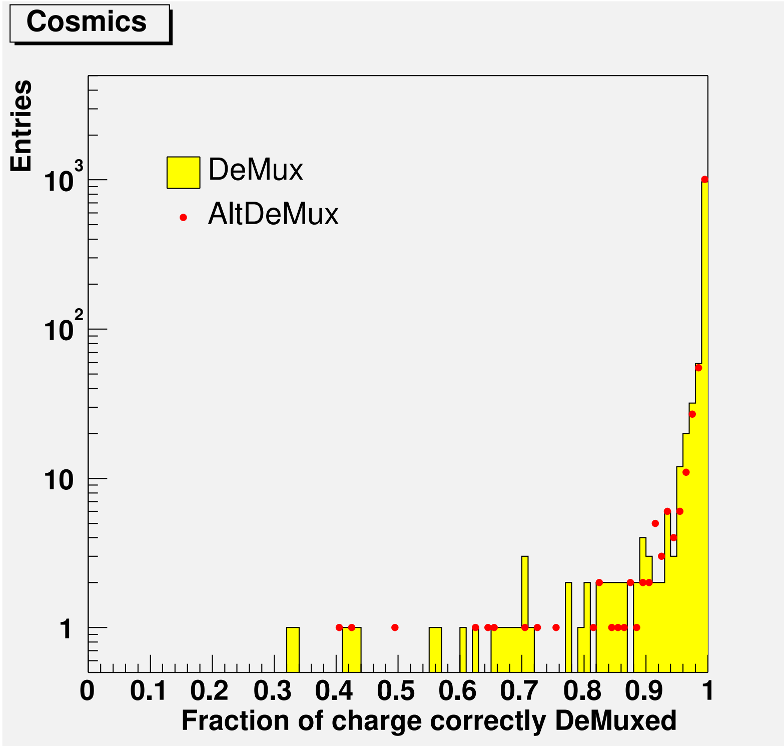
	AltDeMux	DeMux
f_Q	98.2%	95.5%

★ **AltDeMux somewhat better**

★ **AltDeMux designed for Atmospheric neutrinos (less biased towards tracks)**

Single Cosmic Muons





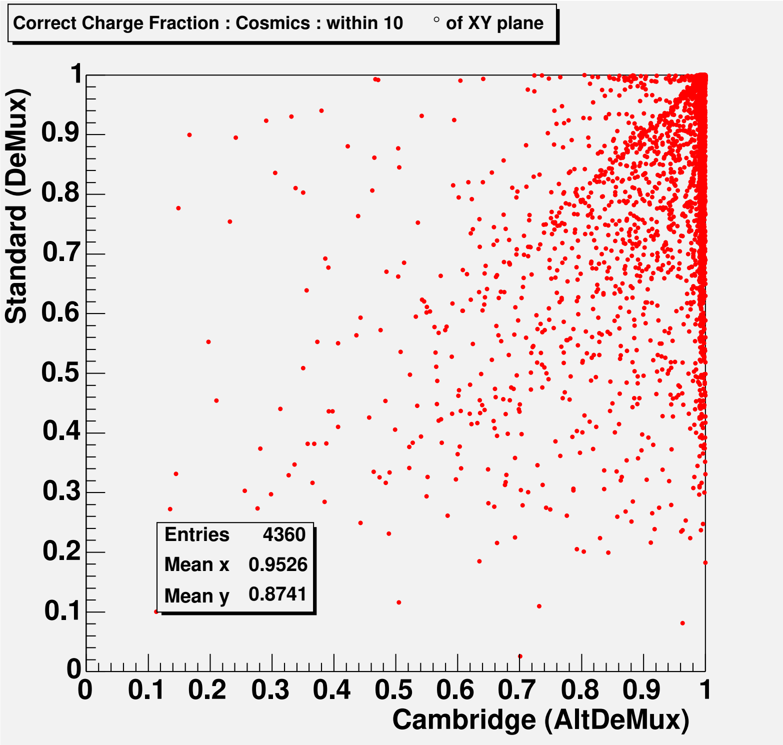
	AltDeMux	DeMux
f_Q	99.1%	98.6%

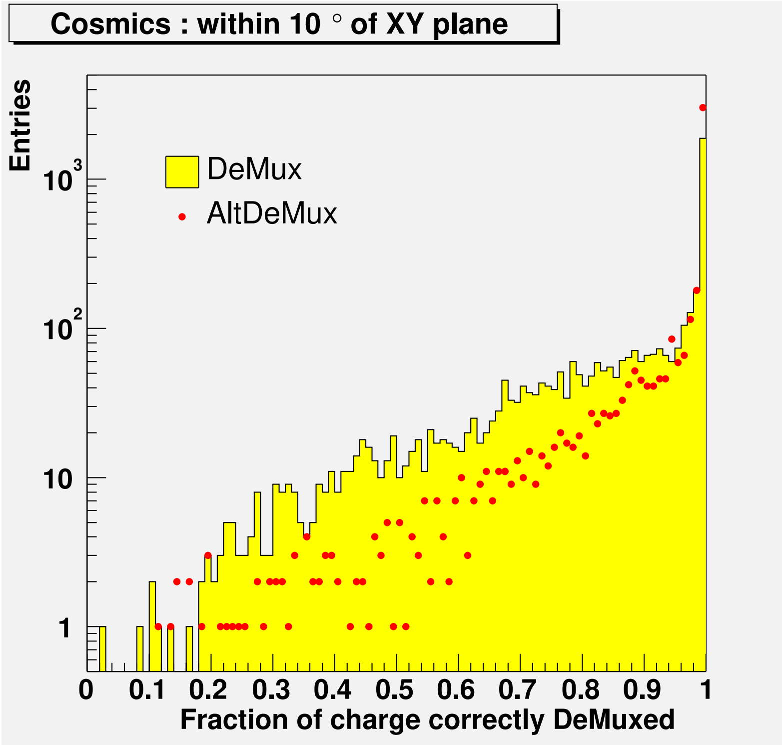
- ★ **Very similar**
- ★ **What about potential background to atmospheric neutrinos ?**

Sneaky cosmics

$E_\mu < 10$ GeV and within 10° of xy-plane

(dominant source of background in Atmos ν analysis

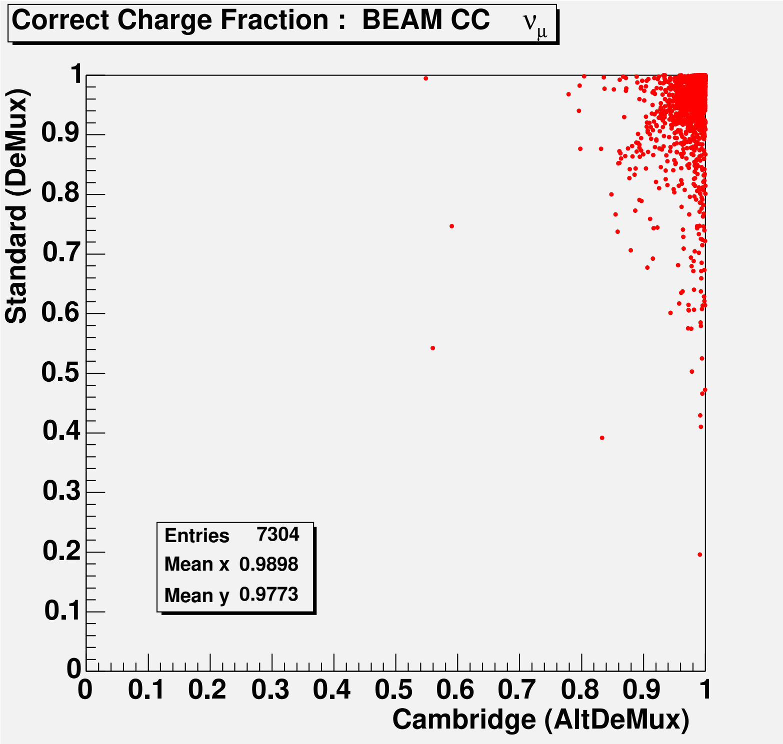


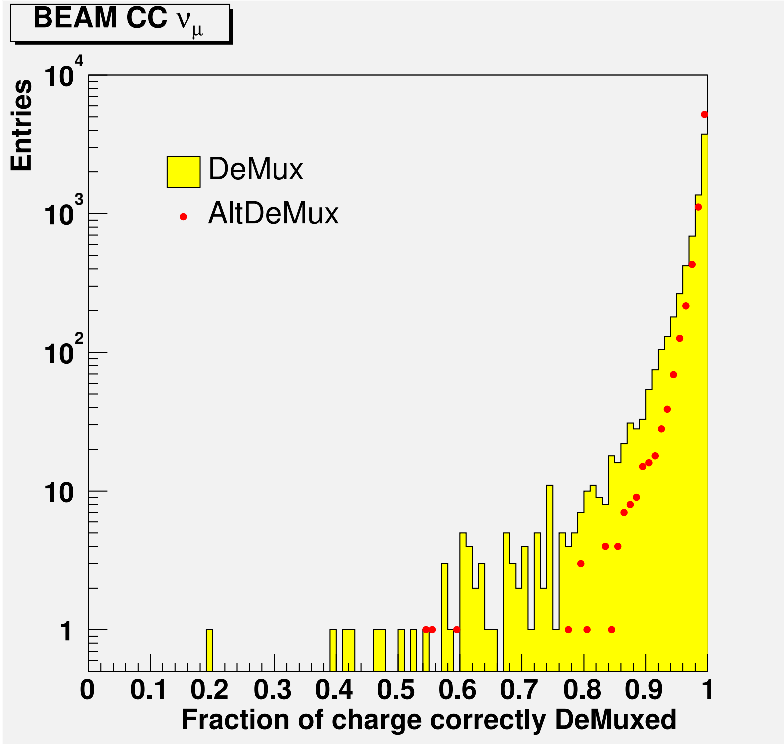


	AltDeMux	DeMux
f_Q	95.3%	87.4%

- ★ **AltDeMux peforms significantly better**
- ★ **again less biased towards straight tracks**

Beam ν_μ CC events





	AltDeMux	DeMux
f_Q	99.0%	97.7%

★ **Similar peformance**

★ **AltDeMux slightly better**

Performance : Summary

	AltDeMux	DeMux	Gain
Atmospheric ν_μ CC	98.2%	95.5%	$+2.7\%$
Atmospheric ν NC	96.1%	92.7%	$+3.4\%$
Single Cosmic Muons	99.1%	98.6%	$+0.5\%$
Sneaky Cosmic Muons	95.3%	87.4%	$+7.9\%$
Beam ν_μ CC	99.0%	97.7%	$+1.3\%$
Beam ν_μ CC $E_\nu < 5$ GeV	98.1%	97.2%	$+0.9\%$
Beam ν_μ NC Events	98.2%	95.5%	$+2.7\%$

- ★ AltDeMux places slightly more charge in correct place
- ★ However real test comes in relative sensitivity of physics analysis.....
- ★ + AltDeMux is ~ 20 times faster than Standard code.

  e.g. for data : 0.075 s/cosmic c.f. 1.43 s/cosmic
- ★ + AltDeMux tags $\sim 88\ \%$ of cross-talk hits
- ★ <u>BUT</u> much more sensitive to calibration - particularly timing

❹ Conclusions and Future Work

Conclusions:

★ AltDeMux works well, performance/speed

★ Viable alternative to Standard code, but need to evaluate impact on physics performance

★ Different strengths/weaknesses
e.g. use DeMux for cosmics, AltDeMux for atmospherics ?

★ Now have 2 high quality DeMuxers ! Use in tandem to reject DeMux failures.

Future Work:

★ Reaching limit of approach... not much left to do

★ but still room for improvement - have yet to fully optimize cuts etc.

★ Develop a “beam neutrino” mode

★ Tidy up multiple muon reconstruction

★ New code will be in CVS soon (before October 7th)