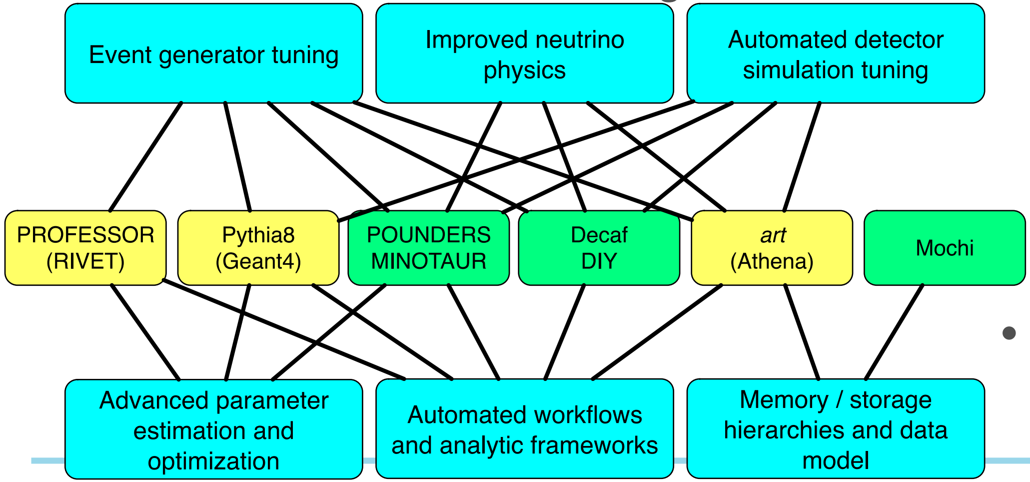
HEP Data Analytics on HPC

Jim Kowalkowski
SciDAC-4 PI Meeting 2018
23 July 2018

Project Goals

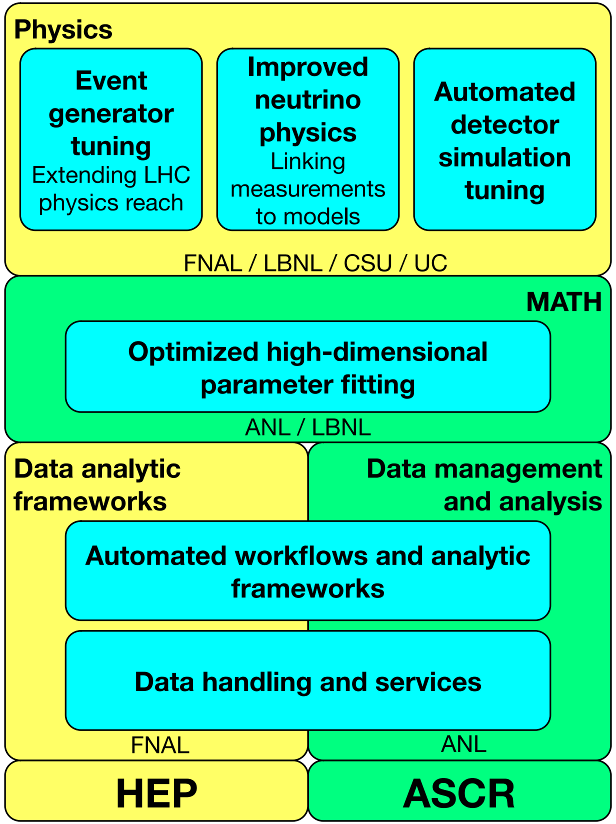
- Extend physics reach of LHC and neutrino experiments
  - Event generator tuning
  - Neutrino oscillation and cross-section measurements
  - Detector simulation tuning



- Transform how these physics tasks are carried out through ASCR math and data analytics
  - High-dimensional parameter fitting,
  - Workflows supporting automated optimization
  - Distributed dataset management storage and access (*in situ*) for experiment data
  - Introduction of data-parallel programming within analysis procedures
- Accelerate HEP analysis on HPC platforms

Collaboration

- Collaboration between DOE Office of High Energy Physics and Advanced Scientific Computing Research (ASCR supports the major US supercomputing facilities)
  - LHC and neutrino physics: N. Buchanan (CSU, NOvA/DUNE), P. Calafiura (LBNL, LHC-ATLAS), Z. Marshall (LBNL, LHC-ATLAS), S. Mrenna (FNAL, LHC-CMS), A. Norman (FNAL, NOvA/DUNE), A. Sousa (UC, NOvA/DUNE)
  - Optimization: S. Leyffer (ANL), J. Mueller (LBNL)
  - Workflow, Data Modeling: M. Paterno (FNAL), T. Peterka (ANL), R. Ross (ANL), S. Sehrish (FNAL)
- J. Kowalkowski – PI (FNAL)



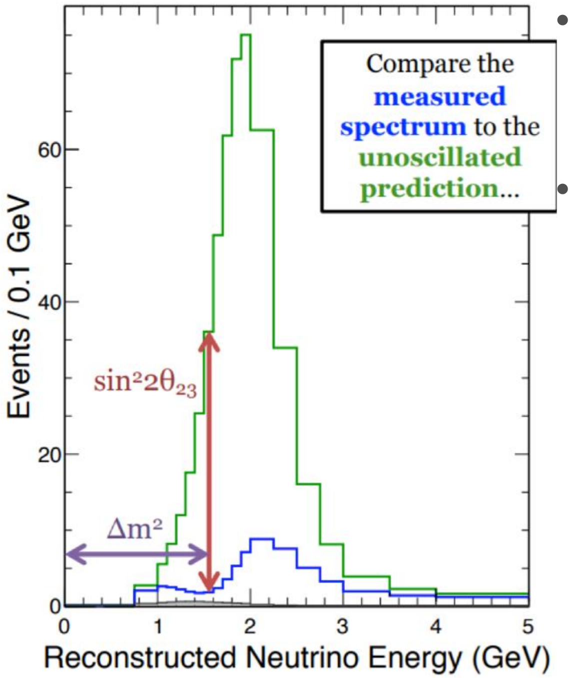
http://computing.fnal.gov/hep-on-hpc/

Accomplishments

- First NOvA neutrino oscillation analysis using NERSC
  - Time-to-result improved by 50x; first round completed within 16 hours
  - Used ~30M hours on Cori (and part of Edison) across two runs
- Prototype event store (**HEPnOS**) built for serving data to HEP analysis codes
- Data-parallel NOvA pre-analysis event selection procedure
  - NOvA accepted ownership of HDF conversion software for their data
- Improved understanding of ATLAS and CMS data through generator tuning with Pythia, Rivet, and Professor
  - Evolution of generator tuning algorithms, optimization of data selection, and development of DIY workflow
  - Generator tuning on unexploited LHC jet data and detector simulation tuning
- Community interactions:
  - CHEP: Event selection, Rational polynomial approximations in Professor, NOvA analysis

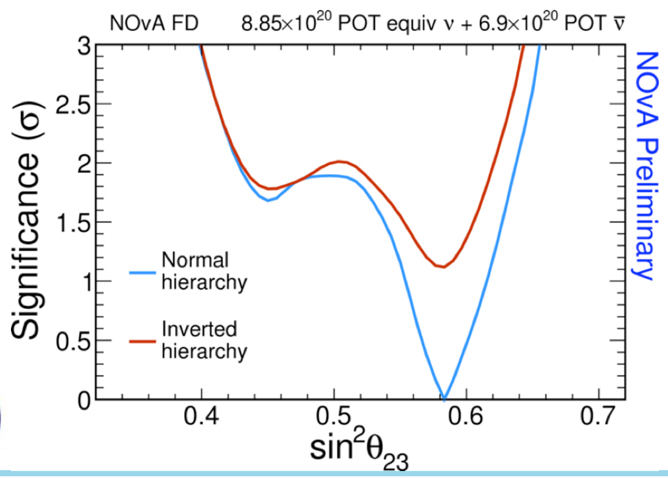
HEPnOS: https://xgitlab.cels.anl.gov/sds/HEPnOS/wikis/home

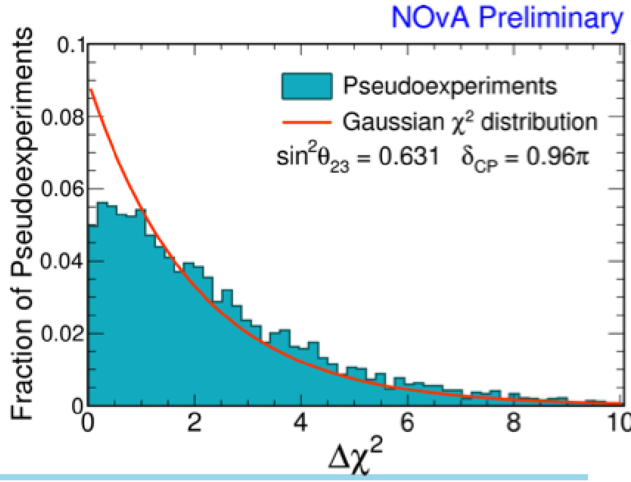
NOvA Neutrino Oscillation Measurements



$$P(\nu_\mu \to \nu_\mu) \simeq 1 - \sin^2(2\theta_{23}) \sin^2\left(1.267\Delta m^2_{32}\frac{L}{E}\right)$$

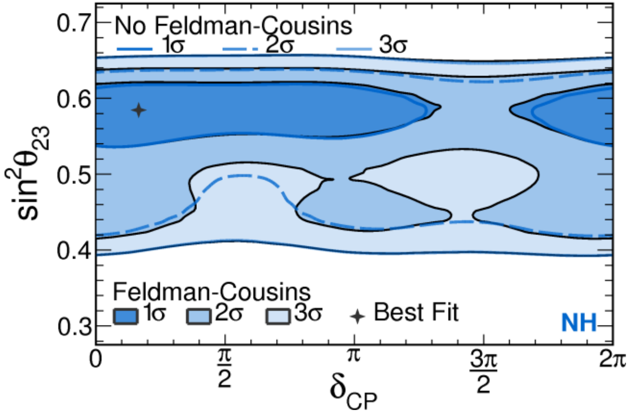
- Compare data with neutrino oscillation hypothesis
  - Extract best-fit oscillation parameters and associated confidence intervals
  - Compute rejection significance for non-optimal parameter values
- Cannot assume gaussian errors for oscillation measurements
  - (1) Low statistics and (2) parameters probed near physical boundaries
  - Require computationally-intensive calculation for confidence intervals





NOvA Neutrino + Antineutrino Analysis

- **NOvA uses some of the most complicated fitting procedures in neutrino physics**
  - Simultaneous multi-dimensional fits for neutrino and anti-neutrino data
  - Complexity of high dimensional parameter space requires billions of functional fits
  - Multi-Universe techniques utilized for proper statistical corrections



Sensitivity contours under the Gaussian statistical computations and under a Feldman-Cousins corrected computation. The corrected contours can reveal large islands in the parameter space where sensitivity is greatly improved.

- Large-scale analysis campaigns carried out at NERSC Cori for the first time
  - First run occurred May 7th, over 1.1 million running jobs
  - Second round of calculations occurred May 24th (both Cori and Edison)
  - Consumed 37M CPU-hours in 42 hours over both runs
  - New facilities and procedures enthusiastically received by NOvA collaboration – desire accelerating transfer of other key analyses
- NOvA revealed first set of electron antineutrino appearance results on June 4th at the Neutrino 2018 conference

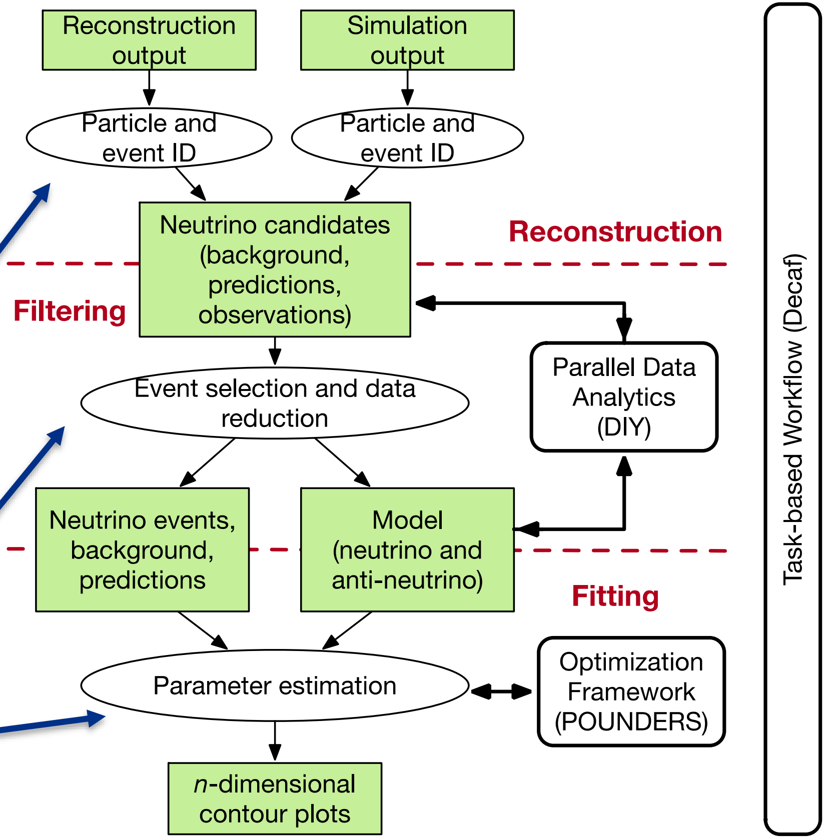
http://news.fnal.gov/2018/07/fermilab-computing-experts-bolster-nova-evidence-1-million-cores-consumed/

NOvA Analysis run: Project support

- Required coordinated effort from SciDAC-4, Fermilab, the NOvA collaboration, and NERSC staff.
  - Analysis and technical effort directed by SciDAC/NOVA laboratory and university Postdocs and students
  - Provided excellent training ground for utilizing HPC centers and tools
- HEPCloud enabled large-scale resource provisioning, workload management, and monitoring at NERSC
- Broke several records
  - Accuracy: 8x higher resolution than any prior result
  - Turnaround: 50x faster - results reviewed in <24 hours
  - Scale: ~1M active cores – biggest Condor pool ever
- This work completes a first year major milestone:
  - Forms baseline for analysis calculations with current data, providing major scientific results
  - Reproduced 2017 results for validation
  - Future HPC refactoring will be compared with this result.

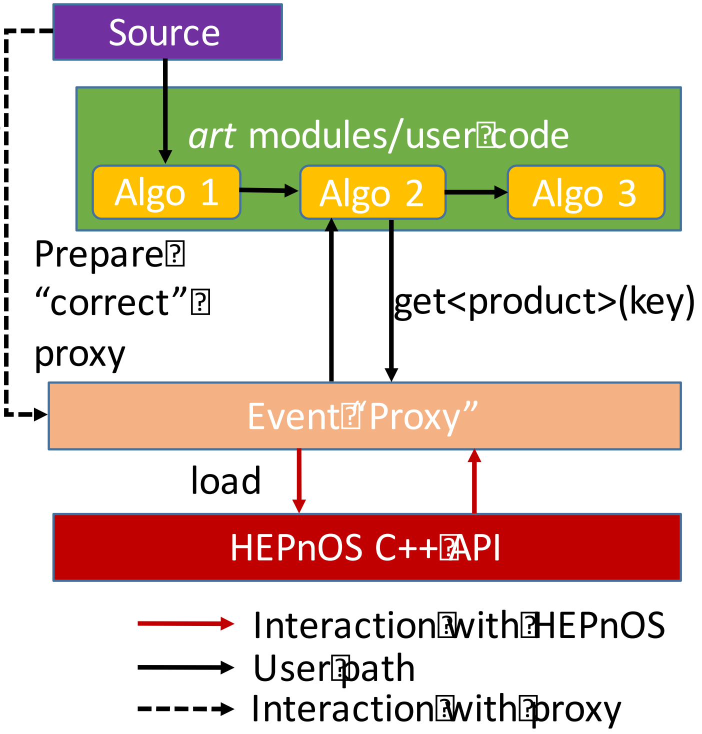
Neutrino analysis workflow

- Advancements using HPC leadership facilities
  - Analysis using full dataset across all layers, managed using tools and techniques developed by **RAPIDS** institute
  - Utilize multi-dimensional fitting procedures from **FastMATH** institute
- Initial tasks
  - Basic data models for NOvA and LArSoft and datasets mapped into HPC NVRAM-based hierarchical storage systems using ASCR services and tools through the **Mochi** project
  - Demonstration of fast event selection using **DIY**
  - Full automation of Feldman-Cousins analysis prescription with **Decaf**



HEP data management

- Make high-volume reconstructed physics object data available to analysis workflows
  - Leverage existing modular frameworks and extensible data models
  - Starting point: Use actual LArSoft *Tracks*, *Hits*, *Associations* from ProtoDUNE simulation
- Allow facility services to distribute data at any scale, using existing abstractions
  - Runtime ROOT replacement using RAPIDS for I/O
  - Include all levels (or layers) of data aggregation with metadata
  - Data distribution and data parallelism implicit to user
- Application access
  - Exploit event independence



Event currently interacts with art/ROOT File

HEPnOS: Fast Event-Store for HEP

Goals:

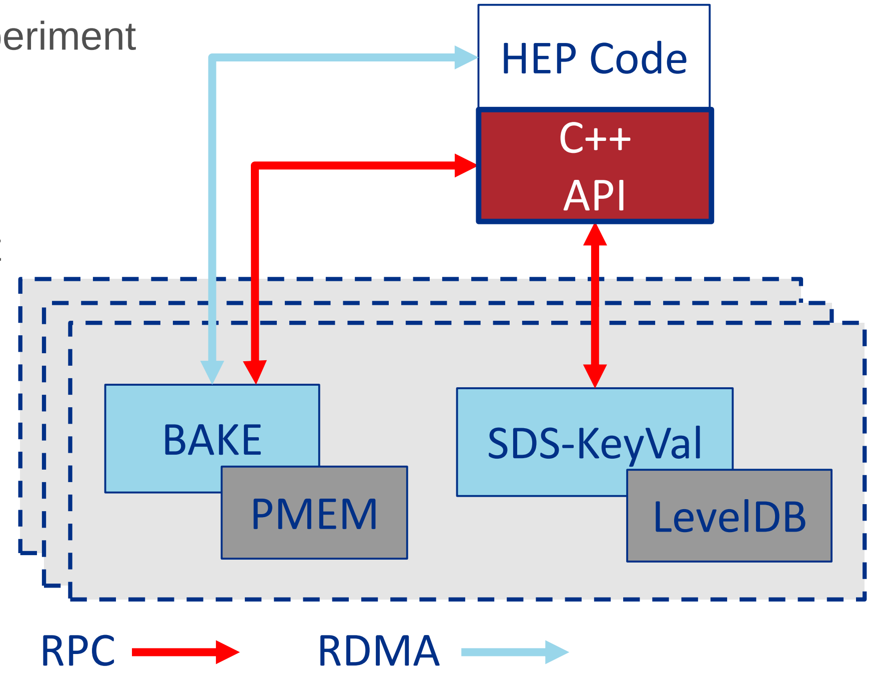
- Manage physics event data from simulation and experiment through multiple phases of analysis
- Accelerate access by retaining data in the system throughout analysis process
- Reuses components from Mochi ASCR R&D project

Properties:

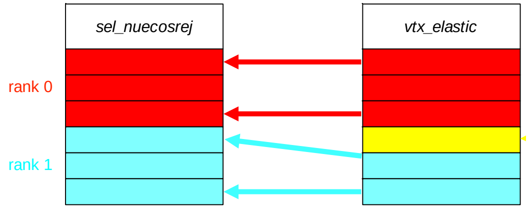
- Write-once, read-many
- Hierarchical namespace (datasets, runs, subruns)
- C++ API (serialization of C++ objects)

Components:

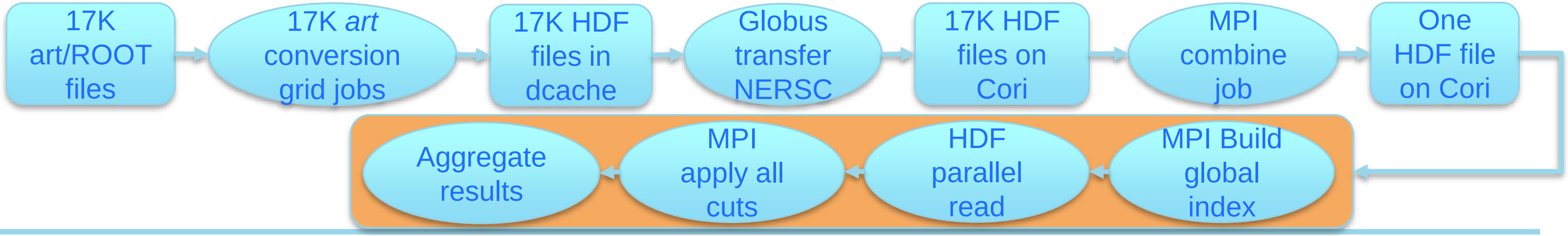
- Mercury, Argobots, Margo, SDSKV, BAKE, SSG
- **New code: C++ event interface**
  Map data model into stores



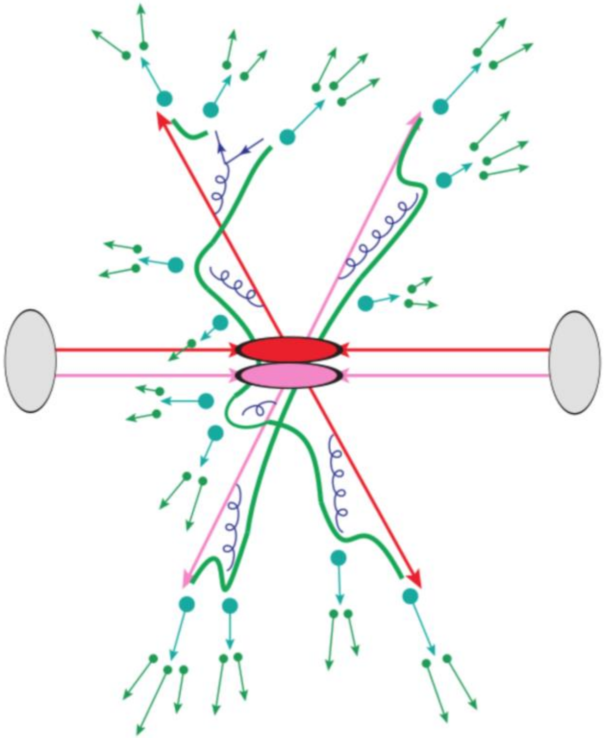
Parallel event pre-selection



- Motivation
  - Fast assessment of event selection used for final analyses
  - Measure effects of event filtering
- Current situation
  - NOvA slice data held in 17K ROOT files across
  - ~27 million events are reduced to tens using ROOT macros applying physics “cuts”
- New method
  - Data prepared for analysis using workflow shown below
  - End state: >50 groups (tables), each with many attributes
- First selection procedure uses Python/MPI/HDF/Pandas on HPCs
  - Global index allow data alignment across tables within one rank
  - Simple composition of cut expressions with Pandas
  - Data parallelism implicit



Event generators

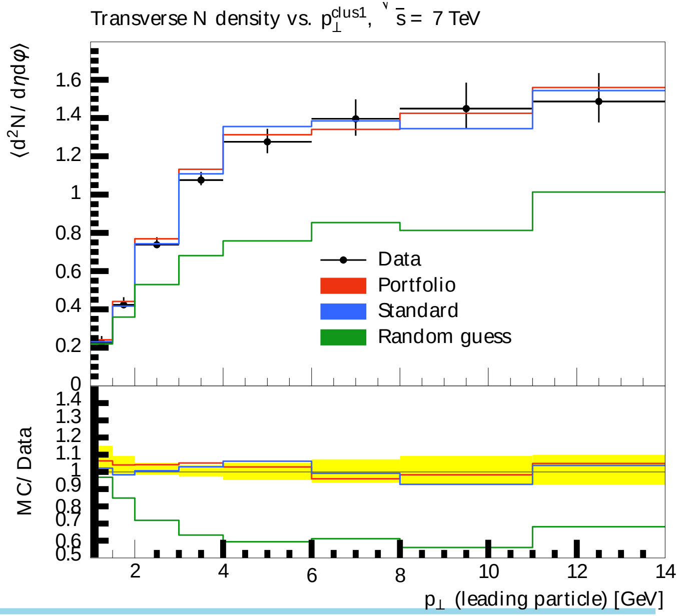


Cartoon event generator output

- Event generators are numerical models used in HEP for describing particle collisions
- 100s of tunable parameters reflecting years of modeling wisdom, made to reproduce experimental measurements
- Professor: a system for tuning a set of parameters to a set of observations; widely used in HEP
  - Relies on hand-picked parameters and observations
  - Only limited sets of parameters can be simultaneously tuned because of computing costs
  - Only simple event generators can be integrated because of computing costs
- Goals: automate, optimize, expand tools, and tune more parameters, exploit neglected data

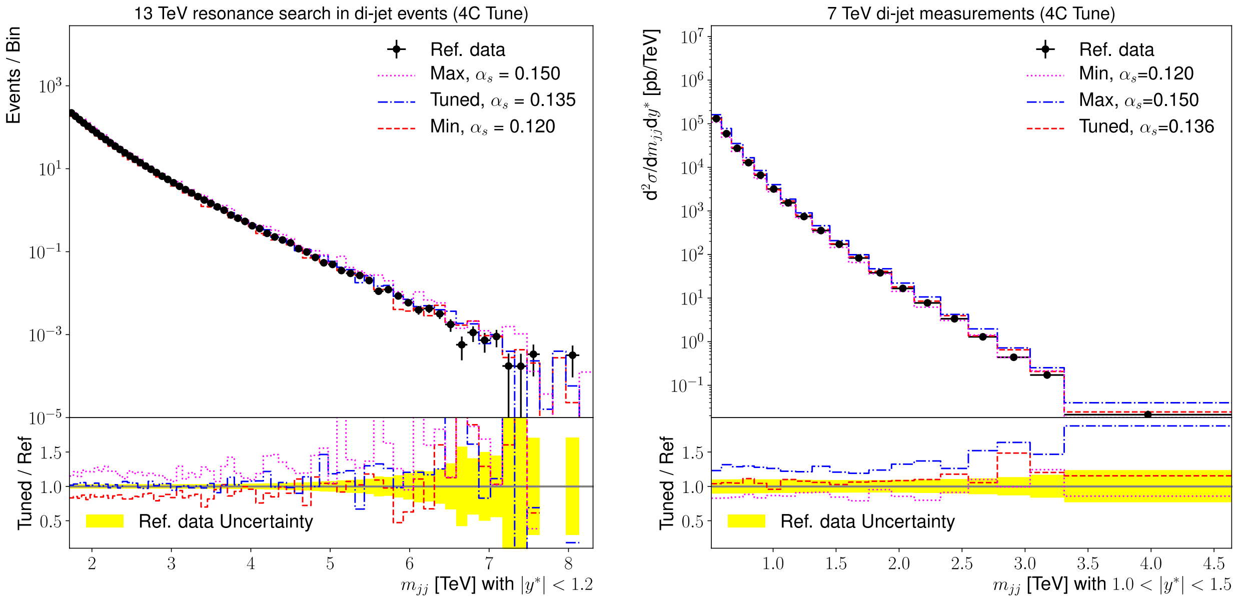
Optimizations

- *Optimization effort:* develop new methods to efficiently identify and optimize only the most relevant tuning parameters in event generator application
- *FASTmath connection*: develop methods for efficient high-dimensional computationally expensive simulation optimization that are general enough to be applicable to a wide range of science problems
- *Our approach*: Formulation as “outer loop” optimization problem
  - Pragmatic approach for balancing deficiencies in physics modeling across a large variety of data
- *Early results*: Optimization performs better than manual data selection



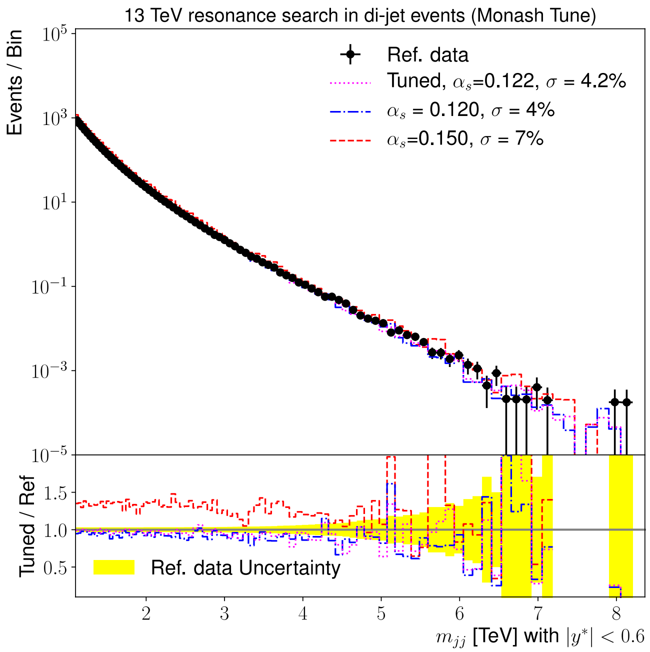
Tuning with data from search analyses

- Generator tuning normally performed with "unfolded" data
  - Specialized measurements (e.g. underlying event, jet properties)
  - Wide variety of data available not previously used for tuning



- **Use fast simulation to model detector effects, and tune directly to search data**
  - First results compare well to tune with measurement
- Expands data that can be used and kinematic range
  - Tune event generators in phase space regions most interesting to LHC searches

Tuning of experimental effects



Tune of jet resolution based on ATLAS dijet search data

- The same mechanism used for tuning event generators can be applied to detector simulation
  - Here: proof of principle with fast simulation, using the same tools and workflow as before
- Parameters normally taken from papers published by the experiments
  - Labor-intensive process; not always applicable to search regions, where these fast simulations are most used
- Can be extended to provide an LHC search-data based fast simulation tune

Coming soon (end of summer) …

- Parallel NOvA analysis event pre-selection run on HPC facilities
  – HDF with Python, then with C++/DIY application
- First version of Feldman-Cousins correction with DIY
- HEPnOS and large-scale LArSoft dataset load and access
  – ProtoDUNE simulation: use of track/hit objects
  – LArIAT waveform – test using DSP app using DIY
- Generator tuning run

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Scientific Discovery through Advanced Computing (SciDAC) program.

Engagements

- RAPIDS
  - DIY: now within generating tuning and neutrino analysis applications, soon will be used within event selection and physics object access applications
  - Mochi: used within HEPnOS
  - Decaf: will first be doing an evaluation for overall tuning workflow
- Stefan Hoeche's SciDAC project
  - Helping define HDF event format and workflows for parallel data access
- FASTMath
  - Combinatorics, binary constraint satisfaction problems: path and schedule optimizations
- HEP community
  - Pythia, Professor, Rivet, art framework, gallery, ROOT, other generators

Potential synergies

- RAPIDS
  - Northwestern University: parallel data access with netCDF
  - HDF Group: improved C++ (14, 17, and beyond), data modeling tools and schema aids
  - Performance tuning: will need help with parallel FS access tuning and vectorization of analysis codes
- FASTMath
  - Sparse grids and MCMC alternatives, high dimensional integration of expensive function
- HEP experiments: Neutrino community and GENIE tuning