Status of ND-LAr Samples and Systematics

Andrew J. Mogan
on Behalf of ND-LAr Group

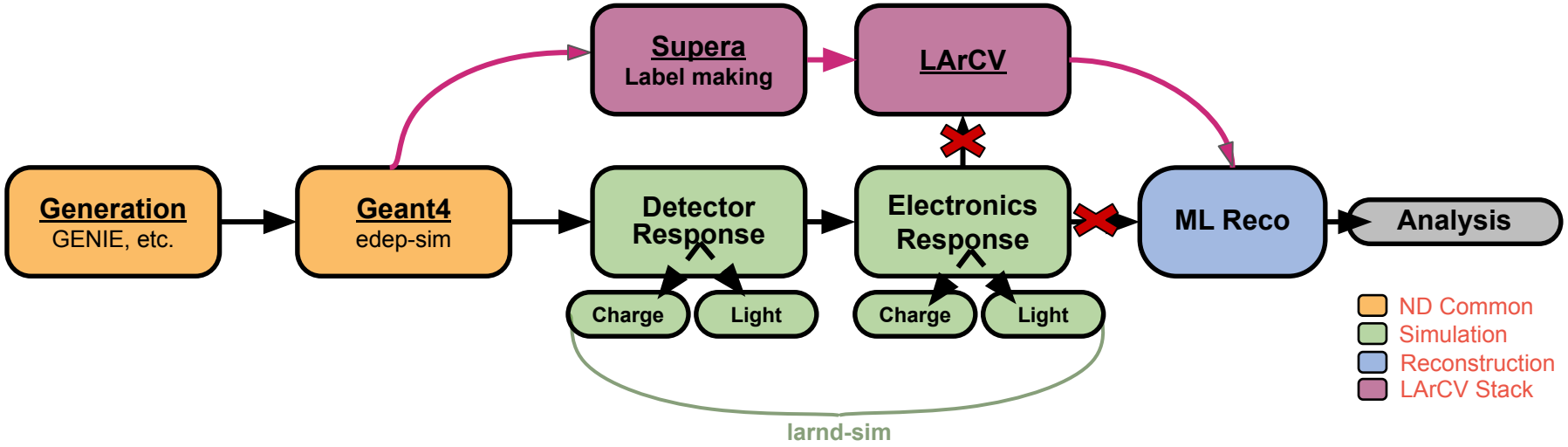
DUNE Oscillation Analysis Workshop

8/15/22

CSU

The Challenge of End-to-End Sim/Reco

- Full end-to-end reconstruction with the machine learning reconstruction (ML reco) requires larndsim output to be compatible with larcv
- larcv looks for ROOT TTrees containing specific objects which larndsim does not create

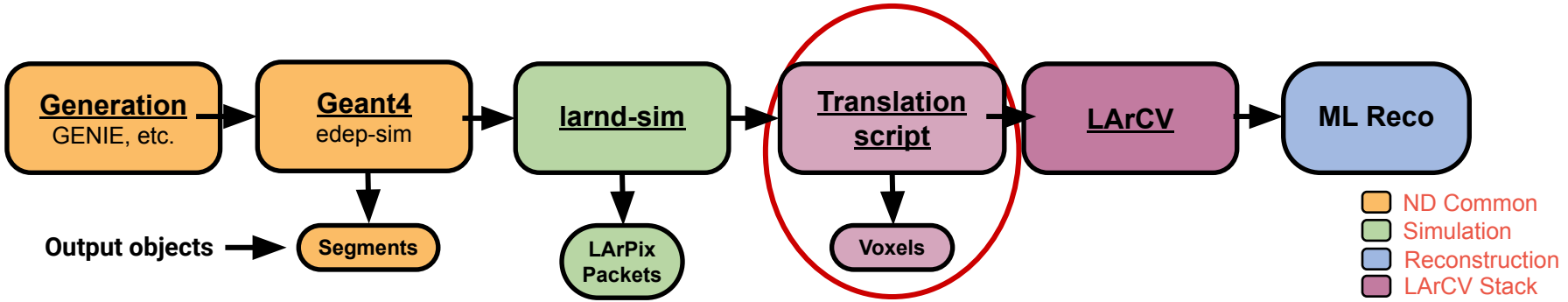


Previously on ND-LAr...

- Detector systematics handled in an ad hoc way
  - Waveforms smeared directly from edep-sim
  - Detector simulation not connected to machine learning reconstruction (ML reco)
- Now, we seek to do things the “right” way
- One of the primary goals for this workshop is to make progress toward connecting detector simulation and ML reco
  - Requires full, end-to-end production chain

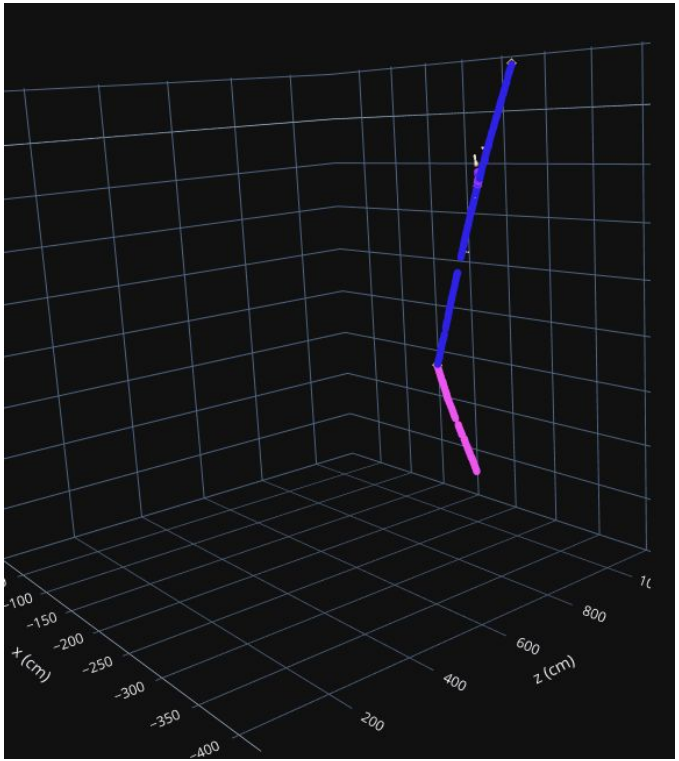
What's Been Done So Far

- Translation script takes the output of larndsim and creates objects needed for larcv
  - Converts LArPix packets to larcv::Voxel() objects
- Current version is very ad hoc…working on a more robust version now

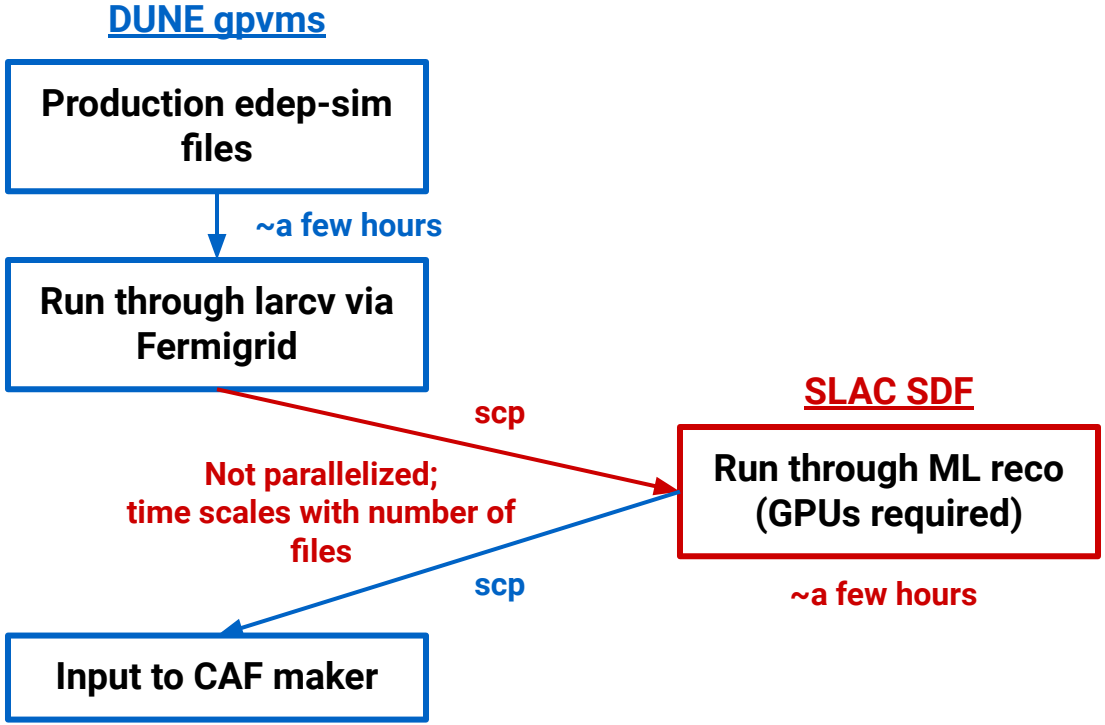


Translating larnd-sim to larcv

- We've shown a few events successfully reconstructed using the tentative translation script (right)
- Larnd-sim is written entirely in python…no LArSoft
- Outputs numpy arrays in .h5 format
- Supera can translate LArSoft->larcv, but not larnd-sim->larcv (yet)
- More input-agnostic version of Supera now exists: SuperaAtomic
- Current task is to write some python module that interfaces with SuperaAtomic and larnd-sim and translates between the two
  - Previous translation script did not use Supera



ML Reco Production Workflow (for now)



File Transfer Time

- scp transfers at about 35 MB/s
  - For ~10 GB file, that's about 5 minutes
  - For ~100 files, that's about 8 hours
  - Production samples are *much* larger than 100 files
  - This is not long-term feasible

```
amogan@sdf-dtn01:~$ scp /sdf/group/neutrino/amogan/data/dune_nd_production_2022_v2/reco_output/ND-LAr_TMS_SAND_TDR_FHC/neutrino.*reco*.npz amoga
@dunegpvm11.fnal.gov:/pnfs/dune/scratch/users/amogan/larcv/dune_nd_reco_prod_Aug2022_workshop/
neutrino.0.reco.npz                                                                          18% 2102MB  35.7MB/s   04:24 ETA
neutrino.0.reco.npz                                                                         100%   11GB  35.9MB/s   05:20
neutrino.10.reco.npz                                                                        100%   11GB  35.9MB/s   05:22
neutrino.11.reco.npz                                                                        100%   11GB  35.7MB/s   05:25
neutrino.12.reco.npz                                                                        100% 9791MB  36.1MB/s   04:31
neutrino.13.reco.npz                                                                        100%   11GB  36.0MB/s   05:20
neutrino.14.reco.npz                                                                        100%   12GB  36.0MB/s   05:36
neutrino.15.reco.npz                                                                        100%   11GB  35.9MB/s   05:24
neutrino.16.reco.npz                                                                        100%   11GB  36.1MB/s   05:13
neutrino.17.reco.npz                                                                        100%   11GB  36.1MB/s   05:15
neutrino.18.reco.npz                                                                        100%   12GB  36.2MB/s   05:27
neutrino.19.reco.npz                                                                        100%   11GB  36.1MB/s   05:07
neutrino.1.reco.npz                                                                          10% 1187MB  34.3MB/s   04:43 ETA
```

What's Available Now

- Intention was to have 100 files with ~3000 events each
- Some hiccups along the way…
  - Run information got scrambled due to "user error," so these were basically unusable
- Chris M. managed to salvage 10 files for CAF making:
  `/dune/data/users/marshalc/lblworkshop_cafs/`
- Most recent production has 74 files with 222,605 events total
  - `/pnfs/dune/persistent/users/amogan/larcv/lartpc_mlreco3d_output/`
  - Grid inefficiencies and file I/O errors when copying from SDF…not yet investigated
- One of our immediate-term priorities is to streamline this production pipeline
  - Move everything to SDF?
  - QMUL GPUs from Fermigrid?
  - Tolerate the Wilson clusters limited GPU capacity?
  - More efficient copy to/from SDF? (bbcp, Globus, etc.)
- New DUNE Slack channel: #nd-lar-ml-reco-prod

CAF Making

- Note: I am not an expert on CAF making (that's why I'm here!)
- ND CAF maker takes the ML reco output files as input
- ND CAF maker tutorial from December 2021 here
  - Naturally, this is somewhat outdated
  - The CAFs pointed out in this tutorial are now obsolete
- Some proposed updates to ND CAF maker here
  - Also outdated…

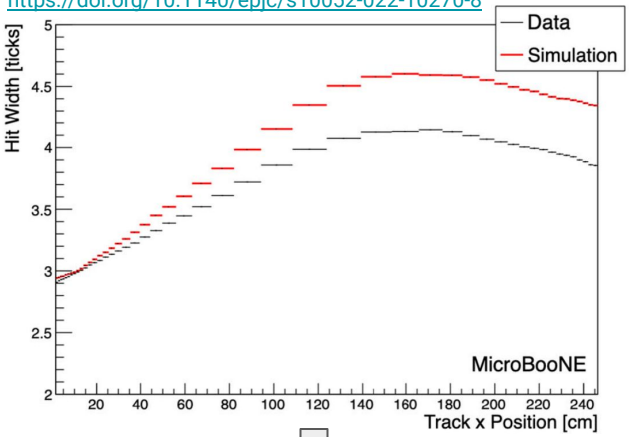
Thinking Ahead: Detector Systematics

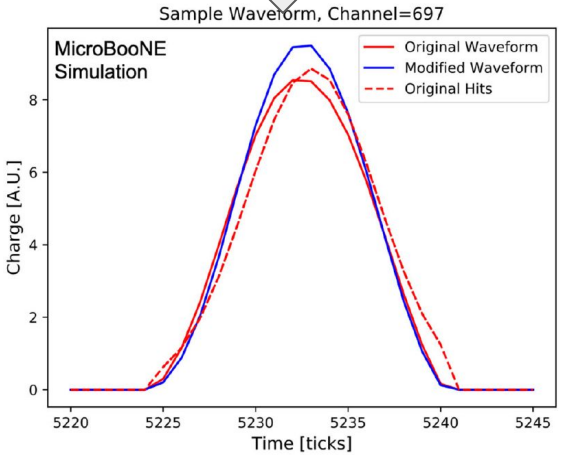
- For DUNE to meet its proposed physics goals, a robust suite of systematic uncertainty evaluations is necessary
- Typical detector systematics flow:
  - Vary some physics parameter in simulation (diffusion, lifetime, recombination, etc.)
  - Variations may impact signal waveforms (e.g. E-field distortions will not)
  - Run modified waveforms through reconstruction, examine impact
  - Construct covariance matrix, etc.
- Recent alternative methods use some form of parametrization on simulated events to avoid re-running detector simulation
  - See for example the recent publication on MicroBooNE “wire modification” method

Detector Systematics

- The primary motivation for connecting simulation and reconstruction
- Historically, detector systematics are evaluated by varying parameters in simulation and examining the impact on high-level quantities
  - This involves re-running detector simulation which is *very* computationally expensive
- Newer methods may involve parametrizing detector effects
  - See new MicroBooNE publication on "wire modification" technique
  - Parametrization extracted from waveform comparisons between data and MC

https://doi.org/10.1140/epjc/s10052-022-10270-8

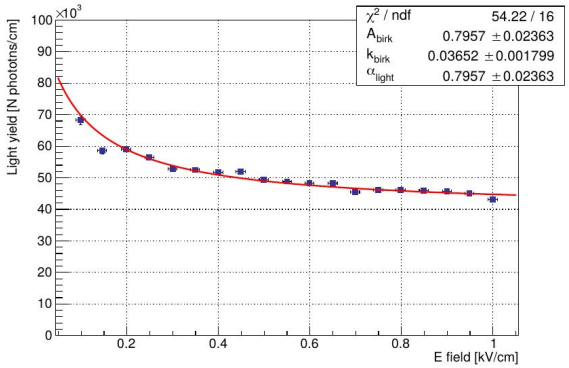


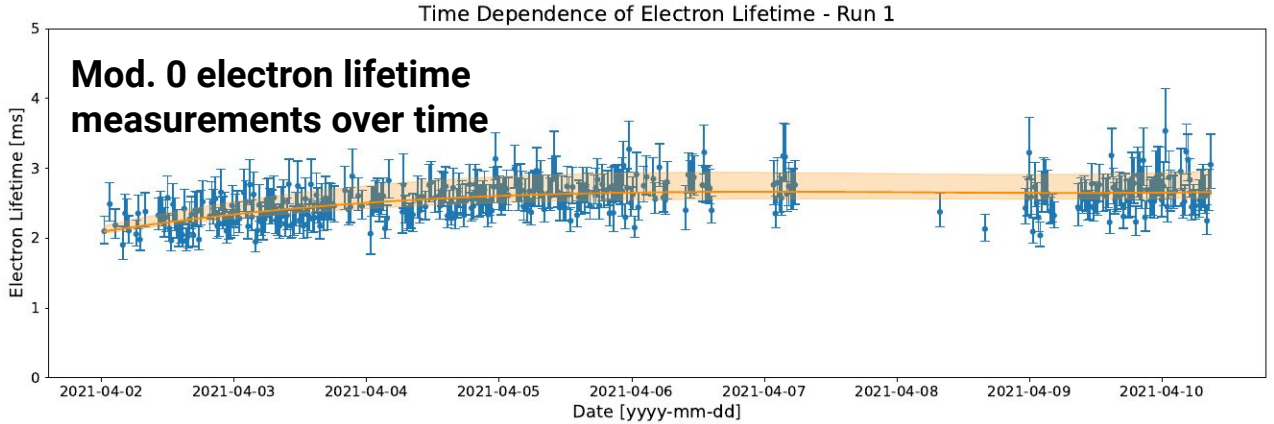


Detector Parameter Variations

- We can use Module 0/1 data to extract reasonable variations for some parameters
  - Electron lifetime, recombination model, light yield, etc.
- Electron diffusion parameters can be estimated from recent measurements

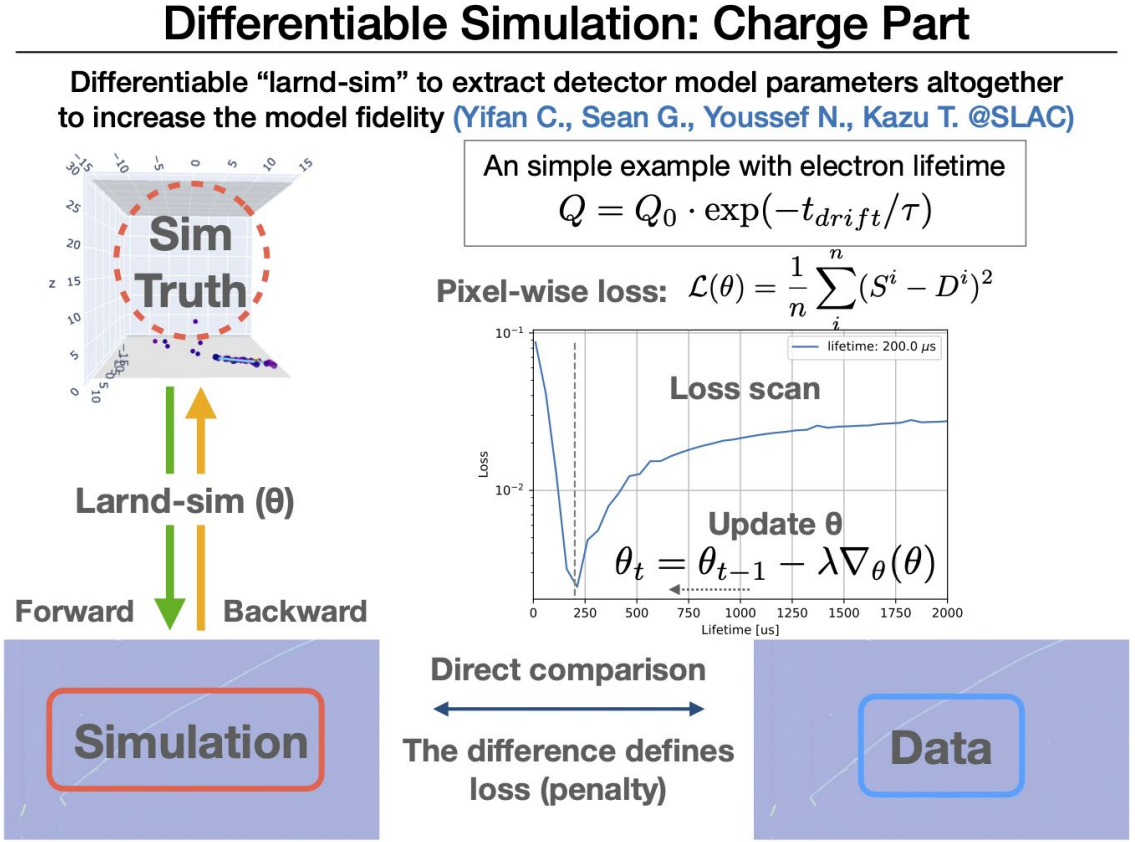
Mod. 0 Light yield vs. E-field





Differentiable Simulator

- Work underway from SLAC team
  - Caveat: I am not at all an expert on this
- Use gradient descent to extract model parameters
- Proof-of-principle on electron lifetime
- Possible method of automated parameter tuning



Summary and Goals for this Week

- Exercise the (tentative) ML reco production chain
  - Feedback will be very helpful
  - Already caught one bug with file run numbers
- Work on connecting larnd-sim to ML reco
- Immediate priority is to streamline the production chain
  - We recently got some new graduate students interested in this!