# Workflow developments for interpolation grids with NNLOJET and APPLfast

Klaus Rabbertz (KIT)
(APPLfast team)

# Outline

- Workflow overview
- CPU time & statistical precision
- Interpolation in x (Master student Johannes Gäßler)
- Workflow on lxplus (Summer student Christiane Mayer)

# The Components

- **NNLOJET:**
  Theory program providing NNLO predictions
- **APPLfast:**
  Common interface of APPLgrid & fastNLO to pQCD theory programs
- **APPLgrid, fastNLO:**
  Creation of fast interpolation grids

*NNLOJET*

NNLOJET, A. Gehrmann-de Ridder et al., Z+jet: JHEP07 (2016) 133,
Framework: T. Gehrmann et al., PoS RADCOR2017 (2018) 074.

APPLfast, D. Britzger et al., ch. I.3 in arXiv:1803.07977.

the APPLgrid project fastNLO

APPLgrid, T. Carli et al., EPJC66 (2010) 503,
FastNLO, D. Britzger et al., arXiv:1208.3641.

- **Luigi & Luigi Analysis Workflow (LAW):**
  Design of distributed, pipelined analysis workflow with inter-dependencies
- **NNLO LAW Analysis & Website:**
  Example application to NNLO calculations with integrated plot server

Luigi law luigi analysis workflow

Luigi (Spotify): https://github.com/spotifi/luigi
LAW: https://github.com/riga/law

LAW workflow with NNLOJET & APPLfast
https://gitlab.etp.kit.edu/qcd-public/law-analysis

M. Santos Correa, Master Thesis, EKP-2019-00003.pdf.

# Luigi and Spotify

**Python package to build complex pipelines of batch jobs:**

- **Features:**
  - **Dependency resolution**
  - **Workflow management**
  - **Visualisation**
  - **Handling failures**
  - **Command line integration**
  - **...**

# *Luigi analysis workflow*

**Built on top of Luigi, adds abstraction of run locations, storage locations, and software environments:**

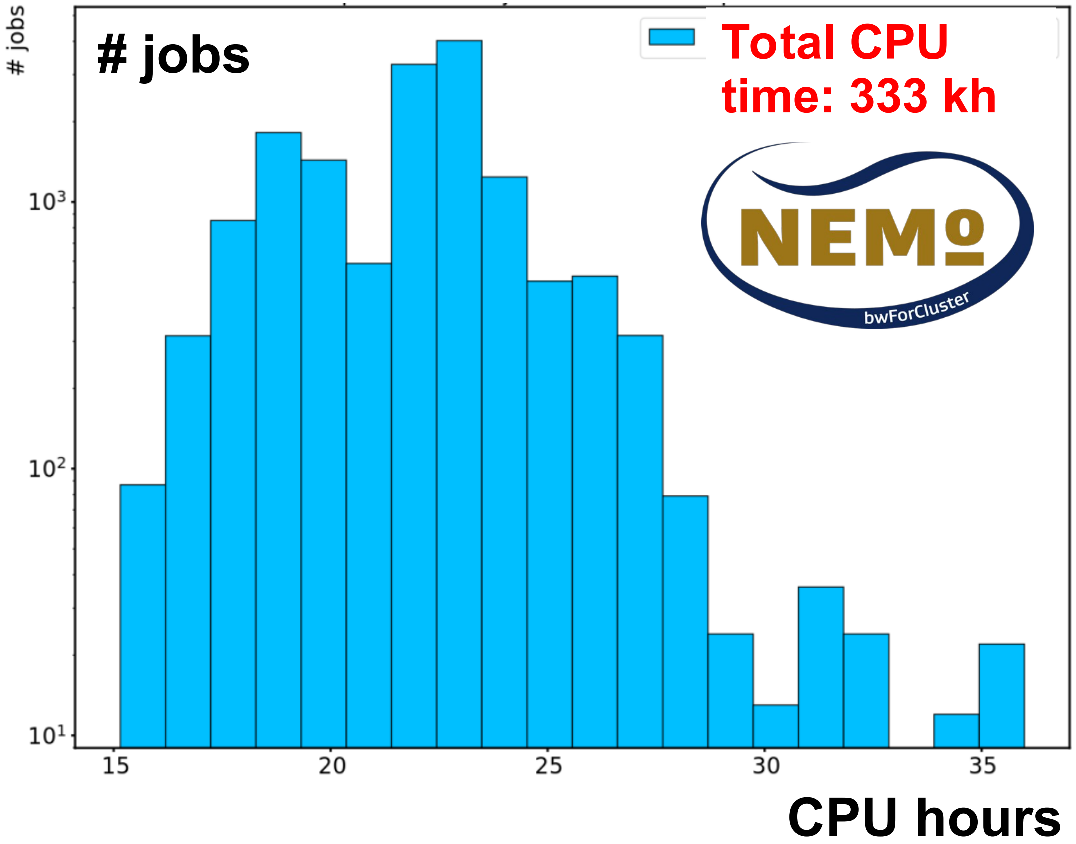
- **Features:**
  - **Remote targets with automatic retries and local caching**
    (WebDAV, HTTP, Dropbox, WLCG protocols: srm, xrootd, dcap ...)
  - **Automatic submission to batch systems within tasks**
    (HTCondor, LSF, gLite, Slurm (not tested yet), ...)
  - **Environment sandboxing, configurable on task level**
    (Docker, Singularity, ...)

# A typical workflow

- **1. Preprocessing:** Check of interpolation quality — FTE time
  - Short test jobs to check interpolation settings (& optimise if necessary)
- **2. NNLOJET Warm-up:** Vegas integration optimisation — CPU & FTE time
  - 1 long (multi-core) job per process
- **3. APPLgrid/fastNLO Warm-up:** Adapt x- and scale-grids to accessed phase space (exact strategy differs between APPLgrid & fastNLO) — FTE time
  - Only phase space provided from NNLOJET → significant speed-up
- **4. Interpolation grid production:** — A lot of CPU time!
  - Thousands of parallel jobs
- **5. Postprocessing:** Statistical evaluation and combination of all produced grids … — CPU time
  - Job to combine all grids and estimate statistical uncertainty
- **6. Provide final closure & result plots** — No time
  - Automated plotting scripts

# The investment

## Typical total runtime of a grid production




**Total # of ~24h jobs: O(15000)**

**Non-CMS CPU resources from:**
**NEMO Freiburg, MPI Munich, CERN Theory**

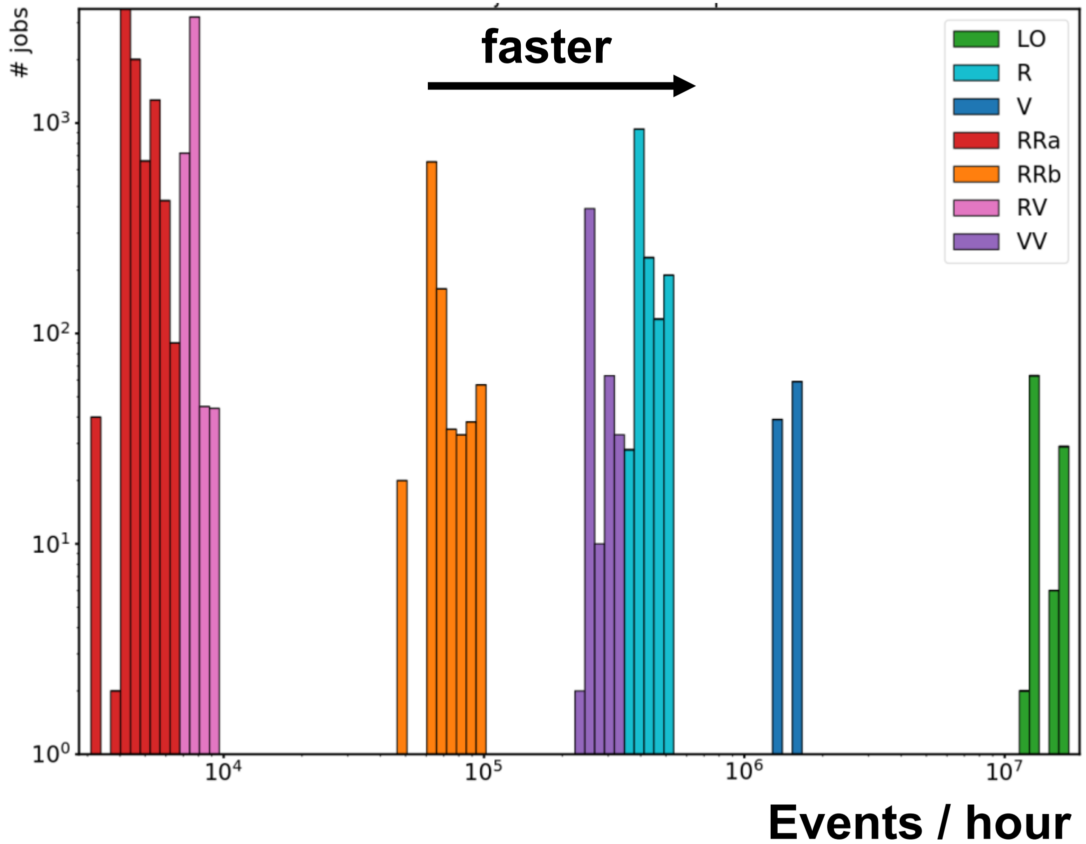
## "Events" / hour

**RRa, RV: most expensive**

**O($10^3$-$10^4$)**

**LO, NLO, VV: easy**

**O($10^5$ - $10^7$)**

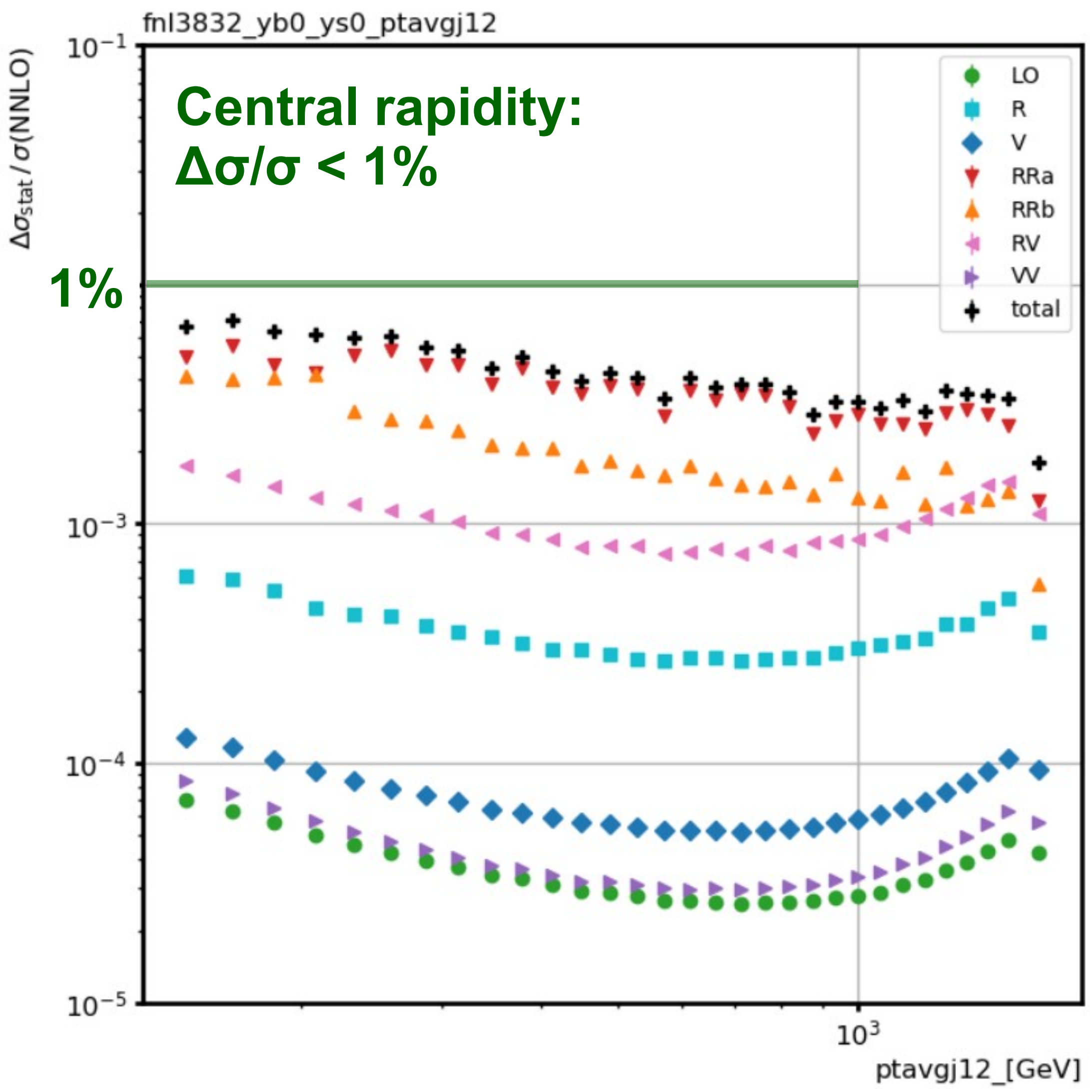


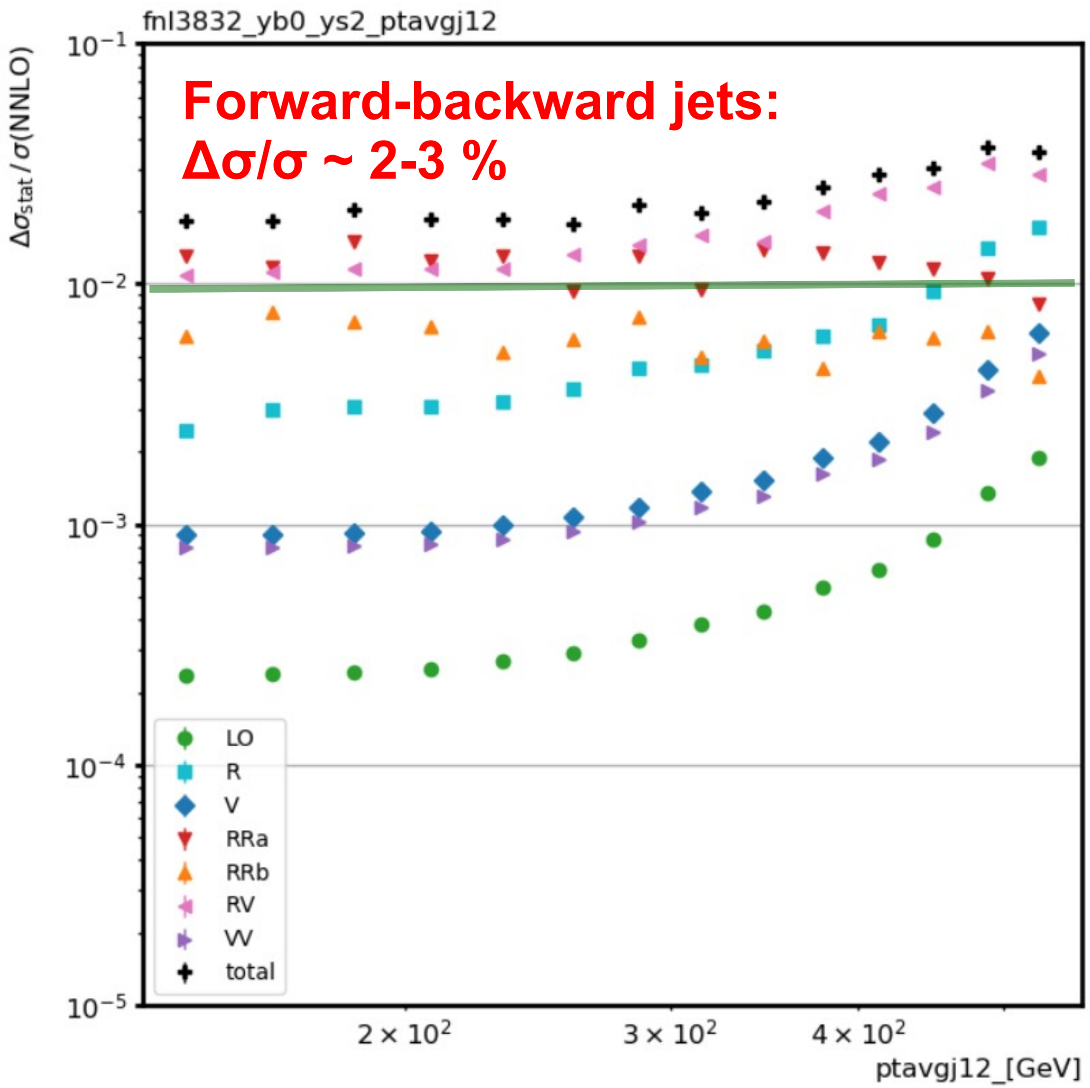
# Return on investment

**Relative numerical uncertainty of NNLO 3D dijet cross section**

**Dominated by RRa and RV channels**

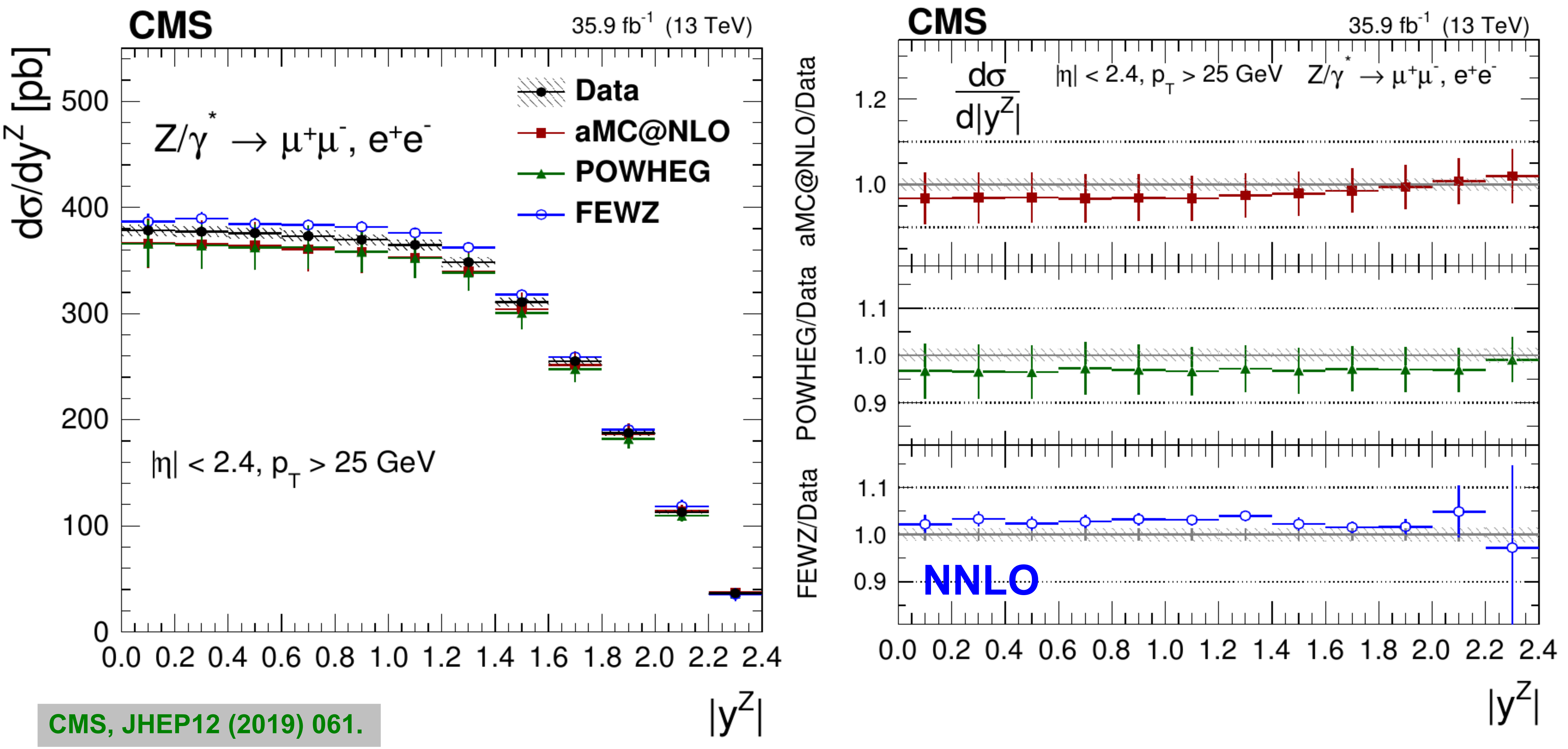
**Numerical uncertainty provided inside grids**



# Summer student project

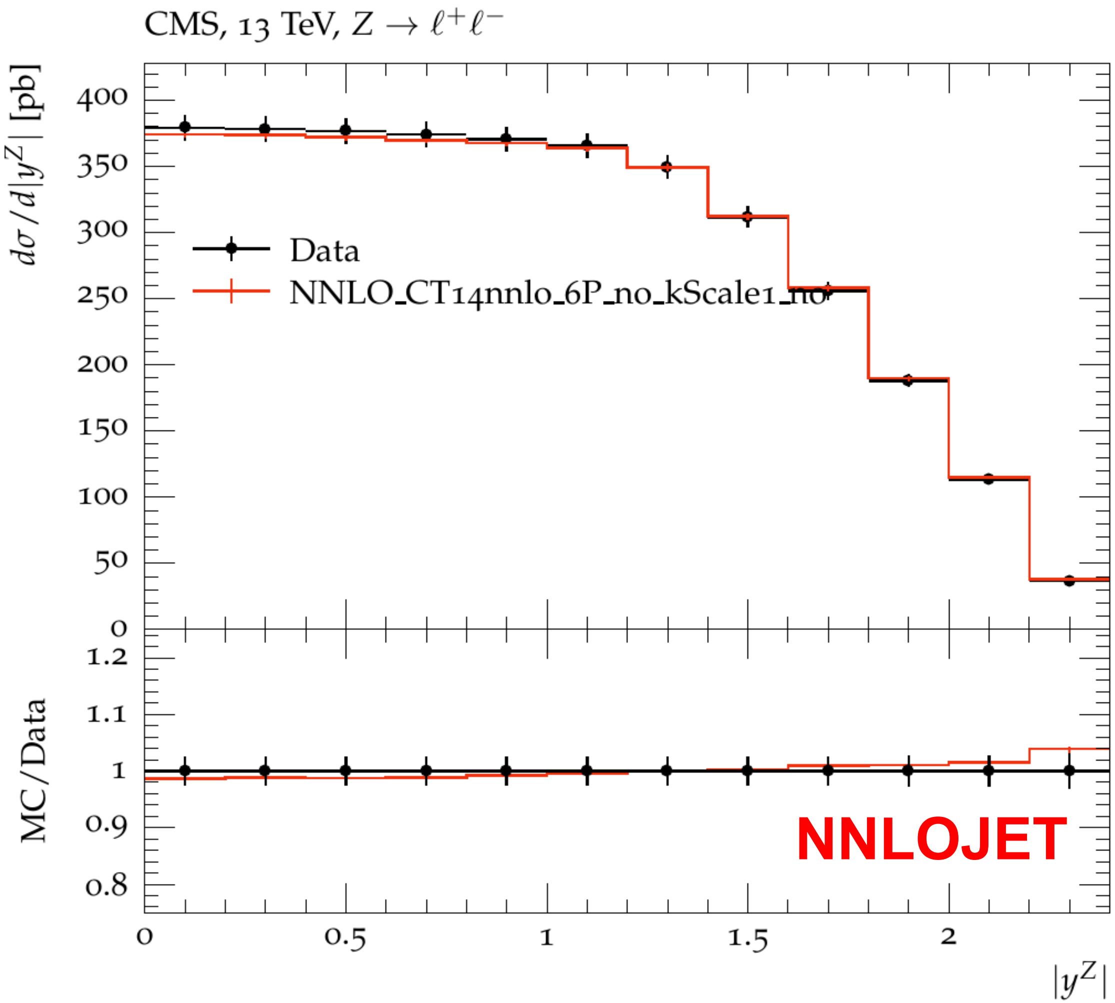
**Christiane Mayer: - Establish grid production workflow (LAW) on lxplus**
**(with Alexander) - Test automated VEGAS grid optimisation**

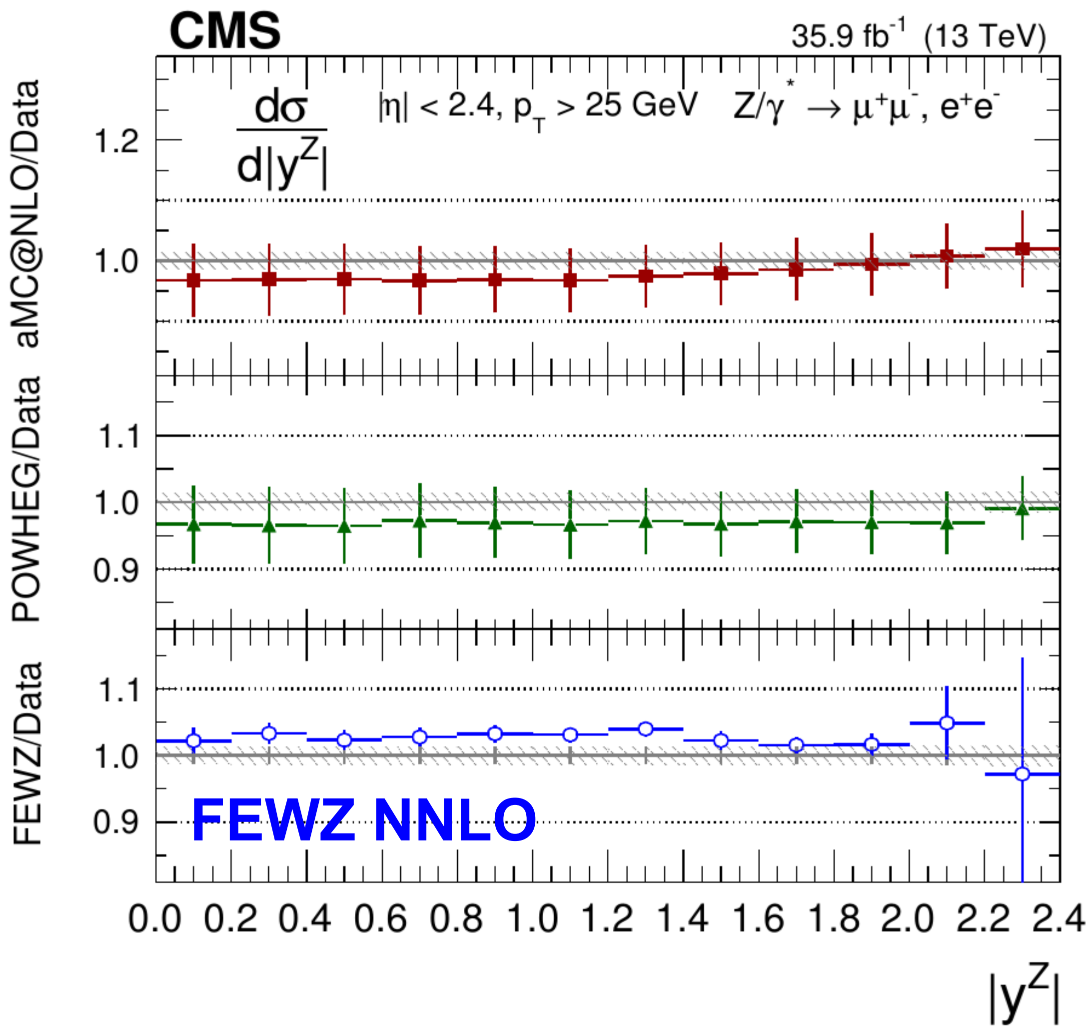



CMS, JHEP12 (2019) 061.

# Data comparison

**Christiane Mayer:** - Ran over night
**(with A. Huss, TH)** - Nicely agrees with CMS data
- Slightly lower than FEWZ



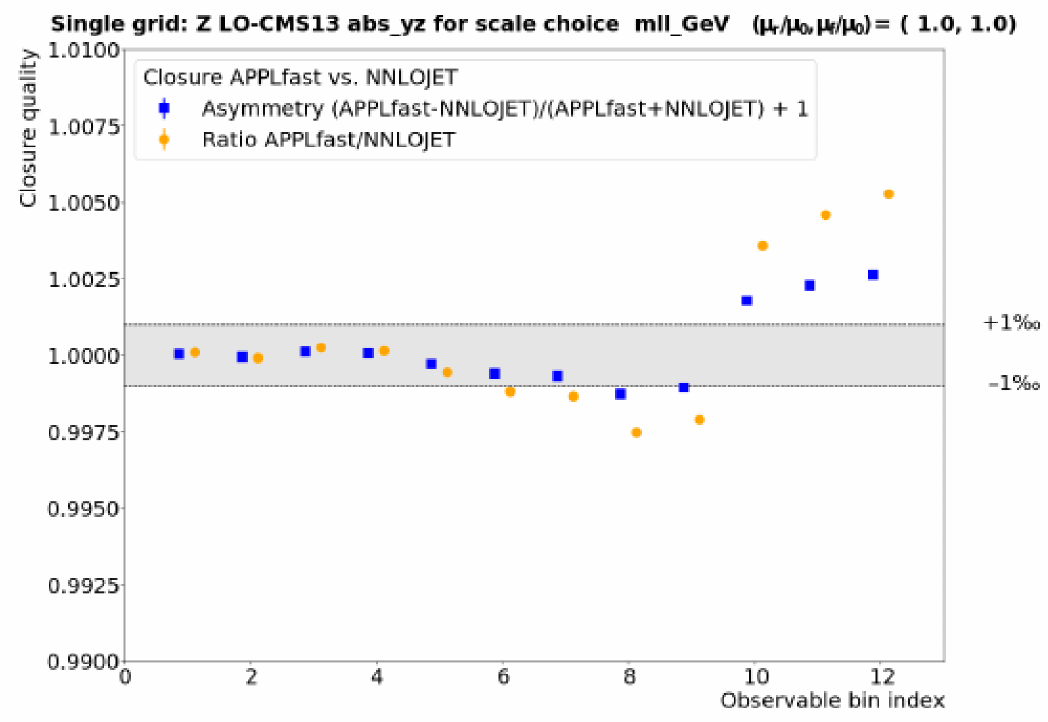

Christiane Mayer

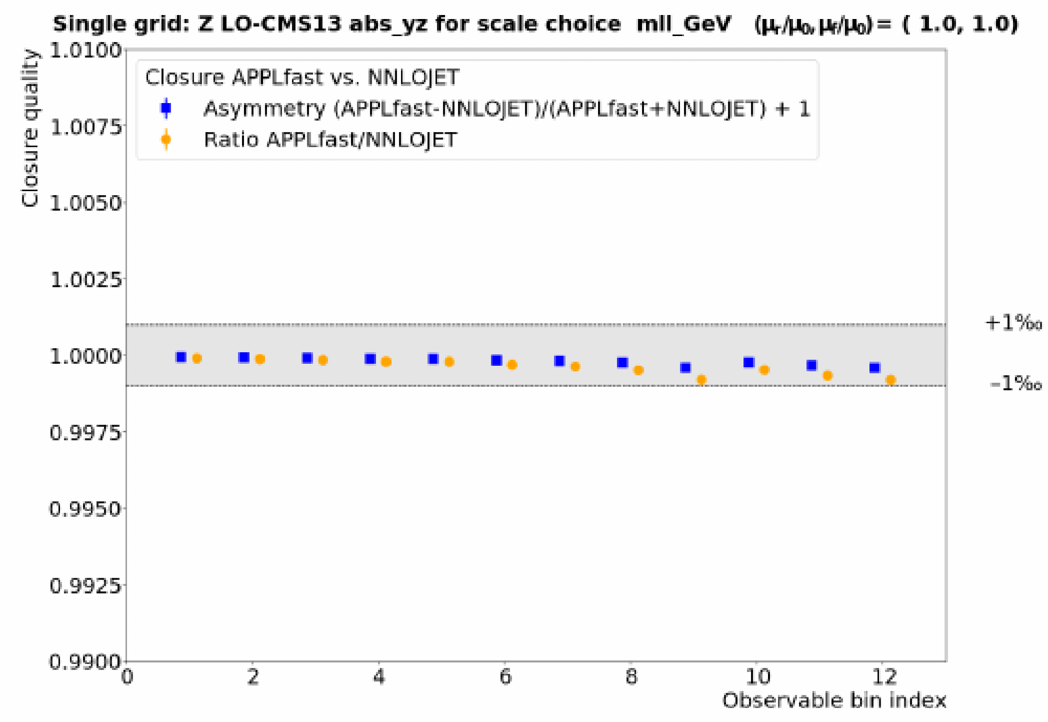
# Grid closure vs. NNLOJET

- **No. of nodes vs. grid size**
- **Ex. DY varying x node distribution**
  - **Left: Equidistant in -sqrt(-log(x))**
  - **Right: Equidistant in log(x)**

| support nodes per magnitude | table size [MB] |
|---|---|
| 8 | 50 |
| 12 | 100 |
| 16 | 180 |

Table 1: Increasing Table Size

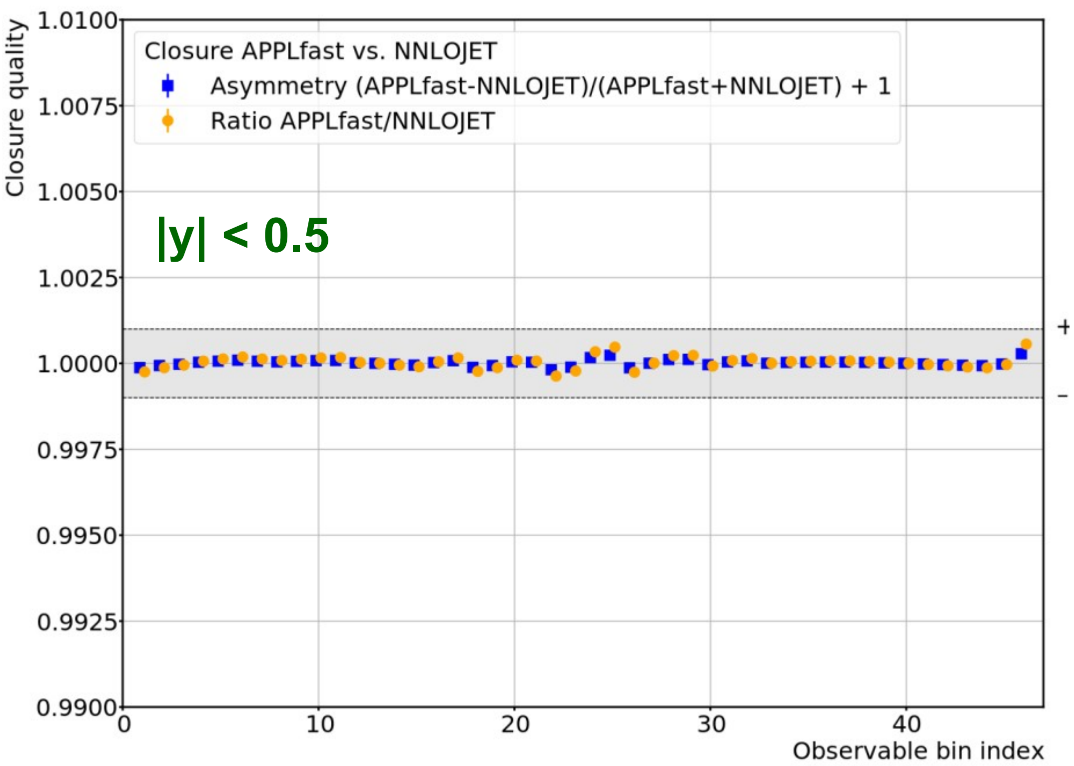


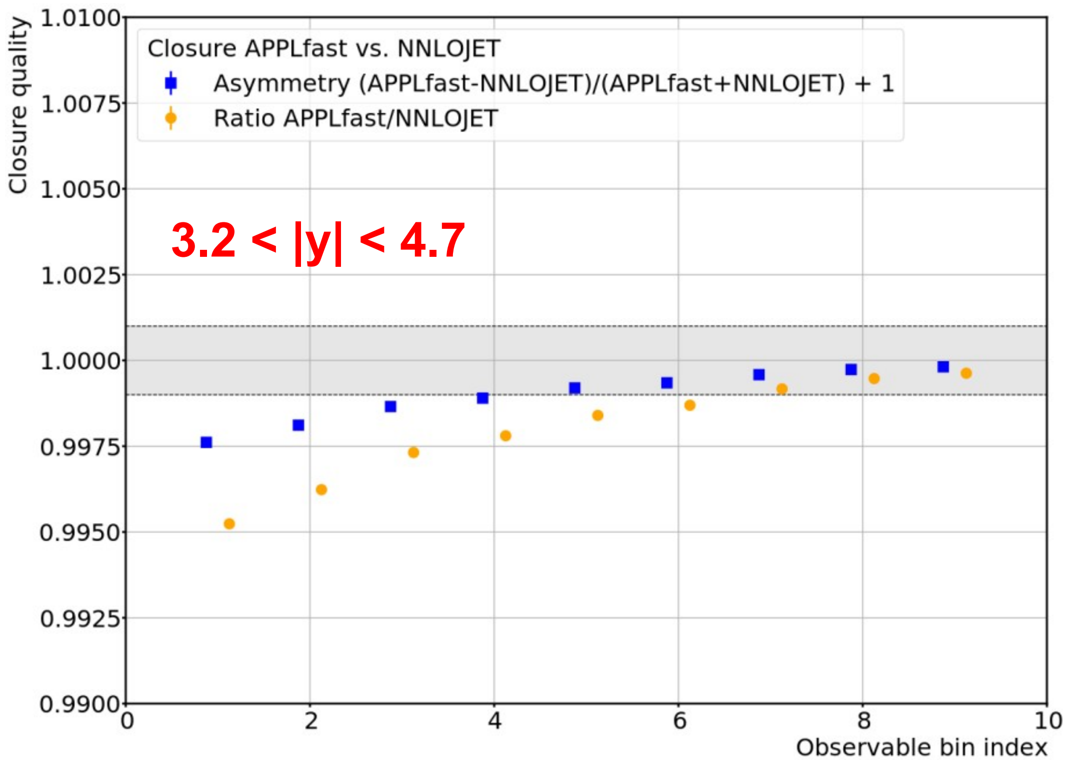




Christiane Mayer

# Grid closure vs. NNLOJET

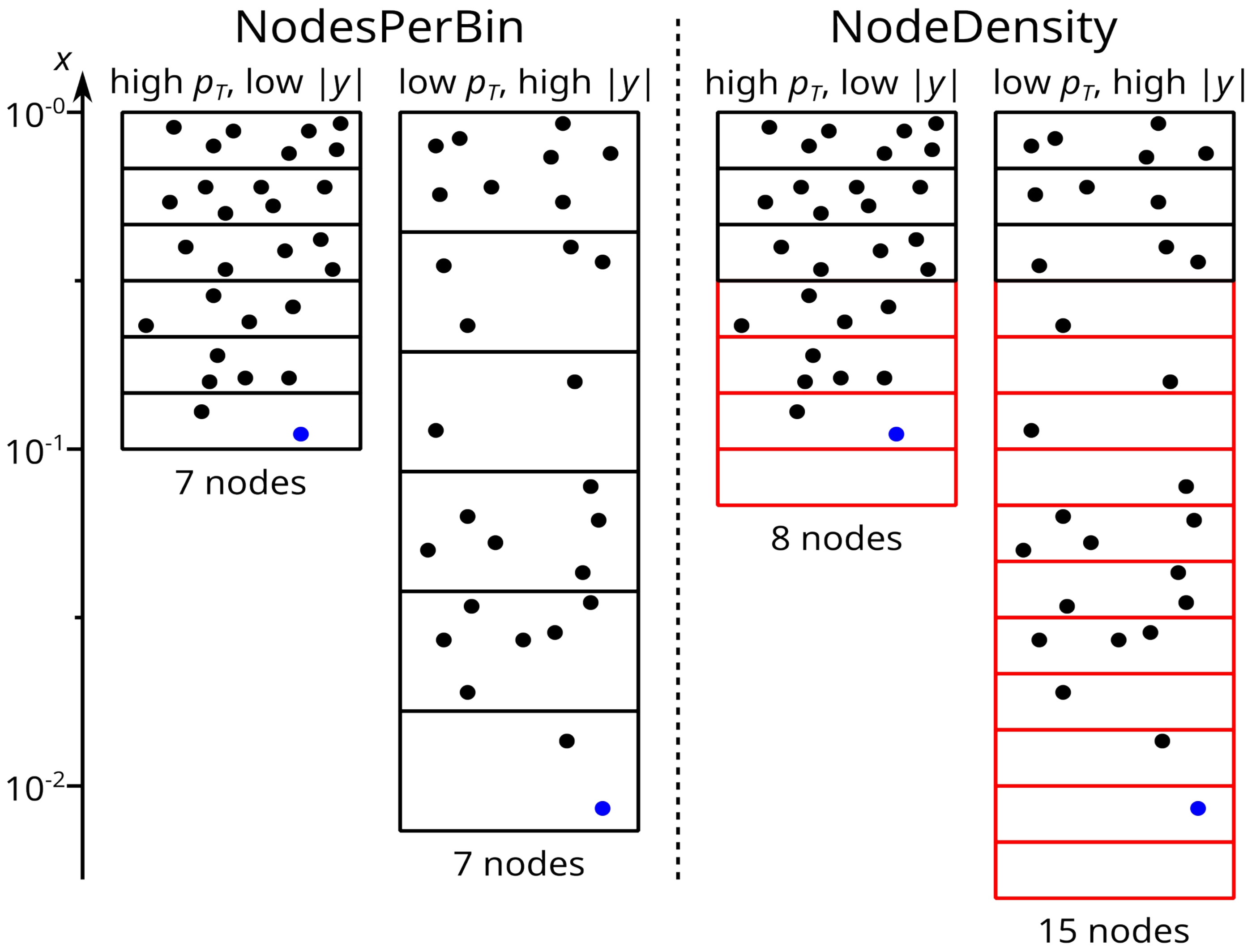
- **Example of CMS inclusive jets at 8 TeV cms energy**
  - **Interpolation in x fine for |y|<3; setting not optimal for larger rapidity**
    - **Use x node density setting instead of fixed no. of nodes (works already)**
    - **Automated extension of x range on-the-fly (→ next slides)**



# Fixed # Nodes vs. Node Density

- $x$: fixed node density instead of fixed node number
- Natural upper $x$ bound at 1
- Start with 4 nodes, expand dynamically to fit minimal $x$




Johannes Gäßler

# x-node extension

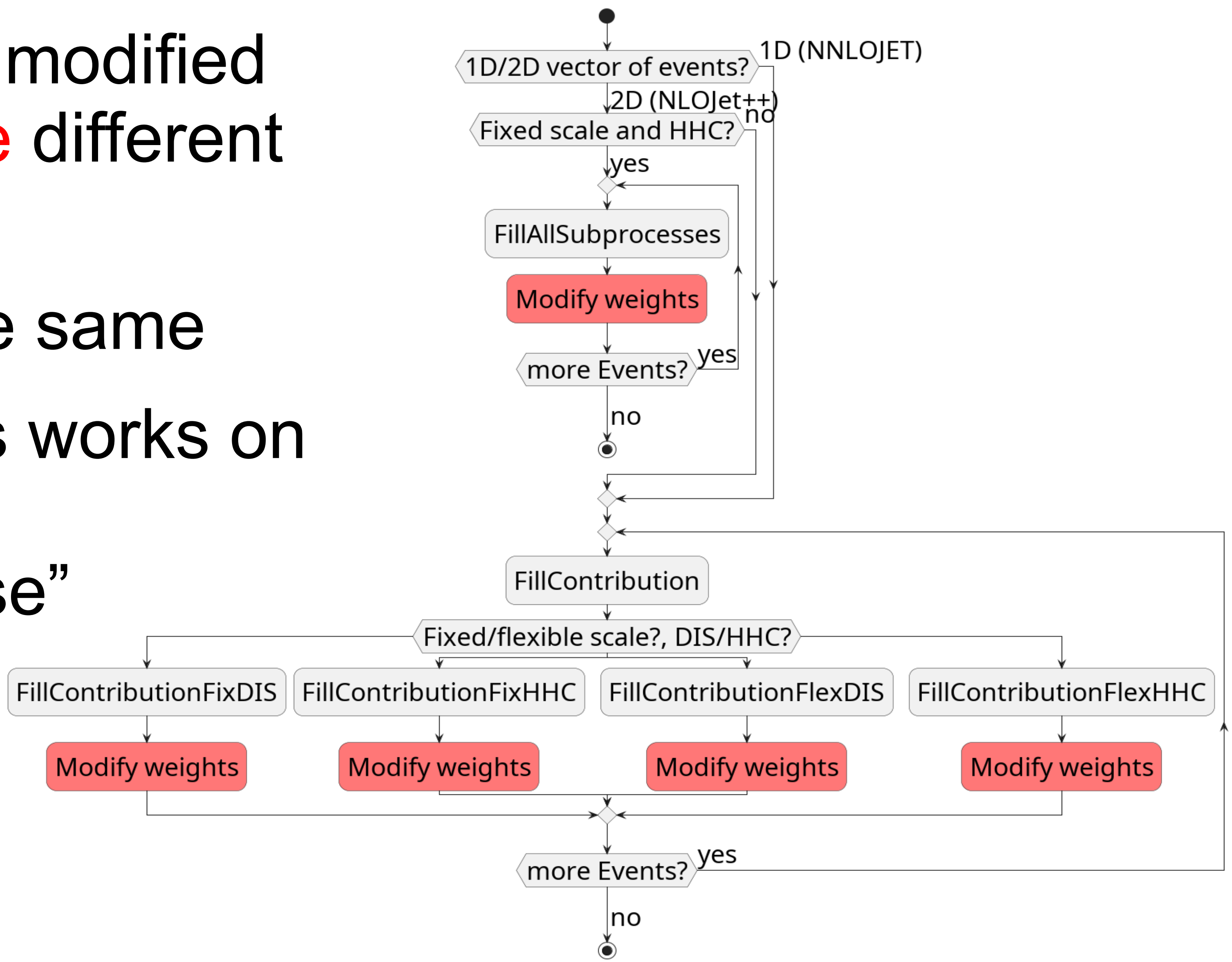
- Adding more $x$ nodes to storage array a little tricky ...
- Deep inelastic scattering: $\begin{pmatrix} 1 & 2 & 3 \end{pmatrix} \to \begin{pmatrix} 0 & 1 & 2 & 3 \end{pmatrix}$
- Proton-proton: $$\begin{pmatrix} 1 & & \\ 2 & 3 & \\ 4 & 5 & 6 \end{pmatrix} \to \begin{pmatrix} 0 & & & \\ 0 & 1 & & \\ 0 & 2 & 3 & \\ 0 & 4 & 5 & 6 \end{pmatrix}$$
- Proton-antiproton: $$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \to \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 \\ 0 & 4 & 5 & 6 \\ 0 & 7 & 8 & 9 \end{pmatrix}$$

**array indexing**

Johannes Gäßler

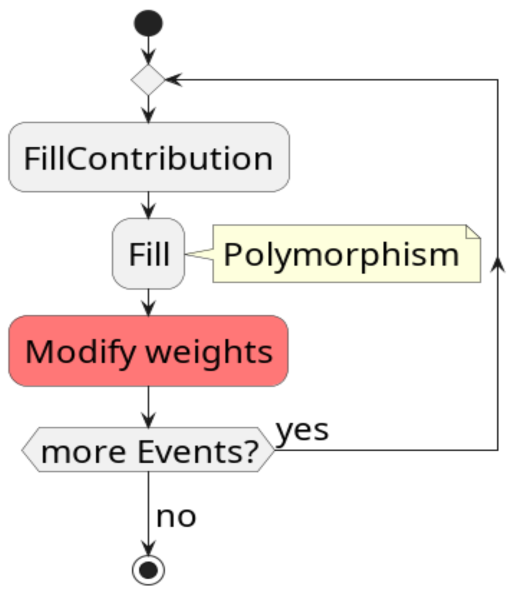
# Implementation

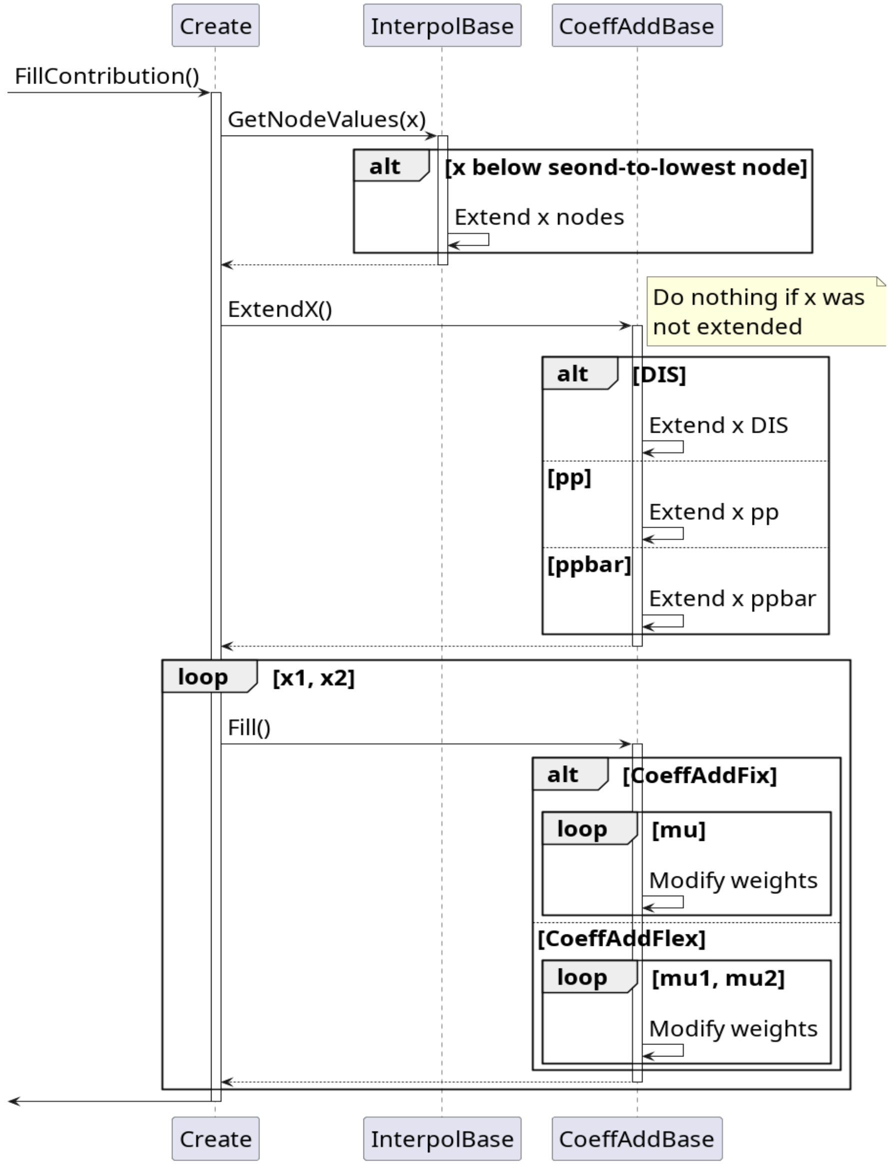
- Original code modified weights in five different places
- Code 80% the same
- “Create” class works on attributes of “CoeffAddBase” classes




Johannes Gäßler

- Deduplicated code as part of density implementation
- Object-oriented design: polymorphism, encapsulation
- Also some speedup for flexible scale




Johannes Gäßler

# Performance optimizations

- fastNLO code deduplicated and refactored
  - In addition some code speedup
- Test at NLO with NLOJet++
  - Helps significantly since ~90% of runtime spent in fastNLO
- First test with NNLOJET & APPLfast2 for modules 2
  - RRa channel <10% of runtime spent in APPLfast
- More checks to follow …
- Would also be interesting to see gain when not accessing LHAPDF anymore (after closure established)

Valgrind

Johannes Gäßler

# Summary & Outlook

- Grid production workflow successfully used for most of the published pp grids
- Summer student project:
  - Adaptation of LAW workflow to lxplus and production of grids for Drell-Yan
  - Test automisation of Vegas grid production
- Master student work:
  - Workflow simplifications (x-interpolation) and performance improvements in progress
- Outlook
  - Adapt workflow at KIT and lxplus to NNLOJET modules 2 & APPLfast2
  - Test and run workflow on acquired HPC resources (HoreKa at KIT), also with Slurm
  - Include additional workflow & performance improvements
  - Provide documentation for more user friendliness …

# Backup Slides

# Ingredients

- **Theory: Today interface to** (others NLOJet++, MCFM, ...)
  - **NNLOJET:** T. Gehrmann et al., RADCOR2017 PoS (2018) 074, arXiv:1801.06415.
  - **Inclusive jets:** J. Currie et al, PRL 118 (2017) 072002; JHEP 10 (2018) 155.
  - **Dijets:** J. Currie et al., PRL 119 (2017) 152001; A. Gehrmann-de Ridder et al., PRL 123 (2019) 102001.
  - **Full-colour NNLO:** X. Chen et al., arXiv:2204.10173. (Not yet included in grids!)

- **Tools:**
  - **APPLfast interface:** D. Britzger et al., EPJC 79 (2019) 845, arXiv:1906.05303.
  - **fastNLO:** D. Britzger et al., Proc. DIS2012 (2012) 217, arXiv:1208.3641.
  - **APPLgrid:** T. Carli et al., EPJC 66 (2010) 503, arXiv:0911.2985.
  - **xfitter:** S. Alekhin et al., EPJC 75 (2015) 304, arXiv:1410.4412.

# Interpolation concept

## Implemented in APPLgrid & fastNLO

## Use interpolation kernel

- Introduce set of n discrete **x-nodes**, $x_i$'s being equidistant in a function f(x)
- Take set of **Eigenfunctions $E_i(x)$** around nodes $x_i$

→ Interpolation kernels

- Actually a rather old idea, see e.g.

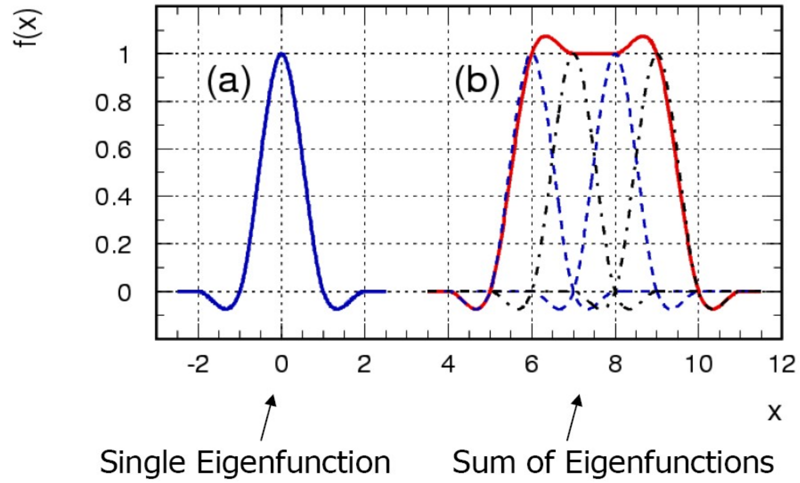
  C. Pascaud, F. Zomer (Orsay, LAL), LAL-94-42

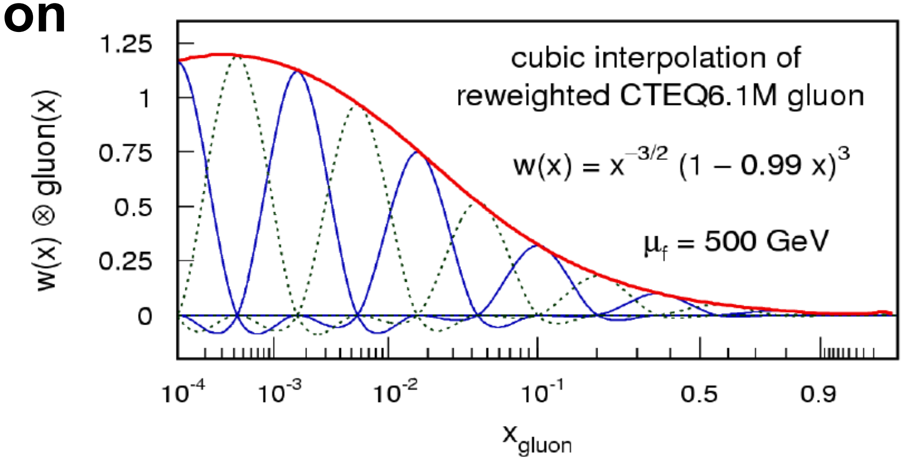
→ Single PDF is replaced by a linear combination of interpolation kernels

$$f_a(x) \cong \sum_i f_a(x_i) \cdot E^{(i)}(x)$$

→ Then the integrals are done only once

→ Afterwards only summation required to change PDF

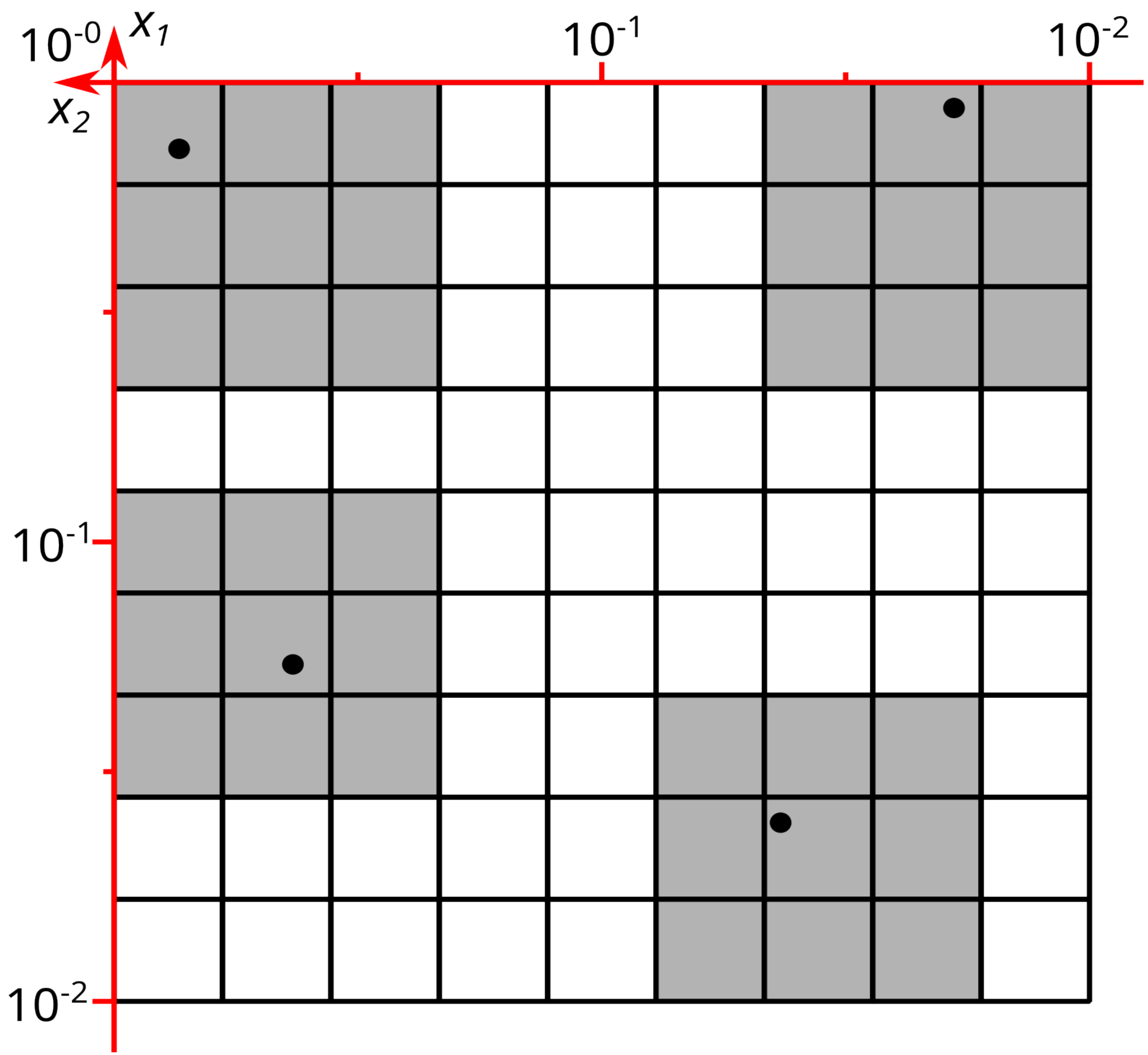



Tabulate the convolution of the perturbative coefficients with the interpolation kernel

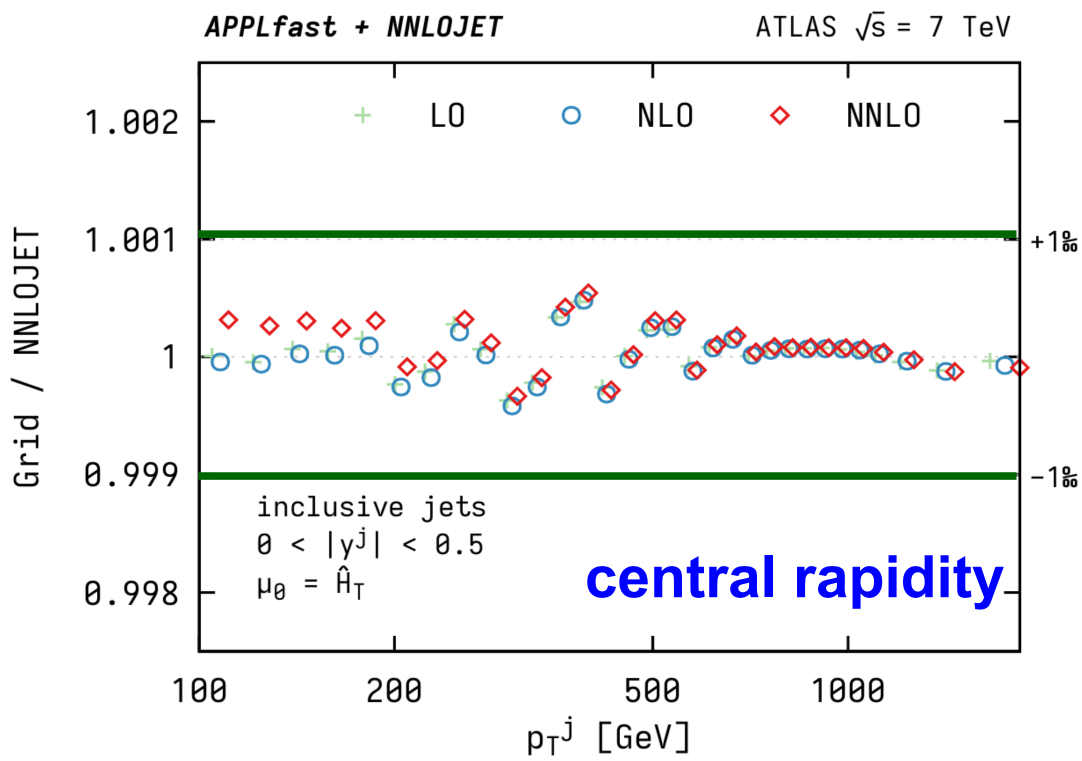
# Node Weight Spread

- Weights spread over 16 $(x_1, x_2)$ nodes
- Special treatment of boundaries
- No lower boundary for node density
- Need to add one more $x$ node




Johannes Gäßler

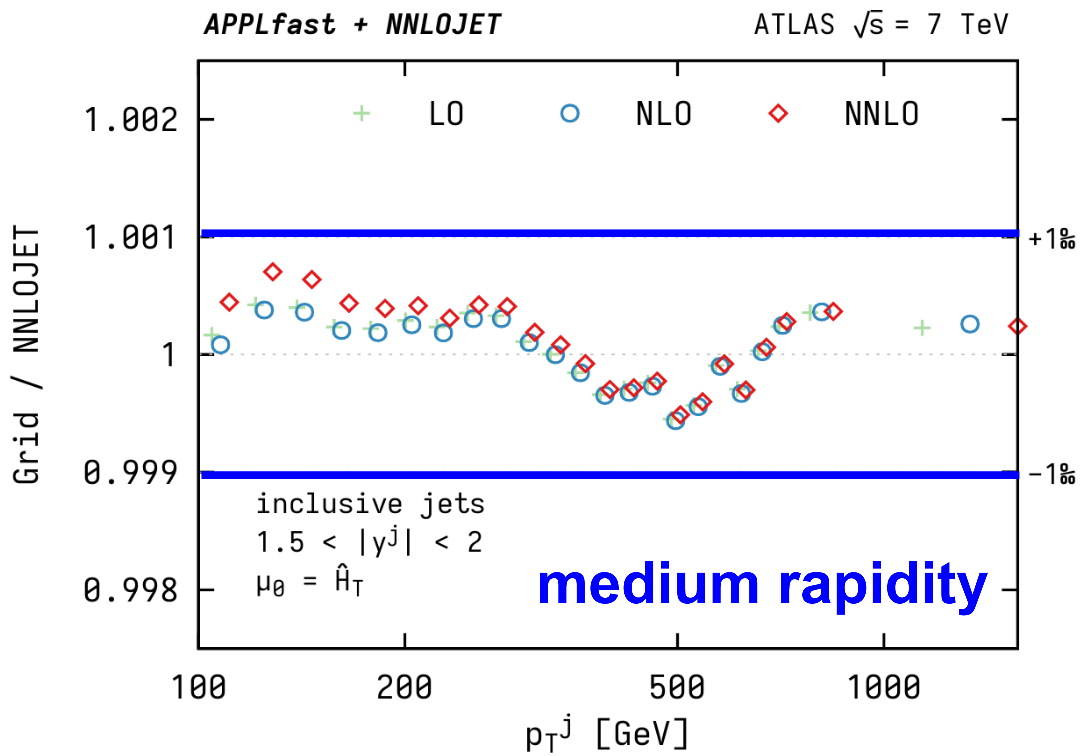
# Grid closure vs. NNLOJET




**ATLAS inclusive jets at 7 TeV**

**Generally aim at closure better than 1 ‰ at each level, LO, NLO, NNLO**

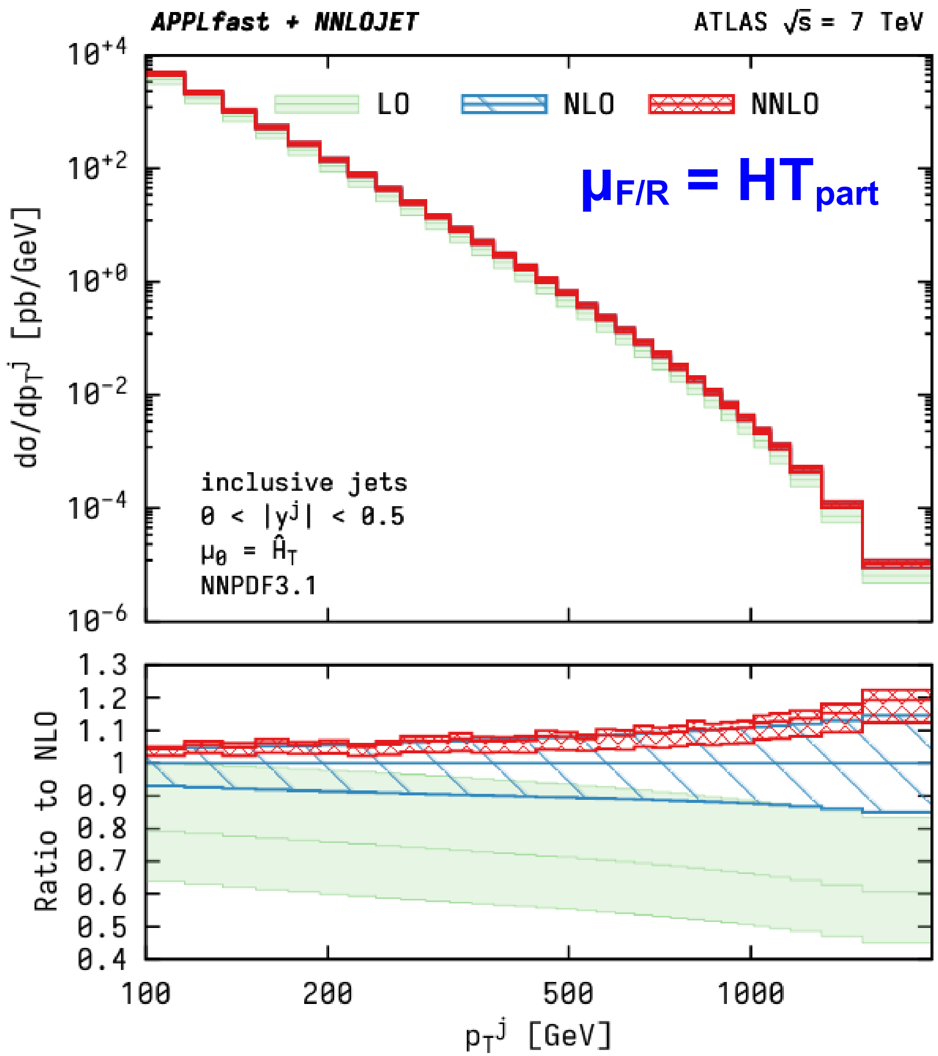
**Closure deteriorates somewhat towards phase space limits; exceptionally may exceed 1 ‰ at phase space edges**



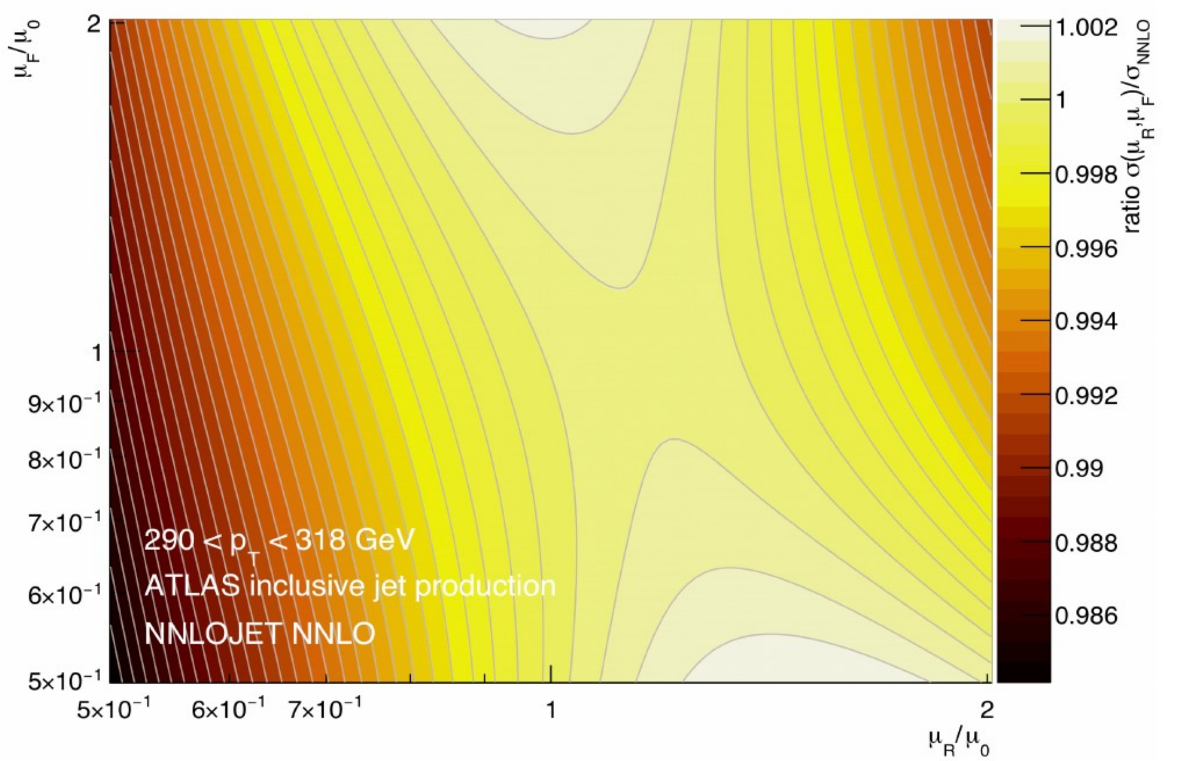
# Scale dependence

**ATLAS inclusive jets at 7 TeV**

**Scale uncertainty bands: LO, NLO, NNLO**




**Full 2-dimensional scale dependence in $\mu_{F/R}$ for each bin**

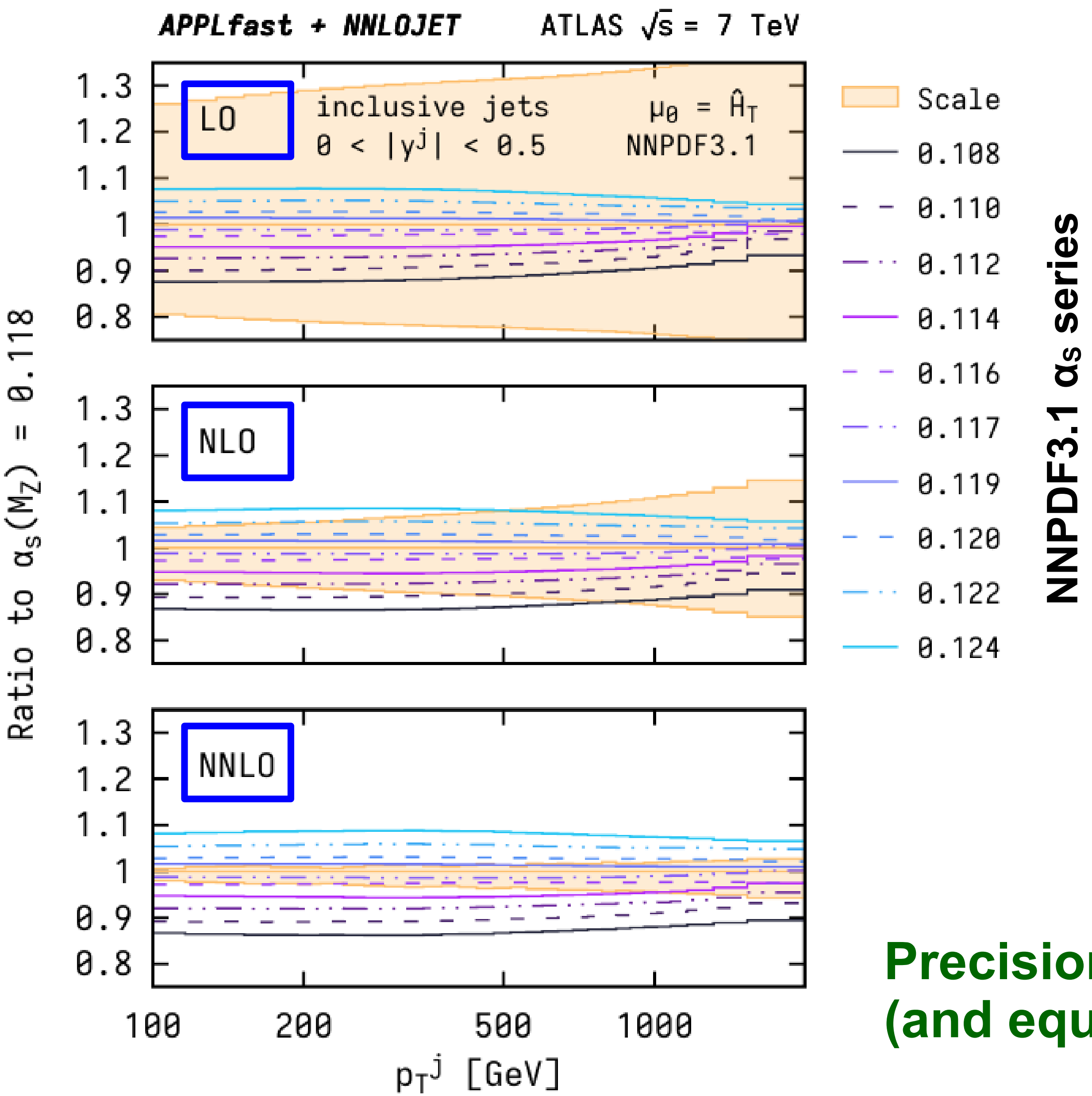



**Alternative scale possible: $p_{Tjet}$ instead of $HT_{part}$**

**From Snowmass report: arXiv:2203.13923**

# $\alpha_s$ dependence

**ATLAS inclusive jets at 7 TeV**




**Scale uncertainty versus**
**$\alpha_S$ dependence**
**(0.108 – 0.124)**
**at each order**

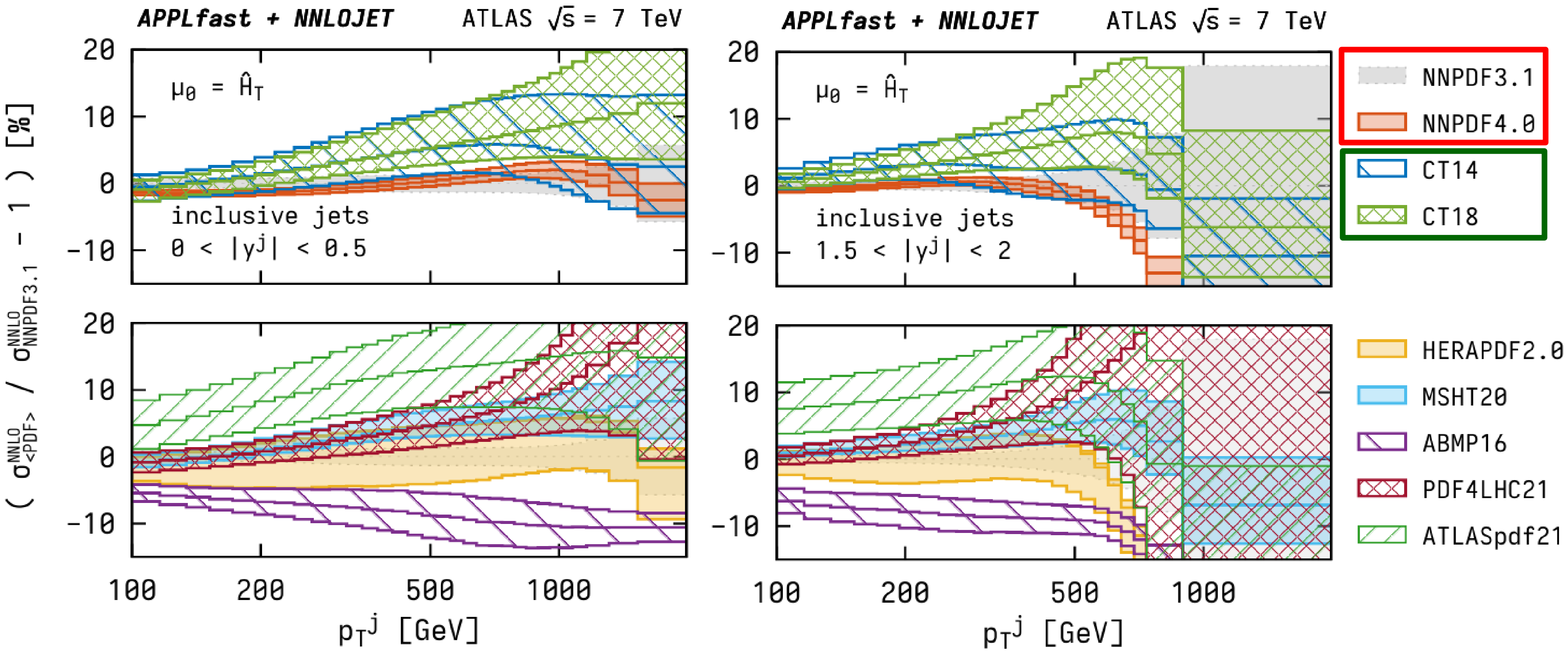
**Precision determinations require NNLO (and equally accurate data!)**

# PDF dependence

**ATLAS inclusive jets at 7 TeV**

**PDF uncertainty bands for selection of 9 PDF sets**

**central rapidity** **"Central" PDF: NNPDF3.1** **medium rapidity**




**Interesting feature for ΔPDF: CT14 → CT18 larger**

**NNPDF 3.1 → 4.0 much smaller**

# Inclusive jet datasets

- **Seven inclusive jet datasets from ATLAS & CMS, 2D in $p_T$ and y**
  - **Four centre-of-mass energies, three jet radii R**
  - **Two central scales for $\mu_{R/F}$**

**Sample plots in this talk**

| Data | $\sqrt{s}$ [TeV] | $\mathcal{L}$ [fb$^{-1}$] | no. of points | anti-$k_T$ $R$ | kinematic range [GeV] | fiducial cuts | $\mu_{R/F}$-choice |
|---|---|---|---|---|---|---|---|
| CMS [30] | 2.76 | 0.00543 | 81 | 0.7 | $p_T^{jet} \in [74, 592]$ | $\|y\| < 3.0$ | $p_T^{jet}, \hat{H}_T$ |
| ATLAS [28] | 7.0 | 4.5 | 140 | 0.6 | $p_T^{jet} \in [100, 1992]$ | $\|y\| < 3.0$ | $p_T^{jet}, \hat{H}_T$ |
| CMS [31] | 7.0 | 5.0 | 133 | 0.7 | $p_T^{jet} \in [114, 2116]$ | $\|y\| < 3.0$ | $p_T^{jet}, \hat{H}_T$ |
| ATLAS [32] | 8.0 | 20.3 | 171 | 0.6 | $p_T^{jet} \in [70, 2500]$ | $\|y\| < 3.0$ | $p_T^{jet}, \hat{H}_T$ |
| CMS [33] | 8.0 | 5.6<br>19.7 | 248 | 0.7 | $p_T^{jet} \in [21, 74]$<br>$p_T^{jet} \in [74, 2500]$ | $\|y\| < 4.7$ | $p_T^{jet}, \hat{H}_T$ |
| ATLAS [34] | 13.0 | 3.2 | 177 | 0.4 | $p_T^{jet} \in [100, 3937]$ | $\|y\| < 3.0$ | $p_T^{jet}, \hat{H}_T$ |
| CMS [35] | 13.0 | 36.3<br>33.5 | 2 × 78 | 0.4<br>0.7 | $p_T^{jet} \in [97, 3103]$ | $\|y\| < 2.0$ | $p_T^{jet}, \hat{H}_T$ |

# Dijet datasets

- **Four dijet datasets from ATLAS & CMS, 2D in $m_{12}$ and $y^*$ or $y_{max}$, or 3D in $\langle p_{T12} \rangle$, $y^*$, $y_b$**
  - **Three centre-of-mass energies, three jet radii R**
  - **One central scale for $\mu_{R/F}$, except for 3D data with two**

**Sample plots in this talk**

| Data | $\sqrt{s}$ [TeV] | $\mathcal{L}$ [fb$^{-1}$] | no. of points | anti-$k_T$ $R$ | kinematic range [GeV] | fiducial cuts | $\mu_{R/F}$-choice |
|---|---|---|---|---|---|---|---|
| ATLAS [55] | 7.0 | 4.5 | 90 | 0.6 | $m_{12} \in [260, 5040]$ | $\lvert y_1 \rvert, \lvert y_2 \rvert < 3.0$<br>$[p_{T,1}, p_{T,2}] > [100, 50]$GeV<br>$y^* < 3.0$ | $m_{12}$ |
| CMS [31] | 7.0 | 5.0 | 54 | 0.7 | $m_{12} \in [197, 5058]$ | $\lvert y \rvert < 5.0$<br>$[p_{T,1}, p_{T,2}] > [60, 30]$GeV<br>$\lvert y_{max} \rvert < 2.5$ | $m_{12}$ |
| CMS [49] | 8.0 | 19.7 | 122 | 0.7 | $\langle p_{T1,2} \rangle \in [133, 1784]$ | $\lvert y \rvert < 5.0$<br>$p_{T,1}, p_{T,2} > 50$GeV<br>$\lvert y_1 \rvert, \lvert y_2 \rvert < 3.0$ | $p_{T,1} \exp(0.3\, y^*)$<br>$m_{12}$ |
| ATLAS [34] | 13.0 | 3.2 | 136 | 0.4 | $m_{12} \in [260, 9066]$ | $\lvert y_1 \rvert, \lvert y_2 \rvert < 3.0$<br>$p_{T,1}, p_{T,2} > 75$GeV<br>$\langle p_{T1,2} \rangle > 100$GeV<br>$y^* < 3.0$ | $m_{12}$ |