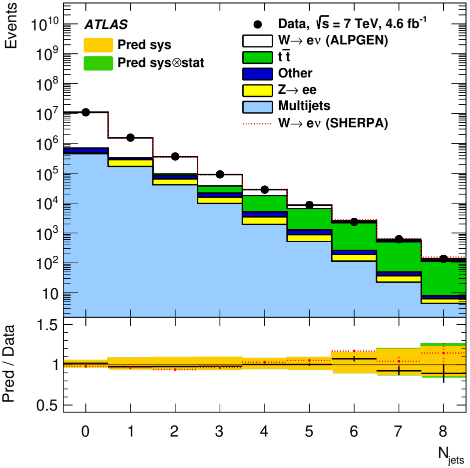
High-multiplicity multi-jet merging with HPC technology

Stefan Hoeche, Xiangyang Ju, Jim Kowalkowski, Stephen Mrenna, Tom Peterka, Stefan Prestel, **Holger Schulz**

Physics Event Generator Computing Workshop, CERN, November 28, 2018

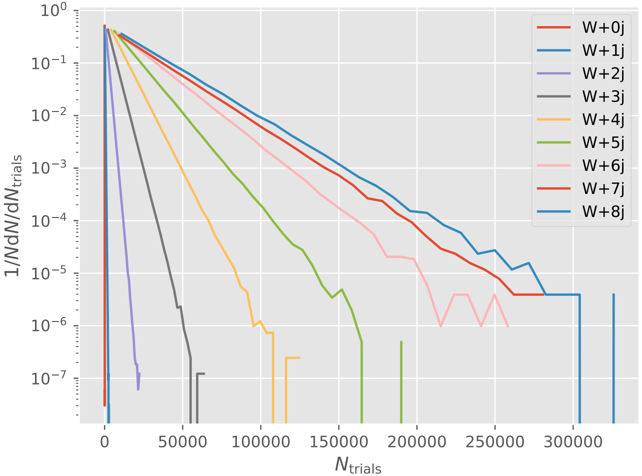
Motivation

arXiv:1409.8639 [hep-ex]



- LHC experiments can see 8 jets
- High precision predictions for e.g. searches should reflect that
- Can we do this on HPC?

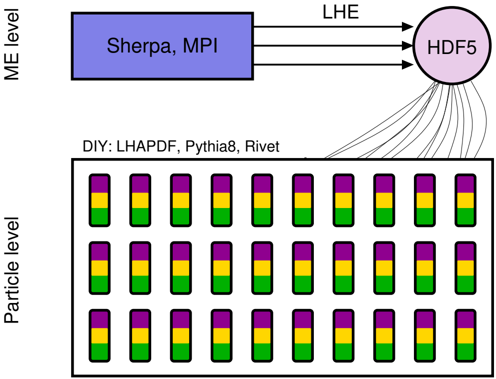
Trials in (LO) ME level events



- Distribution of trials gets flatter with number of jets.
- Huge variation of Matrix Element (ME)-level compute time.
- Traditional Sherpa way of doing all in one go just does not scale. (See also T. Childers et al. doi:10.1088/1742-6596/898/7/072044)

Our approach to event generation on HPC

- Use Sherpa to generate ME-level events (Les Houches like format)
- XML output is not a good solution for HPC machines
- Use HDF5 instead:
  - Parallel write and read
  - Binary storage of data, built-in compression
- Particle level event generation and merging with Pythia8 — we use ASCR's DIY technology for MPI parallelisation here



HDF5 storage

Dataset	data type
NPARTICLES	int
SCALE	double
AQCD	double
...	
NPLO	double
NPNLO	double
WEIGHT	double
TRIALS	double

Table: Event properties

Dataset	data type
ID	int
STATUS	int
MOTHER1	int
COLOR1	int
PX	double
...	
LIFETIME	double
SPIN	double

Table: Particle properties

Dataset	data type
START	size_t
END	size_t

Table: Lookup-table

- Trivial (parallel) storage of properties in 1D datasets of basic types
- Trivial (parallel) access by index, connection between event and particle properties by lookup table

Technicalities

- Requirement: LIBHDF5 (apt-get / dnf install, standard on HPC)
- Header-only library HighFive **github.com/BlueBrain/HighFive**
- N.b. very nice python library H5PY, works beautifully with numpy (used this initally to convert LHE XML files to hdf5 but this is quite cumbersome)

- Header-only library DIY used in particle level simulation **http://diatomic.github.io/diy**
- Computing model based on "blocks"
- Does all the low-level MPI communication for you

W+jets example

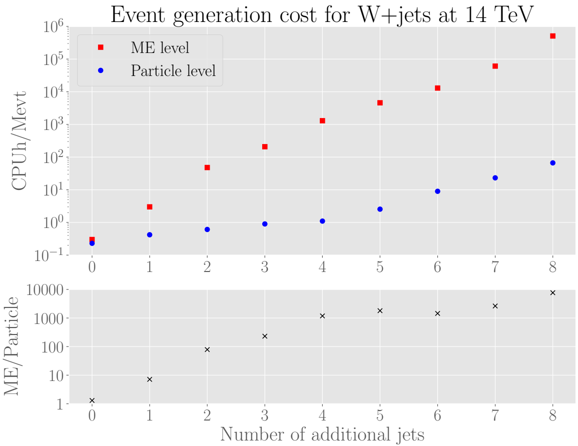
- W+jets at $\sqrt{s} = 14$ TeV simulation.
- Merging scale is at 20 GeV.
- The simulation is at leading order, the merging scheme is CKKW-L.
- ME-level event generation done at SLAC cluster of Xeon E5 CPUs.
- Particle level event generation on NERSC Cori using Haswell nodes.

N_{jets}	0	1	2	3	4	5	6	7	8
N_{events}	65M	32M	16M	8M	4M	2M	1M	500k	250k
HDF5 (9) [GB]	7.1	4.9	3.0	1.8	1	0.6	0.3	0.2	0.1
HDF5 (0) [GB]	26	16	9.1	5.2	2.9	1.9	1.2	0.62	0.25

- Number of quarks limited to ≤ 6 for $N_{\text{jets}} = 6, 7$
- Number of quarks limited to ≤ 4 for $N_{\text{jets}} = 8$

CPU cost analysis

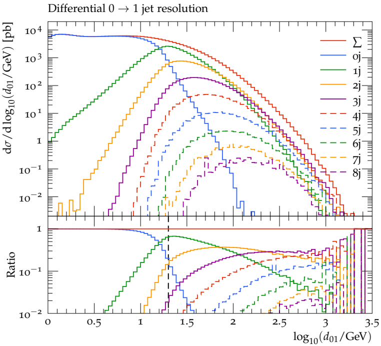
- Process ME-samples with different jet multiplicities separately.
- Compare ME-level and particle level event simulation.
- Note that the measure is CPUh per 1M events



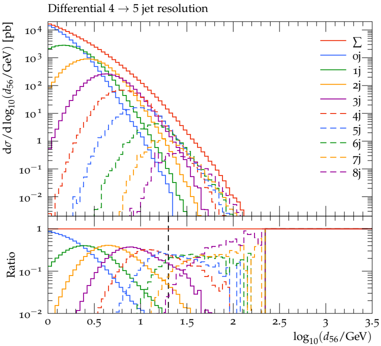
Benefits

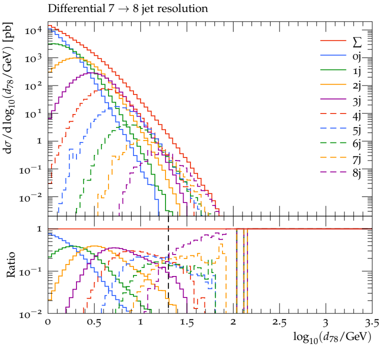
- The CPU expensive part of the simulation is stored in a parton-shower independent format.
- Running the particle level simulation now cheap in comparison, allows e.g.
    - PDF re-weighting
    - All sorts of variation studies
    - Tuning and similar parameter space exploration
- Can think of a hybrid strategy for event generation:
    - Do low multiplicity as per usual
    - Generate higher multiplicities with this approach

Jet rates


Differential 0 → 1 jet resolution
Differential 2 → 3 jet resolution
dσ/d log10(d01/GeV) [pb]
dσ/d log10(d23/GeV) [pb]
Ratio
log10(d01/GeV)
log10(d23/GeV)
Σ
0j
1j
2j
3j
4j
5j
6j
7j
8j

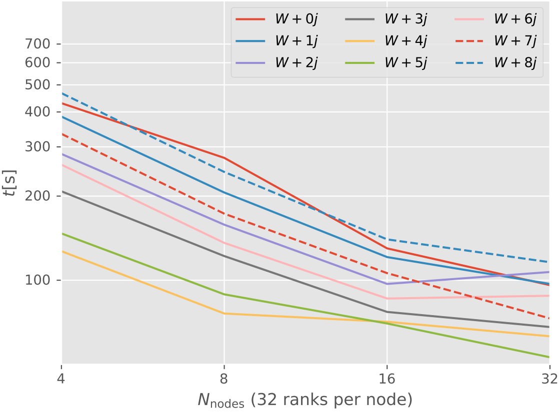
Jet rates


Differential 4 → 5 jet resolution
dσ/d log10(d56/GeV) [pb]
Ratio
log10(d56/GeV)
∑
0j
1j
2j
3j
4j
5j
6j
7j
8j


Differential 7 → 8 jet resolution
dσ/d log10(d78/GeV) [pb]
Ratio
log10(d78/GeV)
∑
0j
1j
2j
3j
4j
5j
6j
7j
8j

Scaling

- Scaling of pure *particle level* event generation for *total* samples
- Software stack compiled on NERSC Cori (gcc7.3), measurements done on Haswell nodes
- N.b. with 16 nodes (512 ranks): 15 minutes — with HepMC+Rivet as in plots above: 25 minutes



Summary and outlook

- Prototype for relatively efficient merged LO W+8j event simulation workflow
- For pragmatic reasons: Sherpa for ME level event generation and Pythia8 for particle level simulation
  - Store CPU expensive part (ME-level) on disk
  - Particle level run-time up to 4 orders of magnitude faster than ME
  - Main technologies used for parallelisation: DIY and HDF5
- Although we use technology aimed at HPC architectures, the code runs well on laptops, clusters etc.
- Want to understand scaling better, investigate with vtune
- Look at Z+jets, higgs, ttbar next.
- Would a hybrid strategy for event generation be a good idea?

Acknowledgement

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

Timing and memory usage (Sherpa 3.x.y + HDF5)

LO ME level event generation only (Comix; $\gamma, Z, h, \mu, \nu_\mu, \tau, \nu_\tau$ off)

Process $W^+ +$	1j	2j	3j	4j
RAM Usage	21 MB	43 MB	48 MB	85 MB
Init/startup time	<1s / <1s	<1s / <1s	2s / <1s	32s / <1s
Integration time	8×4m26s	16×16m42s	32×20m26s	64×1h32m
MC uncertainty	0.22%	0.46%	0.89%	0.97%
Unweighting eff	$6.59 \cdot 10^{-3}$	$7.50 \cdot 10^{-4}$	$2.71 \cdot 10^{-4}$	$1.47 \cdot 10^{-4}$
10k evts	1m 2s	15m 5s	1h 3m	5h 56m

Numbers generated on dual 8-core Intel® Xeon® E5-2660 @ 2.20GHz

Process $W^+ +$	5j	6j*	7j*	8j†
RAM Usage	189 MB	484 MB	1.32 GB	1.32 GB
Init/startup time	3m5s / 1s	24m52s / 5s	3h6m / 18s	5h55m / 29s
Integration time	128×4h38m	256×13h53m	512×19h0m	1024×23h8m
MC uncertainty	1.0%	0.99%	2.38%	4.68%
Unweighting eff	$9.56 \cdot 10^{-5}$	$7.66 \cdot 10^{-5}$	$7.20 \cdot 10^{-5}$	$7.51 \cdot 10^{-5}$
10k evts	24h 40m	2d 11h	10d 15h	78d 1h

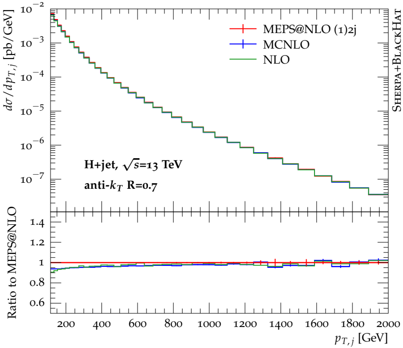
Numbers generated on dual 8-core Intel® Xeon® E5-2660 @ 2.20GHz

*,† Number of quarks limited to ≤6/4

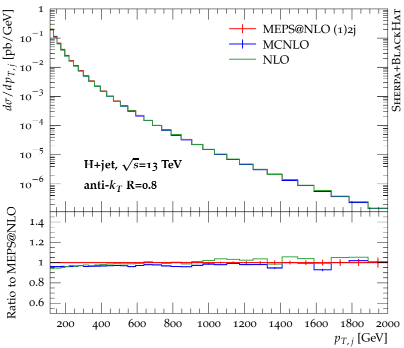
Plans for NLO event generation

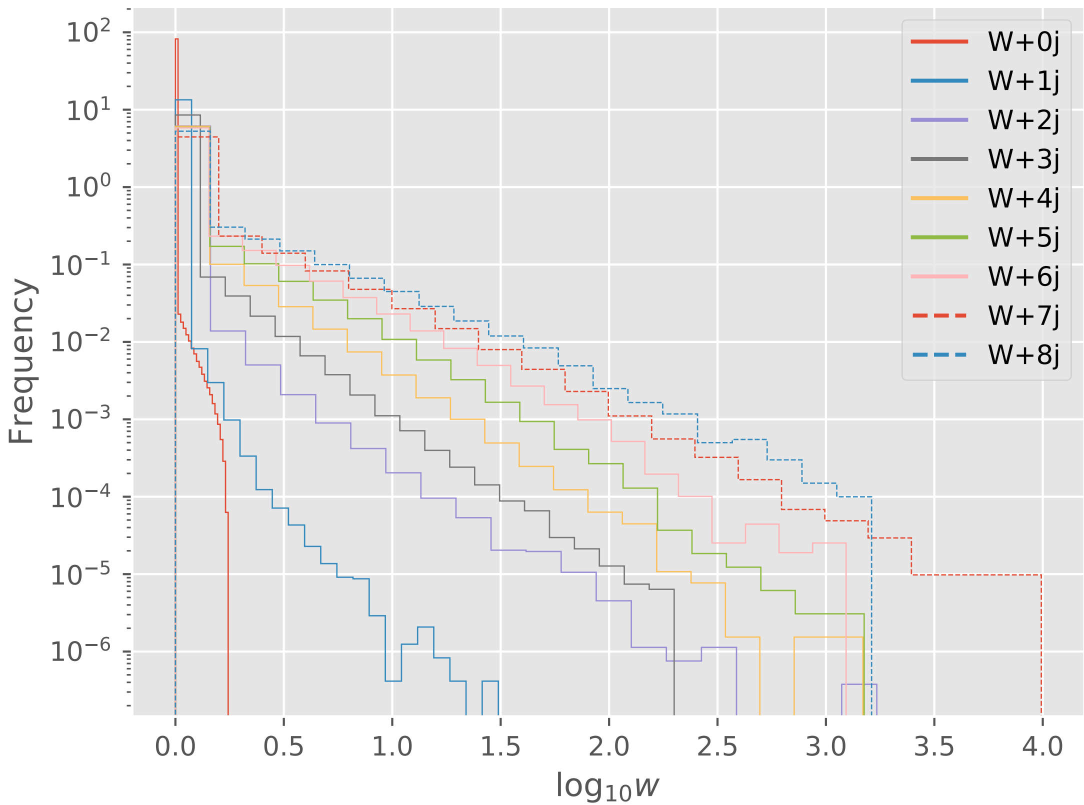
- For large class of processes, NLO fixed-order and MC@NLO agree well with each other and with MEPS@NLO (↗ e.g. plots below)
- Indicates best technical option: Store MC@NLO simulated events
  - Pro: Parton-shower independent results
  - Con: Restricted possibility for variations

- H+jet @ LHC 13 TeV



- Z+jet @ LHC 13 TeV





Frequency
10^{2}
10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
$\log_{10} w$
W+0j
W+1j
W+2j
W+3j
W+4j
W+5j
W+6j
W+7j
W+8j