Path to trigger resources for FD

Artur Sztuc
a.sztuc@ucl.ac.uk
University College London

23 Apr 2024

DEEP UNDERGROUND
NEUTRINO EXPERIMENT

Three paths to figure out the trigger resources for DUNE:

- **On paper:** We can guestimate TP rates, TA rates (at least HighE). Assume each DAQModule needs it's own CPU, we can preliminary calculate number of CPUs, RAM etc.
- **PD2:** especially NP02 (with new trigger layout), will be a good test of the triggering system, and the amount of resources it uses per APA/CRP.
  - Large caveat: PD2 are surface detectors!
- **Simulation:** Once we simulation & integration in the trigger, we can simply pass FD-Dune-simulated tpstream to the replay application.

There are some caveats to all three, but the simulation would be probably the most proper way.

- Trigger algorithms only use 1 core each per APA (at the moment). 10 Algorithms means 10 cores per APA at least.
- TPTee, TATee, TABuffer, TAZipper, TCTee, all can have their own thread, so: 10×3 for TAMaking per APA, $+ 10 \times 4$ for TCMaking (not per APA), assume 150 APAs, we need $\sim$4500 cores before MLT for FD-HD.
  - 6 algos is minimum, but we might need $\sim$20, so $\sim$9000 cores
- This is with the old v4 trigger setup, new v5 setup might need less, but need to check.
- Ethernet speed can be guestimated from the number of TP/s, TA/s and TC/s expected, and so can RAM.
- This is the quickest & least precise path. Doesn't answer all the questions.
  - The CPU speed required? Maybe don't need 5GHz but we need to measure.

- PD2 will allow us to measure the performance of our trigger algorithms in a live system – although we can use replay application for this too.
- PD2 data will come from 4 APAs/CRPs, also allowing us to see how the TC-making performance scales.
- We can use that data to measure the resources used by the trigger:
  - How many CPUs are we using?
  - To what extent are we utilizing these CPUs?
  - How much RAM are we using, and how does it scale with TP/TA rates?
- TP rate/unit in FD might be different. More 39Ar means more TPs, less cosmics means less TPs... really need simulation here.
- PD2 VD, NP02, will be more representative too, since it will use the new readout-like trigger system.

- Probably the most reliable here, but also most time-taking.
  - Luckily work on this is slowly ramping up!
- We could simulate TPs in HD and VD FD DUNE detectors, and save them as tpstream files.
- We would then use the replay application to simulate live FD detector run, and measure the amount resources we're using!
- Of course, we might need to scale down, since 150 APAs/CRPs might be beyond our current capabilities.

- The biggest Ethernet requirement: sending TPs from the readout to the trigger. If TA-making is run on TP-making machines, goes to 0.
- We still don't know if we will end up running algorithms on each plane separately or not...
- With "on-paper" and "pd2" we can only use/assume TA/TC makers we have, which are nowhere near similar to what we want for FD.
- If we go with the "simulation", then we should start thinking about algorithms we actually want to use in FD, and use them for this task.
  - HighE algo will be easy to write – something based on ADC-integral, number of TPs and time-over-threshold, make TA if any of these is met.
  - LowE is more difficult and I currently have no ideas.
  - We might run multiple LowE algorithms. How about supernova? Should we assume 10-20 algorithms in total?
- Of course, we might need more resources than what we calculate.
  - Chat if get a hot channel before we mask it out? Could fill our buffers/RAM and we miss supernova. Extra resources for emulation/simulation?