Developments in APPLfast

D. Britzger (Heidelberg), C. Gwenlan (Oxford), A. Huss (CERN),
K. Rabbertz (KIT), M. Sutton (Sussex)

What is APPLfast?

- Started as common project of NNLOJET, APPLgrid, and fastNLO authors at QCD@LHC in London
- Interface between NNLOJET and fast grid technology - APPLgrid and fastNLO
- Aimed to be the least obtrusive as possible for both ends of the interface
- Intended to be reusable by other theory programs

NNLOJET+APPLfast Workflow

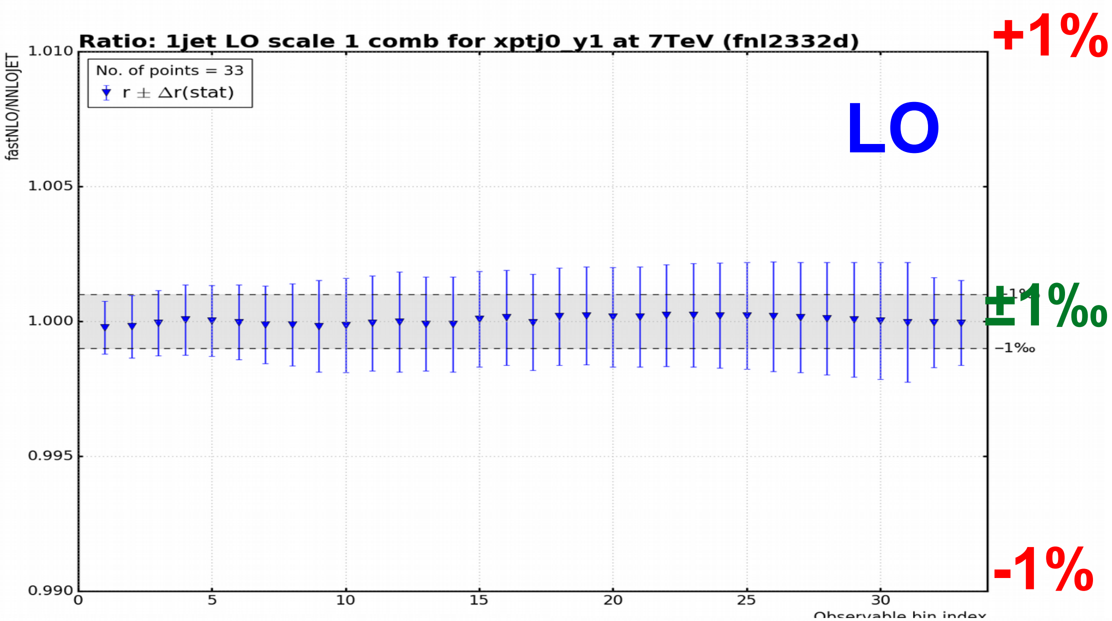
- **1. Preprocessing:** Check of interpolation quality
  - Short test jobs to check interpolation settings (& optimise if necessary) O(10 h)
- **2. NNLOJET Warm-up:** Vegas integration optimisation
  - 1 long (multi-core) job per process O(100 h)
- **3. APPLgrid/fastNLO Warm-up:** Adapt x- and scale-grids to accessed phase space (exact strategy differs between APPLgrid & fastNLO)
  - Only phase space provided from NNLOJET → significant speed-up O(100 h)
- **4. Interpolation grid production:**
  - Thousands of parallel jobs O(250 kh)
- **5. Postprocessing:** Statistical evaluation and combination of all produced grids …
  - Job to combine all grids and estimate statistical uncertainty O(100 h)
- **6. Validate, validate, and validate** O(? h)
- **7. Present final results :-)** 30 min :-)

Step 6: Validation

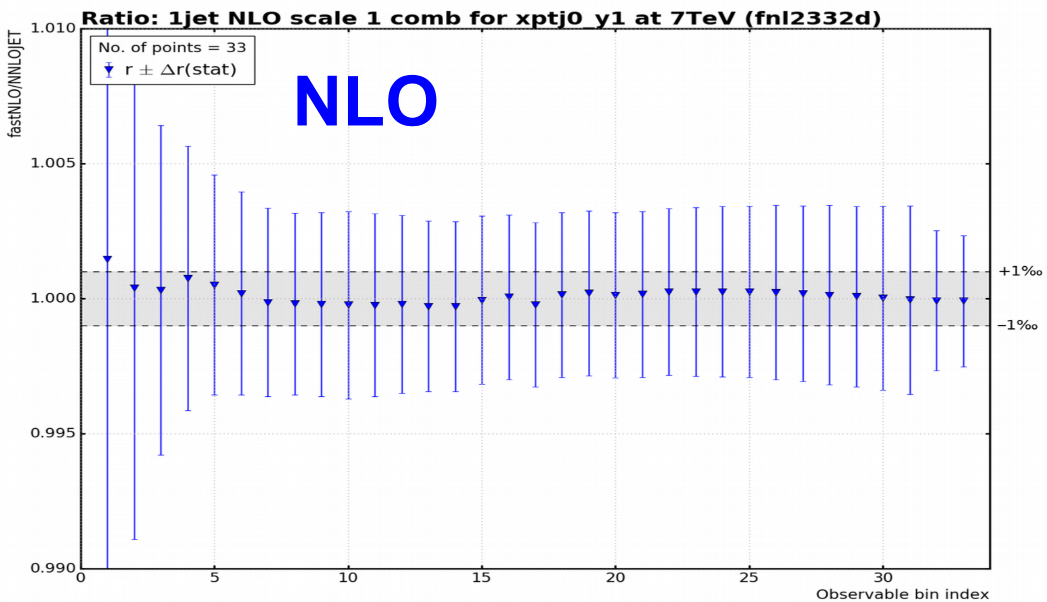
- Check every aspect you can or you will be hit by Murphy's law!
  - Check each contribution (LO, R, V, RRa, RRb, RV, VV) separately
  - Check interpolation in x-space for single grids
  - Check interpolation in scales for single grids
  - Compare merged grids to NNLOJET for each contribution
  - Compare final merged grids for each order to NNLOJET
  - … more checks/comparisons, e.g. to other programs
  - ?

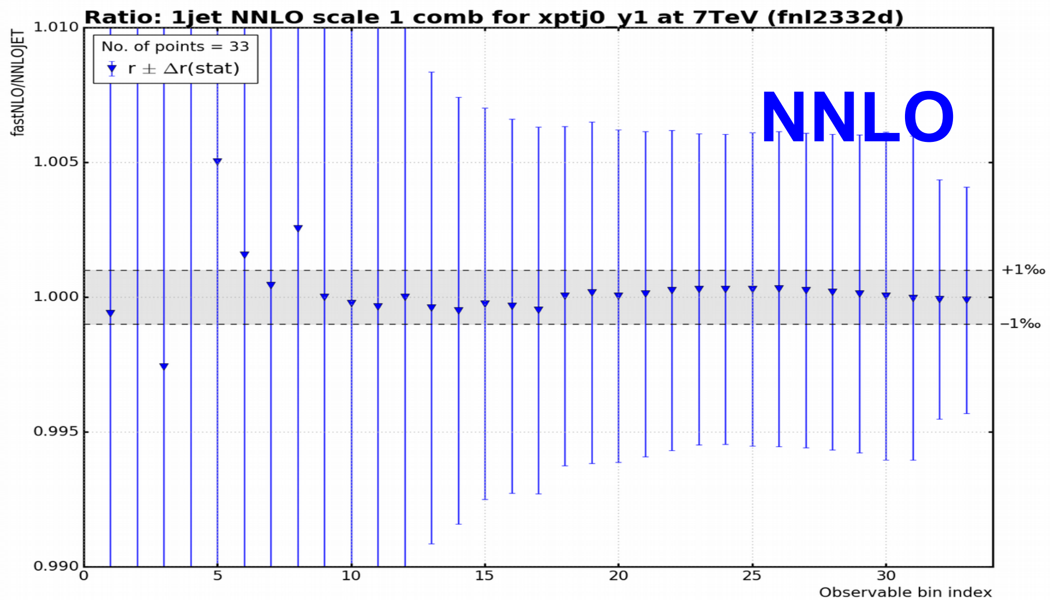
Inclusive jet pT – combined grid

Closure APPLfast / NNLOJET
(December 2017)



Error bars:
stat. uncertainty estimate from NNLOJET

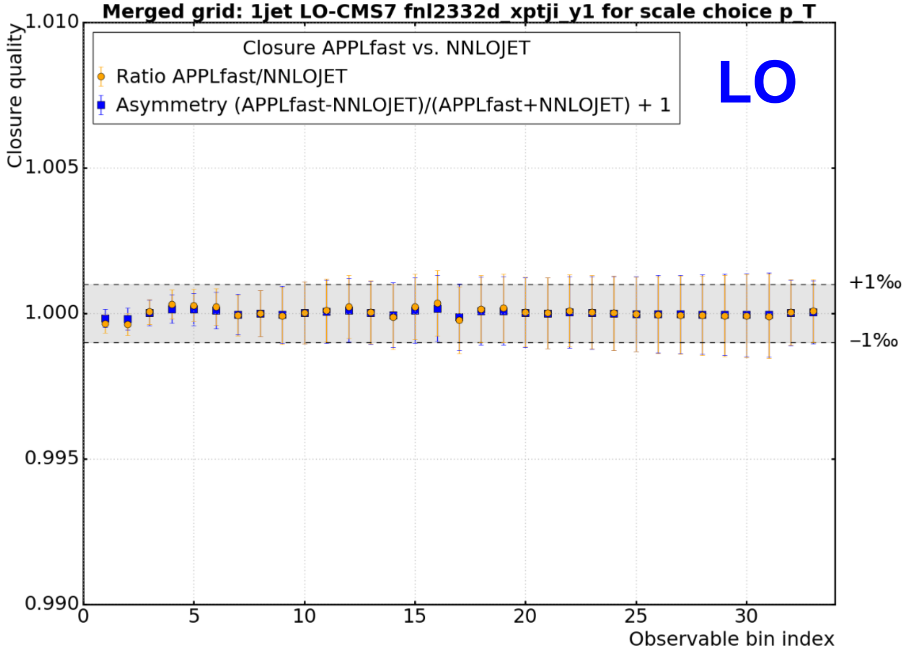




Inclusive jet pT – combined grid

Closure APPLfast / NNLOJET
(March 2018)

Error bars:
stat. uncertainty estimate from NNLOJET

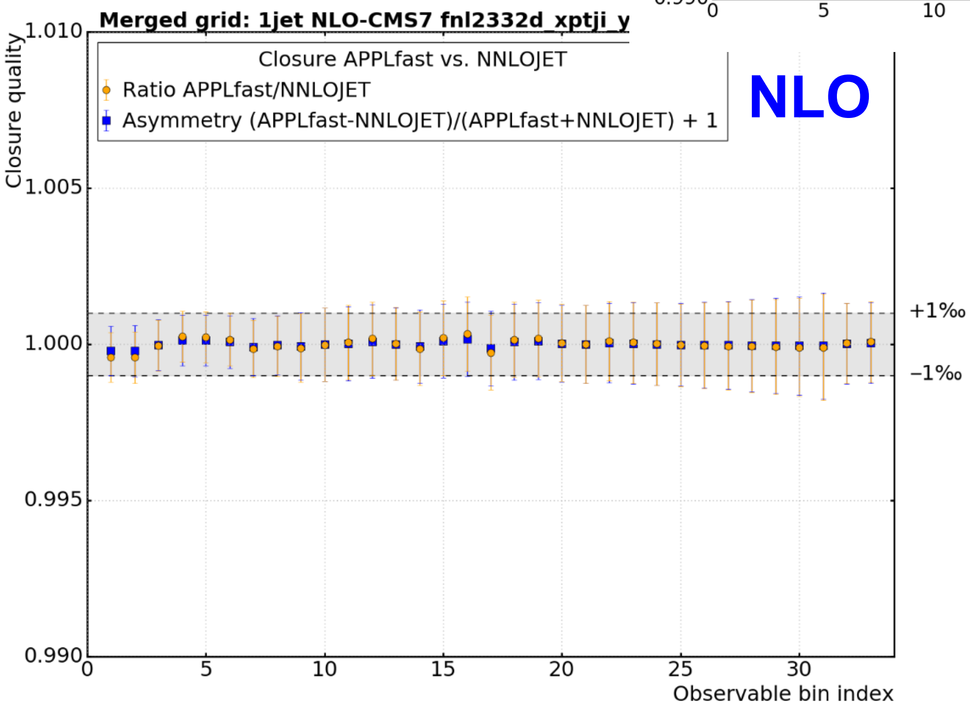


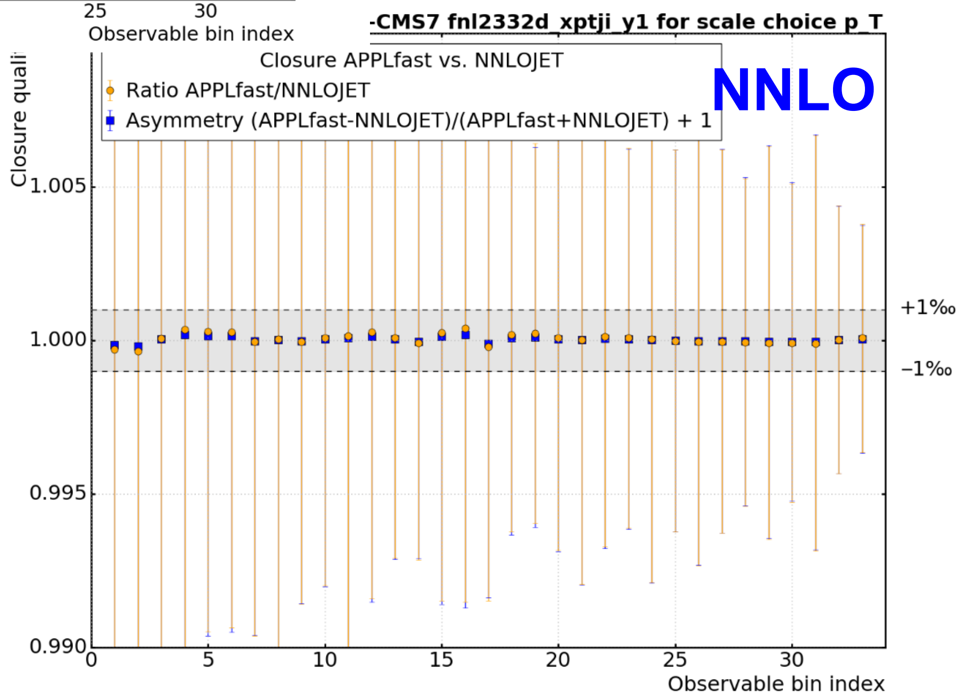
+1%

±1‰

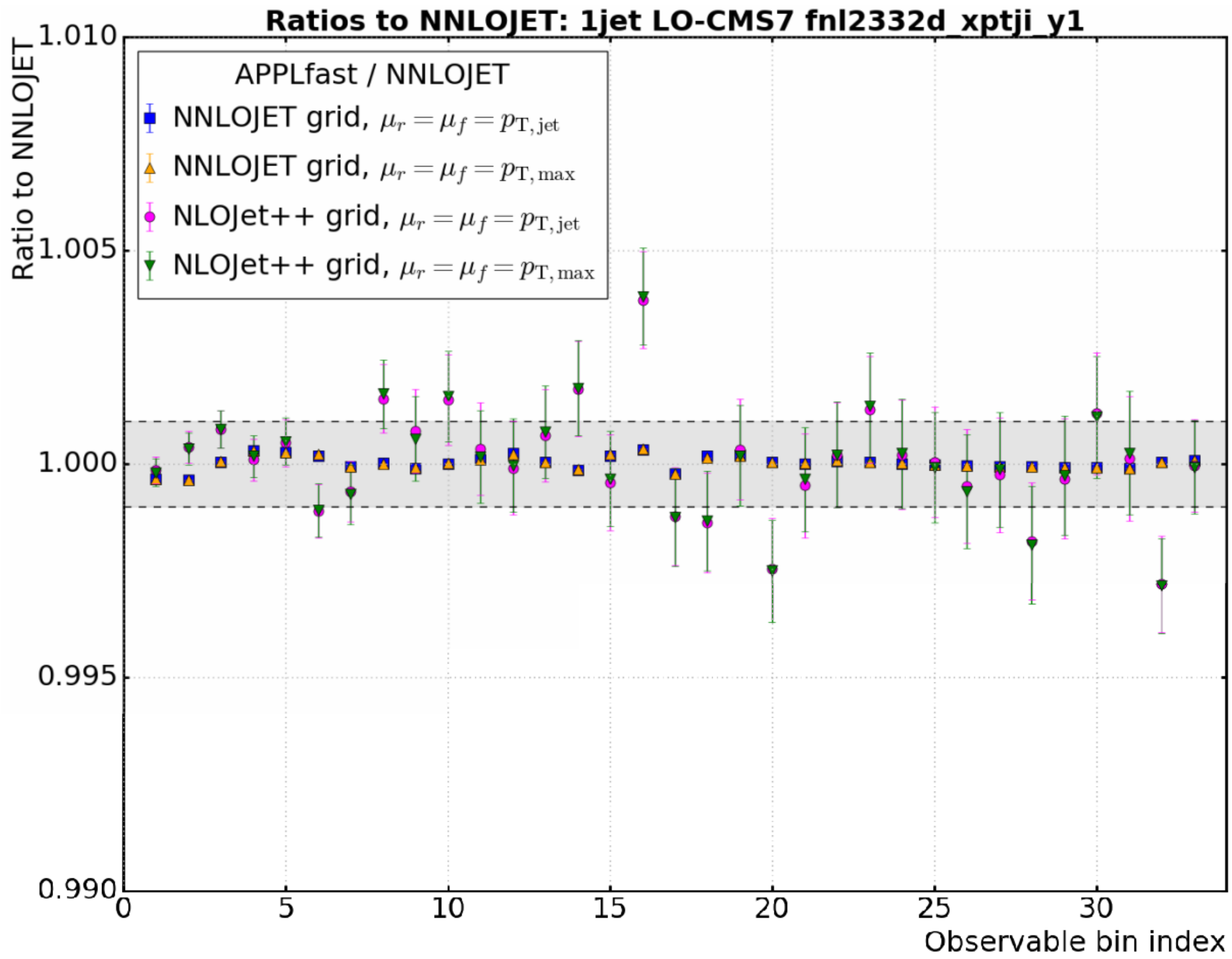
-1%

One issue for $\mu_r \neq \mu_f$ fixed, numerical precision improved!
Waiting for many more NNLO jobs to complete!





Crosscheck with NLOJet++, LO

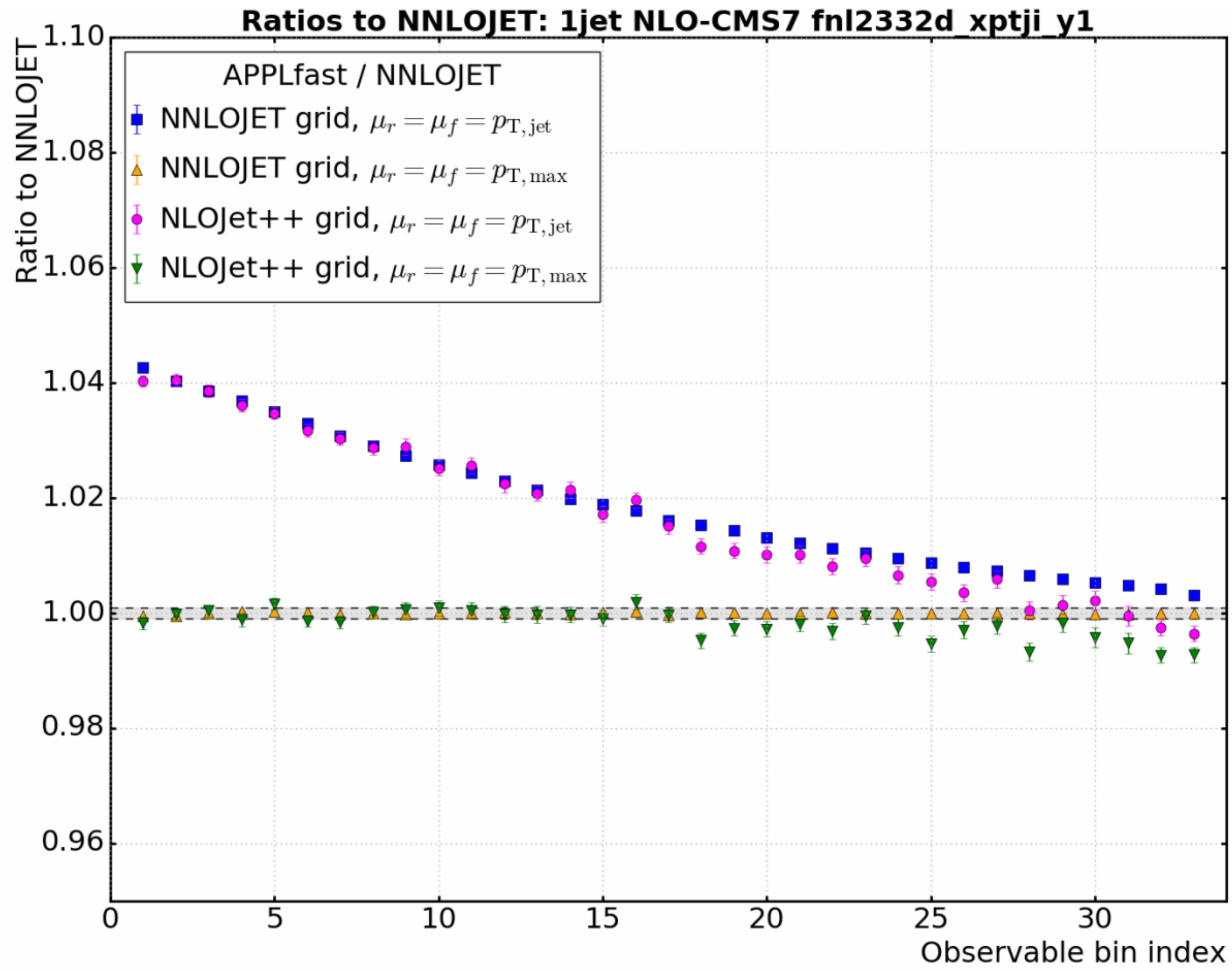


Ratio always to NNLOJET with scale $p_{T,max}$

±1‰

Error bars:
stat. uncertainty estimate from NNLOJET & NLOJet++

Crosscheck with NLOJet++, NLO



Ratio always to NNLOJET with scale $p_{T,max}$

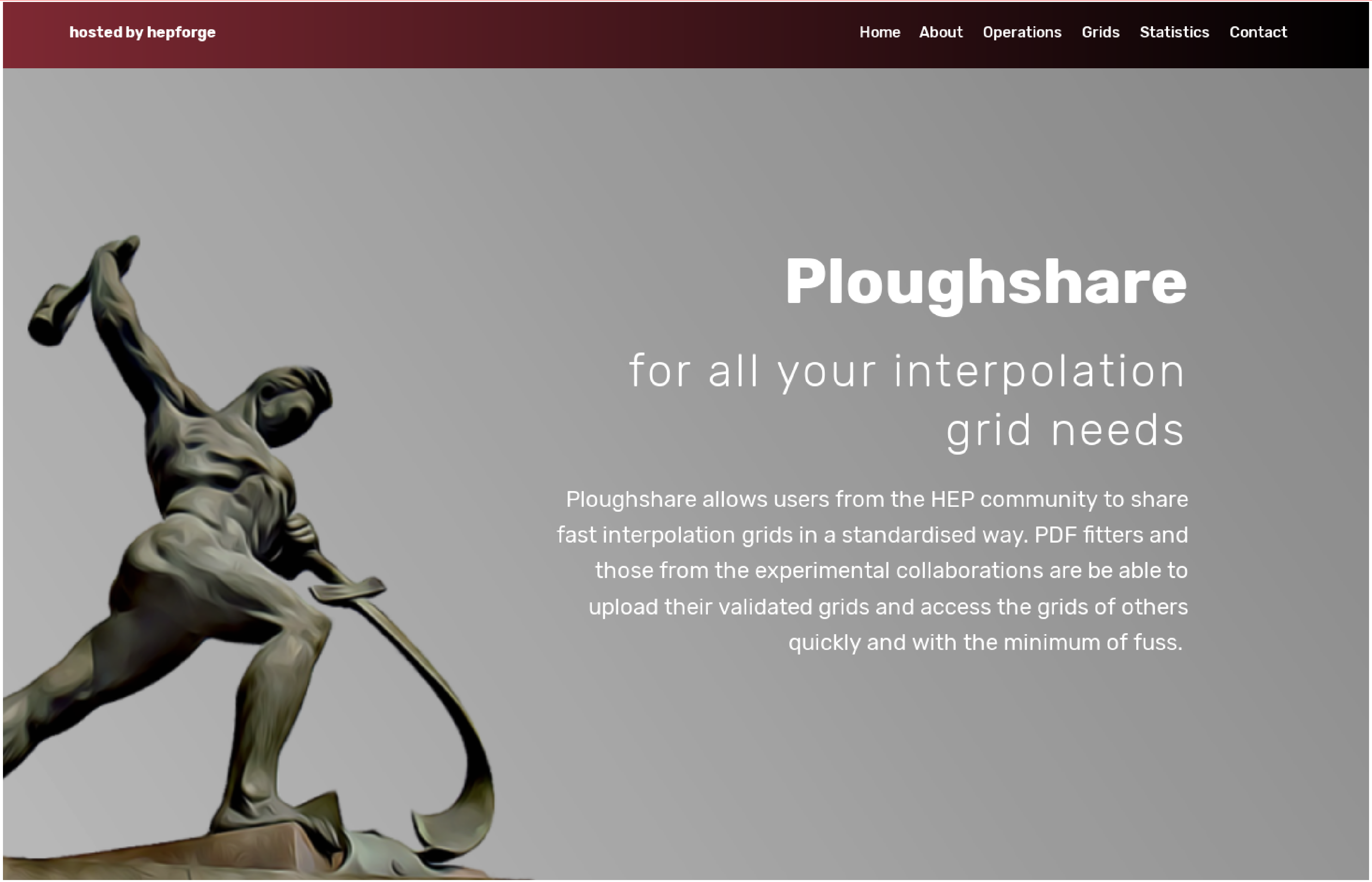
±1‰

Error bars:
stat. uncertainty estimate from NNLOJET & NLOJet++

NNLO Outlook, March 2018

- NNLOJET provides NNLO in common interface for:
  - Z incl., Z+jet, W incl., pp jet+dijets, H incl., H+jet, DIS jet+dijets, e+e- 3jets
  - W+jet published
- APPLgrid+fastNLO interface (NNLO-Bridge) is working
- Numerous adaptations implemented by all sides
- Large-scale productions tested for Z+jet and DIS jet and pp jet
- DIS NNLO results & α_s published with H1: Eur. Phys. J. C, 2017, 77, 791
- Received final combination prescription for NNLOJET results last Nov.
  - Removes fluctuations from bad cancellations
  - Weighted interpolation table merging implemented
- Production for CMS pp inclusive jets at 7 TeV almost finished, previous plots have full NLO and ~20% of NNLO interpolation grids
- Plan more grids for inclusive jets, Z+jet, dijets in next months

Grid distribution – Ploughshare



Grid distribution – Ploughshare

What is Ploughshare ?

1 **Quick to use** - a web based utility for the automated distribution of fast inperpolation grids for the high energy physcis community.

2 **Secure storage** - registered users can upload grid files and corresponding standard format configuration files to describe the grids and physics processes and these are added to a central repository.

3 **Automatic distribution** - a standard utility library will be provided to download any required grids automatically in user code.

A utility for the community Ploughshare allows users to share their grids, so it is important that the provenance of the grids is guaranteed. This is achieved by allowing only registered users to upload their validated grids. Subsequently however, anyone is free to download and use the grids as they wish.

Fast operations summary

Navigate quickly to some of the primary operations you might be interested in

Download grids
Veiw all the lovely grids which are available for download

Upload grids
Upload grids using the standard web interface

Download grid code
Get the code for the automated download of multiple grids

Settings
How to set up the automated code for the grid downloads

- New hep forge package where registered users upload grids with some documentation
- Registered(!) user gets FAME or BLAME
- Automated job treats the upload:
  - Add to the appropriate location in the file system
  - Generate relevant lists, and display web pages
- Provides a user interface for automated download with a simple line of code
- Have expression of interest from many stakeholders …
- Proof of concept is there, need to bring it to life now.
  HELP is WELCOME!

Summary

- APPLfast interface (NNLO-Bridge) and interpolation is working
- Large-scale productions tested for Z+jet, DIS jet, and pp jets
- Combination of grids with weights à la NNLOJET implemented
- Production of a series of APPLgrid and/or fastNLO grids for various processes has started
- Final grids will be made available on Ploughshare

Thank you for your attention!

Backup

Motivation

- Interpretation of experimental data requires reasonably fast theory
- Often: Repeated computation of same cross section:
  - Different PDF sets; PDF uncertainties
  - Variations of renormalisation & factorisation scales μ_R, μ_F
  - Variation of $\alpha_s(M_Z)$
  - SM parameter fits (→ xFitter)
- Jet cross section calculations at NLO were slow → initial reason for interpolation grids
- Nowadays NNLO in general very demanding!
  - Need procedure for fast repeated computations of higher order cross sections
  - Use interpolation grids like from fastNLO or APPLgrid (both are interfaced to xFitter :-))

Interpolation concept

Implemented in APPLgrid & fastNLO

APPLgrid, Carli et al., Eur. Phys. J. C, 2010, 66, 503.
fastNLO, Britzger et al., arXiv:0609285, 1208.3641.

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

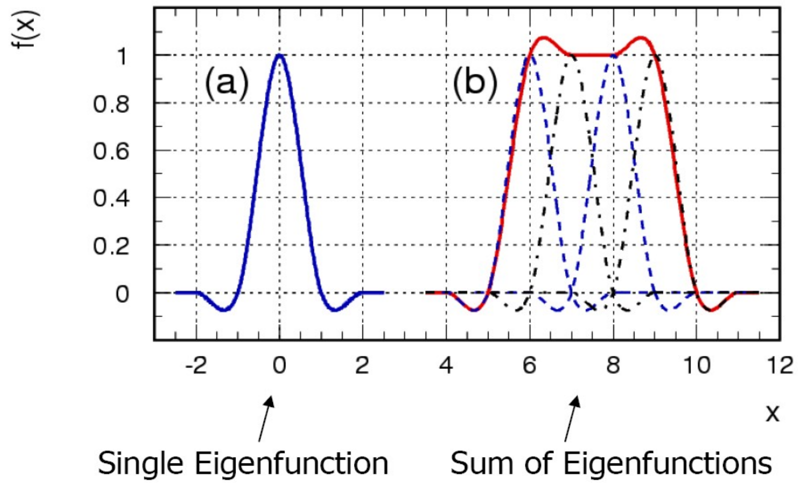
- Actually a rather old idea, see e.g.
  C. Pascaud, F. Zomer (Orsay, LAL), LAL-94-42

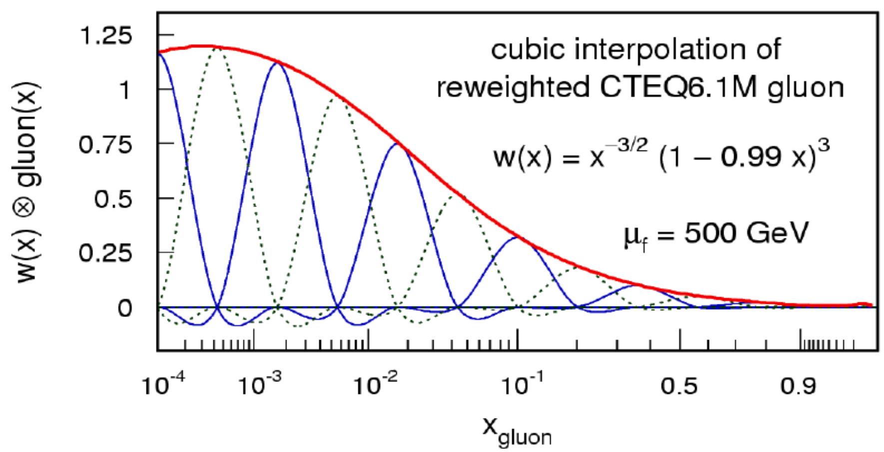
→ Single PDF is replaced by a linear combination of interpolation kernels

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

→ 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

Flexible-scale tables

- Storage of scale-independent weights enable full scale flexibility also in NNLO
  - Additional logs in NNLO

$$\omega(\mu_R,\mu_F) = \underbrace{\omega_0 + \log(\mu_R^2)\omega_R + \log(\mu_F^2)\omega_F}_{\text{log's for NLO}} + \underbrace{\log^2(\mu_R^2)\omega_{RR} + \log^2(\mu_F^2)\omega_{FF} + \log(\mu_R^2)\log(\mu_F^2)\omega_{RF}}_{\text{additional log's in NNLO}}$$

  - Store weights: w_0, w_R, w_F, w_{RR}, w_{FF}, w_{RF} for order α_s^{n+2} contributions

- Advantages
  - Renormalization and factorization scale can be varied *independently* and by *any* factor
    - No time-consuming 're-calculation' of splitting functions in NLO necessary
  - Only small increase in amount of stored coefficients

- Implementation
  - *Two* different observables can be used for the scales
    - e.g.: H_T and $p_{T,max}$
    - or e.g.: p_T and $|y|$
    - …
  - *Any function* of those *two observables* can be used for calculating scales

Step 4: Mass Production

- **NNLOJET + APPLfast**
  - **Massive parallelised computing on Virtual Machines with 24h lifetime**
  - **Example with fastNLO, APPLgrid example in progress**

Job Type	# Jobs	Events / Job	Runtime / Job	# Events	Total Output	Total Runtime
LO	10	140 M	20.6 h	1.4 G	24 MB	206 h
NLO-R	200	6 M	19.0 h	1.2 G	1.3 GB	3800 h
NLO-V	200	5 M	21.2 h	1.0 G	1.2 GB	4240 h
NNLO-RRa	5000	60 k	22.5 h	0.3 G	26 GB	112500 h
NNLO-RRb	5000	40 k	20.3 h	0.2 G	27 GB	101500 h
NNLO-RV	1000	200 k	19.8 h	0.2 G	6.4 GB	19800 h
NNLO-VV	300	4 M	20.5 h	1.2 G	2.0 GB	6150 h
Total	11710	---	---	5.5 G	64 GB	248196 h

3 times 11710 grids/tables + all NNLOJET output!
Final 3 files for analysis are O(10 MB) each.