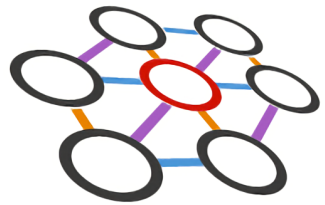
LArTPC Simulation and Signal Processing Updates for FD Vertical Drift

Haiwang Yu (BNL) for the Wire-Cell Team

FD simulation and reconstruction meeting

Nov. 22, 2021



Outline

- Configuration update for Vertical Drift
- Memory/ROOT IO issues scaling up to more CRMs/APAs
- Progress with DNN-ROI finding in Wire-Cell Signal Processing
- Portable Parallelization development for Wire-Cell LArTPC simulation

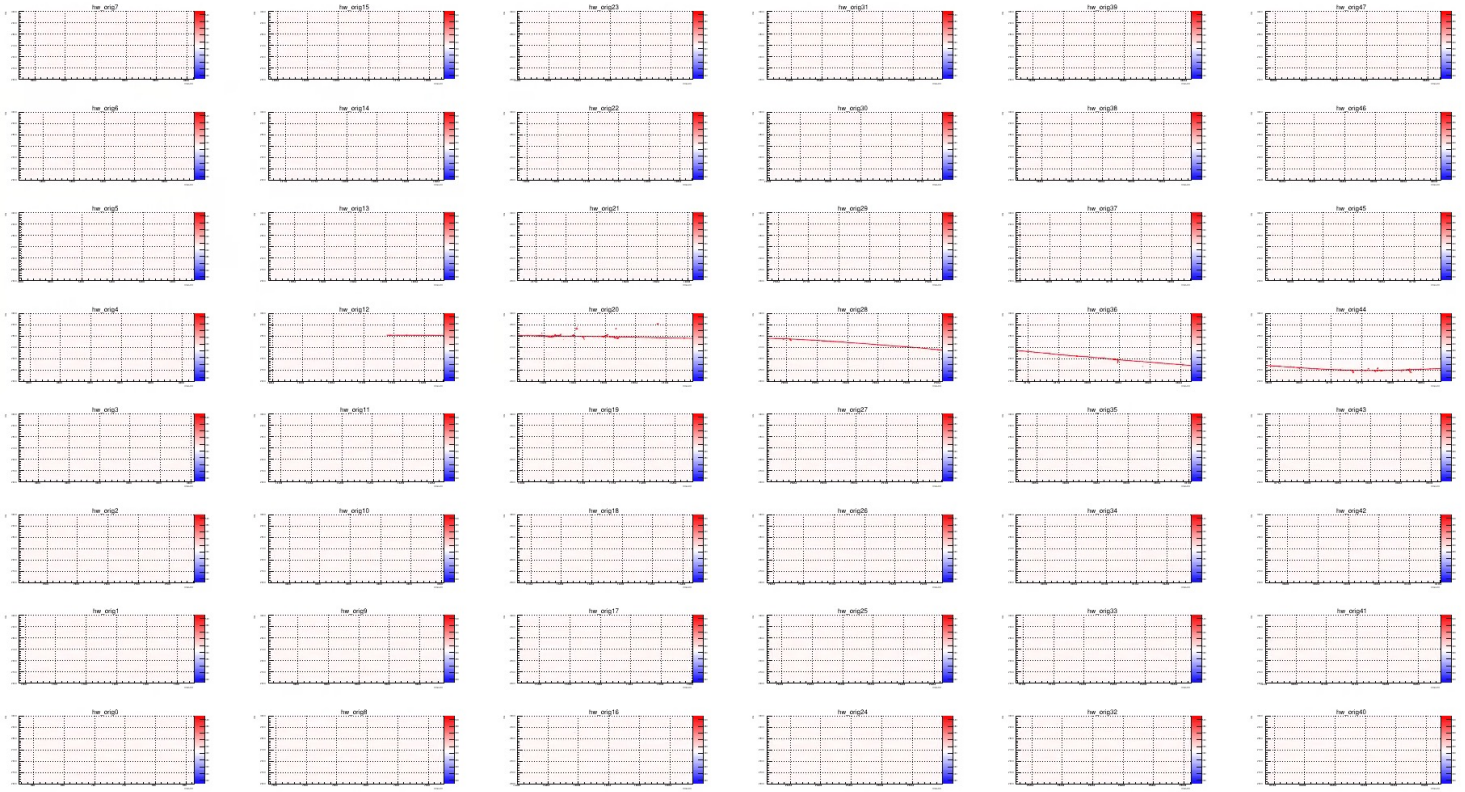
Wire-Cell Simulation and Signal Processing configuration for Vertical Drift

Update:

- Switched from compiled json to jsonnet
- Parsing most parameters from dune services instead of manually
  - only drift speed is still hardcoded
- New configuration for 1x8x6 workspace
- Available in dunetpc v09_36_00_01
  - May need manually set $WIRECELL_PATH for now

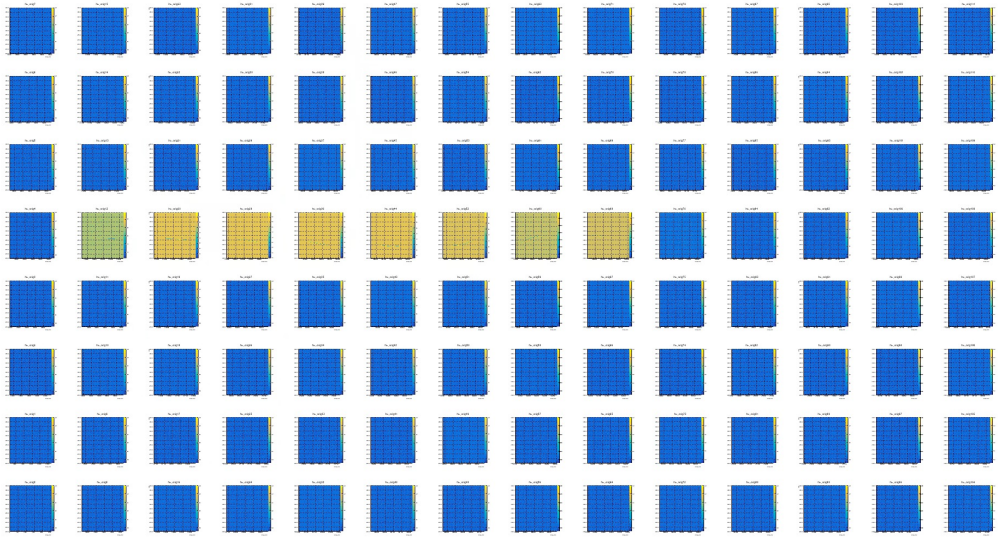
fcl block for 1x8x6 sim

```
structs: {
  # number of time samples
  nticks: @local::dunefdvd_detproperties.NumberTimeSamples
  # Longitudinal diffusion constant [cm2/ns] 4.0e-9
  DL: @local::dunefd_largeantparameters.LongitudinalDiffusion
  # Transverse diffusion constant [cm2/ns] 8.8e-9
  DT: @local::dunefd_largeantparameters.TransverseDiffusion
  # Electron lifetime [us] #10.4e3
  lifetime: @local::dunefdvd_detproperties.Electronlifetime
  # Electron drift speed, assumes a certain applied E-field [mm/us]
  driftSpeed: 1.60563
  # G4RefTime [us]
  G4RefTime: @local::dunefd_detectorclocks.G4RefTime
  # response plane [cm]
  response_plane: 18.92
  # number of CRMs, 36 for 1x6x6, 112 for 1x8x14
  ncrm: 48
}
```



ROOT IO error for Vertical Drift 1x8x14

Magnify Output



TrigReport ---------- Event summary ----------
TrigReport Events total = 1 passed = 1 failed = 0

TrigReport ---------- Modules in End-path ----------
TrigReport Run Success Error Name
TrigReport 1 1 0 out

TimeReport ---------- Time summary [sec] -------
TimeReport CPU = 106.881483 Real = 106.905577

MemReport ---------- Memory summary [base-10 MB] ------
MemReport VmPeak = 11023 VmHWM = 9767.18

%MSG-s ArtException: PostEndJob 11-Nov-2021 20:52:11 EST ModuleEndJob
---- EventProcessorFailure BEGIN
EventProcessor: an exception occurred during current event processing
---- FatalRootError BEGIN
Fatal Root Error: TBufferFile::WriteByteCount
bytecount too large (more than 1073741822)
ROOT severity: 3000
---- FatalRootError END
---- EventProcessorFailure END
---- FatalRootError BEGIN
Fatal Root Error: TTree::SetEntries
Tree branches have different numbers of entries, eg sim::SimChannels_tpcrawdecoder_simpleSC_wclssim. has 0 entries while EventAuxiliary has 1 entries.
ROOT severity: 2000
---- FatalRootError END
%MSG
Art has completed and will exit with status 1.

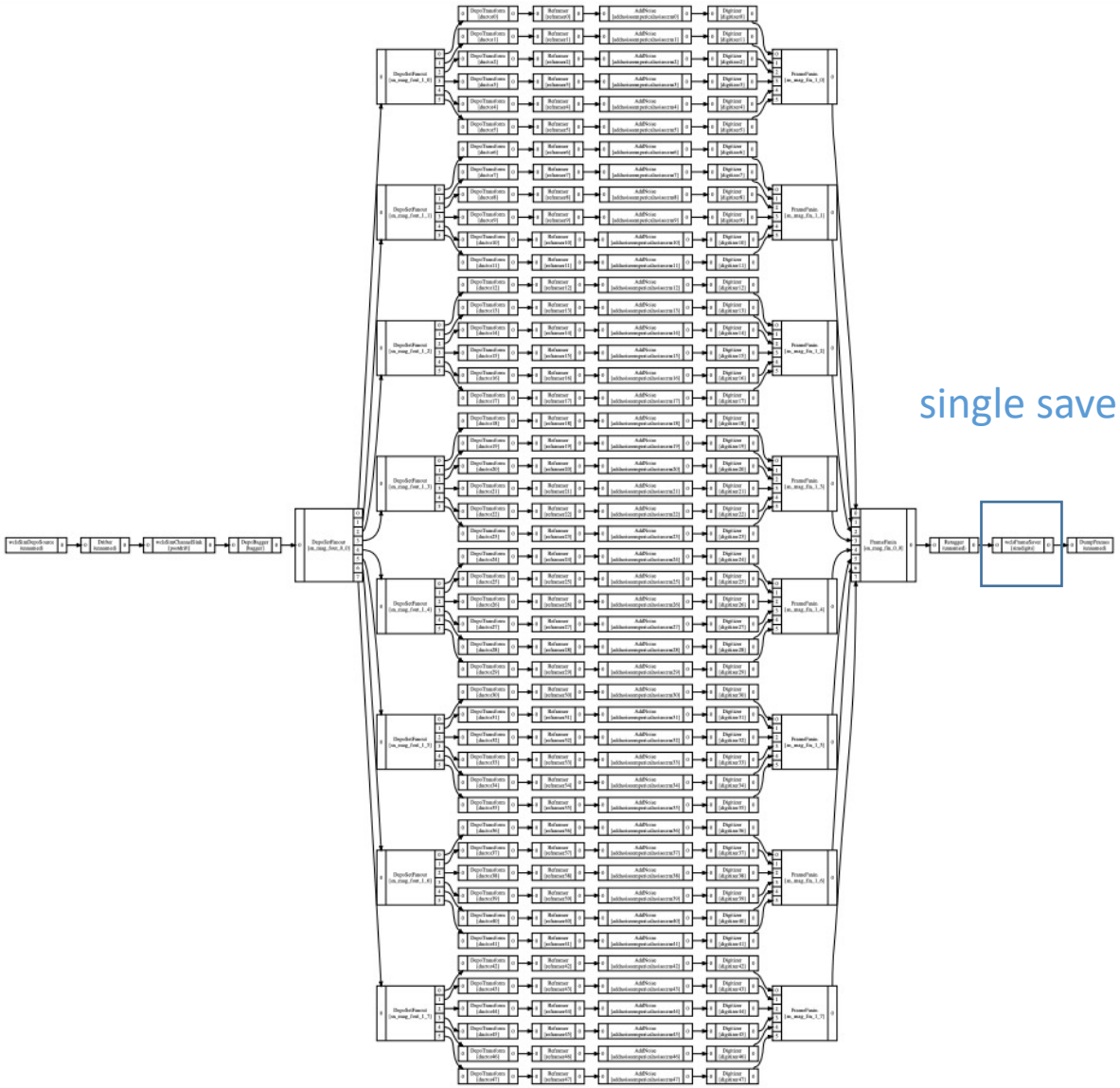
2^30

Current strategy is to combine all CRMs/APAs into one container

PROCESS NAME	MODULE LABEL..	PRODUCT INSTANCE NAME.....	DATA PRODUCT TYPE.....	.SIZE
GenieGen....	generator.....	.....	std::vector<simb::GTruth>.....	1
GenieGen....	TriggerResults	.....	art::TriggerResults.....	....-
GenieGen....	generator.....	.....	std::vector<sim::BeamGateInfo>.....	1
GenieGen....	rns.....	.....	std::vector<art::RNGsnapshot>.....	0
GenieGen....	generator.....	.....	std::vector<simb::MCTruth>.....	1
GenieGen....	generator.....	.....	art::Assns<simb::MCTruth,simb::MCFlux,void>.....	1
GenieGen....	generator.....	.....	std::vector<simb::MCFlux>.....	1
GenieGen....	generator.....	.....	art::Assns<simb::MCTruth,simb::GTruth,void>.....	1
G4.....	elecDrift.....	.....	std::vector<sim::SimChannel>.....	..379
G4.....	PDFastSim.....	.....	std::vector<sim::OpDetBacktrackerRecord>.....	..480
G4.....	rns.....	.....	std::vector<art::RNGsnapshot>.....	5
G4.....	largeant.....	LArG4DetectorServicevolTPCPlaneUInner	std::vector<sim::SimEnergyDeposit>.....	0
G4.....	PDFastSim.....	Reflected.....	std::vector<sim::SimPhotonsLite>.....	..480
G4.....	PDFastSim.....	.....	std::vector<sim::SimPhotonsLite>.....	..480
G4.....	IonAndScint...	.....	std::vector<sim::SimEnergyDeposit>.....	.2872
G4.....	largeant.....	LArG4DetectorServicevolTPCActiveInner	std::vector<sim::SimEnergyDeposit>.....	.2872
G4.....	TriggerResults	.....	art::TriggerResults.....	....-
G4.....	largeant.....	.....	std::vector<simb::MCParticle>.....	..692
G4.....	largeant.....	LArG4DetectorServicevolTPCInner.....	std::vector<sim::SimEnergyDeposit>.....	0
G4.....	largeant.....	LArG4DetectorServicevolTPCPlaneVInner	std::vector<sim::SimEnergyDeposit>.....	0
G4.....	largeant.....	LArG4DetectorServicevolTPCPlaneZInner	std::vector<sim::SimEnergyDeposit>.....	0
G4.....	PDFastSim.....	Reflected.....	std::vector<sim::OpDetBacktrackerRecord>.....	..480
G4.....	largeant.....	.....	art::Assns<simb::MCTruth,simb::MCParticle,sim::GeneratedParticleInfo>	..692
G4.....	largeant.....	LArG4DetectorServicevolTPCActiveOuter	std::vector<sim::SimEnergyDeposit>.....	0
wclssim.....	tpcrawdecoder.	daq.....	std::vector<raw::RawDigit>.....	41472
wclssim.....	tpcrawdecoder.	simpleSC.....	std::vector<sim::SimChannel>.....	41472
wclssim.....	TriggerResults	.....	art::TriggerResults.....	....-
wclssim.....	plopper.....	bogus.....	std::vector<sim::SimEnergyDeposit>.....	..713

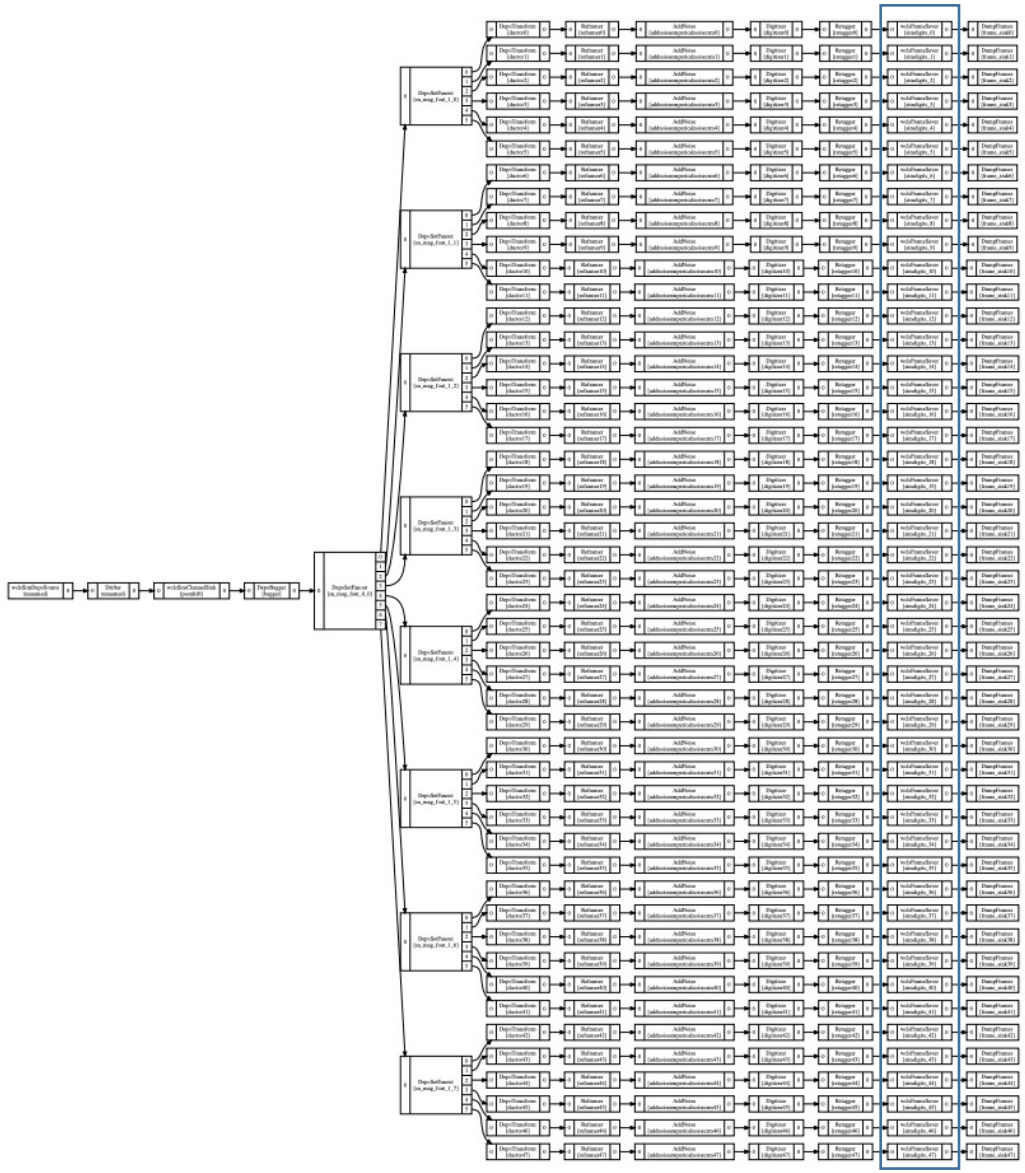
Wire-Cell: Use multiple FrameSaver?

Vertical Drift 1x8x6



single saver

multi saver



art::Event::put() is thread safe

Kyle's Talk:
https://indico.fnal.gov/event/20453/contributions/57777/

Module example

- We want to create a track from a collection of hits

```
void TrackMaker::produce(art::Event& e)
{
  auto const& hits = e.getValidHandle<Hits>(tag_);
  unique_ptr<Track> track = trackFromHits(*hits);
  e.put(move(track));
}
```

- Assuming `trackFromHits` does not update any state, then this `produce` function is thread-safe—i.e. it can be called concurrently with different `art::Event` objects.
- Why?
  - *art* guarantees that product retrieval and insertion is thread-safe
  - the produce function above modifies no state of the TrackMaker object

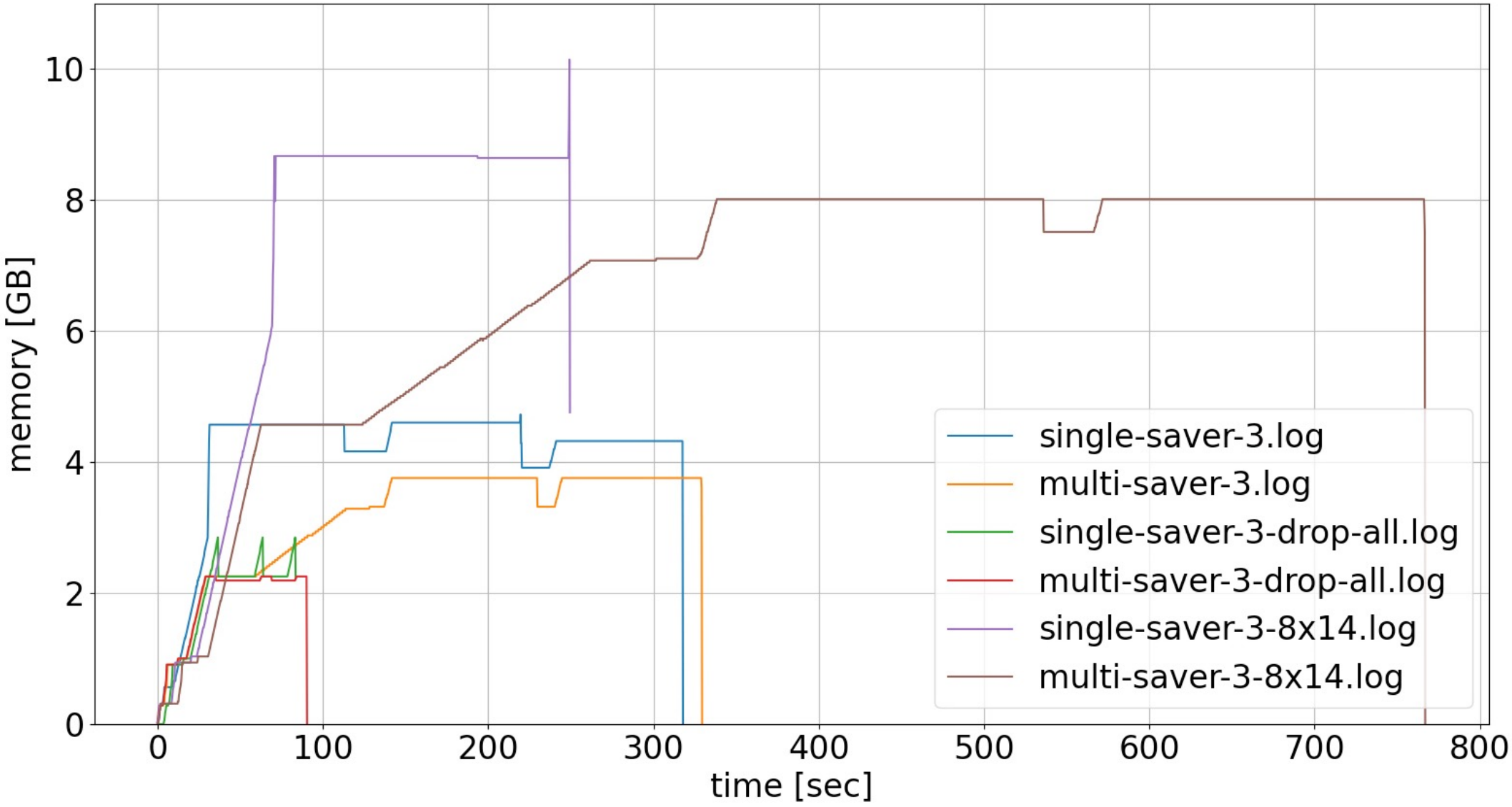
Fermilab

57 6/25/19 Kyle J. Knoepfel I LArSoft Workshop 2019

Example of multi- FrameSaver

wclssim.....	tpcrawdecoder.	daq18.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq16.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq14.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq12.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq10.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq46.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq25.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq44.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq27.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq42.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq21.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq40.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq8.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq23.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq6.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq4.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	simpleSC.....	std::vector<sim::SimChannel>.....	41472
wclssim.....	tpcrawdecoder.	daq29.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq2.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq0.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq32.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq30.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	TriggerResults	.....	art::TriggerResults.....	....-
wclssim.....	tpcrawdecoder.	daq36.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq34.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq38.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq35.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq37.....	std::vector<raw::RawDigit>.....	..864
wclssim.....	tpcrawdecoder.	daq31.....	std::vector<raw::RawDigit>.....	..864

Test with 3 events simulation VD-1x8x6, VD-1x8x14



Thoughts on scaling up to full DUNE FD

Holding lossless LArTPC noise from all APAs in **one art::event/std::vector** will create memory and ROOT IO issues when scaling up

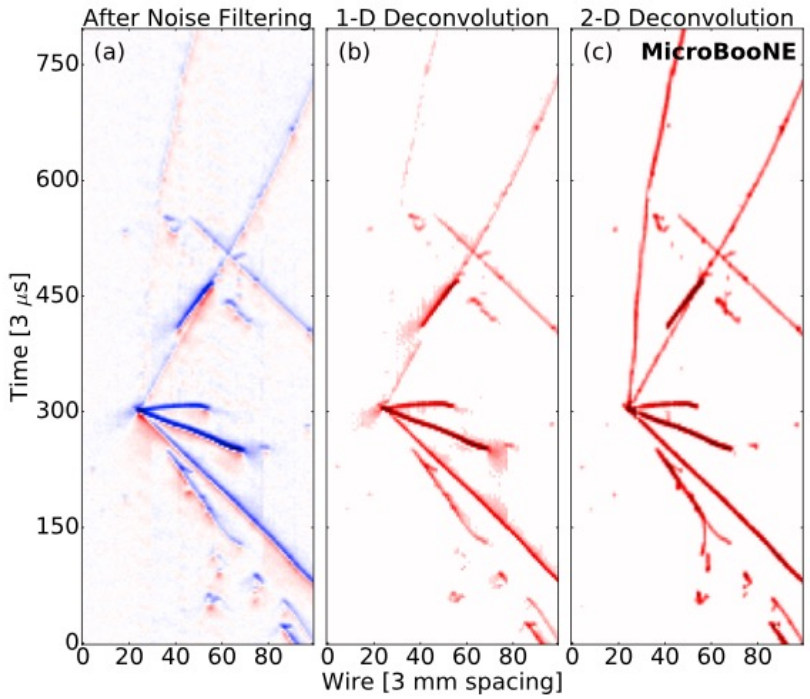
- Per APA vector could resolve the ROOT IO issue

LArTPC handlings (sim, sigproc, etc) are independent between APAs

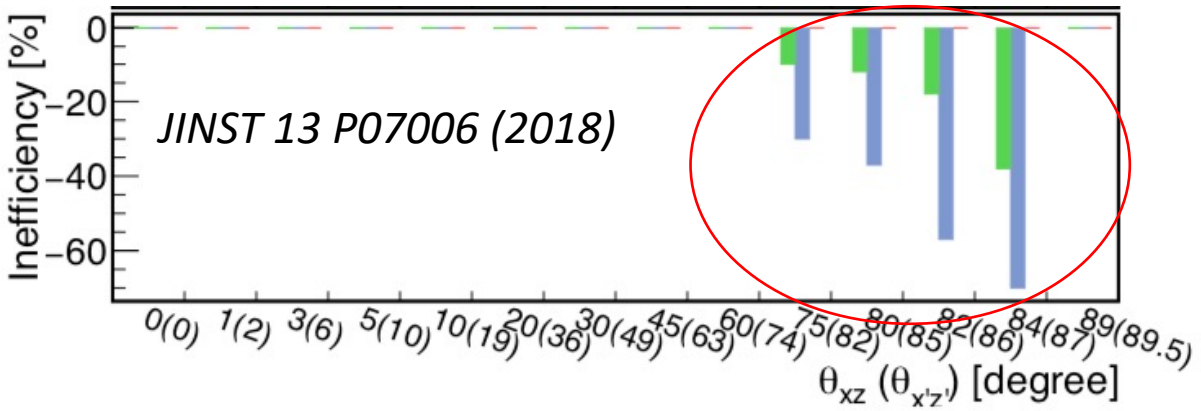
- The issue seems to be that art needs a whole event loaded in memory – can we load partial event in art?
- Wire-Cell Toolkit (Data Flow Programing) can process APAs independently while can do aggregation when needed – an alternative way of handling noise transient/persistent?

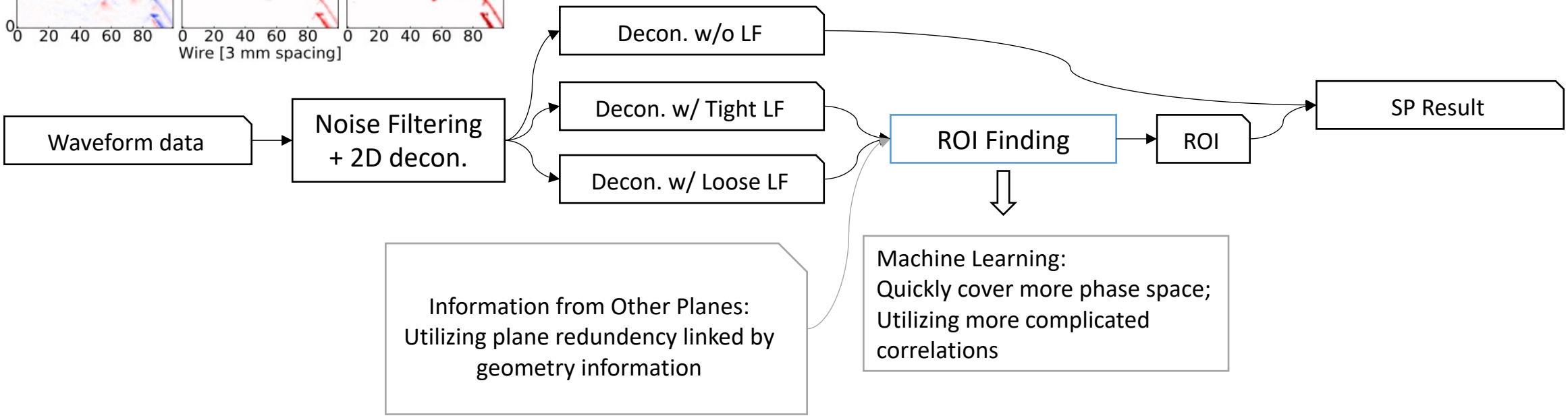
Improving Signal Processing with Deep Learning and Multi-Plane Information

JINST 13 P07006 (2018)



MicroBooNE simulation evaluation

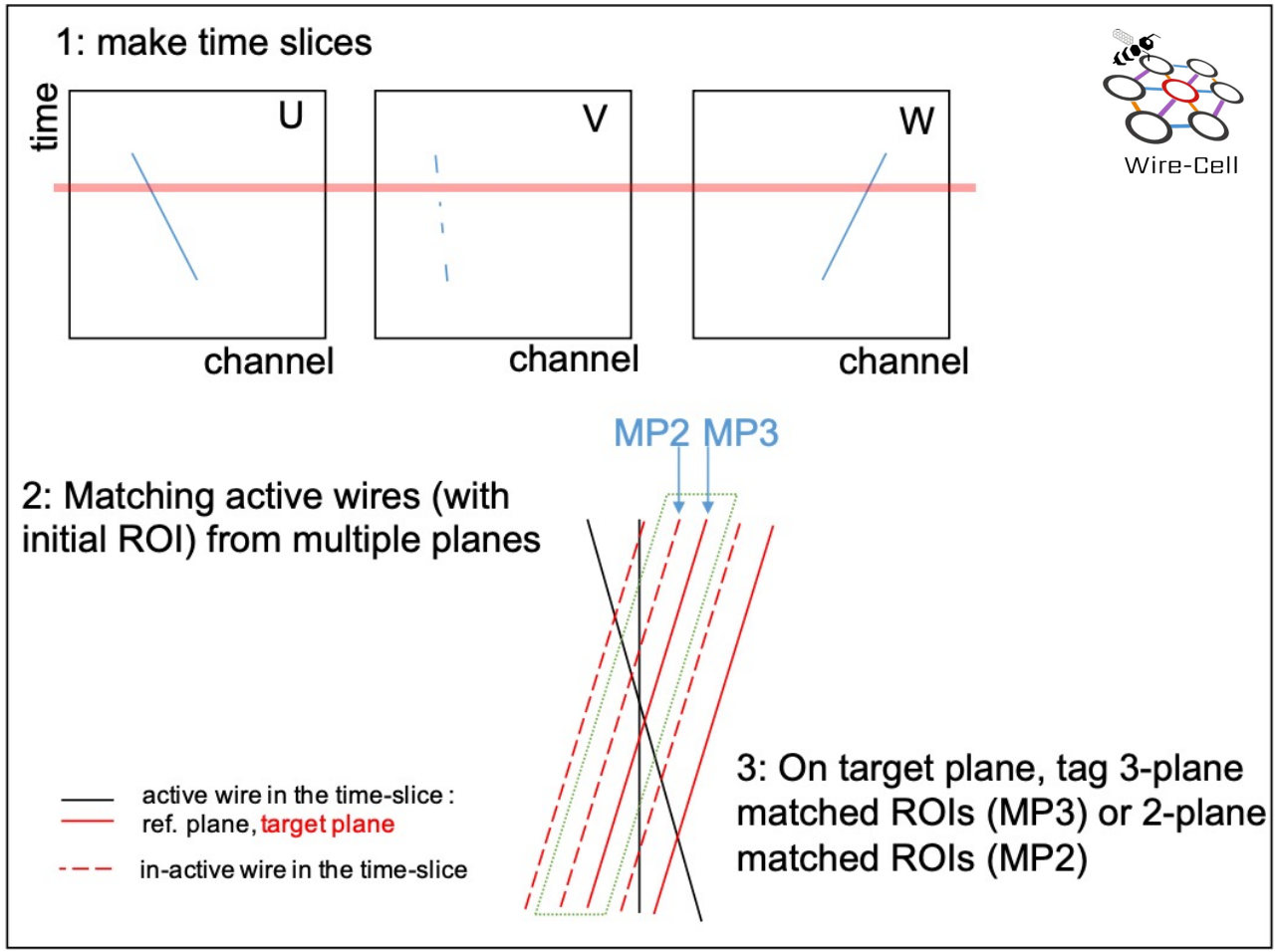




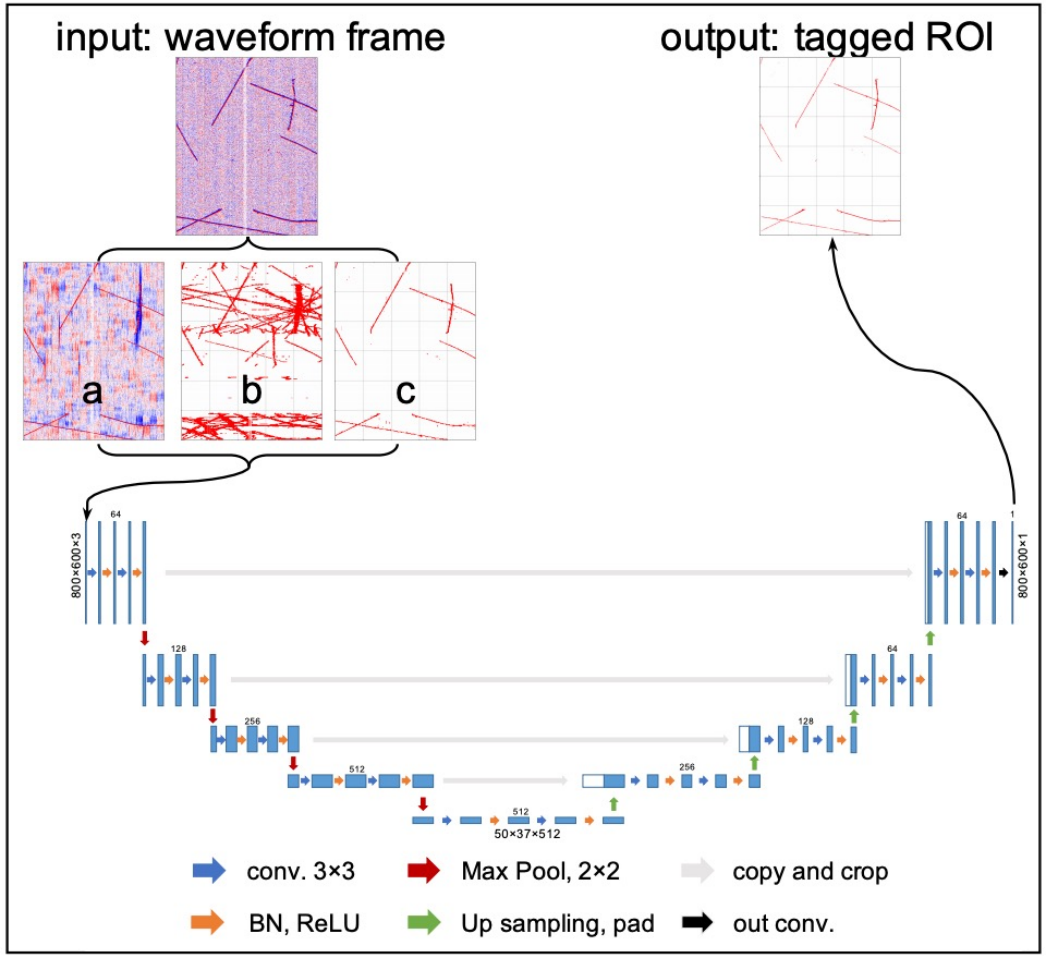
DNN ROI finding with 3-plane information

JINST 16 (2021) 01, P01036

Multi-plane information in Signal Processing

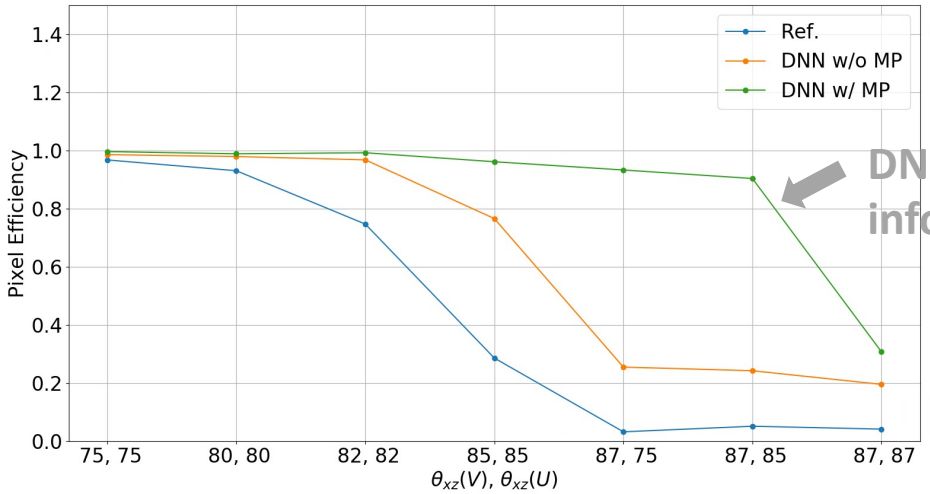


DNN ROI finding with multiple input channel

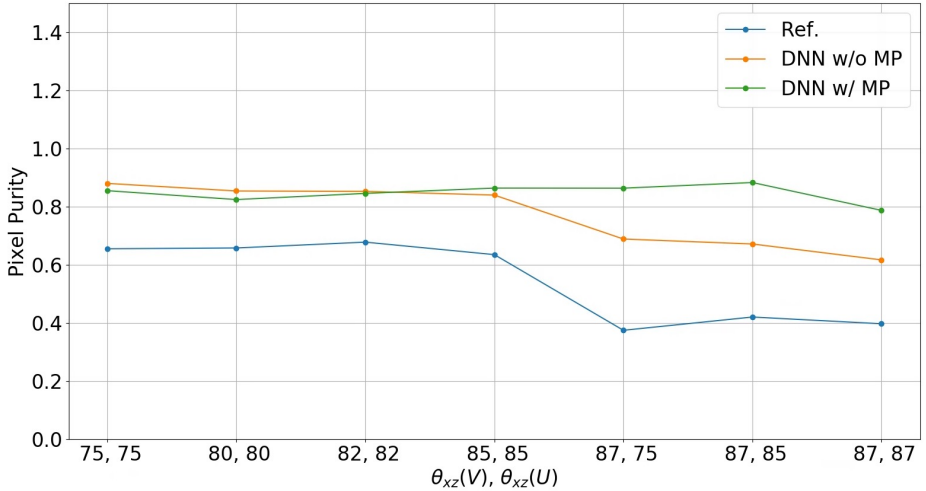


DNN ROI finding with 3-plane information (II)

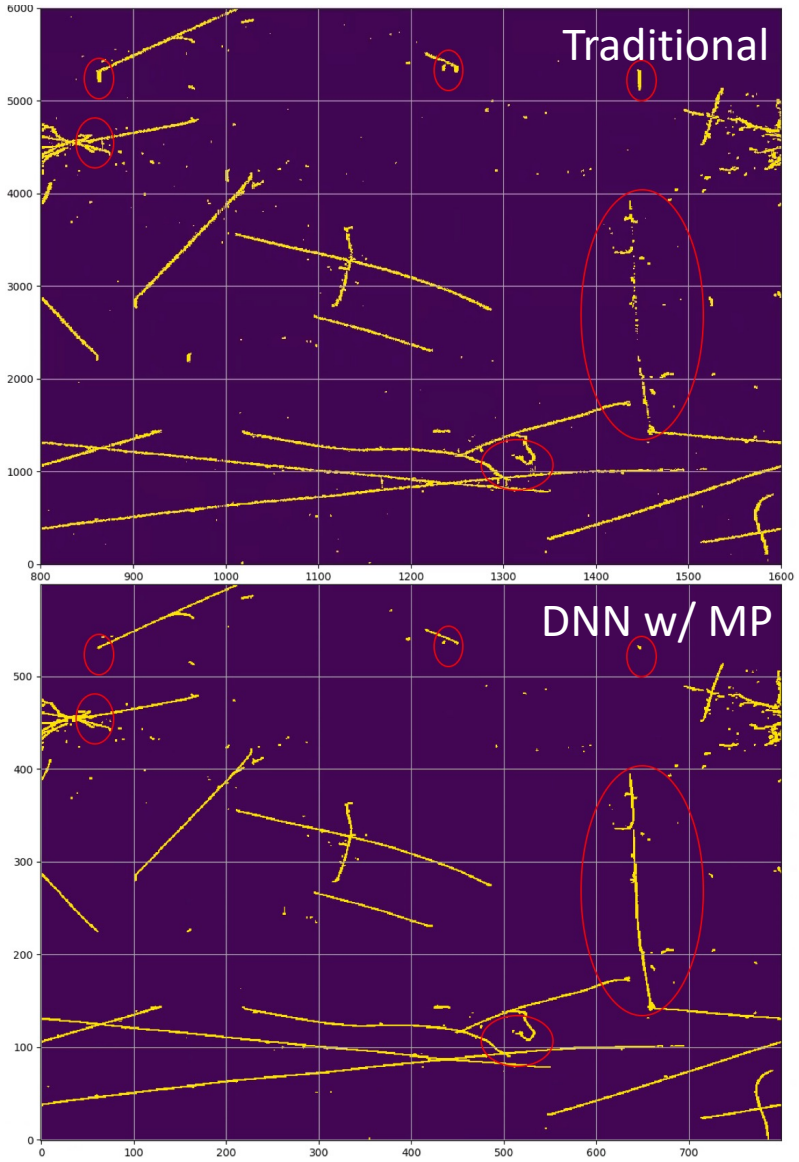
ProtoDUNE simulation
ROI finding on V plane (2nd induction)



DNN With 3-plane information



tested on ProtoDUNE data



Progress with DNN-ROI finding in Wire-Cell Signal Processing

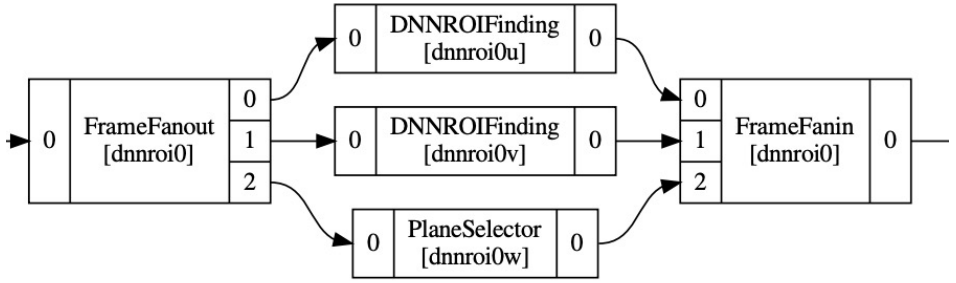
Finished full configuration for full workspace

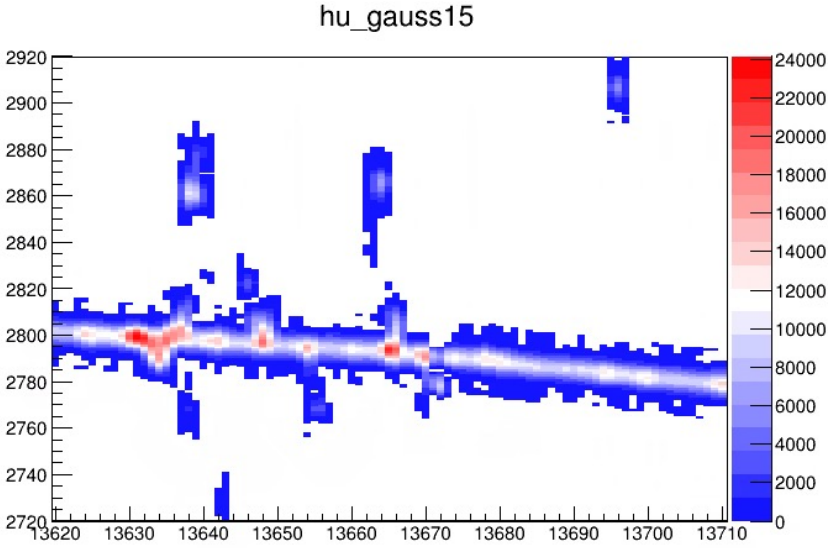
Semaphore type service for better resource control – B. Viren

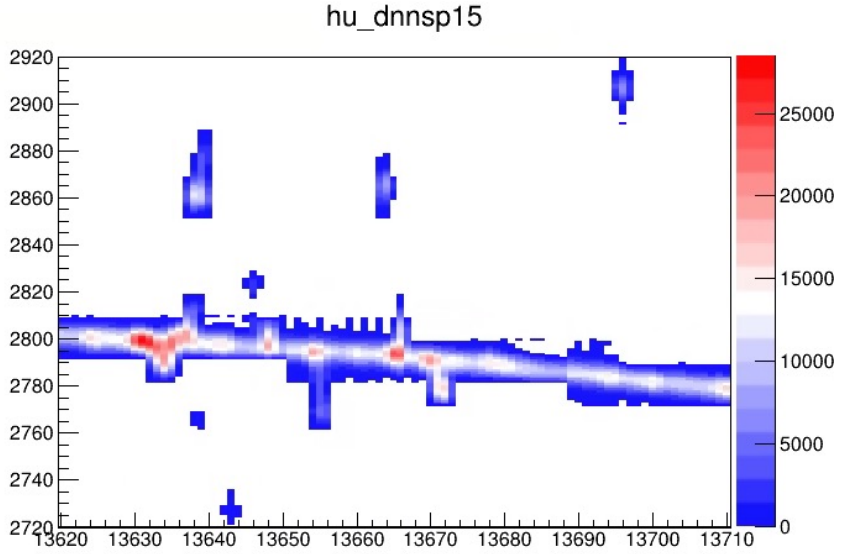
- semaphore: counting lock

Previous model trained on PDSP seems still working on Vertical Drift

- More tests and dedicated training on-going





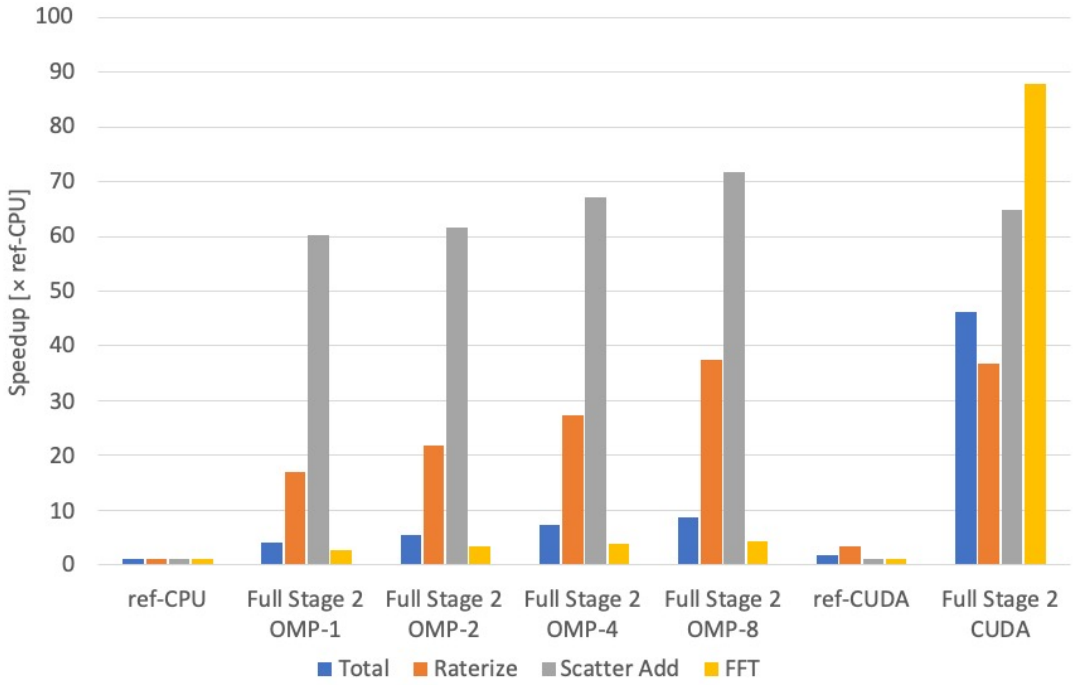


Portable Parallelization development for Wire-Cell LArTPC simulation

- Heterogenous computing (GPU, FPGA) could significantly speed up some algorithms.
- Some future HEP computing facilities are likely to have accelerators.
- Writing code with portable performance across multiple architectures could be beneficial
  - Kokkos, SYCL, Alpaka, OpenMP/ACC, Parallel STL
  - HEP-CCE-PPS

Wire-Cell 2D Conv. Simulation Kokkos Porting (unfinished) Speedup

- H. Yu, Wire-Cell PPS talk at vCHEP21
- Intel i9-9900K, NVIDIA RTX 2080Ti



Full porting finished recently!
Kokkos impl. vs. CPU-ref