CalWire and Deconvolution Update

LarSoft Meeting

Dec. 7, 2011

H. Greenlee

CalWire Module

- CalWire module performs signal shaping of wire signals during reconstruction.
  - Deconvolutes field + electronics response.
    - Converts induction plane pulses from bipolar to unipolar.
  - Also fits “exponential tail.”
    - Not sure what this is, exactly.
    - Argoneut only.
  - Is experiment-specific.
    - CalWireT962.
    - CalWireMicroBooNE.
  - Microboone version is broken.

Convolution and Deconvolution Kernels

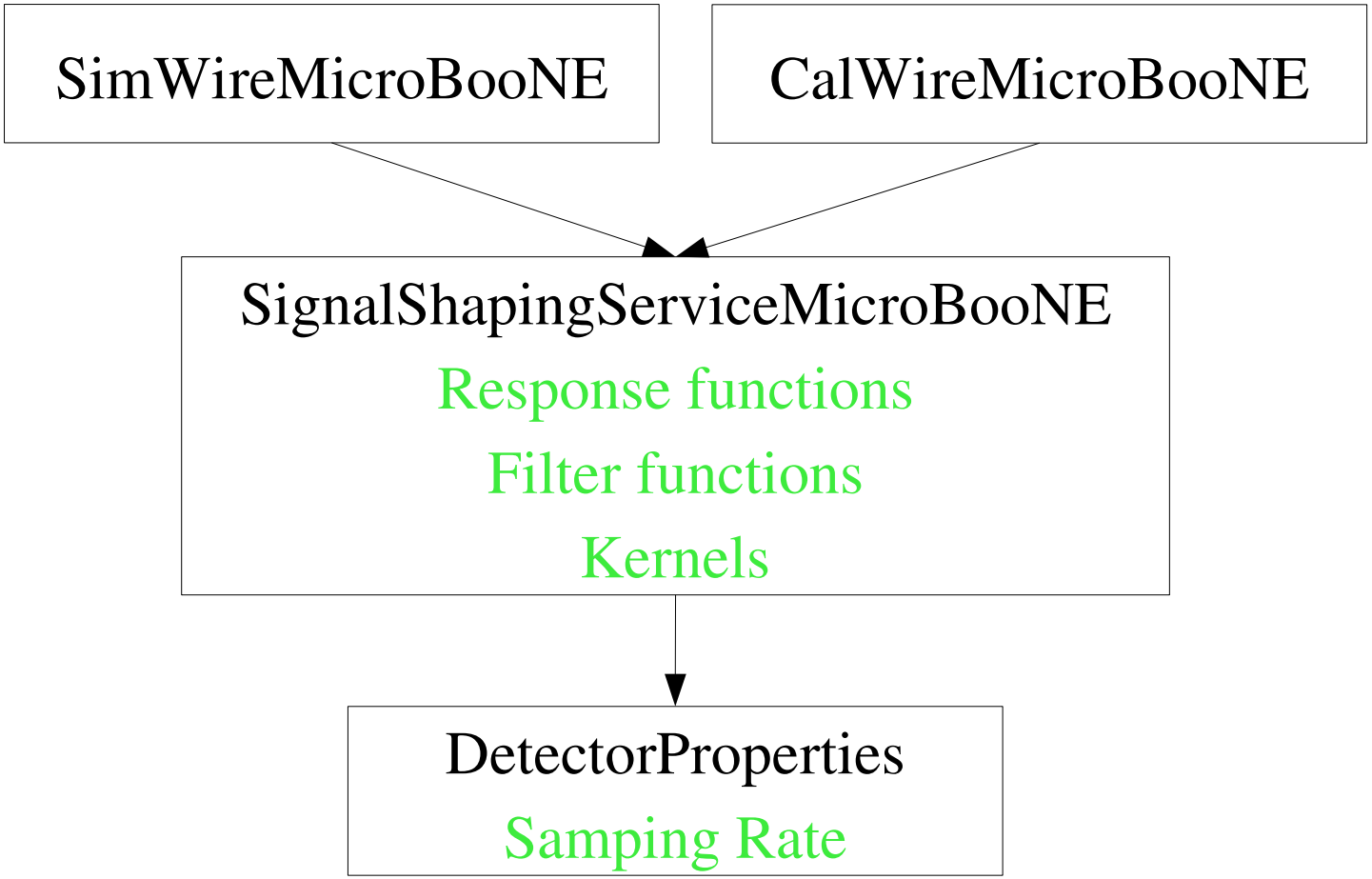
- Convolution kernel.
  - $R(f) = FT\{R(t)\}$.
  - Response function $R(t)$ is convolution of field and electronics response.
- Deconvolution kernel.
  - $K(f) = F(f) / R(f)$.
  - Filter function $F(f)$ is needed to suppress frequencies where $R(f)$ is zero or dominated by noise.

Why CalWireMicroBooNE is Broken

- The deconvolution kernel is dependent on -
  - Sampling rate.
  - Response functions.
- Current versions of CalWire (including CalWireMicroBooNE) read a precalculated deconvolution kernel from a data file.
  - Data file has not been updated, even though sampling rate and response functions have changed.
- A better solution than updating the precalculated deconvolution kernel is to calculate the deconvolution kernel on the fly during job initialization.

Strategy

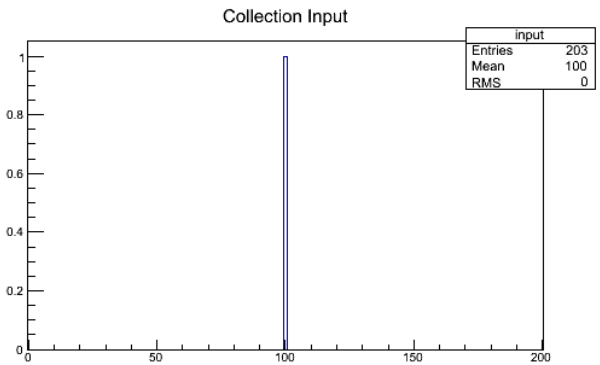
- Response functions, filter functions, and kernels are owned by a service.

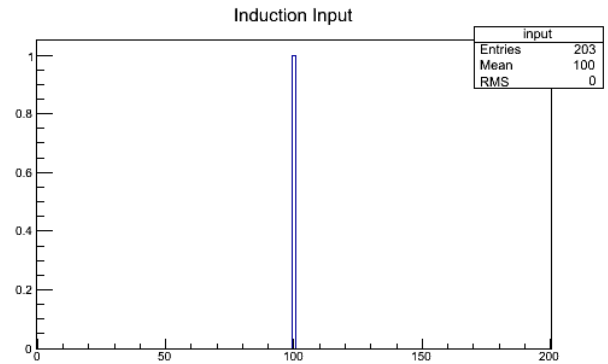


Current CalWireMicroBooNE Performance

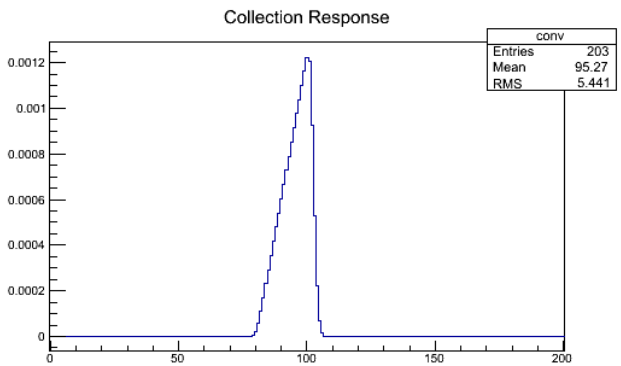
t_s=198 ns

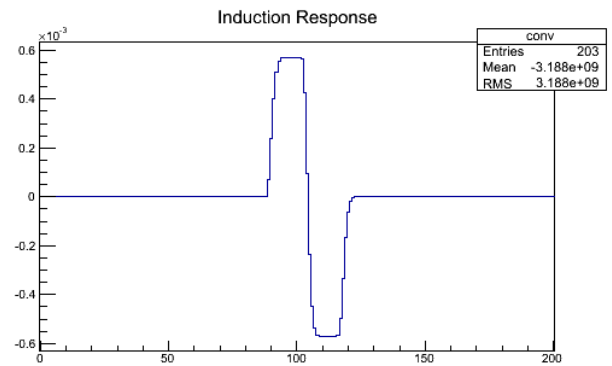
Input



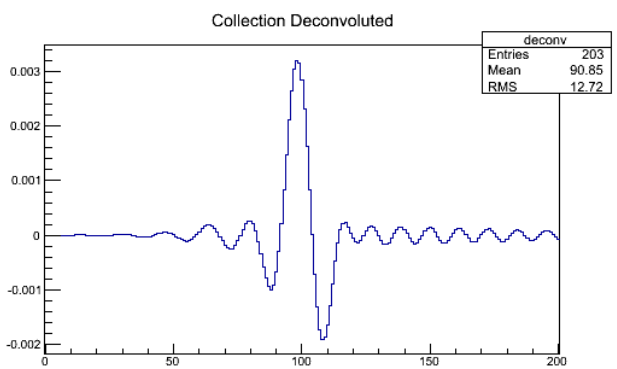


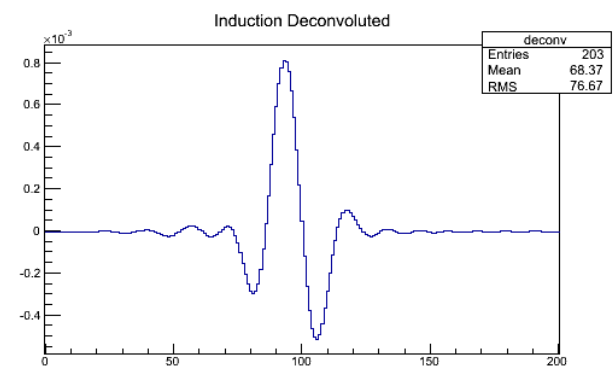
Convoluted





Deconvoluted

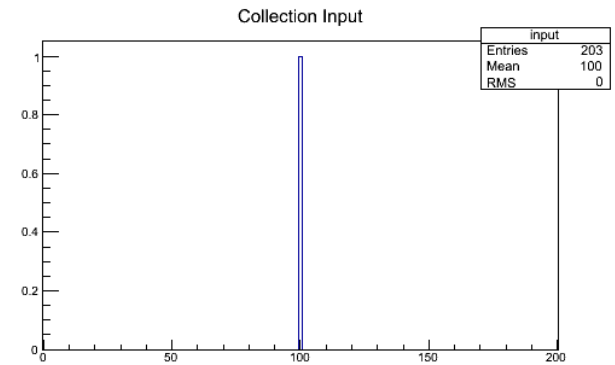


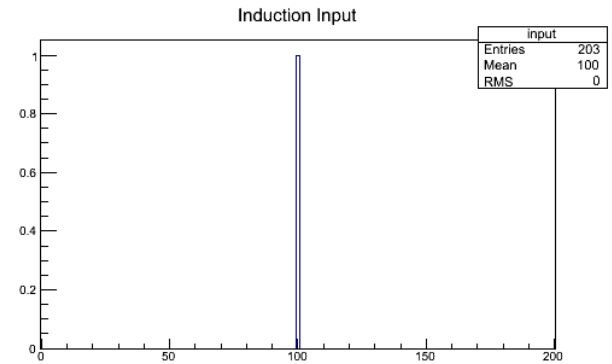


Current CalWireMicroBooNE Performance

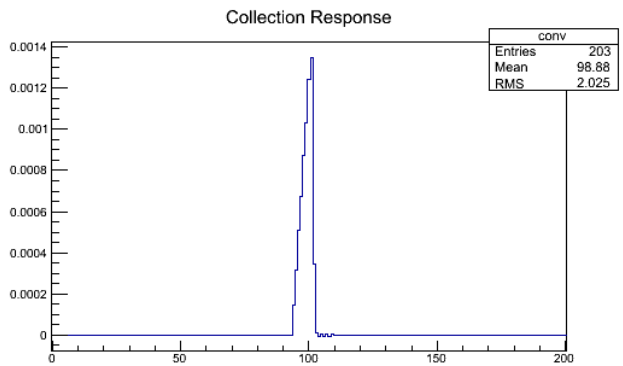
t_s=500 ns

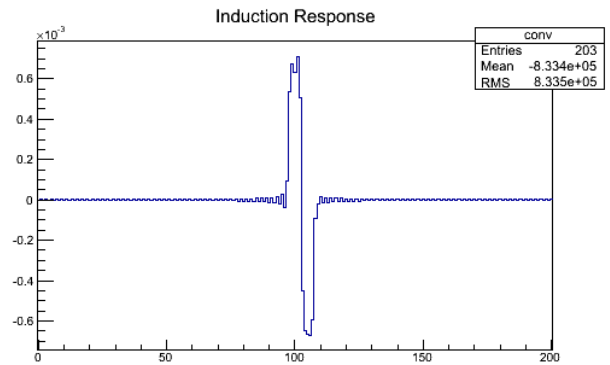
Input



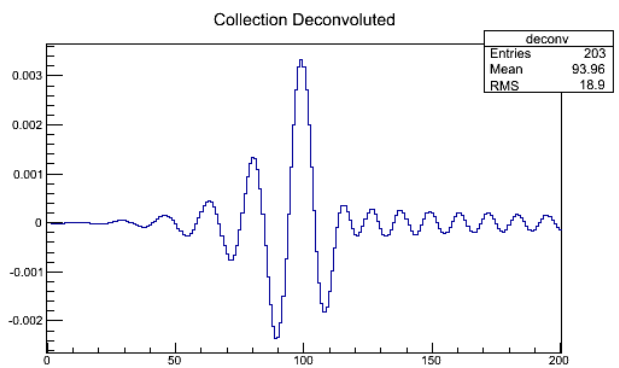


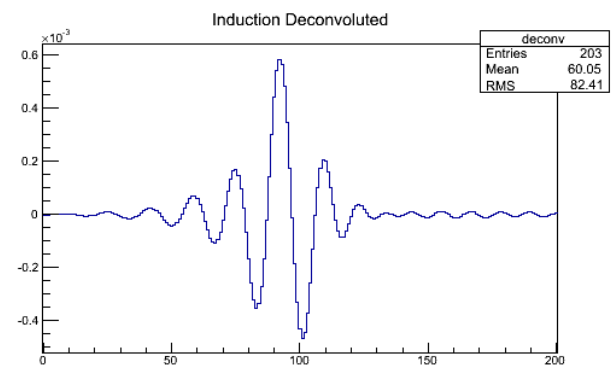
Convoluted





Deconvoluted





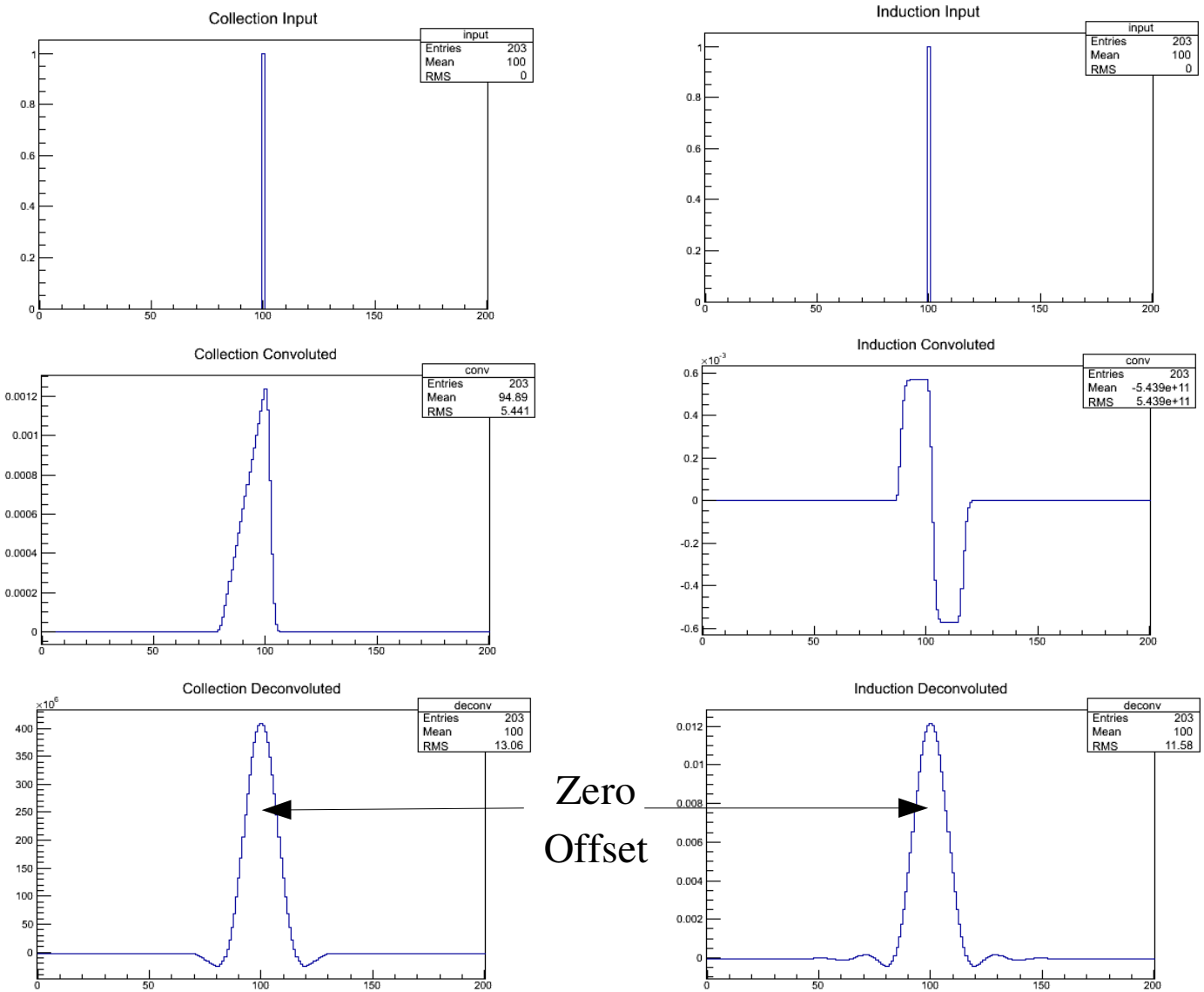
Updated CalWireMicroBooNE Performance

t_s=198 ns

Input

Convoluted

Deconvoluted



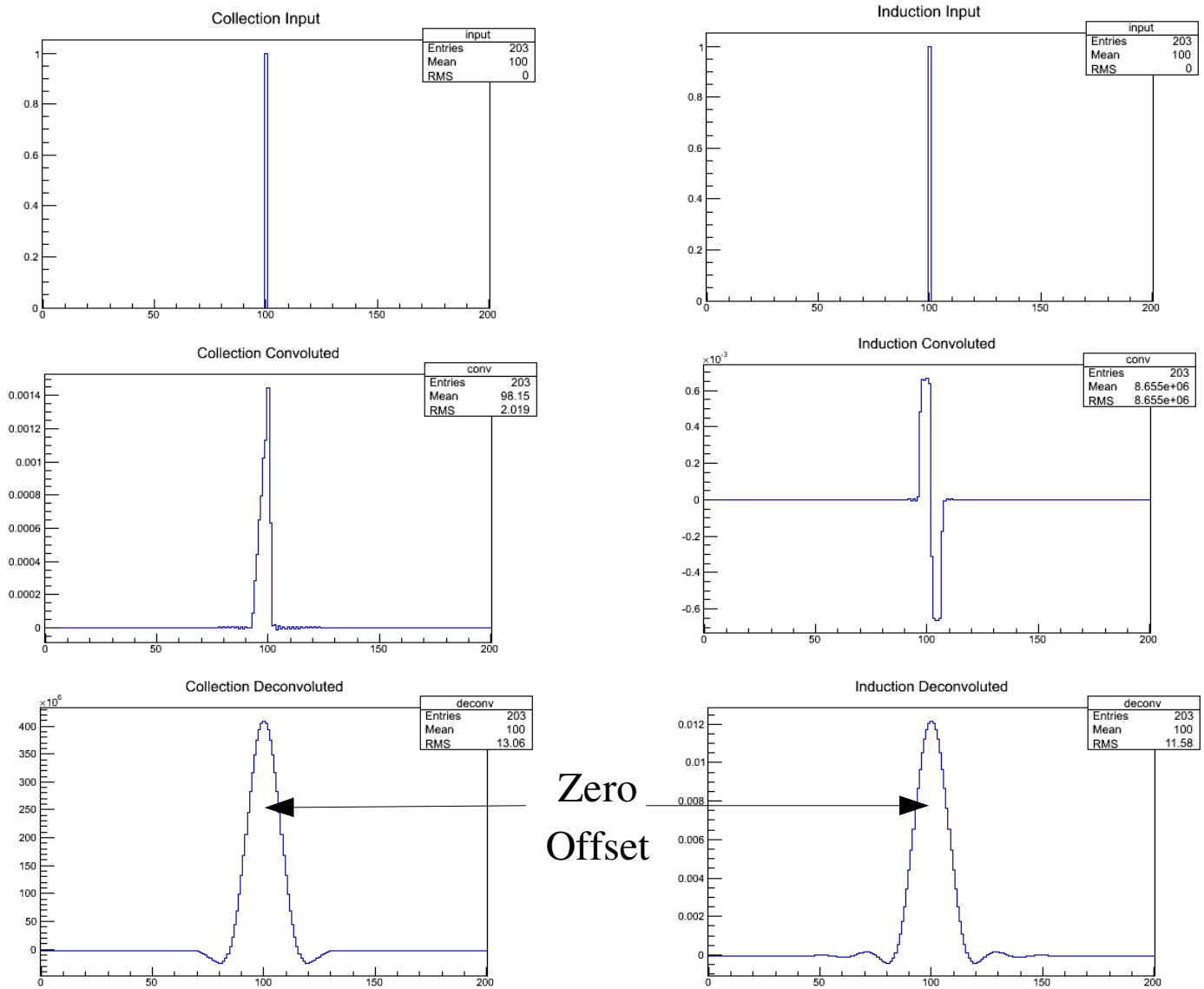
Updated CalWireMicroBooNE Performance

t_s=500 ns

Input

Convoluted

Deconvoluted



Choice of Filter Function

- Current filter functions in SignalShapingServiceMicroBooNE copied from CalData/MicrobooneResponseFile (added by Brian P.).
  - Differs from filter functions stored in precalculated data file.
- Filter function implemented as root parameterized function (TF1) defined using fcl parameters.

```
IndFilter:       "gaus*exp(-0.5*pow(x/[3],[4]))"
IndFilterParams: [ .454, 0.0, 150.0, 200.0, 8.0 ]
ColFilter:       "(x>0)*[0]*exp(-0.5*(((x-[1])/[2])^2)^[3])"
ColFilterParams: [ 79258.0, 0.0, 150.0, 1.5 ]
```

- Deconvoluted pulse shape (modulo noise) is convolution of filter function and input pulse.
  - Shaping induced time offset is zero if done consistently.

Choice of Filter Function (cont.)

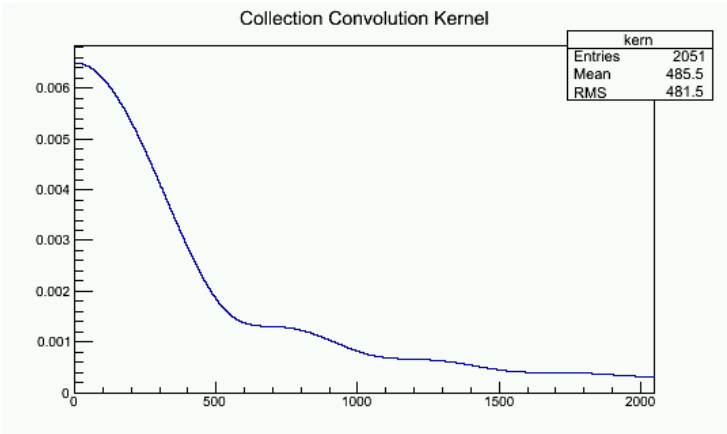
- There is may well be room to tune the filter function.
- According to image processing theory (Wiener deconvolution), the optimal filter function is
  - $F(f) = |R(f)|^2 / (|R(f)|^2 + |N(f)|^2)$
    - $R(f)$ = Response function.
    - $N(f)$ = Noise spectrum.
  - It is not obvious whether something like a Wiener deconvolution filter makes sense to try (you would need a model for the noise spectrum).

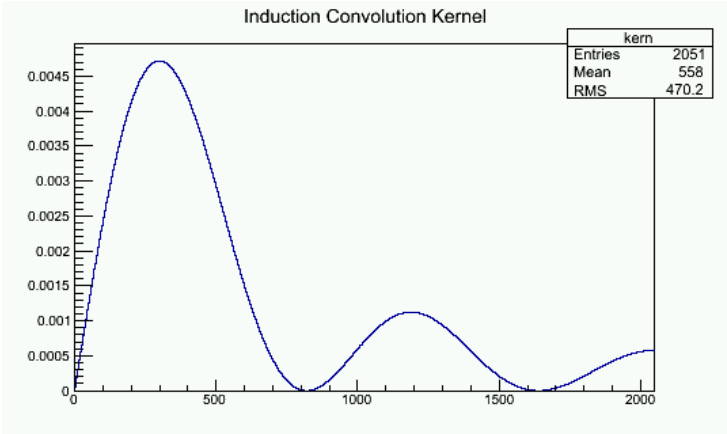
Convolution Kernel and Filter Functions

t_s=500 ns

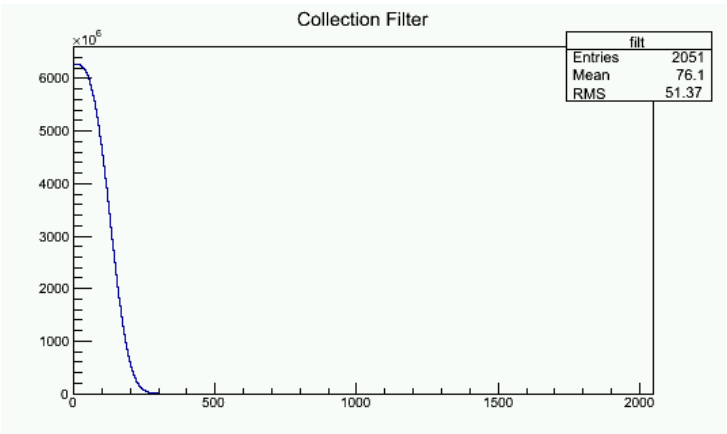
Collection Plane | Induction Planes

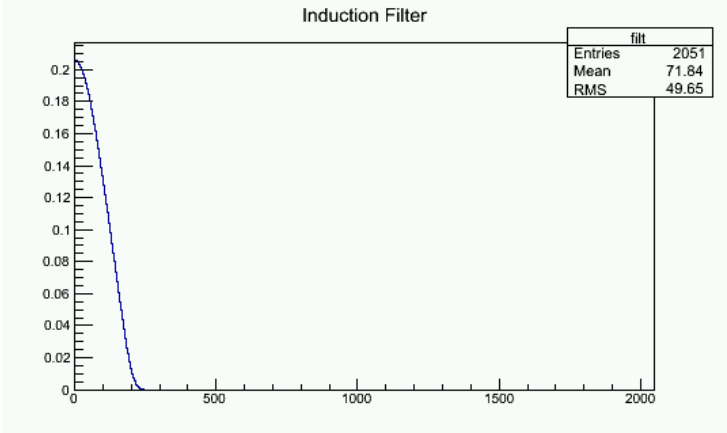
$|R(f)|$





$F(f)$





- Filter functions are clearly narrower than they need to be.
  - Were probably tuned for t_s=198 ns.

Status

- Initial version of code in my private test release (nothing committed to svn yet).
  - SignalShapingServiceMicroBooNE added in DetSim package.
  - CalWireMicroBooNE updated.
  - SimWireMicroBooNE not updated.
- To do.
  - Optimize filter function.
  - Test reconstruction chain.
  - Field response needs to be calculated better.
  - Do FFTHitFinder parameters need to be updated?