WCT Simulation Developments and larsoft integration

Brett Viren

Physics Department

BROOKHAVEN NATIONAL LABORATORY

Yale Visit to BNL – 12 Apr 2018

Outline

Developments in Wire-Cell Simulation and Toolkit
- Signal
- Noise
- Mixing
- DFP Graph

Integration of WCT sim to LArSoft

Developments in Wire-Cell Simulation and Toolkit

- Signal
- Noise
- Mixing
- DFP Graph

Integration of WCT sim to LArSoft

Simulated Signal

The four D's: **deposit** → **drift** → **(in)duction** → **digitize**

- User-provided: wire geometry + 2D field responses (batteries included for MB).
  - Also, parameterized wire geometry generation for MB, DUNE, etc.
- Includes some idealized depo sources: point-like (includes ^{39}Ar spectrum), track-like line source, user-configurable parameters ($\vec{r}$, $\vec{dr}$, $\frac{dE}{dX}$).
  - → Geant4 depos not yet supported but see LArSoft integration slides
- The `Drifter` reference implementation can apply ionization and recombination if depos in terms of energy. Or, can accept depos already in terms of #e's.
- Three `Ductor` implementations provided (next slide).
- Simple, linear `Digitizer` with user-configurable parameters (resolution, gains, baseline, range).

Ductors

A "ductor" converts a charge distribution ("depos") near the anode into voltage waveform fragments.

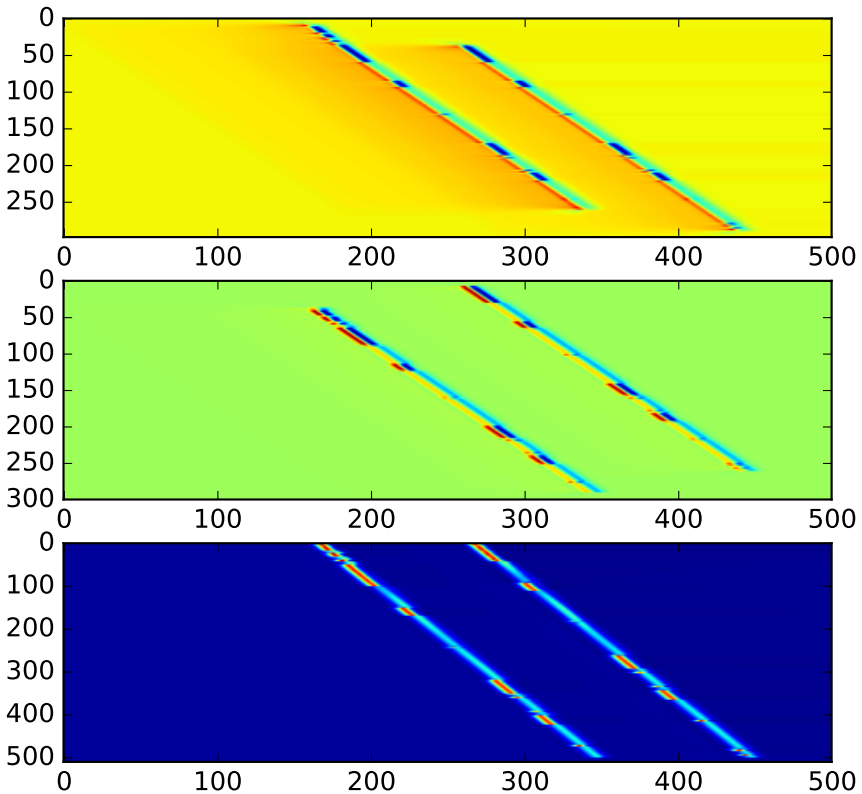
`Ductor` reference implementation using a single set of field responses.

`MultiDuctor` a facade mapping a depo to a `Ductor` via user-configured rules.

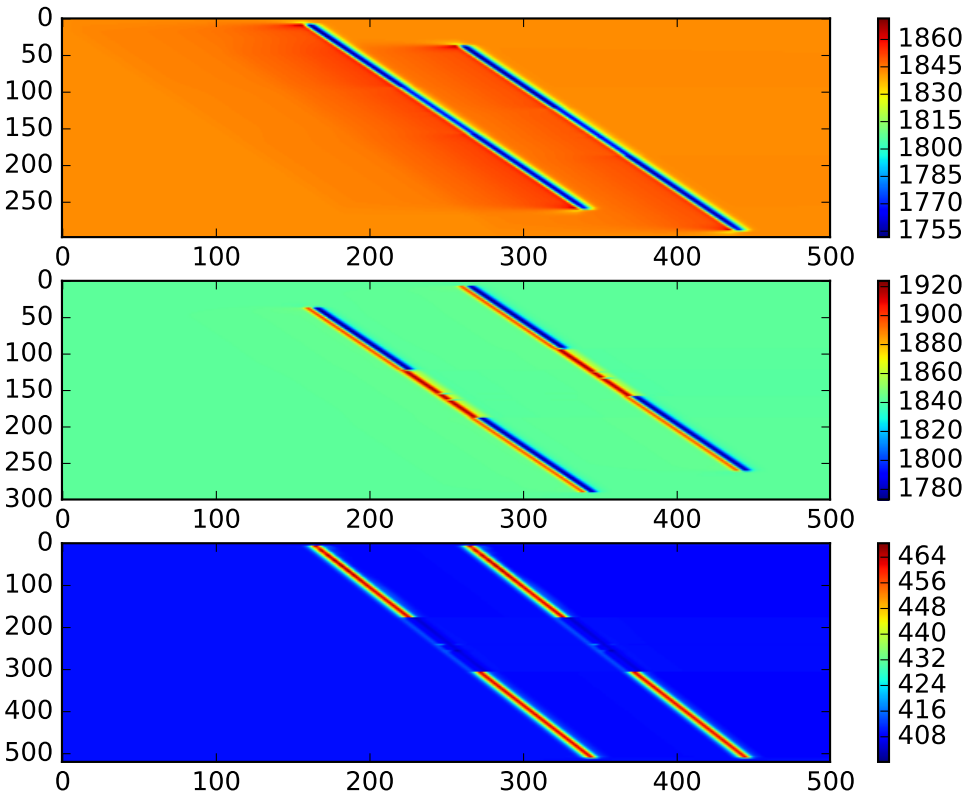
`Truth` approximate the "true" post-deconvolved signal waveforms directly from depos.

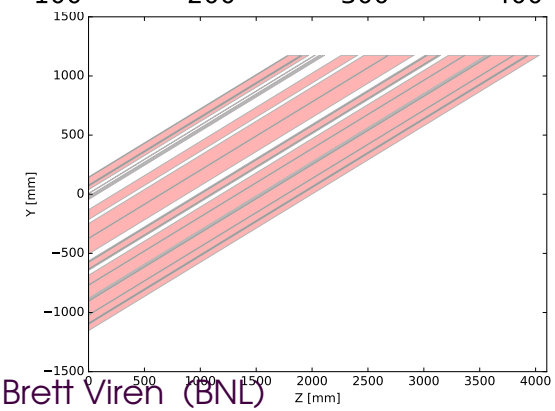
Test tracks, MB grounded wires (chan vs tick)

UV-grounded region

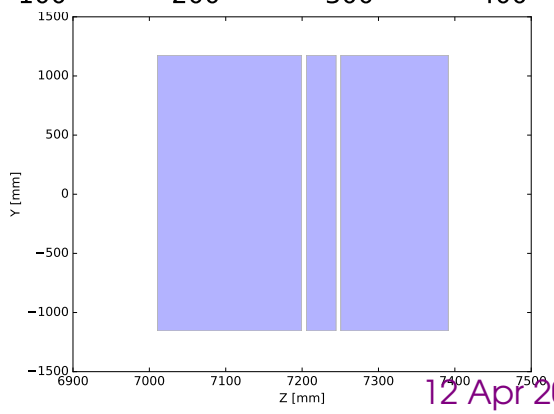


VY-grounded region





(grounded wires)



Developments in Wire-Cell Simulation and Toolkit

- Signal
- Noise
- Mixing
- DFP Graph

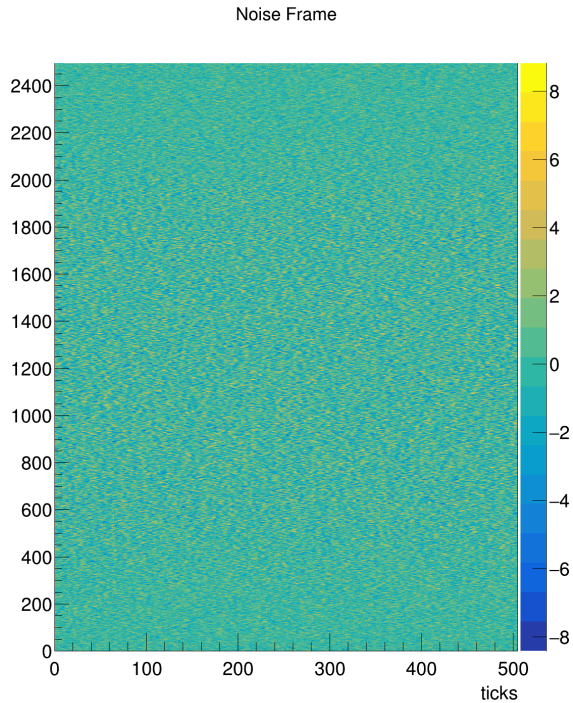
Integration of WCT sim to LArSoft

Simulated Noise

The fifth D: **dissonance**

- Including the noise simulation in a job is a user-configurable optional.
- User-provided mean noise specta as function of wire length.
- Battery included: post-filtered spectra from MicroBooNE noise paper.
- Proper Rayleigh sampling when producing noise waveforms.
  - → todo: generating noise needs many randoms, which is rather slow. Ideas on how to speed this up need implementation and validation.

→ todo: support for modeling coherent noise.



Sampled noise spectra in channel vs tick for MB U-plane channels. Note variation as function of wire length.

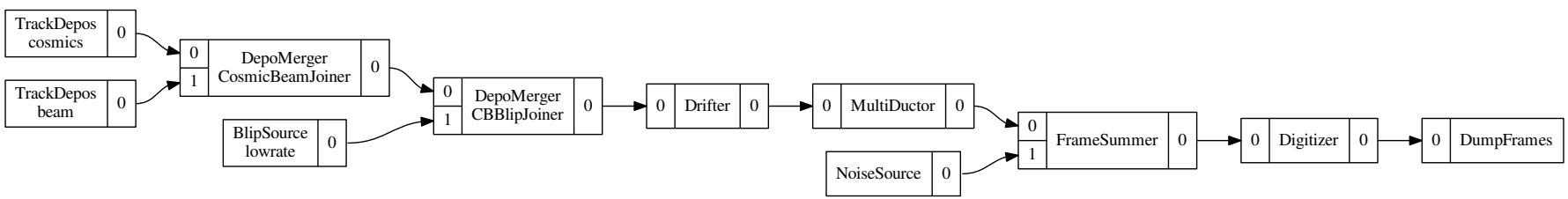
Developments in Wire-Cell Simulation and Toolkit
- Signal
- Noise
- Mixing
- DFP Graph

Integration of WCT sim to LArSoft

Mixing

WCT simulation supports mixing at many levels.

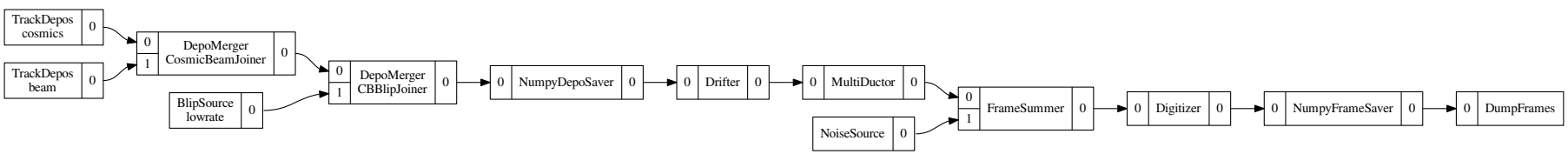
- WCT inherently follows a **data flow processing** (DFP) paradigm.
  - constrast: *art* follows an "event" (blocked) processing paradigm.
- DFP naturally supports "mixing" of data streams.
  - Not just for sim, but any WCT job.
- WCT simulation could mix multiple sources of:
  - depos, each one type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



DFP Graph Execution

New support for DFP graph execution:

- Prior, most WCT "app" components hard-coded execution paths.
  - If you squint at the C++ you might see a directed acyclic graph.
- New: `Pgrapher`, a single-threaded, memory-minimizing DFP graph execution engine
- Can now construct many variants of a full WCT job simply in WCT configuration by effectively "drawing" the graph.
  - Eg, this graph is generated directly from a WCT configuration:



todo: a multi-threaded DFP graph execution engine also exists in WCT but needs validation. It most likely can only reap benefit in stand-alone `wire-cell` on HPC (outside of a monolithic larsoft).

Developments in Wire-Cell Simulation and Toolkit

Integration of WCT sim to LArSoft

WC/LS Integration in General

Wire Cell / LArSoft integration follows a specific design:

- WCT components that **convert** between WCT data interfaces and `art::Event` data objects.
- WCT components that **adapt** LS services.
- The mostly empty `WireCellToolkit_module` which uses,
- The `WCLS_tool` which executes WCT "apps"
  - This `art::Tool` is to *art* what `wire-cell` is to your shell command line.
  - What might be on the command line is provided in FHiCL.
  - Largely identical WCT config works for WC/LS or stand-alone WCT running.

WC/LS Code & Config

- Integration code is in the `larwirecell` package, part of the base LArSoft family.
  - Development is against `master` not the ancient MB "production" branches.
  - Package only holds integration glue and essentially no "real" code except for some "impedance matching" modules.
- Compiled WCT code is provided by the `wirecell` UPS product as an "external" package.
  - Subset of releases made in upstream WCT become UPS'ified.
- WCT configuration files (including "data" files such as for field response, noise spectra) are provided by the user and found by WCT via the `WIRECELL_PATH` env. var.

WC/LS Status and Work Needed

- Noise filter + signal processing is integrated and in use.
  - WCT config uses hard-wired `Omnibus` "app", may move to `Pgrapher`.
- Simulation integration already "done"
  - Full `art::Event` → WCT → `art::Event` for the "4 D's" for single "event".
    - (took four hours of fighting `mrb` and 30 minutes of actual coding)
  - Minor work on WCT needed to better adapt WCT's DFP to *art*'s "event" based processing paradigms.
  - I have copied `SimEnergyDeposit` into my own recent branch of `lardataobj`.
    - Will push to merge my `lardataobj` branch to `master`.
    - For testing, I simply generate depos for a line source.
    - Getting LArG4 or other sources to produce depos left as an exercise for the user.