GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN

Institut für Theoretische Physik

Computational developments of SHERPA and LHAPDF

Max Knobbe
QCD@LHC-22

Gefördert durch
DFG Deutsche Forschungsgemeinschaft

Reference

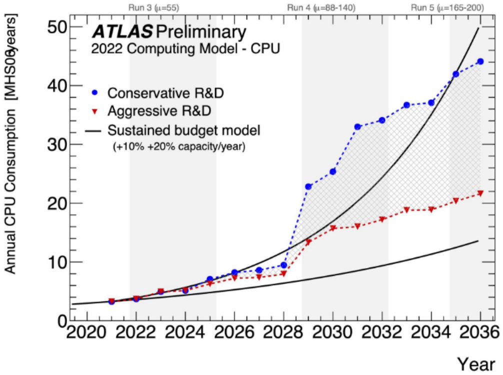
Based on:

Accelerating LHC event generation with simplified pilot runs and fast PDFs

E. Bothmann, A. Buckley, I. A. Christidi, C. Guetschow, Stefan Hoeche, MK, T. Martin, M. Schoenherr

arxiv:2209.00843

Introduction / Motivation



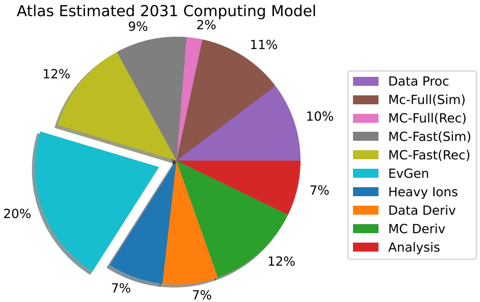


Figure and numbers taken from [CERN-LHCC-2022-005]

- Computing needs are predicted to grow faster than available resources
  $\rightarrow$ Computing budget might limit physics outcome
- Sizeable part of CPU budget spend on event generation (roughly 20%)
- LHC measurements in danger of being limited by Monte Carlo statistics

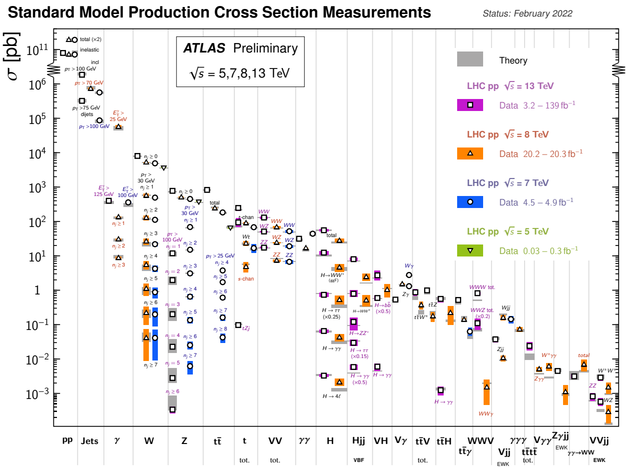
Boiling down the Problem

Expensive MC Samples: V+Jets, $t\bar{t}$+Jets

- Background to essential analysis(es), e.g. Higgs-boson and top-quark measurements
  cf. [2112.09588]
- extremely large event sample sizes
- To reduce significant portion of MC budget, ensure to be efficient for these processes

Performance dependence on the number of multiweights studied using different setups

- baseline MEPS@NLO (no variations)
- + EW_{virt} corrections
- + 7-point variations of factorisation and renormalisation scales in matrix element and parton shower
- + 100 (1000) NNPDF3.0nnlo replicas

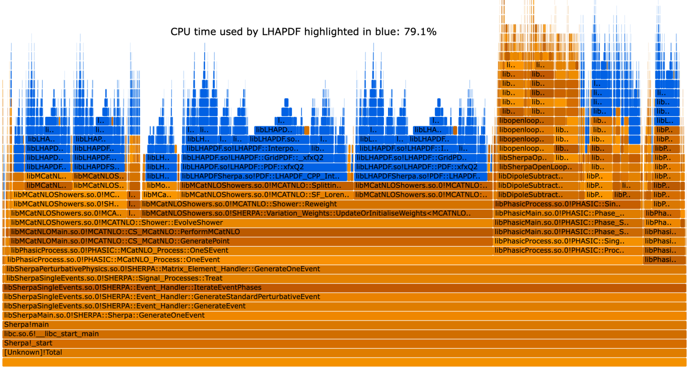


twiki.cern.ch/twiki/bin/view/AtlasPublic/StandardModelPublicResults

Boiling down the Problem

- Base setup:
  - Sherpa 2.2.11
  - OpenLoops 2.1.2
  - LHAPDF 6.2.3
- Per-event CPU time dominated by LHAPDF
  $\to$ Graph shows PDF calls in blue
- Not completely surprising
  $\to$ multiweights not designed with hundreds of variations in mind [arXiv:1606.08753]

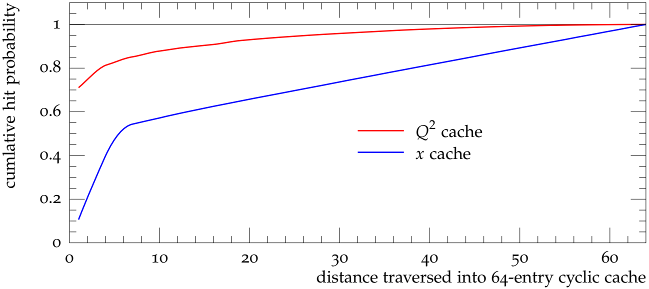
MEPS@NLO $pp \to e^+e^- + 0,1,2j$@NLO $+ 3,4,5j$@LO
$\mathrm{EW_{virt}}$ + scales + 100 PDFs



$\Rightarrow$ Explore two approaches in parallel: make LHAPDF faster and rework LHAPDF call strategy

Improving LHAPDF

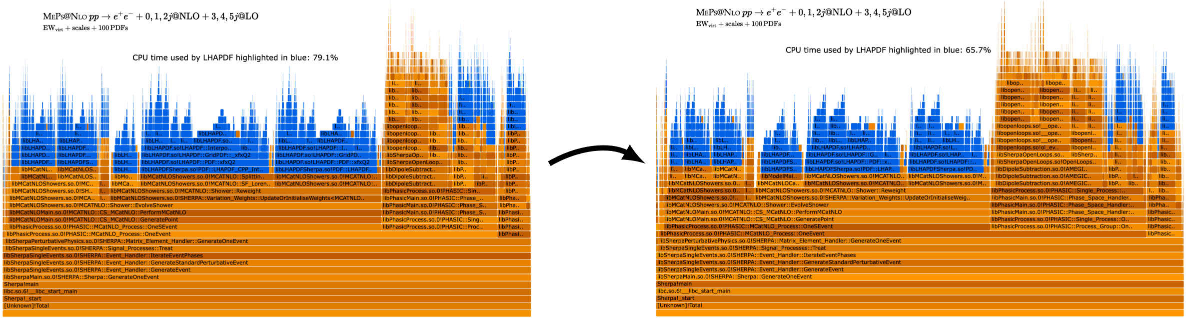
- First approach: cache grid computations
  $\rightarrow$ Introduced in v.6.3.0
  $\rightarrow$ Rendered ineffective by Sherpa call-strategy
  $\rightarrow$ Useful as initial study
- Follow-up release v6.4.0 with improved interpolation logic



- Revised cache implementation with improved memory layout
  $\rightarrow$ (but well-matched call strategy in the generator still crucial)
- Pre-computation of shared coefficients of the interpolation polynomial along (x, Q^2) grid lines
- Results in factor 3 speed-up for single flavour computations
- Can achieve factor 10 speed-up when combining with multi-flavour caching

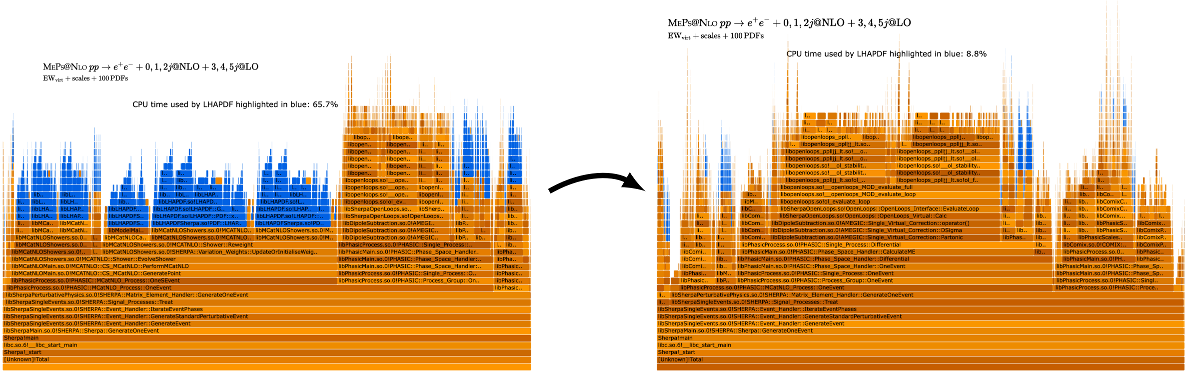
Improving LHAPDF: Results

- ATLAS V+jets setup overall 30% faster using new LHAPDF release



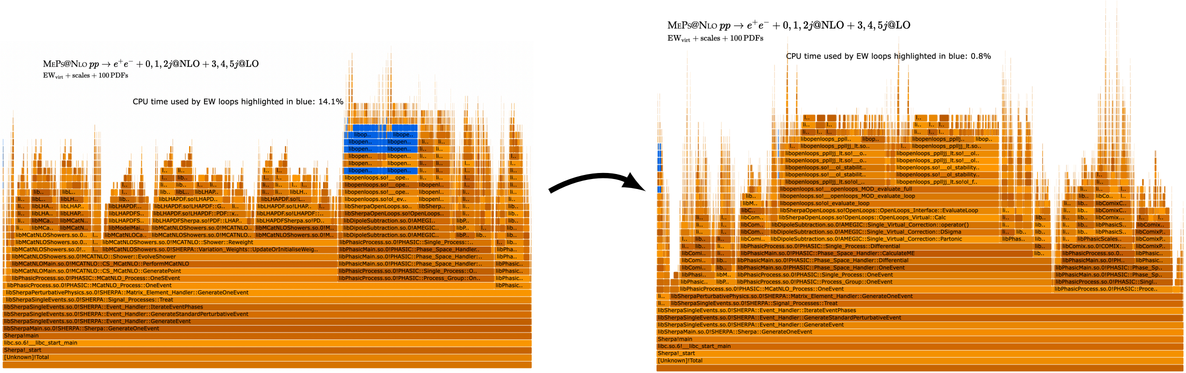
Reworking the Sherpa call strategy: Introducing the pilot-run

- Perform the unweighting using a minimal setup
  $\rightarrow$ once event is accepted, re-calculate accepted event using all the bells and whistles
- Achieves factor 5 speed improvement for ATLAS setup
  (using LHAPDF 6.4.0 yields additional 6% speed-up)
- Pilot run reduces CPU spent on evaluating PDFs to below 10



Reworking the Sherpa call strategy: Introducing the pilot-run

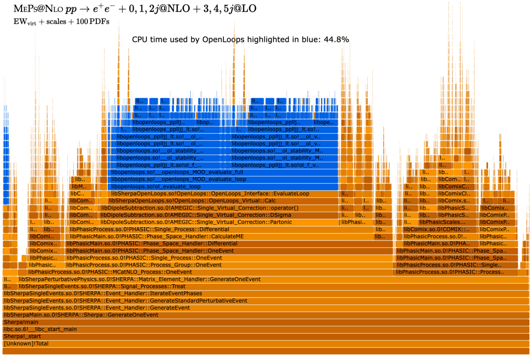
- Can also move EW-corrections out of the unweighting loop
- CPU spent on calculating EW one-loop amplitudes going from 19% down to 0.8% when using the pilot run with the ATLAS V +jets setup
- Remaining CPU time: 40% of the CPU spent on calculating QCD loops



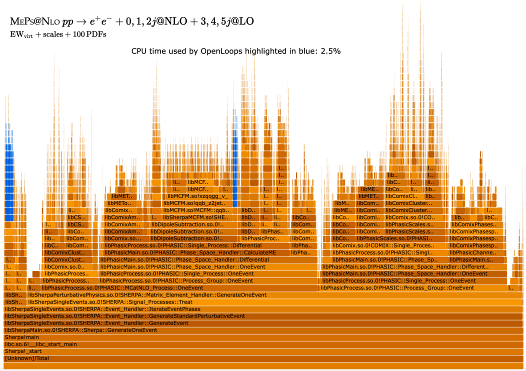
Reworking the Sherpa call strategy: Introducing the pilot-run

- If available use analytic loop amplitudes
  → possible via Sherpa-MCFM interface [arXiv:2107.04472]
- yields additional 35% speed improvement for the V +jets setup

MePs@Nlo $pp \to e^+e^- + 0,1,2j$@NLO $+ 3,4,5j$@LO
EW$_{\text{virt}}$ + scales + 100 PDFs



MePs@Nlo $pp \to e^+e^- + 0,1,2j$@NLO $+ 3,4,5j$@LO
EW$_{\text{virt}}$ + scales + 100 PDFs



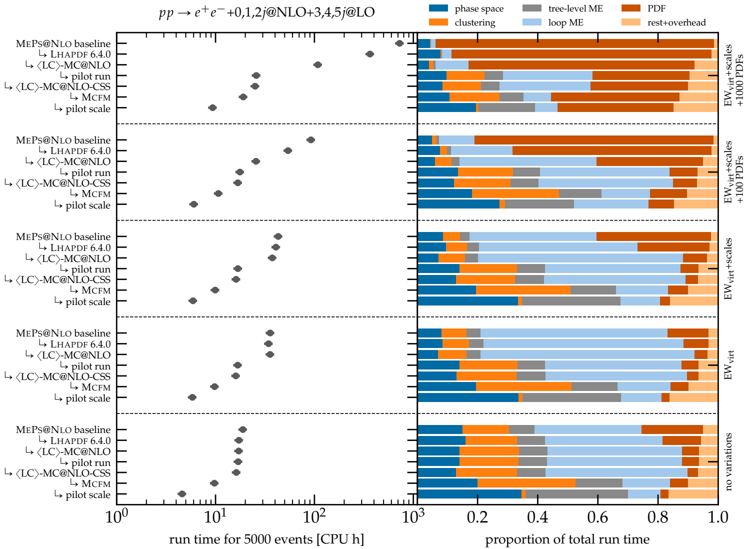
Putting it all together

Study the impact of different improvements sequentially:

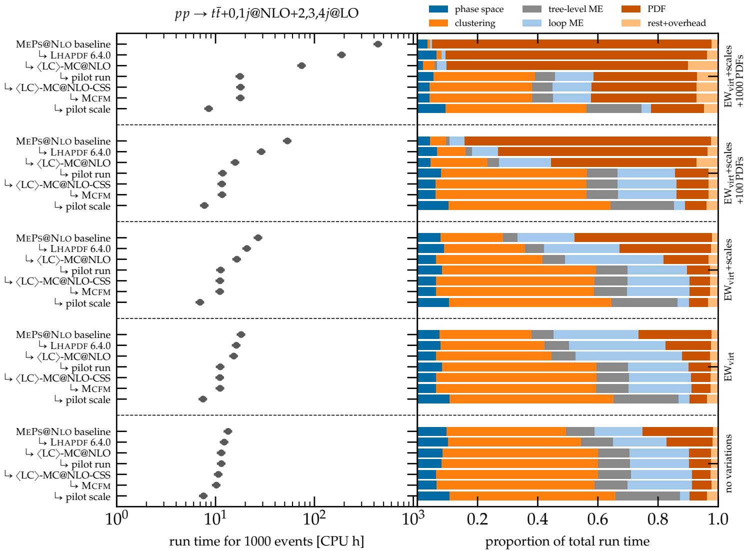
- Improved interpolation strategies in LHAPDF (6.2.3 $\rightarrow$ 6.4.0)
- Replace full-colour spin-correlated S-MC@NLO algorithm with leading-colour spin-averaged $\langle$LC$\rangle$-MC@NLO
- Introduce pilot run in Sherpa (2.2.11 2.2.12)
- Use analytic one-loop amplitudes from MCFM in pilot run
- Use a simplified pilot scale for the unweighting

	$pp \rightarrow e^+e^-$ + jets			$pp \rightarrow t\bar{t}$ + jets		
	runtime [CPU h/5k events]			runtime [CPU h/1k events]		
setup variant	old	new	speed-up	old	new	speed-up
no variations	20 h	5 h	4×	15 h	8 h	2×
$\mathrm{EW}_{\mathtt{Virt}}$	35 h	5 h	6×	20 h	8 h	2×
$\mathrm{EW}_{\mathtt{Virt}}$+scales	45 h	5 h	7×	25 h	8 h	4×
$\mathrm{EW}_{\mathtt{Virt}}$+scales+100 PDFs	90 h	5 h	15×	55 h	8 h	7×
$\mathrm{EW}_{\mathtt{Virt}}$+scales+1000 PDFs	725 h	8 h	78×	440 h	9 h	51×

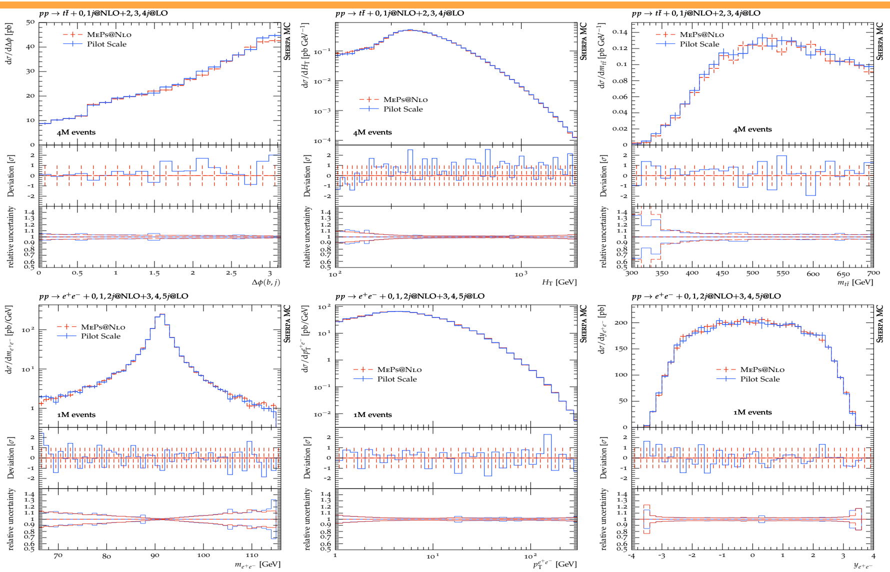
Breakdown of CPU budget in V+jets



Breakdown of CPU budget in $t\bar{t}$+jets

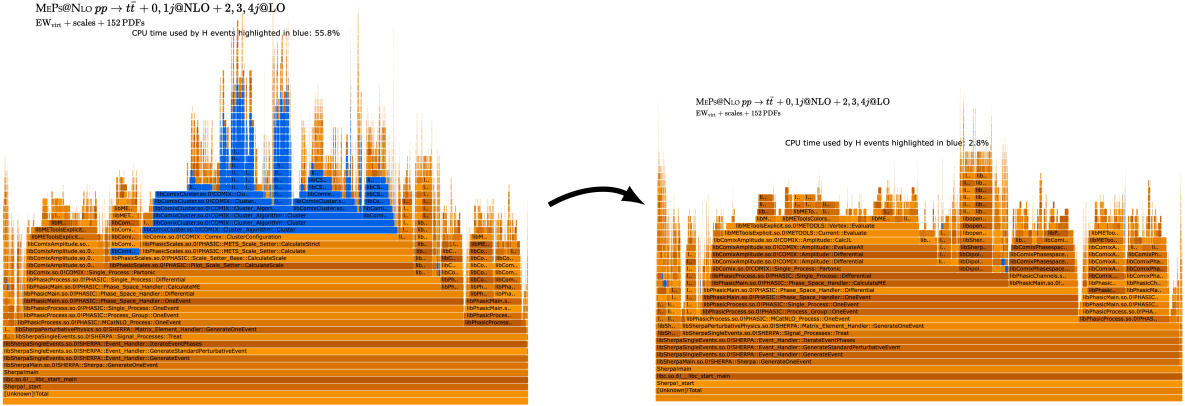


Comparison of MEPS@NLO vs Pilot Scale strategy



Impact Cluster-independent scale definition

- Employ clustering-independent scale definition (HT/2) for H-events in tt+jets
  $\rightarrow$ (already used in V +jets baseline setup)
- Yields additional factor 2 speed-up of the overall run time



Conclusion

In this study:

- Latest LHAPDF release series brings major performance improvements with noticeable impact on overall event-generation run time
- Introduction of pilot run in Sherpa brings a factor 5 improvement
- Using analytic QCD loop amplitudes in the unweighting brings another factor 1.5
- Overall factor 40 speed-up in case of ATLAS baseline configuration
- Achieves major factor-10 milestone set by HEP Software Foundation

Future R&D:

- Still desirable, facilitate more complicated computations $\rightarrow$ Fast turnover times fuel pheno studies and further developments
- New architectures, e.g. GPU's
  $\rightarrow$ Keeping pace with changing HPC landscape
  $\rightarrow$ Developments in both Sherpa & LHAPDF