UPDATE ON SIMULATION ENERGY DEPOSITIONS

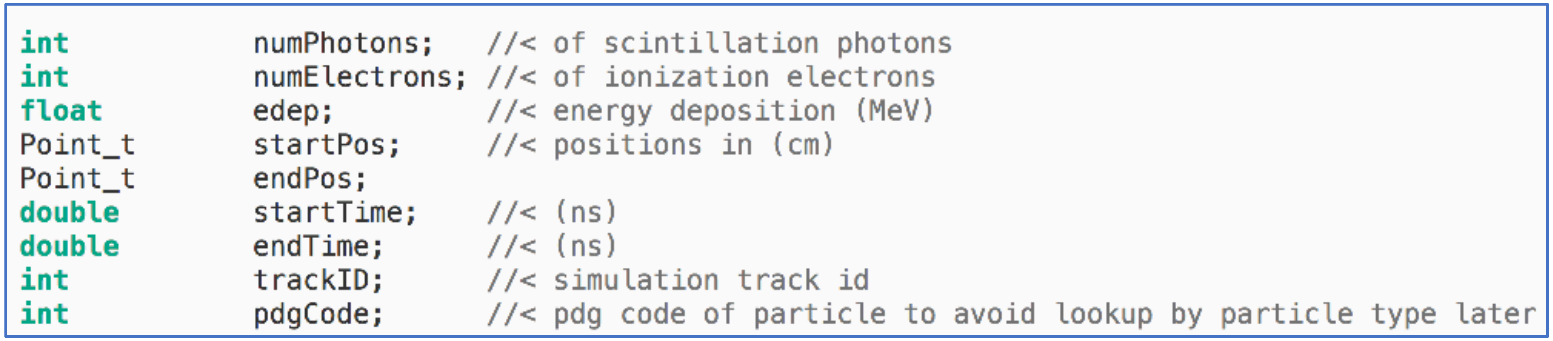
Wesley Ketchum (FNAL)

REVIEW

- Needed to add option for storing EnergyDepositions in LArG4 step for uboone production release
    - *Previously reported on idea/status here*
- **Critical points:**
    - Define energy deposition object
    - Hooks into LArG4 for storing energy deposits
    - Downstream modules to do electron/photon propagation
- **Talk today**
    - Review what has now been merged to develop
        - *feature/wketchum_LArG4Refactor_develop on lardataobj and larsim*
    - Discuss what is needed “operationally” for experiments

Energy Deposition Object

- Using the object created by Bill and Hans for LArG4 refactoring project
- Added pdg code of particle
  - *Enable scintillation by particle type without lookup to MCParticle list*



```
int             numPhotons;   //< of scintillation photons
int             numElectrons; //< of ionization electrons
float           edep;         //< energy deposition (MeV)
Point_t         startPos;     //< positions in (cm)
Point_t         endPos;
double          startTime;    //< (ns)
double          endTime;      //< (ns)
int             trackID;      //< simulation track id
int             pdgCode;      //< pdg code of particle to avoid lookup by particle type later
```

  - *Side note: this now lives on several branches of lardataobj I think, so we're going to have to be careful about merges/conflicts*

HOOKS IN LARG4

- Energy depositions are being tracked and stored in OpFastScintillation process
  - *This is a change from “LArVoxelReadout”*
  - *Requries “FastOptical” physics to be enabled*
  - *Presumably this changes with LArG4 refactoring*
- Important: energy depositions stored in different instance labels by parent volume in current (MicroBooNE-centric) implementation
  - *If G4 volume name contains “TPCActive”, stored in “TPCActive” instance label*
  - *If it doesn’t, stored in “Other” instance label*
    - →Cryostat
- How do other experiments want this?
  - *Hint: some instance labeling on which TPC and which cryostat?*

NEW OPTIONS IN LARG4 PARAMETERS

- FillSimEnergyDeposits
  - *This turns on/off storing the energy deposits, as described*
  - *Default is false (don't do it)*
- NoElectronPropagation
  - *This turns on/off doing the electron propagation in the default way, inside LArVoxelReadout and producing SimChannel collection*
  - *Default is false (leave it on)*
- NoPhotonPropagation
  - *This turns on/off doing the photon propagation in the fast optical physics in the normal way and producing SimPhotons collection*
  - *Default is false (leave it on)*

ADDITIONAL UNRELATED OPTION!

- Added “SparsifyTrajectories” option in LArG4 ***module***
- Calls “SparsifyTrajectories()” function for MCParticles put into the output collection
  - *Calls “Sparsify(0.1)” on MCTrajectory underneath, which removes interior trajectory points that lie within some tolerance of neighboring points (0.1→1 mm)*
- Significantly (factor ~2?) reduces output MCParticle collection size

- Improvement: allowing the tolerance desired to be passed in as an option
  - *Requires new nusimdata release*

Ionization and Scintillation

- I do not calculate ionization electrons and scintillation photons in the LArG4 stage
  - *Set to "-1" on both*
- I have separated out ISCalculationSeparate into a standalone algorithm in larsim/IonizationScintillation
  - *I've dropped the ISCalculation base class here … so it's a single purpose standalone algorithm*
  - *Should be turned into art::Tool for future*
- This algorithm will get used in electron and photon propagation
  - *Initialized with LArProperties (scint yield), DetectorProperties (Efield), SpaceCharge, and LArG4Properties (recombination)*

MODIFIED ENERGY DEPOSITIONS

- WireCell simulation does not incorporate space charge → must be handled externally
  - *Correlated apparent position offsets and differences in recombination*
- New module in ElectronDrift/ShiftEdepSCE_module.cc
  - *Input: edeps*
  - *Output: edeps with shifted positions based on space charge service and n_electrons/n_photons filled from ISCalculationSeparate Alg*

ELECTRON PROPAGATION

- Bill Seligman wrote a refactor of the electron drifitng that was being done in LArVoxelReadout
  - *larsim/ElectronDrift/SimDriftElectrons_module.cc*
- I’ve modified it only slightly
  - *Use the ISCalculationSeparate algorithm*
    - Note: should do an option for using electrons/photons in object if they’re there…
  - *Ignore energy deposits behind the first induction plane (as was being done before…)*
  - *Stripped anything for associations*
- ***This should not use modified energy depositions***
- Output is SimChannels, which should be same as previous simulation

PHOTON PROPAGATION

- I wrote new module to do photon propagation via visibility library
  - *larsim/PhotonPropagation/ PhotonLibraryPropagation_module.cc*
  - *Uses ISCalculationSeparate alg to get number of photons, and then applies library*
  - *Options for inputting RiseTime*
    - Fast and slow risetimes currently set to -1.0 by default (→ no rise time)
- ***This should not use modified energy depositions***
- Output is SimPhotons

NOTES ON ALL THESE MODULES

- These were made and validated against v06_26_01_XX larsoft with MicroBooNE
- **Things lacking/potentially at issue**
  - New added modeling/complications in photon simulation that I have not included in the standalone module
  - BackTracker for photons and charge won't work well without options to specify input labels
    - *BackTracker assumes same module label for MCParticles and SimChannels/SimPhotons*
    - *Photon BackTracker completed untested*
  - Overall testing for other experiments
    - *I assume it won't work out of box and will need some tests/tweaking*

POINTS FOR DISCUSSION

- How will energy depositions be stored/labelled/handled for best use downstream
- What are the additional workflows we will pursue?
  - *Additional optical simulation, like Cherenkov light*
  - *Moving energy depositions inside or among detectors?*
    - Would require rerunning G4 when energy deposition “leaves” volume
  - *Mixing modules?*
- How does this properly evolve under other assumptions?
  - *DUNE SP?*
  - *Dual-phase?*