

# Accelerating LHC event generation with simplified pilot runs and fast PDFs

Enrico Bothmann (Göttingen), Andy Buckley (Glasgow),  
Christian Gütschow (UCL), Stefan Höche (FNAL),  
Max Knobbe (Göttingen), Marek Schönherr (Durham)

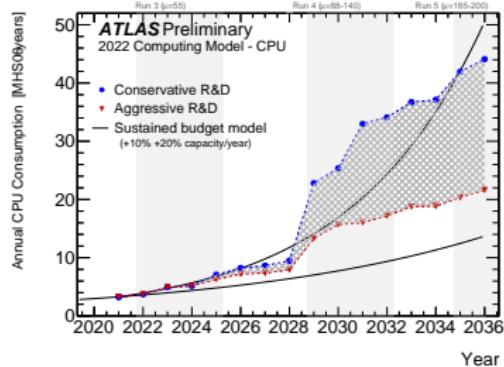
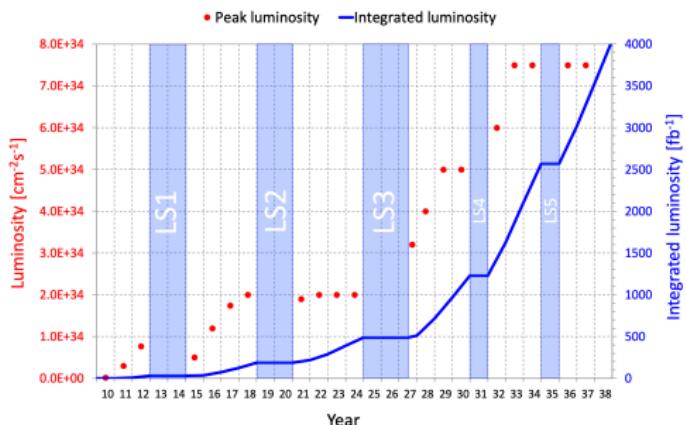
ACAT 2022, Bari, Italy

24 October 2022

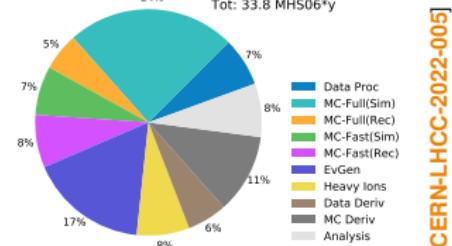


## Expected computing requirements

- latest update to the projected evolution of computing resources sees cost of event generation on par with detector simulation
- LHC measurements in danger of being limited by Monte Carlo statistics



ATLAS Preliminary  
2022 Computing Model - CPU: 2031, Conservative R&D  
24%  
Tot: 33.8 MHS06'y



[CERN-LHCC-2022-005]

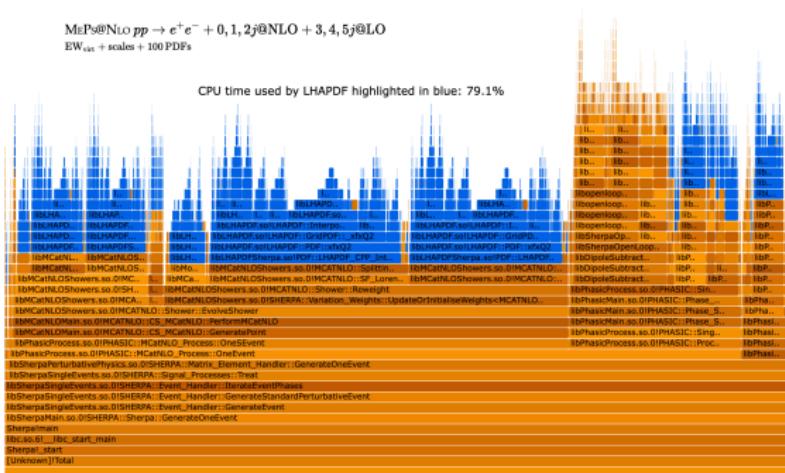
## Systematic profiling

- Most event generation CPU spent on multi-leg NLO calculations [[arXiv:2112.09588](#)]
  - used for main Standard Model processes
  - relevant to measurements and searches alike
  - extremely large event sample sizes
- Study CPU performance of MEPS@NLO calculations for  
 $e^+e^- + 0, 1, 2j$ @NLO+3, 4, 5j@LO and  $t\bar{t} + 0, 1j$ @NLO+2, 3, 4j@LO  
with Sherpa 2.2.11, OpenLoops 2.1.2 and LHAPDF 6.2.3 using VTune 2021.7.1
- performance dependence on the number of multiweights studied using different setups:
  - baseline MEPS@NLO (no variations)
  - + EW<sub>virt</sub> corrections
  - + 7-point variations of factorisation and renormalisation scales  
in matrix element and parton shower
  - + 100 (1000) NNPDF3.0nnlo replicas
- detailed write-up presented in [[arXiv:2209.00843](#)]

## Initial profiling exercises

- first generator CPU profiling done by Tim Martin suggested per-event CPU dominated by LHAPDF

MEPs@NLO  $pp \rightarrow e^+e^- + 0, 1, 2j$ @NLO + 3, 4, 5j@LO  
 EW,  $\gamma\gamma$  + scales + 100 PDEs

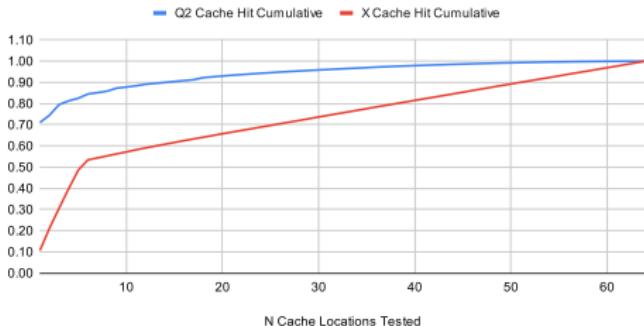


- explore two approaches in parallel: make LHAPDF faster and rework LHAPDF call strategy

## Improving LHAPDF

- first PDF-grid cache introduced in v6.3.0

- rendered ineffective by PDF-call strategy used in Sherpa
- nevertheless useful as case study

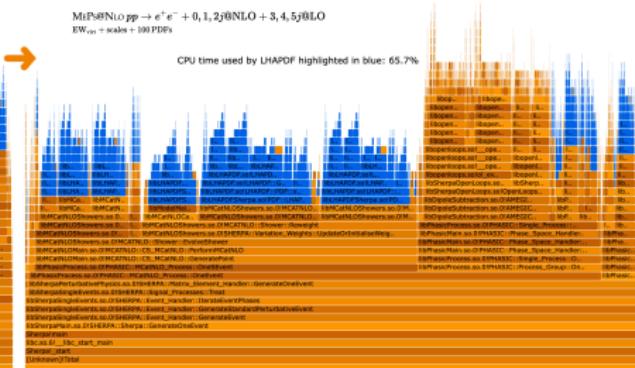
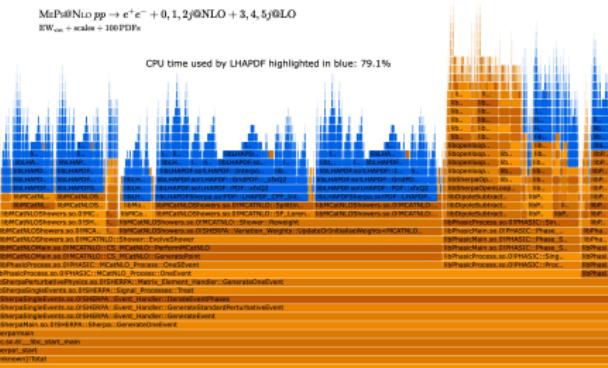


- follow-up release v6.4.0 with improved interpolation logic

- revised cache implementation with improved memory layout (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

## Impact of new LHAPDF

- ATLAS  $V+jets$  setup **overall 30% faster** using new LHAPDF release
  - switching from old ATLAS production default v6.2.3 to new v6.4.0 release

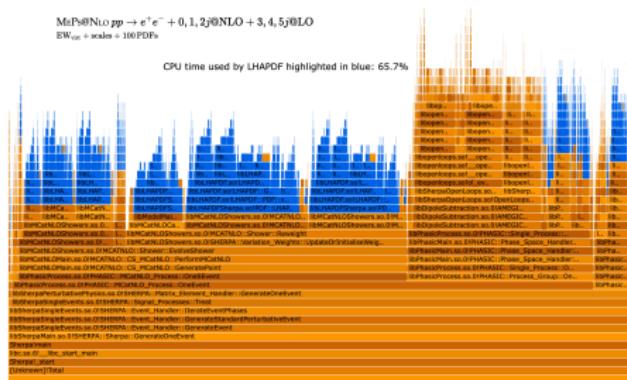


## Internal restructuring in Sherpa 2.2.12: the pilot run

- perform the unweighting using a minimal setup and once an event is accepted, rewind RNG state and 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%

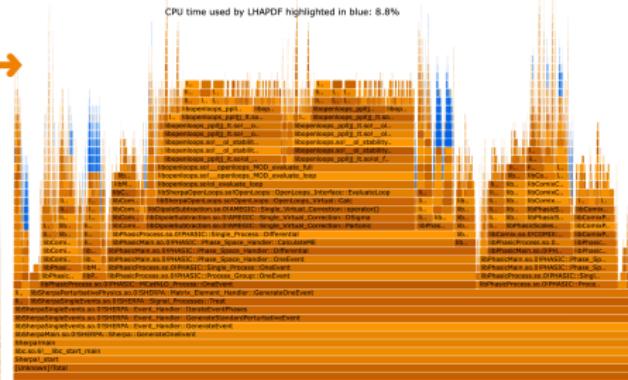
MnPs@NLO  $pp \rightarrow e^+e^- + 0, 1, 2j@NLO + 3, 4, 5j@LO$   
 $EW_{\text{sys}} + \text{scales} + 100 \text{ PDFs}$

CPU time used by LHAPDF highlighted in blue: 65.7%



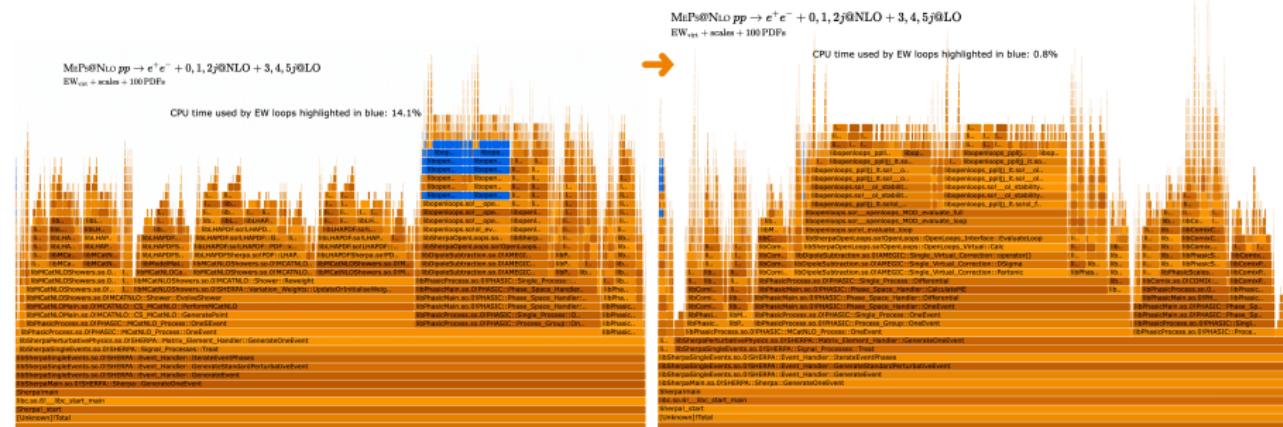
MnPs@NLO  $pp \rightarrow e^+e^- + 0, 1, 2j@NLO + 3, 4, 5j@LO$   
 $EW_{\text{sys}} + \text{scales} + 100 \text{ PDFs}$

CPU time used by LHAPDF highlighted in blue: 8.8%



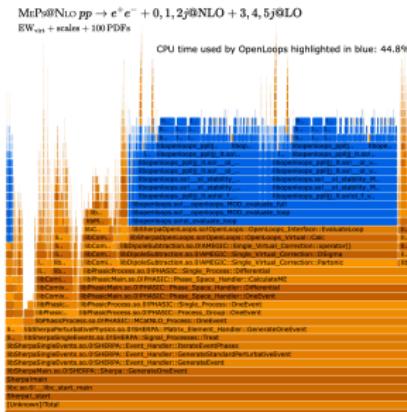
## Internal restructuring in Sherpa 2.2.12: the pilot run

- 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
  - nevertheless,  $\sim 40\%$  of the CPU still spent on calculating QCD loops



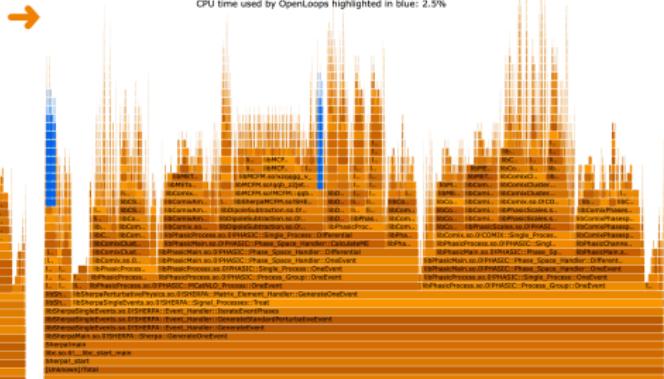
## Analytic vs numerical QCD loop amplitudes

- employ analytic one-loop amplitudes (if available) in the pilot run using Sherpa-MCFM interface [arXiv:2107.04472]
- yields **additional ~35% speed improvement** for the  $V+jets$  setup



MiPs@NLO  $pp \rightarrow e^+e^- + 0,1,2j@NLO + 3,4,5j@LO$   
 $EW_{\text{ext}} + \text{scales} + 100 \text{ PDFs}$

CPU time used by OpenLoops highlighted in blue: 2.5%

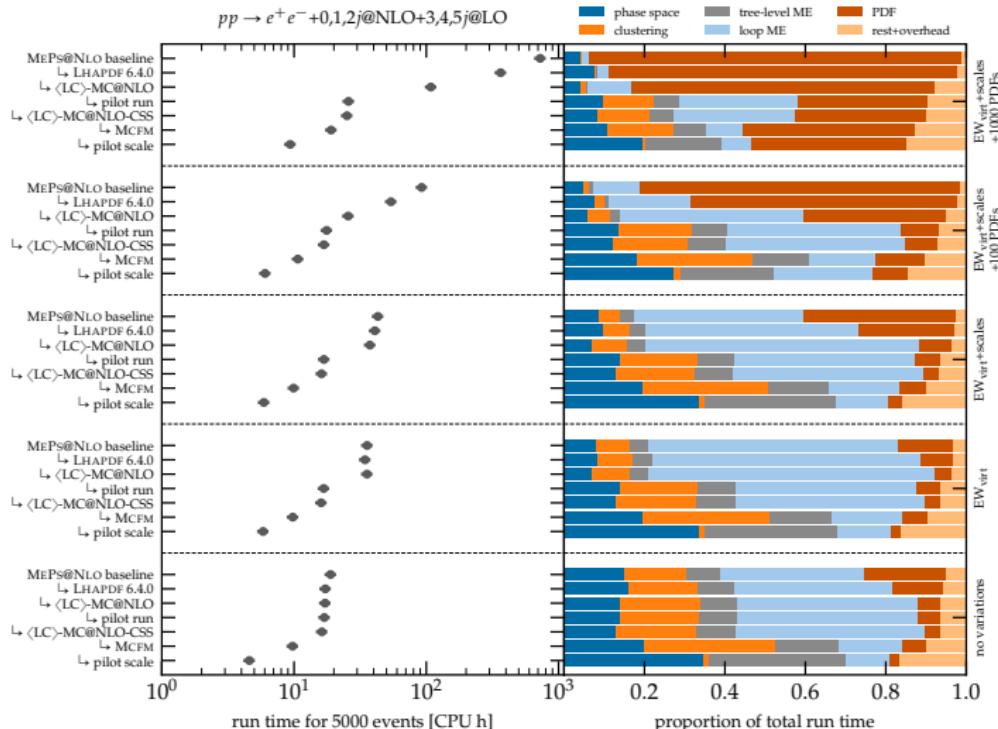


## Full suite of improvements

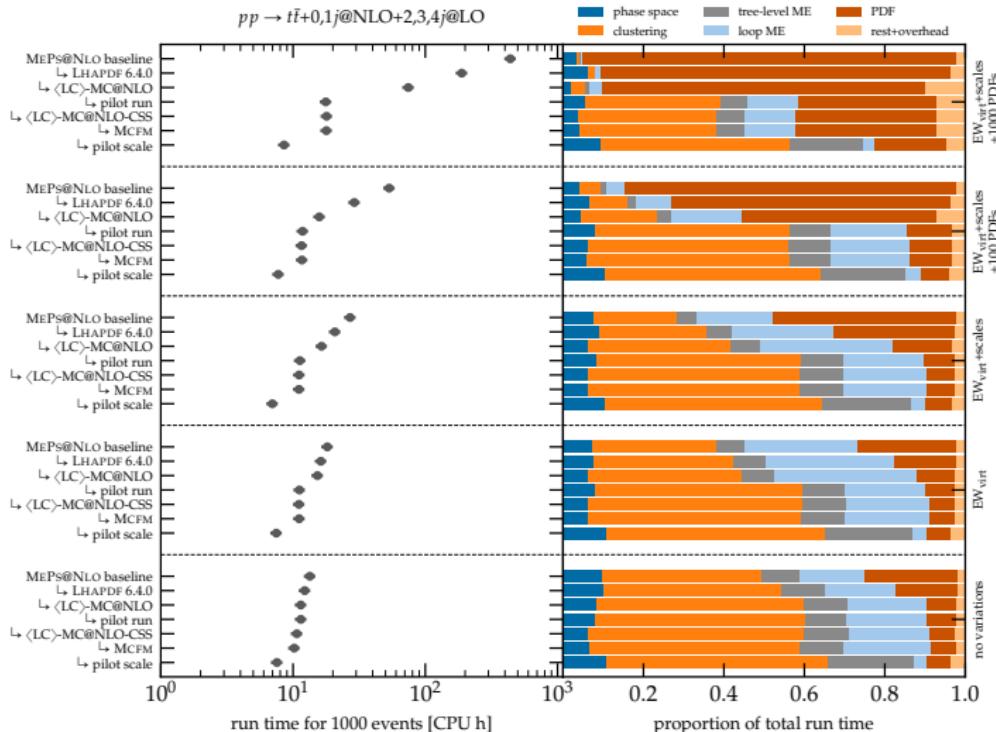
- study the impact of different improvements sequentially:
  - improved interpolation strategies in LHAPDF (6.2.3 → 6.4.0)
  - replace full-colour spin-correlated S-MC@NLO algorithm with leading-colour spin-averaged  $\langle LC \rangle$ -MC@NLO (`NLO_CSS_PSMODE 0 → 1`)
    - this disables subleading colour corrections in the parton shower
  - introduce pilot run in Sherpa (2.2.11 → 2.2.12)
  - defer leading-colour MC@NLO until after the unweighting (`NLO_CSS_PSMODE 1 → 2`)
  - use analytic one-loop amplitudes from MCFM in pilot run
  - use a simplified pilot scale for the unweighting

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

## Breakdown of CPU budget in $V+jets$

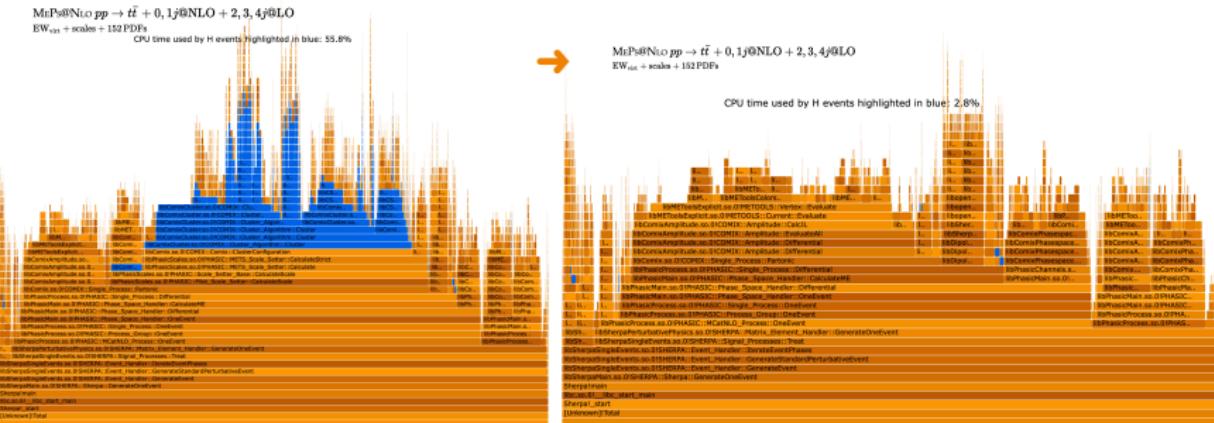


# Breakdown of CPU budget in $t\bar{t}+{\rm jets}$

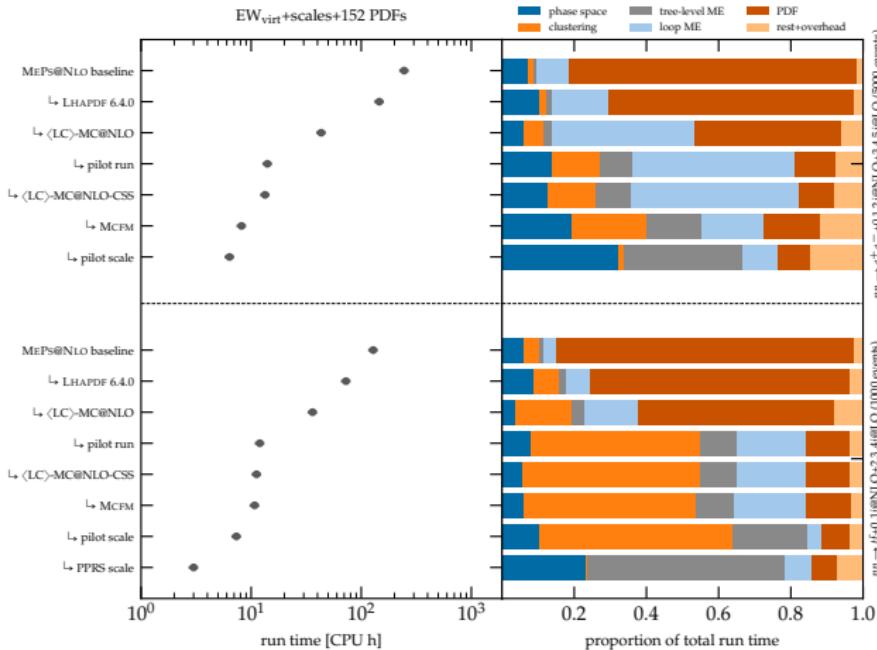


## Cluster-independent scale definition

- employ clustering-independent scale definition ( $H_T'/2$ ) for  $\mathcal{H}$ -events in  $t\bar{t}+{\rm jets}$  (already used in  $V+{\rm jets}$  baseline setup)
- yields **additional factor 2 speed-up** of the overall run time

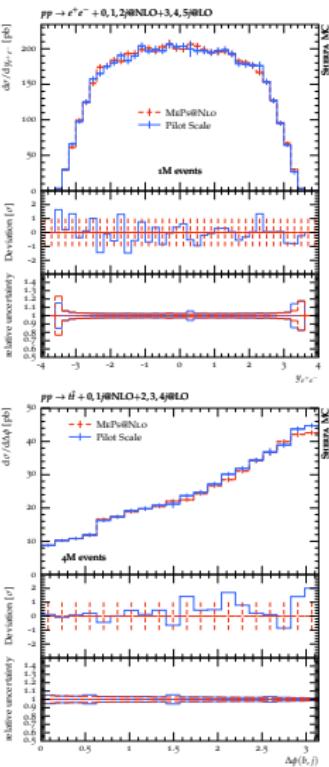
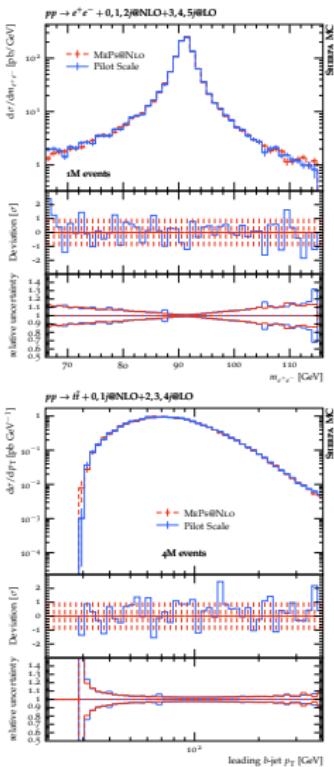
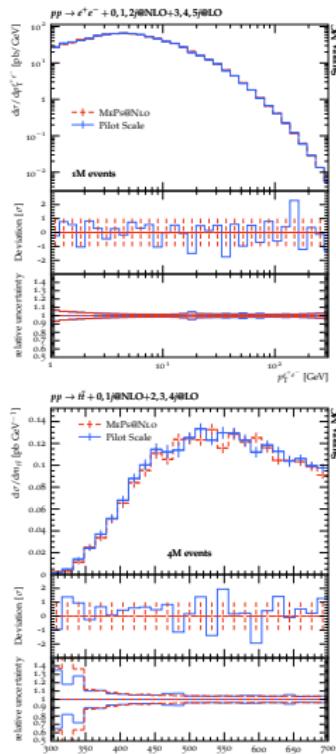


## Case study: latest ATLAS baseline configuration



→ CPU consumption **overall improved by factors of ×39 and ×43** for V+jets and tt+jets

# Comparison of MEPS@NLO vs Pilot Scale strategy

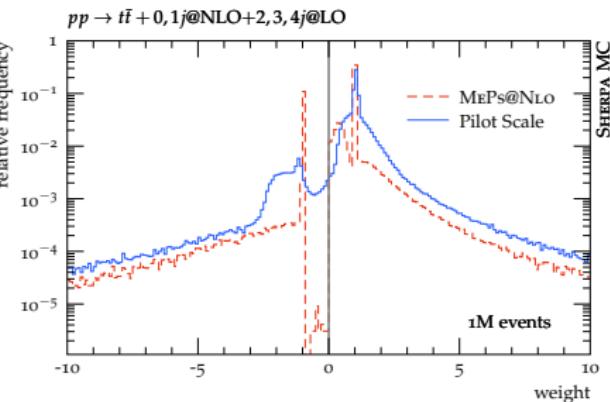
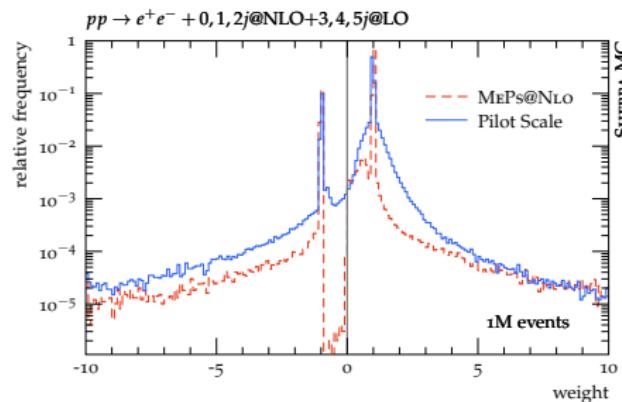


## Summary

- 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

## Weight distribution for pilot scale

- weight distributions for partially unweighted events after matching and merging:



- second unweighting would reduce the efficiency by less than factor 2 for large  $N_{\text{events}}$