

THE OPTICAL DETECTOR: A MICROBOONE PERSPECTIVE

WESLEY KETCHUM (LANL)

CAVEATS

I am not the expert

- I didn't design the code
- I didn't write the code
- I didn't test the code (rigorously)

I know very little, beyond basic details, on the simulation side

SIMULATION

Get scintillation photons from LArG4

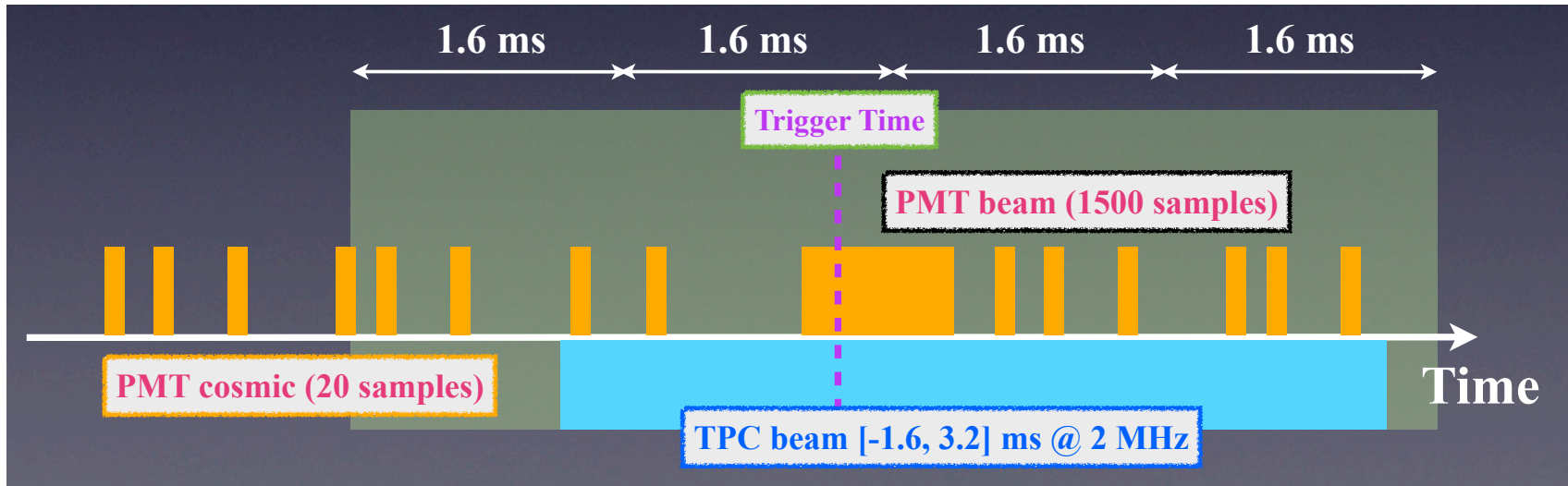
Full simulation

- Cerenkov and scintillation light
- Includes quenching, absorption, scattering, and reflection
- Takes a long time

Fast simulation

- Scintillation light only
- Assumes isotropic production, and uses lookup table to model response on each optical detector (PMT)
- LArIAT work on adding in effects of wavelength shifter

OPTICAL READOUT SIMULATION



r, May 28, 14

Digitization

- OpDetDigitizer being replaced with uboone specific OpticalADCSim

PMT readout is not one long window like TPC

- Broken up into many smaller readout windows
- OpticalFEM being moved to uboonecode

DETAILS OF THE DIGI/READOUT SIM

Let me reiterate: not the expert

But uboone code now...

- Takes in data for non-PMT pulses in optical readout
 - Beam pules, and other auxiliary things
- Includes hooks for channel-by-channel variations in calibration parameters
- Includes all parameters matching uboone readout, with adjustments available via fhicl file
- All of this is in v02_01_00

End output: PMT waveforms

RECONSTRUCTION ALGORITHM

OpFlashFinder_module

- Reads in PMT waveforms and trigger info
- Creates OpHits (light on one PMT) and OpFlashes (light on multiple PMTs at same time)
 - Form optical hits from all pulses
 - Collect hits into proto-flashes (course)
 - Refine proto-flashes and ensure uniqueness among hits
 - Construct `recob::OpFlash` objects from proto-flashes
 - Remove flashes that are likely late light
 - Do bookkeeping for making `OpHit` \leftrightarrow `OpFlash` associations
- Code recently refactored and `OpHit/OpFlash` times modified to be in units of microseconds, relative to trigger time

FLASH-TRACK MATCHING

General idea

- Match tracks to flashes, and use flash-timing to give us out-of-time-tracks

Specifics: BeamFlashCompatibility_module

- Find flash during beam gate ($\rightarrow t_0$)
- Assume all tracks in event produced at t_0 and make “flash hypothesis”
 - Assume all tracks are MIPs for the moment
- Compare flash hypotheses to actual flash, and tag those that are inconsistent
 - Too much light on a PMT or as a whole

DISCUSSION