Wire-Cell Toolkit
Some Q&A

Brett Viren

Physics Department

BROOKHAVEN
NATIONAL LABORATORY

MB AT – 20180503

Outline

Acronyms

WCT	Wire-Cell Toolkit (also WCP for prototype)
WC/LS	Related to integration of WCT and LArSoft
NF	Noise filter
SP	Signal processing
L1SP	Special type of SP using an L1 regularization

WCT Simulation Overview

Signal 4D's: depo→drift→ductor→digitize:

- Point-like energy depositions input.
- Linear drift to wires + long/trans diffusion, absorption.
- Field response induction on wires, electronics shaping.
- Simple, linear ADC model.

Noise ("dissonance" = 5^{th} D?)

- User-provided average **noise amplitude spectra** as a function of wire length and allows "misconfigured" **gain/shaping** for select channels.
  - WCT includes spectra for post-software-filtered noise from MB.
  - Can "misconfigure" signal as well.
- Rayleigh sampling of amplitude, uniformly sampled complex phase.

Signal+Noise summed at voltage level (just before digitizer).
Can run just-signal or just-noise.

Space Charge Effect

WCT simulation currently does not have any SCE model.

- Nominal drift is modeled assuming a uniform electric field up to the wires where field response functions take over.
- SCE model can be applied by shifting initial location depositions just before input to WCT.
- Or, relatively easy for someone to add a WCT component to perform this shift first thing.

Ionization & Recombination

It's up to the user to provide deposition information somehow.

- WCT sim does not generate energy depositions (no G4 yet) except for simple point or ideal line source tests.
  - → of course, user likely wants depos from "real" physics int. sim.
- WCT ultimately needs **number of electrons at a point**.
- If user has *dE* instead of $\#e^-$ there is optional support for calculating ionization and recombination.
  - If just *dE* at a point is given, then assume MIP step.
  - If both *dE* and *dX* then can apply either Birks or Box models.

WCT provides simple JSON file input of depositions as well as input of `SimEnergyDeposit`'s from `art::Event` via the WCT component `SimDepoSource` which is part of `larwirecell`.

Diffusion

A WCT "depo" actually represents a 2D Gaussian. It may start point like but "grows" (longitudinal $\times$ transverse) extent as it drifts in the simulation.

- Diffusion is governed by D_L and D_T parameters
  - $\rightarrow$ (see, eg arXiv:1508.07059v2)
- Electron absorption governed by a single exponential lifetime
  - $\rightarrow$ (see, eg MB note 1003)
- User option to turn off fluctuation of total charge left in a depo after it finishes drifting.

Induction

WCT uses a **long-range, fine-grained induction** model[1]
An element of charge in a depo induces waveform:

- On many wires:
  - Nominally, the 21 nearest wires receive waveform.
  - Actual number is configurable based on coverage of the user-provided field response functions.
- Based on the local field
  - Nominally, drift paths and their fields are defined at $\frac{1}{10}^{th}$ pitch intervals, user defined.
  - Further, the two nearest paths are interpolated to avoid aliasing that would otherwise occur.

Each wire plane is handled separately when performing the 2D (trans $\times$ long) convolution of field response functions and charge distribution, w/ results binned into channel (naturally handles wrapped wires).

[1] If you call this "DIC" then I guess it stands for "Do It (more) Correctly". There's nothing particularly "dynamic" about it.

Simulating with Shorted Wires

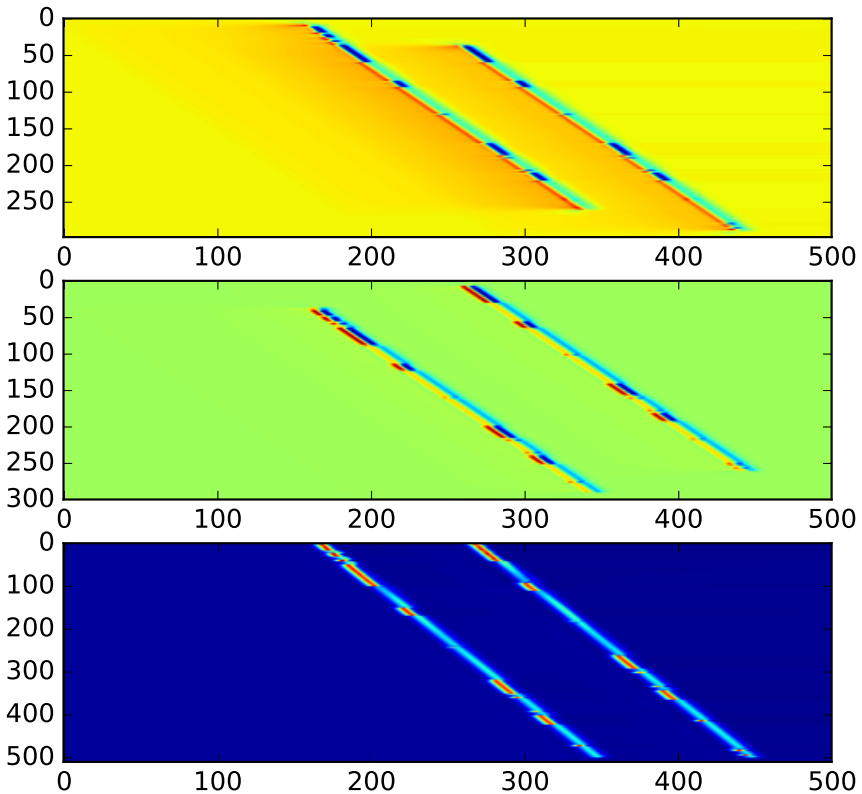
WCT can map a given depo to an induction model based on rules defined in terms of the depo data.

- For example: final or initial location in physical or "wire space", size, charge, time, ID#, etc.
- This feature is used to map each depo to appropriate set of field response functions for the wire region in which it drifts.
- Nominal and UV/VY shorted field response functions provided in WCT (generated with 2D Garfield, Yichen).

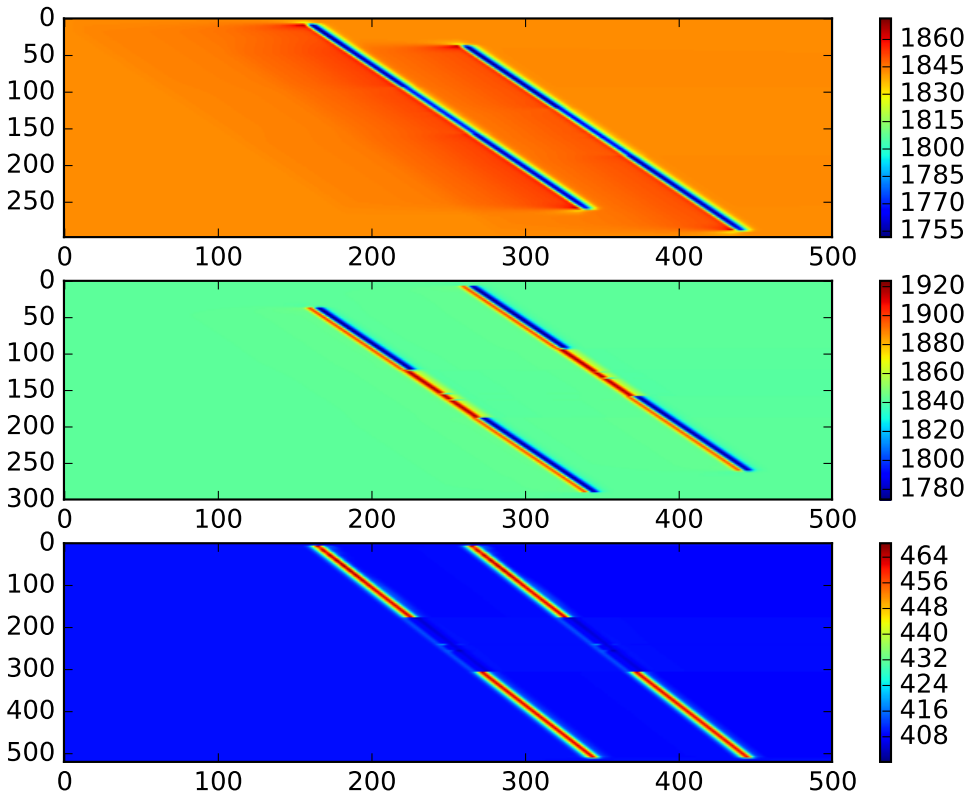
Caveat: long range induction means that the waveforms from a depo near an "edge" of one rule will "spill over" into neighboring regions.

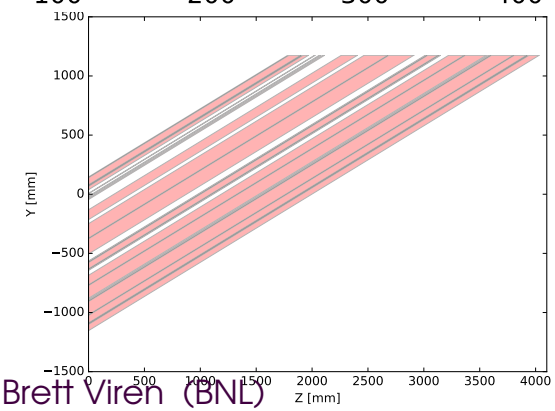
Test tracks, MB grounded wires (chan vs tick)

UV-grounded region

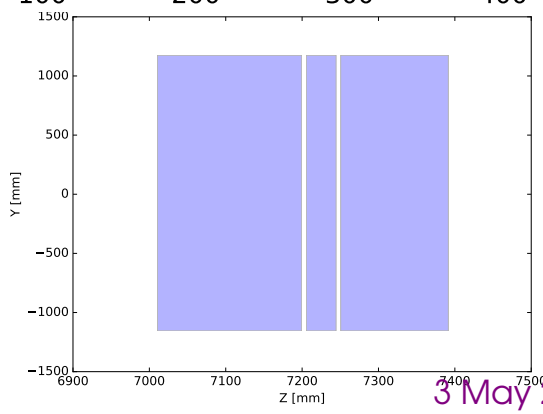


VY-grounded region





(grounded wires)



Simulating with Misconfigured ASICs

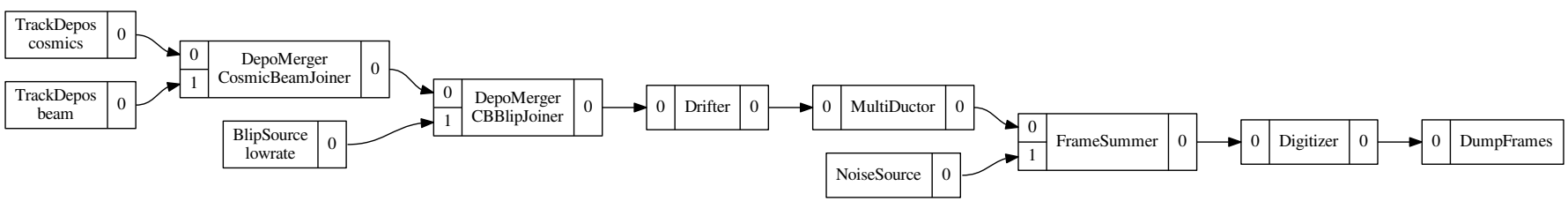
Per channel "misconfiguring" of electronics gain + shaping supported independently for signal and noise:

- **Signal** is initially simulated with uniform gain/shaping. "Misconfiguring" is applied as a follow-on.
  - Convolve a waveform spectra using a frequency filter that is the ratio of "misconfigured/nominal" electronics response spectra.
  - A WCT component will be produced to apply this.
- **Noise** simulation accepts per-channel gain/shaping parameters.
  - Also a scaling term applied to the noise amplitude spectrum associated with the wire length feeding the channel.

Bonus: Cheap Mixing in WCT

WCT simulation supports mixing at many levels.

- WCT inherently follows a **data flow processing** (DFP) paradigm.
- DFP easily supports the "mixing" of data streams.
  - Not just for sim, but any data/process flow.
- WCT simulation could mix multiple sources of:
  - depos, each one type of kinematics (cosmic, beam, ^{39}Ar, dirt)
  - frames, noise+signal, `MultiDuctor`, data+sim



WCT Simulation as DFP graph mixing three sources of depositions and two sources of frames.
Diagram from an actual job, generated automatically and directly from the configuration.

L1 Signal Processing

Exploits a technique from **compressed sensing**.

- Essentially, "fit" a waveform to a mix of unipolar+bipolar response functions, but do it fast.
- L1SP needed for proper handling of signals in the MB shorted wire regions.
- Runs as follow-on to the nominal WCT 2D SP.
- Runs as yet another WCT component
  - Effectively, no user-visible difference in a job other than some signals are better.

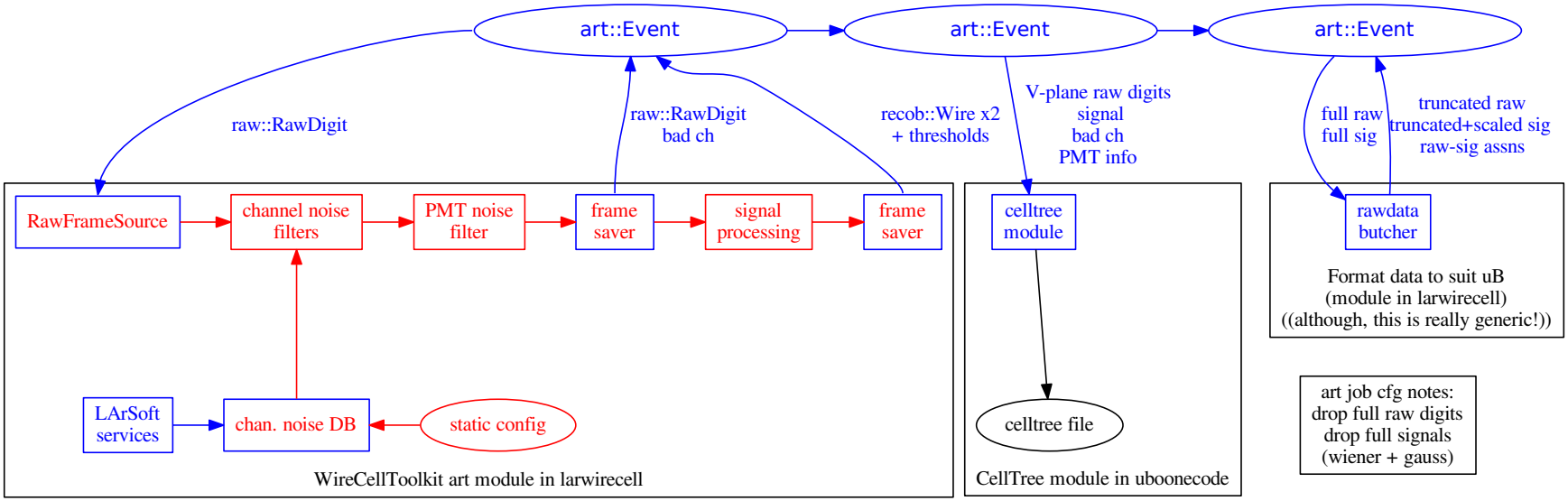
The one and only WC/LS module

`WireCellToolkit_module`

- Is a LArSoft module living in `larwirecell`
- Is fully generic to the make up of the actual WCT job.
  - → Only module needed to run any sort of WCT job inside LS.
- Is mostly empty as it delegates to the "`wcls_main`" `art::Tool`.
  - The tool is to *art* as the `wire-cell` command line program is to `bash`.
  - It's FHiCL parameters correspond to `wire-cell`'s CLI args.
  - Both can pass arbitrary but select user parameters from FHiCL/CLI to WCT config.

N.B. the initial WCT "noise filter module" has long been deprecated and unsupported and its existence has led to someconfusion. I would like to remove it from `larwirecell`.

The current NF+SP WCT job



nf+sp A WC/LS module running WCT NF + SP components, output full-tick `recob::wire`.

celltree a "pure" LS module producing a file for Wire Cell Prototype (WCP) code with signal + V-plane raw, bad ch, PMT info.

butcher a "pure" LS module output truncated-tick and assn'ed `RawDigit` and `recob::Wire` vectors.

"*Why 2 (actually 3) modules instead of 1?*" Not sure how to answer, by if I'm allowed to be cheeky: because they are called "modules" not "monoliths".

Persisting Waveforms

WCT NF output saved as **full-tick** `RawDigit` (full-tick = 9595 ticks, but its configurable)

WCT SP output saved as **full-tick, sparse** `recob::Wire` for "gauss" and "wiener" filtered signal.

butcher saves **truncated-tick** `RawDigit` and `recob::Wire` copies and makes assn's based on channel number. (truncation: drop first 2400, keep next 6400, again configurable)

Our `art` job is configured to **drop** the intermediate full-tick data.

"*What is the proper way to slim these data products?*" I don't know what is meant by "slim" but the `recob::Wire` are produced **sparse**.

WCP SP Regions of Interest, aka "signal"

- At one point in the SP, the algorithm must reject all regions of time in a channel's waveform which are consistent with just noise. What is left is potentially "signal".
- A contiguous period of time in that signal is called an "ROI".
- The WCT SP signal information in ROIs is most naturally represented as a sparse vector, so `recob::Wire` is an obvious choice for LS consumption.

"*How do ROI differ from CalWireROI?*" No idea, I never heard of this class, and it doesn't come up in LArSoft's doxygen documentation.

WCT Configuration

Q *What tunable parameters exist?*

A1 None.

A2 A huge amount.

- WCT configuration allows experts to have a lot of power to configure just about "everything possible" while selectively exposing only a few "knobs" to novices.
- External parameters may be "injected" from the user interface, eg the `wire-cell` program command line or the FHiCL for the `wcls_main` tool for *art* users.

FHiCL injection to WCT Config

The FHiCL for testing integration of WCT simulation into LS:

```
simmer : {
  module_type : WireCellToolkit
  wcls_main: {
     tool_type: WCLS
     apps: ["Pgrapher"]
     plugins: ["WireCellPgraph", "WireCellGen",
               "WireCellSio","WireCellLarsoft"]
     configs: ["bogosim.jsonnet"]
     inputers: ["wclsSimDepoSource"]
     outputers: ["wclsFrameSaver"]
     params: {
        detector: "uboone"
}}}
```

The variable "`detector`" is then available to the WCT configuration.
Its value can be used to control the construction of the final configuration data structure to, eg, to switch between MicroBooNE and DUNE.

WC/LS Integration Status

- Work in `larwirecell` branch **feature/bv_simint**
- WCT simulation now integrated and tested.
- Requires new `SimEnergyDeposit` data object class
  - Copied code related to this class from Hans/Bill+Wes's branch to `larwirecell` branch **feature/bv_simint**
  - Plan to merge both branches simultaneously.
- Once L1SP work is done in WCT itself will make 0.7.x
- Then will make `wirecell` UPS product `v0_7_x` and seek Lynn's bug checking.
- Then request Lynn to merge branches next release.

Goal was this week but now it's more like early next week.