ProtoDUNE/SP Data Reduction Software & Computing Aspects

Brett Viren

Physics Department

BROOKHAVEN
NATIONAL LABORATORY

Reco
2017 Feb 22

Review of Data Reduction

Main idea:

1. Run the required, initial data processing steps:
    1. ADC mitigation (stuck-codes, non-linearity)
    2. Excess noise filtering (if any, fingers crossed)
    3. Signal processing
       (detector response deconvolution and ROI selection)
2. Remove unnecessary (at this point) oversampling.
3. Pack and compress data and save to file.

→ Then, use reduced files as input the remaining data processing.

Reference:

- Xin's talk last meeting
- DocDB 2089 proposal details

Performance Conclusions

- pD/SP can achieve design 4× compression even with MB-like[1] excess noise and $6 - 8\times$ with expected noise levels.
- 150× reduction achieved with just signal-ROI selection.
- 400× reduction by further reclaiming over-sampling
  - using safe rebin-3, can explore more aggressive rebin-4, ...
- Relies on saving reduced output data with:
  - 32 bit in-memory samples re-digitized to 16 bit (can optimize with "dynamic dynamic-range")
  - ROOT compression (negligible CPU even at "level 9")
- Time dominated by signal processing step (required anyways) → 45 seconds / APA / trigger

2.5 PB → 6.5 **TB** (TPC only)

[1] DocDB 2089: estimates made using actual reduction algorithms on a few MicroBooNE events and scaling by number of channels.

Validation with Simulation

- Confirm:
  - → reduction efficacy,
  - → CPU performance and
  - → correctness of output.
- Use new, realistic simulation:
  - Developed as part of the Wire Cell Toolkit.
  - Proper long-range induction and inter-pitch field variance.
  - Noise models from MicroBooNE measurements and "first principles" calculation from Milind that can explore different assumptions.
- Validate non-effect of reclaiming over-sampling.
  - 3-bin sum is safe, 4-bin is on edge.
  - Already demonstrated with toy Gaussian - try it yourself!
  - O.w. win Nobel for disproving Nyquist!

Software Parts Needed for Reduction Process

- Raw"Data Access Library" (DAL)
  - Provided by DAQ "fragment experts", used by many.
  - Fragment **unpackers** needed for reduction process (among others).
  - Fragment **packers** also needed for simulation.
- "Keep up" data reduction processing system
  - Needs job management system (p3s could work).
  - Monitoring of jobs needs integration into shift operations.
- Actual implementation
  - Needed modules: ADC mit., noise filt., sig. proc. and reduced DAL.
  - Most parts already exist in MB, 35t and Wire Cell.
  - Won't know real excess noise types/levels until turn-on.
  - Code needed in reduction process **and** general offline jobs.
  - Noise filt. and sig. proc. is in Wire Cell Toolkit, integration into art/LArSoft is in progress.

ProtoDUNE Keep Up Processing

What does it take to keep up with the protoDUNE data?

- Cycle avg: 10 Hz × 6 APA × 45s/APA = 2700 cores
  - 6750 cores to keep up during spill (25 Hz)
- 20 cores / node ⇒ need 135 nodes
  - Depending on LArSoft RAM, may need as much as 64GB/node!
- Computing environment needs:
  - → dedicated CPU, but only during actual running.
  - → new jobs triggered by new data.
  - → quick, high b/w access to raw data (EOS is a good source).

Nominal request:

∼200 nodes of ∼20 cores and ∼48 GB each

Can we find this?

N.B: full reco takes ∼ 6× more CPU-time. Any delay we accept for reduction, we accept 6× more delay for final result!

Status and To Do

- Initial improved detector response simulation in WCT. Next:
  - implement correct drift physics statistics
  - implement noise models
  - file interface and LArSoft integration
- Integrate noise filter and signal processing. Next:
  - Port signal processing from Wire Cell prototype to toolkit
  - Integrate sig.proc. into LArSoft.
  - Move both noise filt. and sig. proc. to use Art “tools”
- DALs
  - Work with DAQ group on Raw DAL
  - Develop “reduced” DAL

Possible Opportunities

A keep up reduction process opens some possible opportunities:

- Does 400× reduction in TPC data make PDS data dominate data volume?
  - Do we need/want to reduce PDS data?
- Are there other low-CPU, high-I/O algorithms which are convenient to run in coincidence with data reduction?
  - Maybe build data indices?
  - Maybe repack/reformat for faster read-in later?
- Merge the out-of-band Beam Information?
  - There is a Beam Info data stream latency to worry about. (I've heard 20-40 minutes)
  - Will merging/syncing in this job be over-complicated?
  - Maybe wait and merge BI using data-reduced output? At 6.5 TB (for TPC only) we can afford a duplicate copy!