SciDAC-4 pilot project "*HEP Event Reconstruction on Cutting Edge Computing Architectures*"

Accelerating HEP Event Reconstruction

Fermilab: G. Cerati, A. Hall, M. Wang
University of Oregon: B. Norris, B. Gravelle

Motivation

Among the main science drivers, the 2015 P5 report identified the Higgs boson as a new tool for discovery, the physics associated with neutrino mass, and the search for unknown particles and interactions. The P5 panel recommends that such goals are pursued through the completion of the high luminosity LHC (HL-LHC) upgrade program, and the development of a short- and long-baseline neutrino program hosted at Fermilab, which is based on Liquid Argon Time Projection Chamber detectors (LArTPC). Both LHC and neutrino experiments need to reconstruct events that are challenging, although their challenges are different in nature.
Current HEP processing model based on multi-author gigantic sequential C++ code is not adequate for experiments with increasing detector sizes and accelerator intensities. Enabling the efficient usage of modern compute architectures in HEP event reconstruction is vital for achieving their physics goals.

Work

Fermilab and University of Oregon are accelerating HEP event reconstruction using highly parallel architectures. We focus on the novel parallel algorithm for charged particle tracking in CMS, and pioneer the usage of similar techniques for reconstruction in LArTPC detectors. With the use of advanced profiling tools and development techniques, including autotuning, the throughput of the algorithms on the leading parallel architectures (Xeon Phi, GPU) will be maximized and portable implementations for usage at supercomputers and with heterogenous platforms will be explored.

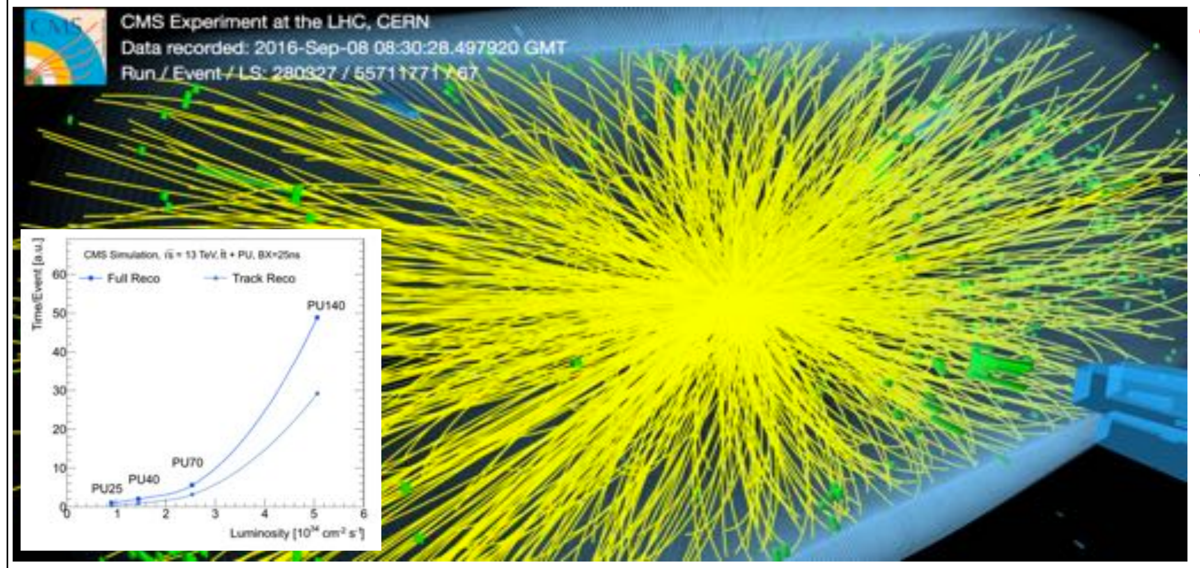
Goals of the project are the following:

1. Identify key algorithms for physics outcome which are dominant contributions for the experiments' reconstruction workflows
2. Characterize and re-design the algorithm to make efficient usage of parallelism, both at data- and instruction-level
3. Deploy the new code in the experiments' framework
4. Explore execution on different architectures and platforms

Related Activities

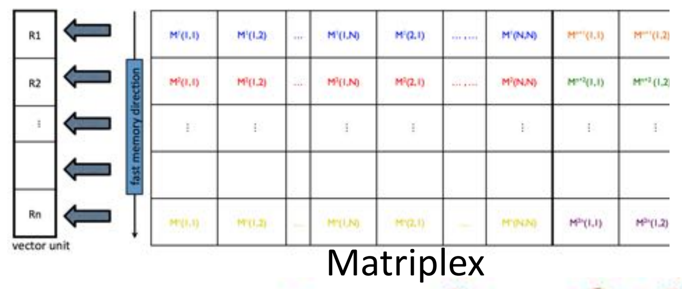
Cornell/Princeton/UCSD collaboration: original authors of parallel tracking prototype for CMS. Our project is working in close contact with this collaboration.
ASCR institutes: Rapids (Platform Readiness)
Other SciDAC projects: Hep.TrkX (tracking with ML)
HEP Experiments: LHC (CMS, Atlas), Neutrino experiments (DUNE, SBN)

CMS Tracking



The problem:

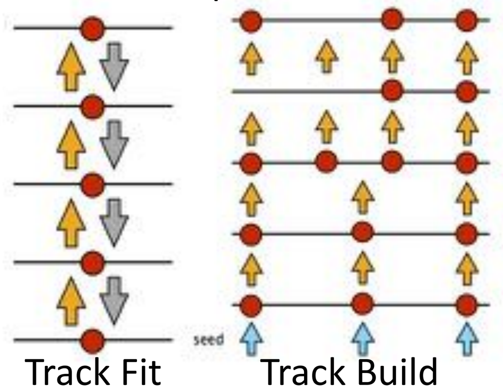
Track reconstruction time diverges with increasing luminosity and becomes unmanageable for data processing at HL-LHC



Matriplex

The algorithm:

Kalman filter-based fit or build. Build is combinatorial search for compatible hits along the track. Vectorized using Matriplex: SIMD processing of multiple candidates. Parallelization with threads at seed level (TBB). Challenges: unpredictable branching, low arithmetic intensity (quick processing of many small objects).



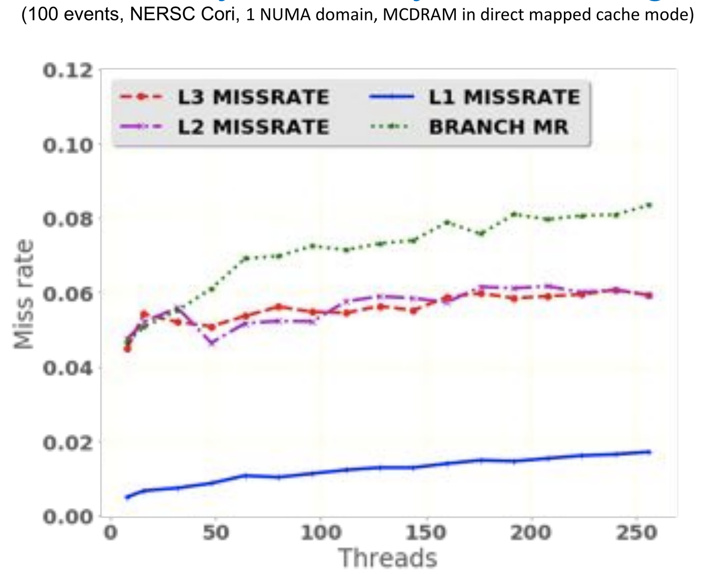
Track Fit Track Build

Performance analysis:

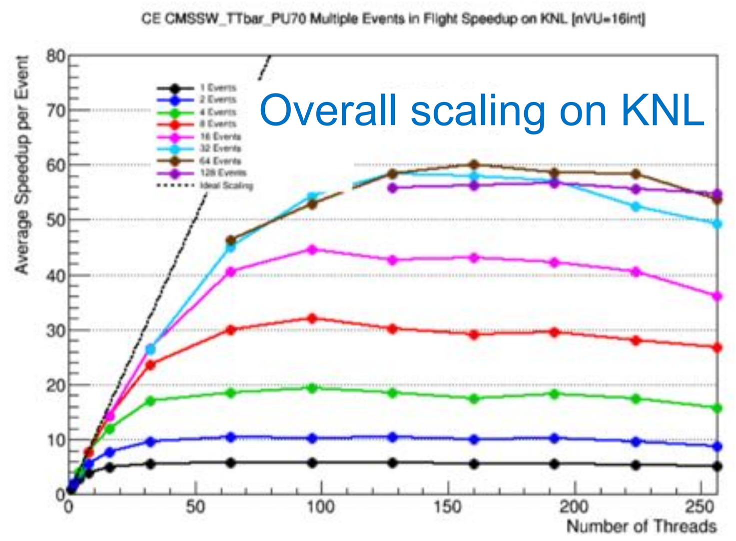
Code profiled with TAU Commander (sampling) and analyzed with Pandas

Memory efficiency and scaling

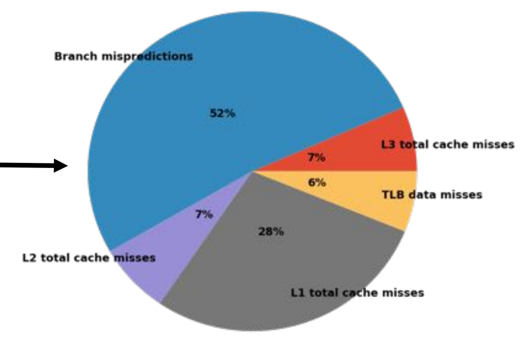
(100 events, NERSC Cori, 1 NUMA domain, MCDRAM in direct mapped cache mode)



Correlations between metrics (Spearman) for 256 threads	Stall Cycles	Total Cycles
Scalar vector ops	86.01%	82.78%
Conditional branches	81.92%	77.70%
L3 misses	81.20%	77.25%
Branch instructions	80.67%	77.00%
Branch mispredictions	79.81%	74.99%
Load/store instructions	79.68%	74.60%
L3 accesses	78.61%	73.99%
L2 total cache misses	77.84%	73.86%
L2 accesses	77.55%	72.79%
Memory acces stalls	77.02%	74.78%
L1 total cache misses	76.43%	71.66%
Unconditional branches	76.31%	71.99%
Vector operations	74.15%	68.52%
TLB data misses	73.86%	68.70%



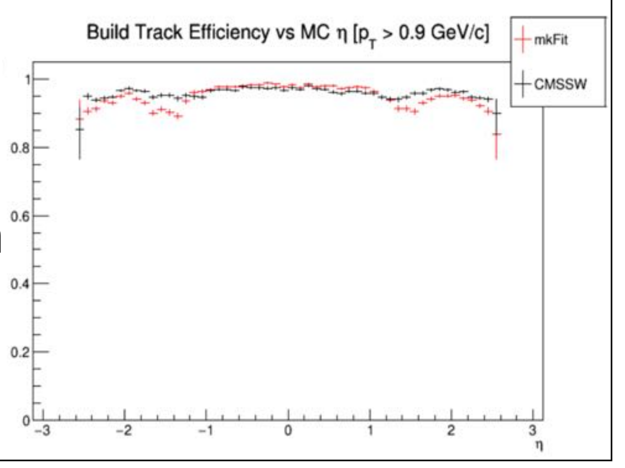
Least scalable functions: Increase in total cycles (from 8 to 256 threads)	
MkBuilder::map_seed_hits()	2.6e+06
Hit::Hit()	1.6e+05
Matriplex<float, 1, 1, 8>::operator[](int)	2.1e+04



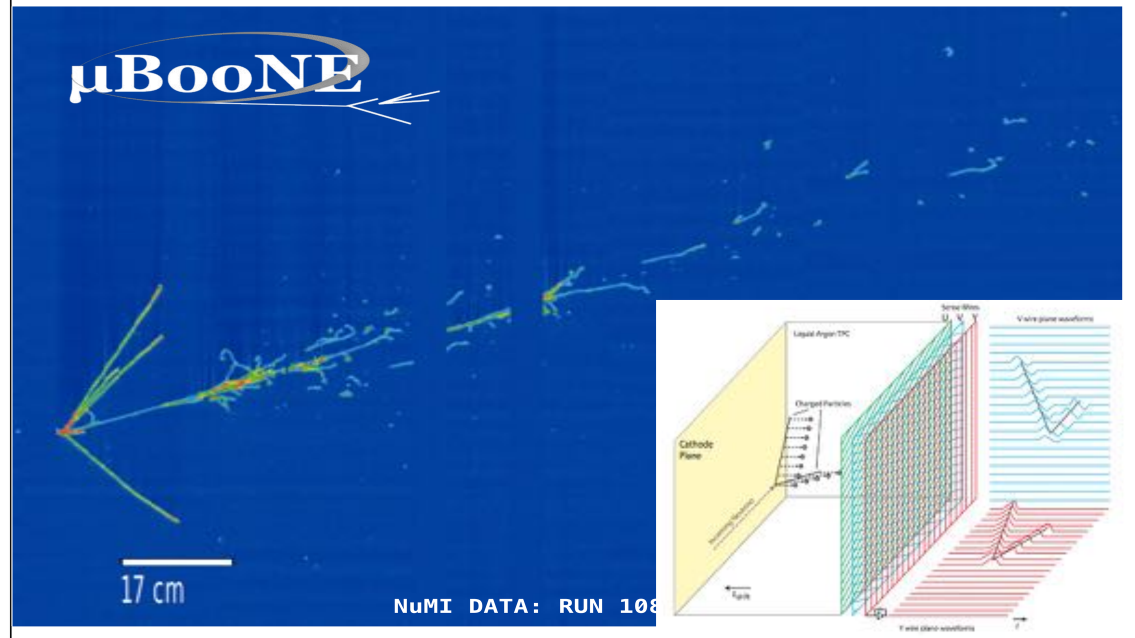
All result from TTbar+70PU sample

Results and next steps:

Vectorization speedup up to 3x. Efficient parallelization when using seeds from multiple collision events. Single thread 10x faster than CMSSW with comparable physics performance. CMSSW integration and further optimizations in progress.



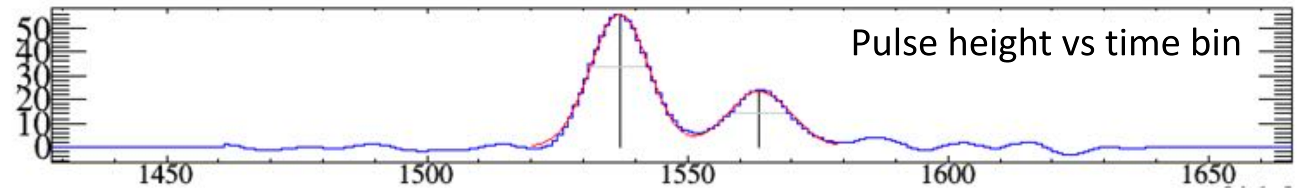
LArTPC Reconstruction



The problem:

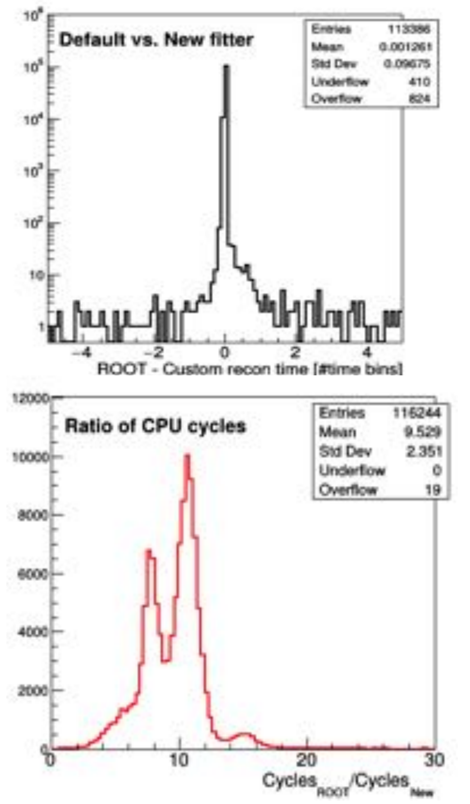
Reconstruction in LArTPC neutrino experiments is challenging due to many possible neutrino topologies, noise, contamination of cosmic rays. Takes O(100) s/ev in MicroBooNE, and future experiments will be much bigger and on more intense beams.

Feasibility study: Hit finding algorithm.



MicroBooNE'sTPC is made of ~8k wires readout at 2 MHz. Signal from charged particles produces Gaussian pulses. Hit finding is the process of identifying such pulses and determine its properties (peak position and width). Suitable for feasibility study to demonstrate parallelism with LArTPC: wires can be processed independently. Currently takes ~15% of the MicroBooNE reconstruction workflow.

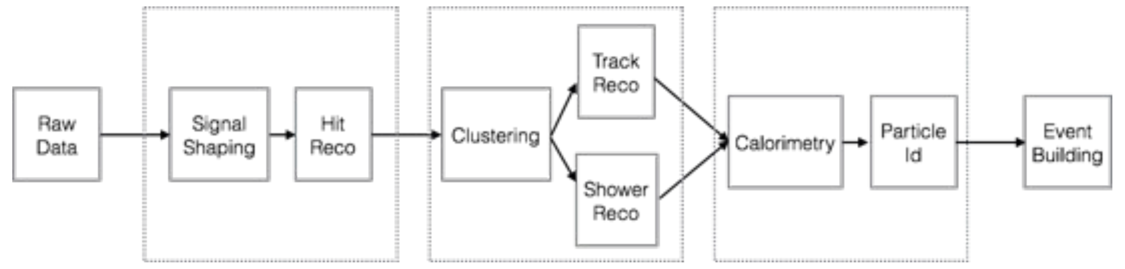
Results:



Replicated experiment's implementation in standalone code for easier testing and optimization. Replaced current Gaussian fit technique (based on Minuit+ROOT libraries) with local implementation of Levenberg-Marquardt minimization. Results nearly identical in terms of hit properties but with large speedup (~8x). Next steps: parallelize at wire level; identify strategy for vectorization; test scalability; port improved version into experiments' codebase.

Plans:

After completing the feasibility study, we will focus on identifying another crucial algorithm. Options could be upstream (signal processing) or downstream (pattern recognition) in the reconstruction chain with respect to Hit finding.



Typical reconstruction chain for LArTPC experiments