status of APPLfast

interpolation grids for NNLO

DIS18, Kobe, Japan, April 2018

D. Britzger (Heidelberg)

C. Gwenlan (Oxford)

A. Huss (CERN)

K. Rabbertz (KIT)

M. Sutton (Sussex)

with support from IP3

motivation, and APPLfast

interpretation of exp. data requires fast theory predictions

- often need **repeated computation of same cross section**, for EG: pdf uncertainties and/or alternative sets; scale variations, μ_R, μ_F; variation of $\alpha_s(M_Z)$; SM parameter fits

jet cross section calcs. at NLO were slow – historical reason for development of interpolation grids

- nowadays **NNLO** in general very demanding!
- **need procedure for fast repeated computations of higher order cross sections** → interpolation grids using **APPLgrid** or **fastNLO**

APPLfast: common project of **APPLgrid**, **fastNLO** and **NNLOJET** authors:

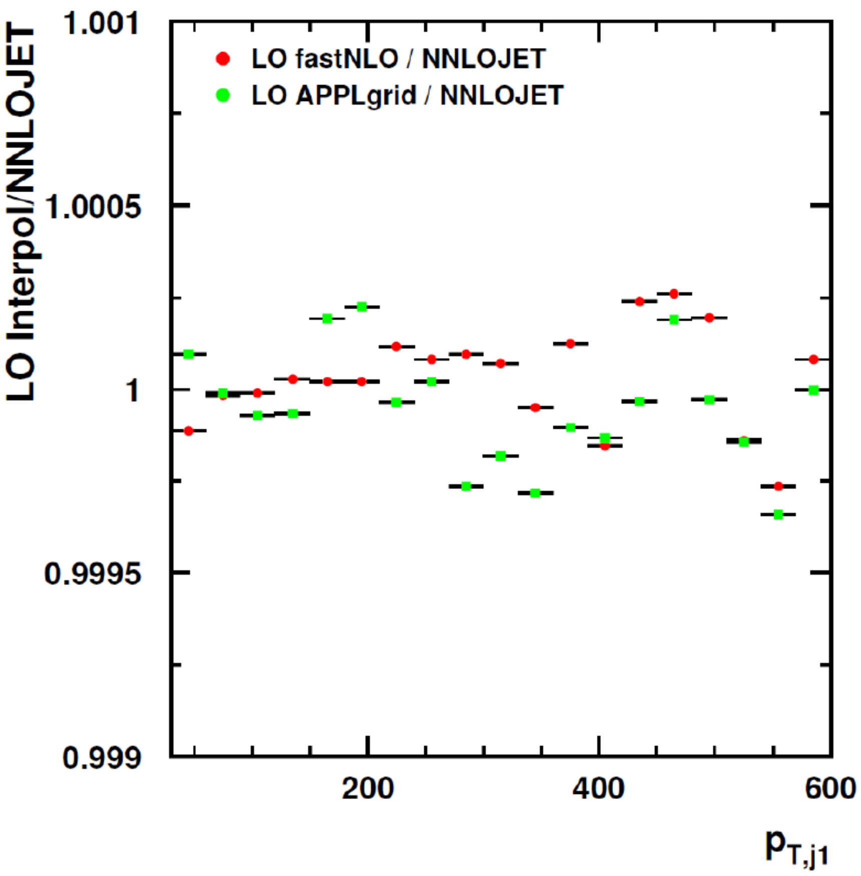
- interface between **NNLOJET** and **fast grid technology**;
- implementation for both APPLgrid and fastNLO;
- aims to be as **unobtrusive** as possible, for both ends of interface;
- **flexible**; intended to be reusable by other theory codes

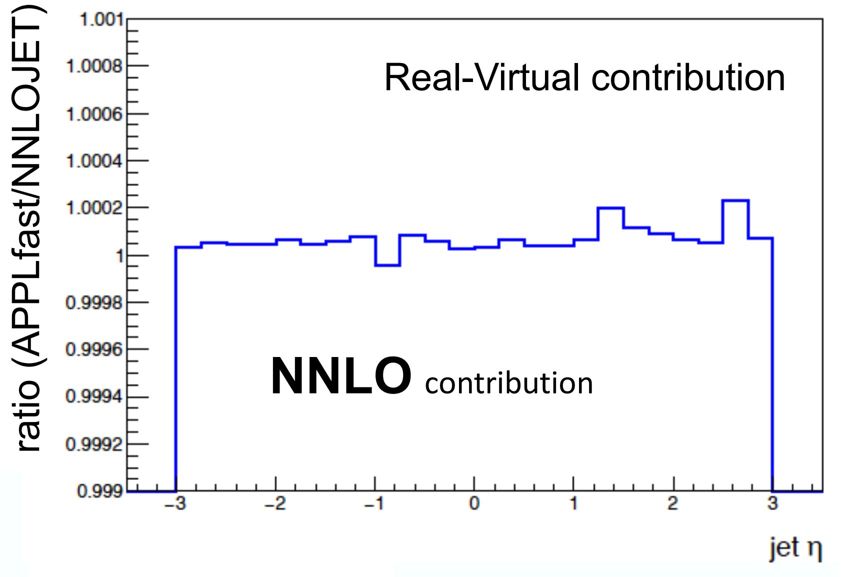
APPLfast grid generation (workflow)

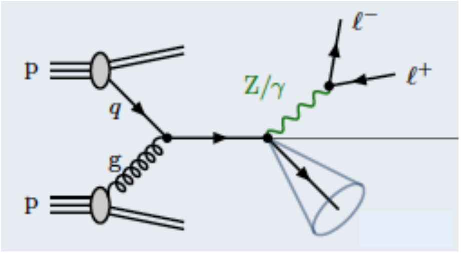
1. **pre-processing**: establish details of parameterisation using short test jobs (EG. number of grid nodes, interpolation order etc.) **O(10 h)**
2. **NNLOJET warm-up:** optimise the NNLOJET VEGAS phase space in dedicated NNLOJET job [1 long (multicore) job per process] **O(100h)**
3. **NNLOJET – APPLfast warmup:** run with grid filling enabled to establish optimised phase space (exact strategy differs between APPLgrid and fastNLO, but this is hidden in interface); only phase space provided by NNLOJET, providing significant speed up **O(100 h)**
4. **grid production run:** thousands of jobs in parallel **O(250 kh)**
5. **post-processing**: statistical evaluation and combination of all produced output sub-grids from production run **O(100 h)**
6. **validation, validation, validation** **O(?? h)**
7. **present final results** **30 mins!**

step 1: pre-processing

M. Sutton at QCD@LHC 2016







Z+Jet test setup:

E_{CM} = 8TeV

Pt_{jet} > 30 GeV

$|y_{jet}|$ < 3

$|y_{ll}|$ < 5

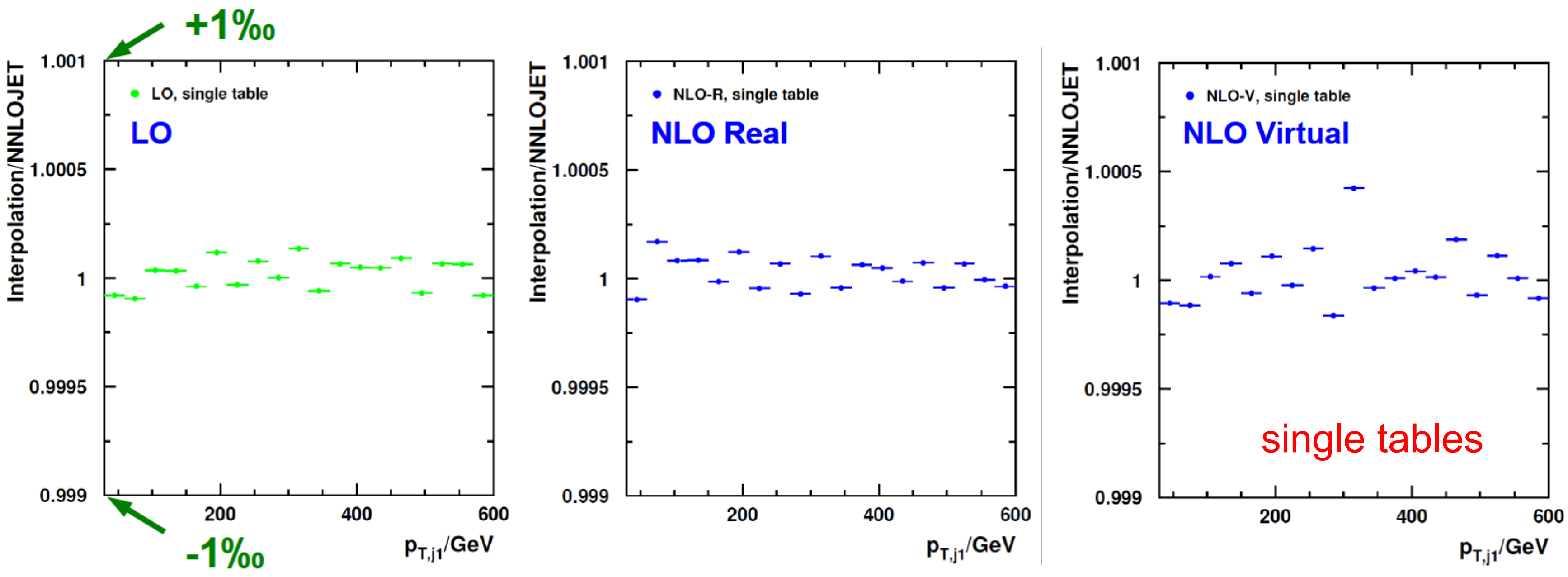
80 < M_{ll} < 100 GeV

$\mu_R = \mu_F = M_Z$ = fixed

Z+Jet approximation test jobs

note y-axis range; **sub-permille agreement** reached at LO, NLO and NNLO in validation jobs

checking LO/NLO interpolation

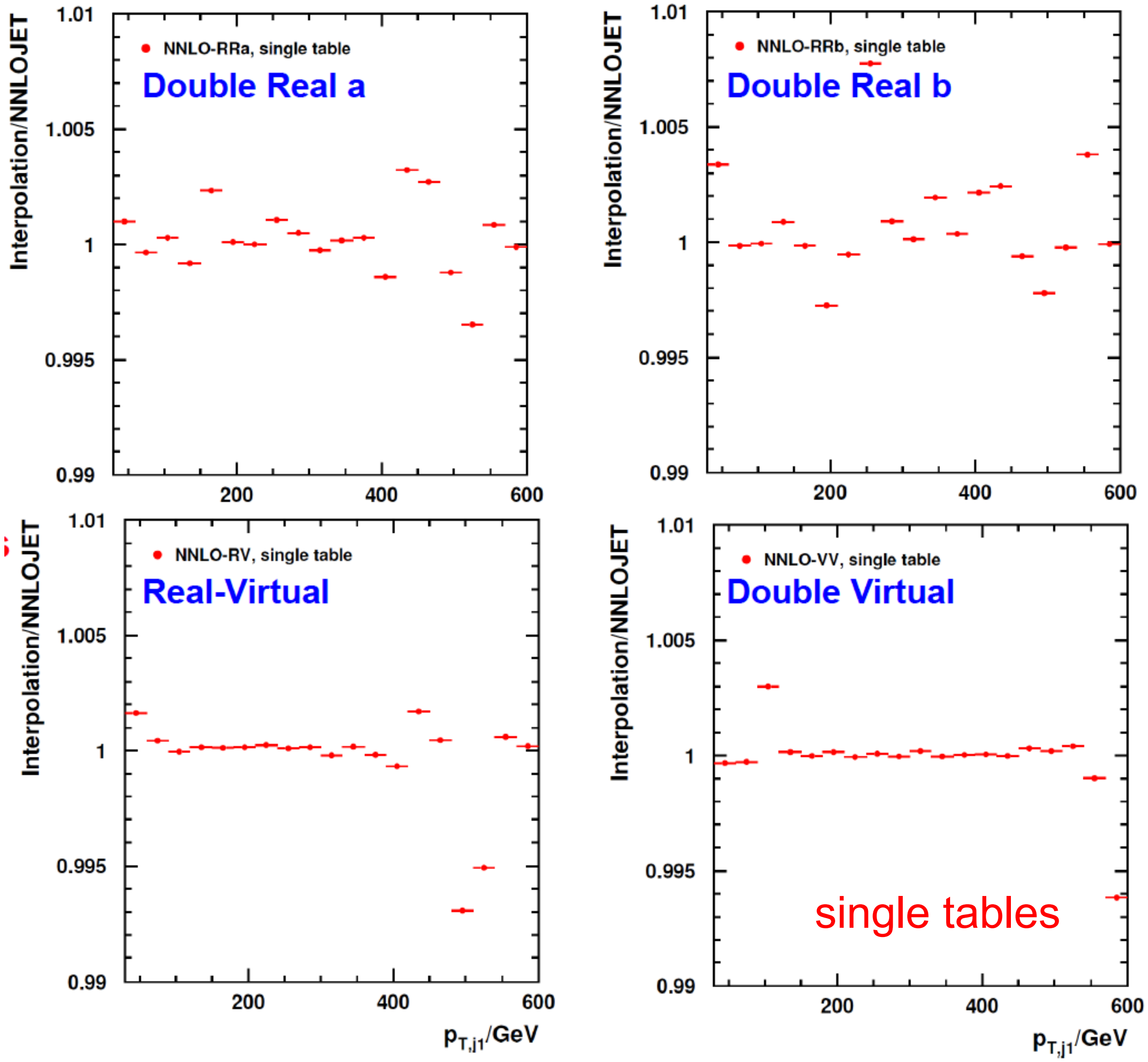


- **check closure of individual grids/tables from each job separately**
- general agreement to much better than per mille level
- at this level, interpolation of LHAPDF may be limiting factor

checking NNLO interpolation

- **agreement to sub percent level**
- some impact of fluctuations visible; to be dealt with in global procedure to combine grids for final cross sections

(a/b indicates a technical phase space separation for RR)



single tables

step 2: initial VEGAS warm up

NNLOJET warmup – no grid generation

one job per cross section contribution type EG. LO, R, V, RR(a,b), RV, VV;

internal NNLOJET multi-threading possible

Job Type	# Jobs	Threads / Job	Events / Job	Runtime / Job	Total Runtime
LO	1	16	32 M	0.35 h	0.35 h
NLO-R	1	16	16 M	1.0 h	1.0 h
NLO-V	1	16	16 M	1.0 h	1.0 h
NNLO-RRa	1	32	5 M	17.5 h	17.5 h
NNLO-RRb	1	32	5 M	20.7 h	20.7 h
NNLO-RV	1	16	8 M	22.4 h	22.4 h
NNLO-VV	1	16	8 M	24.6 h	24.6 h
Total	7	-	-	-	87.6 h

step 3: phase space exploration

APPLfast warmup:

NNLOJET run without CPU-time expensive weight calculation;
at least one job per process needed to determine phase space limits individually;
jobs can be parallelised if necessary

Job Type	# Jobs	Events / Job	Runtime / Job	# Events	Total Runtime
LO	5	500 M	12 h	2.5 G	60 h
NLO-R	5	300 M	18 h	1.5 G	90 h
NLO-V	5	500 M	13 h	2.5 G	65 h
NNLO-RRa	10	50 M	13 h	0.5 G	130 h
NNLO-RRb	10	50 M	15 h	0.5 G	150 h
NNLO-RV	5	300 M	19 h	1.5 G	90 h
NNLO-VV	5	500 M	12 h	2.5 G	60 h
Total	45	---	---	11.5 G	645 h

(presented tables used for extensive testing; overkill for normal use)

step 4: mass production

NNLOJET+APPLfast warmup:

massive parallelised computing on virtual machines with 24h lifetime;

example with fastNLO:

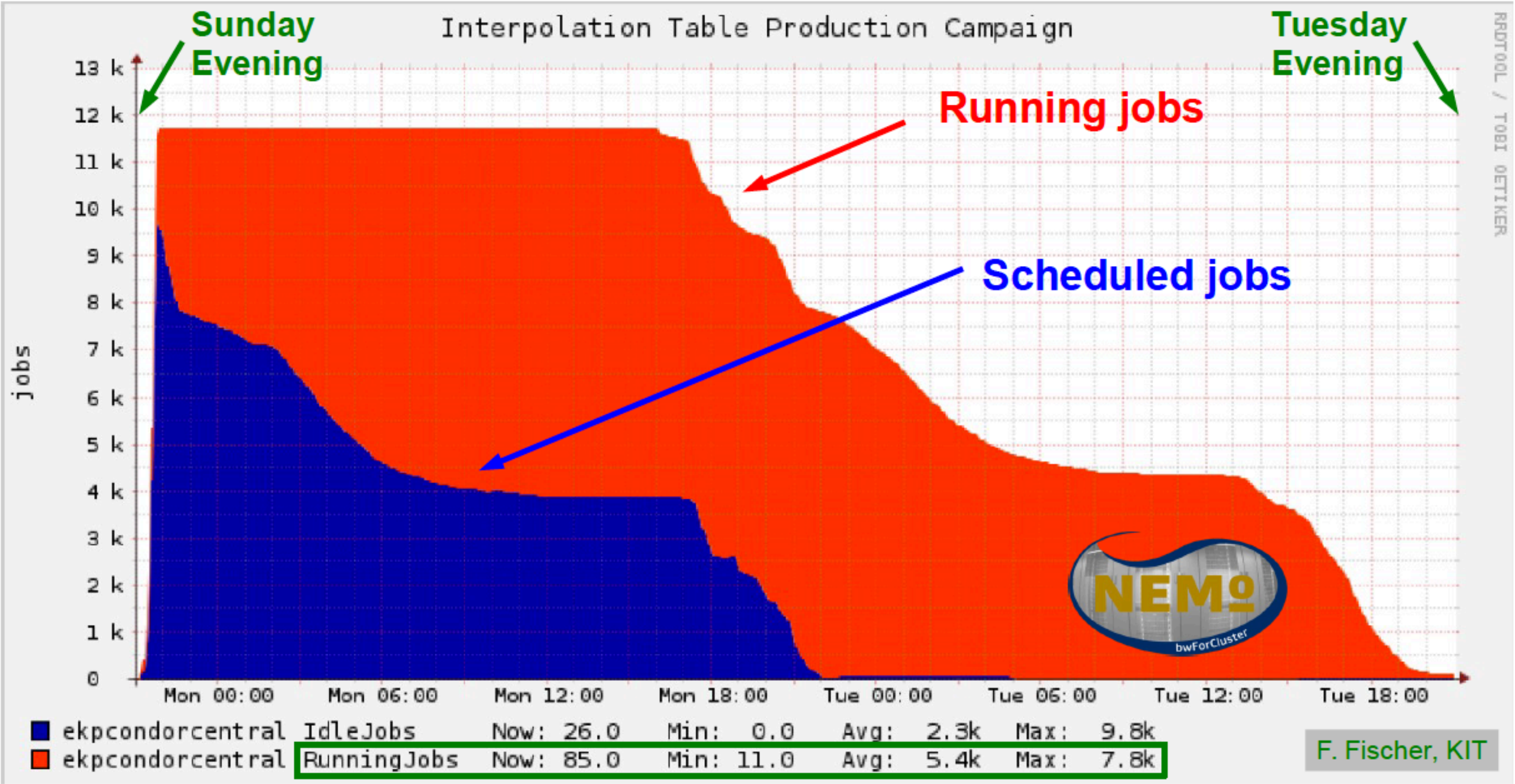
Job Type	# Jobs	Events / Job	Runtime / Job	# Events	Total Output	Total Runtime
LO	10	140 M	20.6 h	1.4 G	24 MB	206 h
NLO-R	200	6 M	19.0 h	1.2 G	1.3 GB	3800 h
NLO-V	200	5 M	21.2 h	1.0 G	1.2 GB	4240 h
NNLO-RRa	5000	60 k	22.5 h	0.3 G	26 GB	112500 h
NNLO-RRb	5000	40 k	20.3 h	0.2 G	27 GB	101500 h
NNLO-RV	1000	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 × 11710 grids/tables + all NNLOJET output!

Final 3 files for analysis are O(10MB) each

production campaign

optimised scenario: finished in two days with 7800 parallel jobs at maximum



Sunday Evening

Tuesday Evening

Running jobs

Scheduled jobs

F. Fischer, KIT

(thanks to bwHPC and the NEMO HPC cluster team in Freiburg)

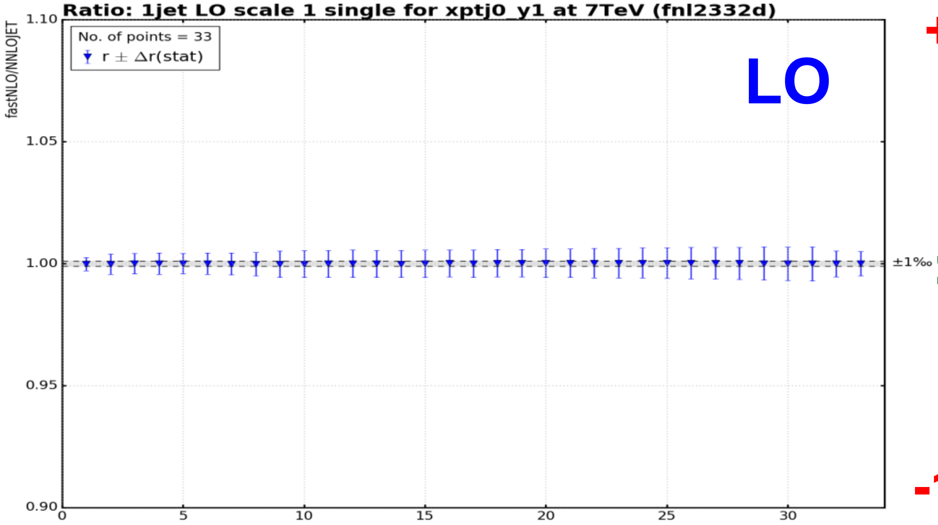
step 5: post-processing

- **checking, purging, combining**
- check again interpolation quality for individual grids
- run NNLOJET combination script ⟶ weight tables
- weighted merging of grids
- check and treat potential remaining unsuppressed fluctuations
- **… do some interesting physics**

step 6: validation

- **check every aspect we can think of… else, Murphy’s Law!**
- check each contribution (LO, R, V, RR(a,b), 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 and comparisons EG. to other programs
- ETC ETC.

inclusive jet pt – single grid



+10%

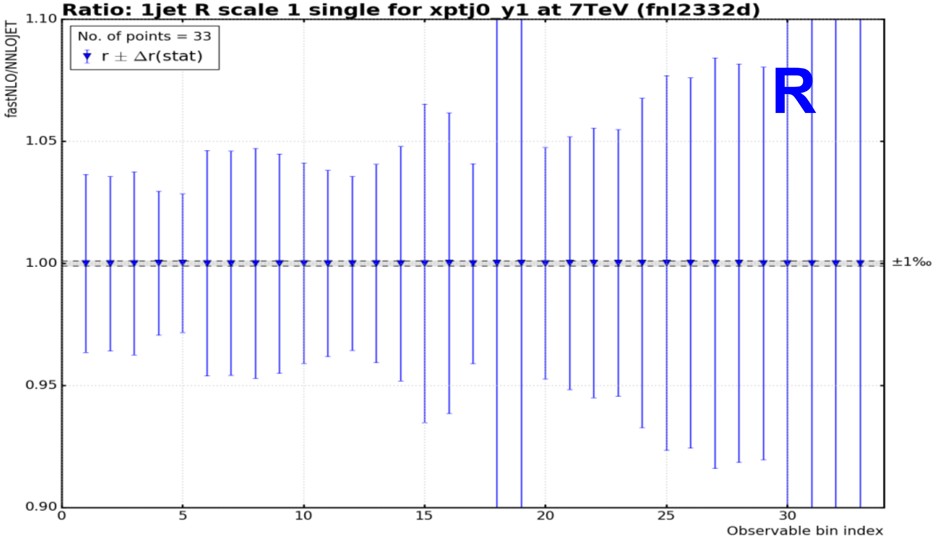
±1‰

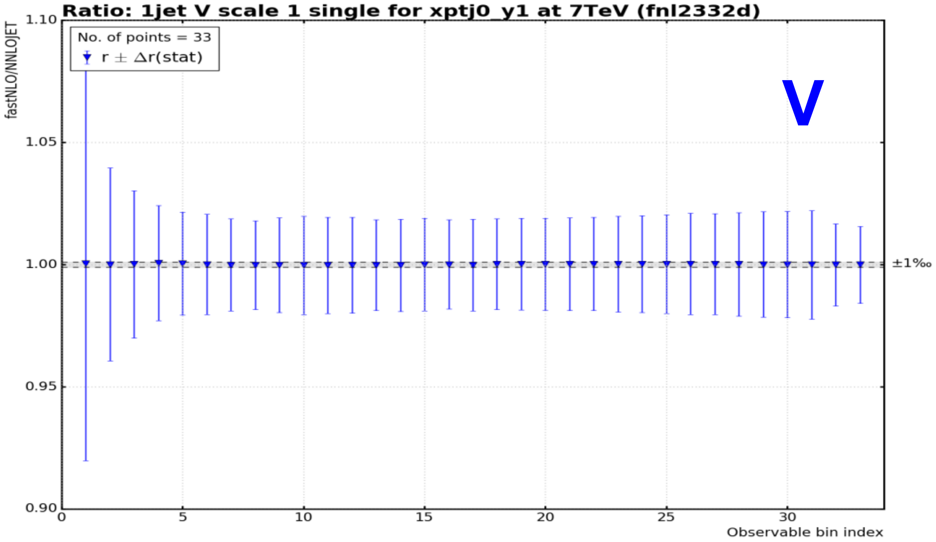
-10%

ratio **APPLfast/NNLOJET**

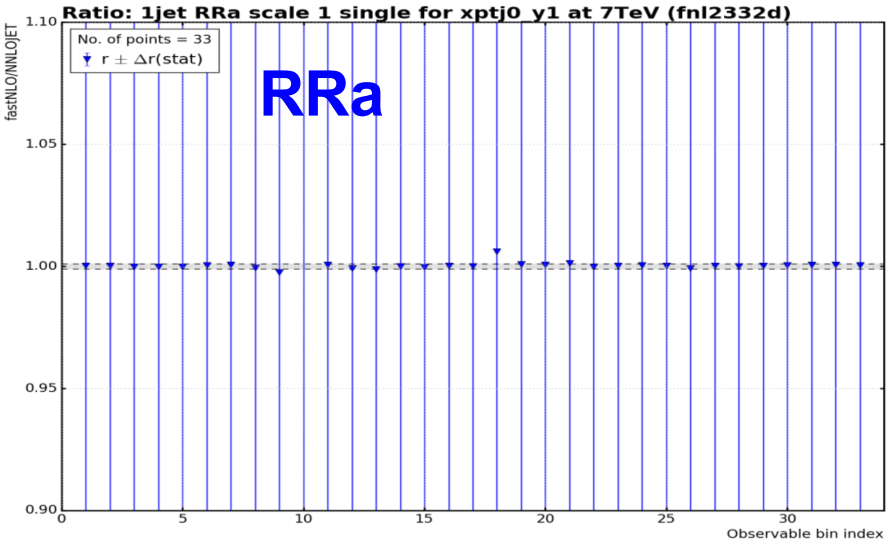
error bars:

stat. uncertainty estimate from NNLOJET





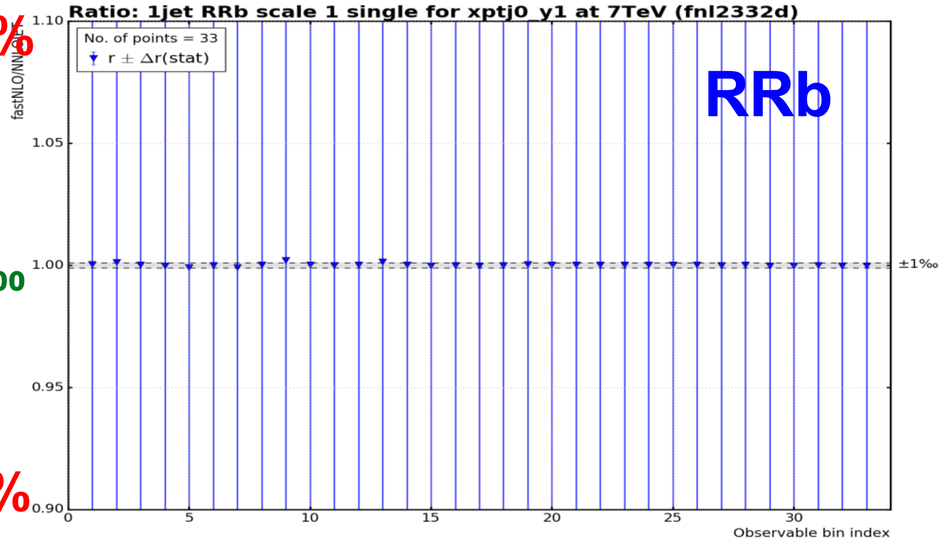
inclusive jet pt – single grid

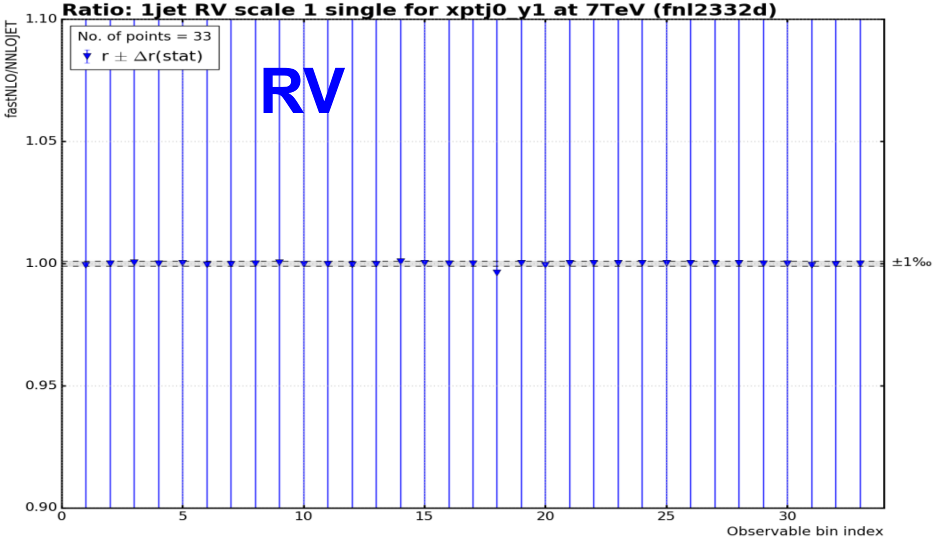


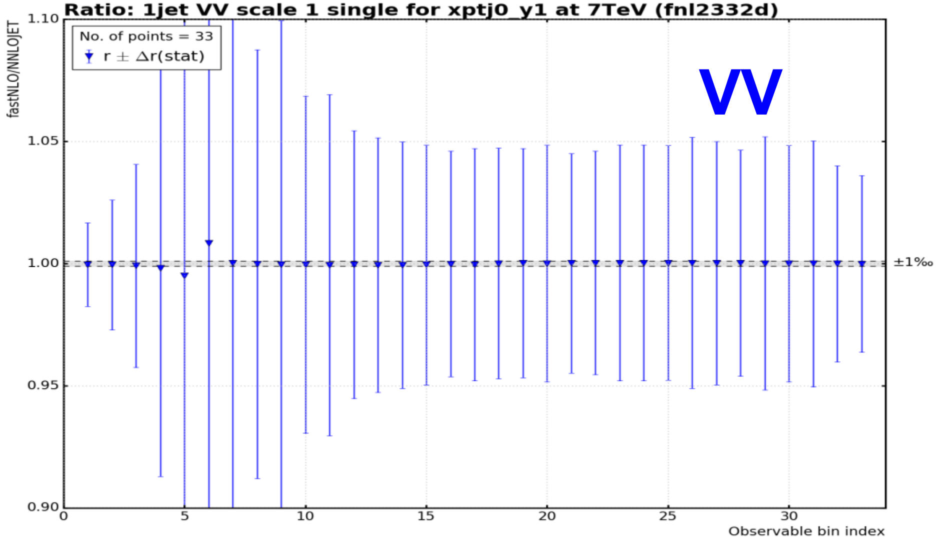
+10%

±1‰

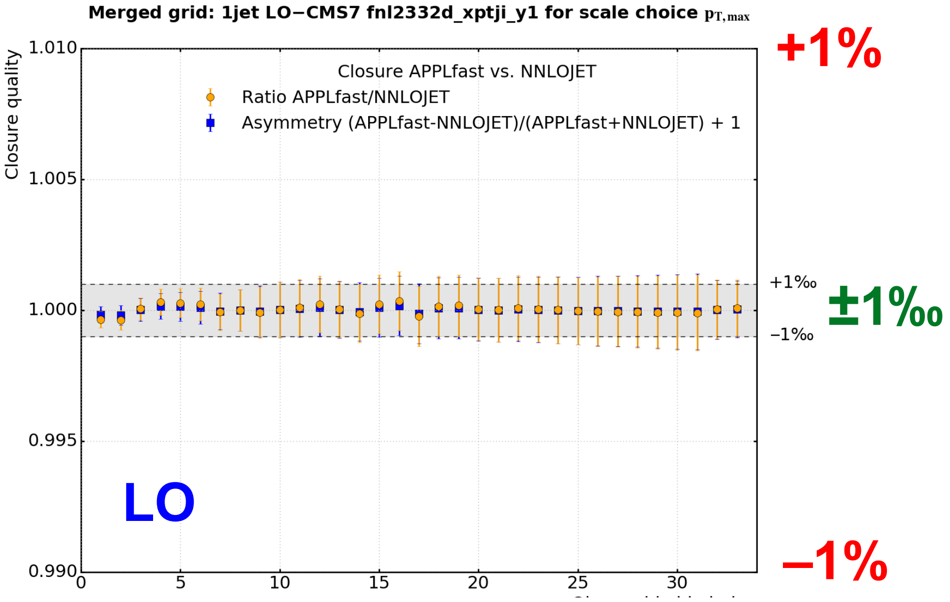
-10%





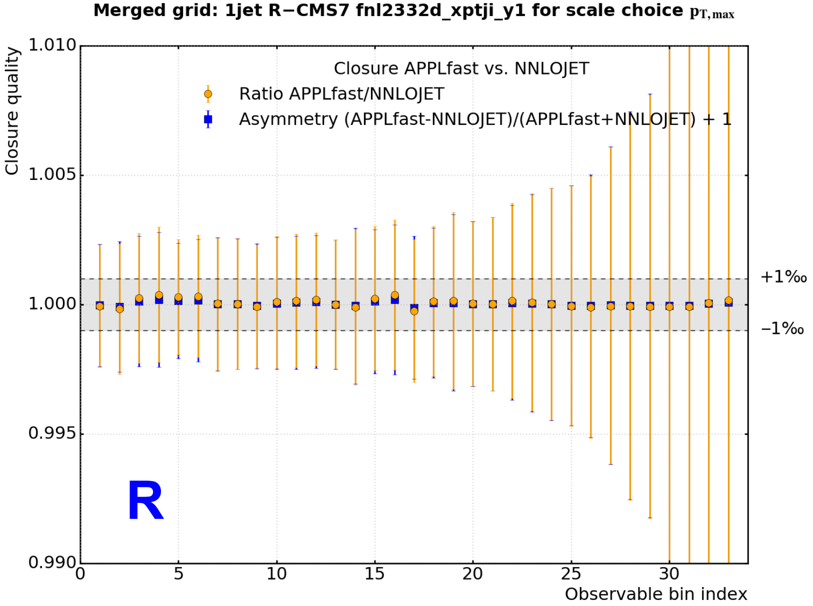


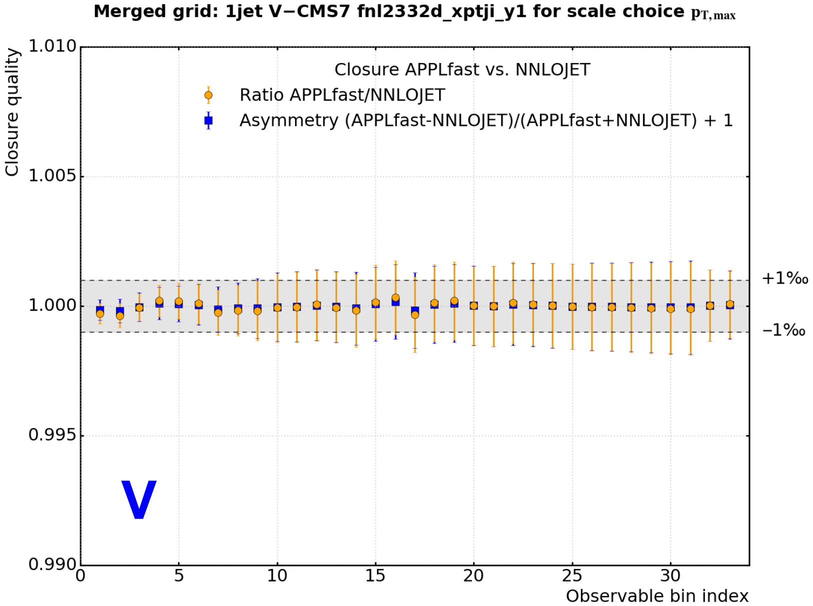
inclusive jet pt – combined grid


Merged grid: 1jet LO−CMS7 fnl2332d_xptji_y1 for scale choice $p_{T,\mathrm{max}}$

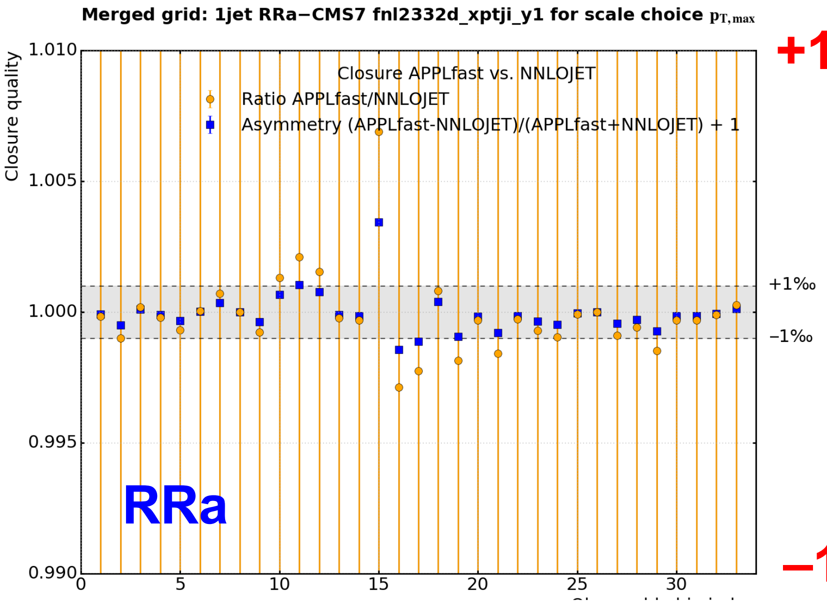
closure **APPLfast/NNLOJET**

error bars: statistical uncertainty estimate from NNLOJET


Merged grid: 1jet R−CMS7 fnl2332d_xptji_y1 for scale choice $p_{T,\mathrm{max}}$


Merged grid: 1jet V−CMS7 fnl2332d_xptji_y1 for scale choice $p_{T,\mathrm{max}}$

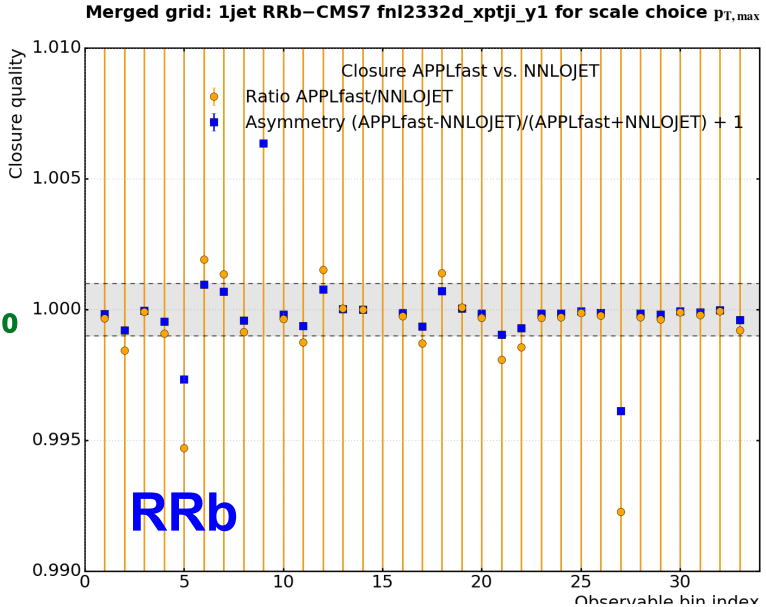
inclusive jet pt – combined grid

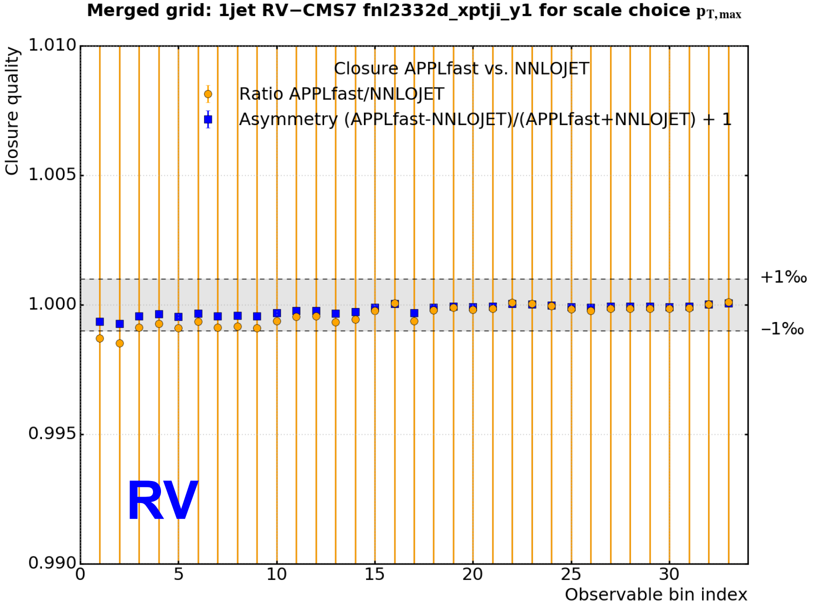

Merged grid: 1jet RRa−CMS7 fnl2332d_xptji_y1 for scale choice $p_{\mathrm{T,max}}$

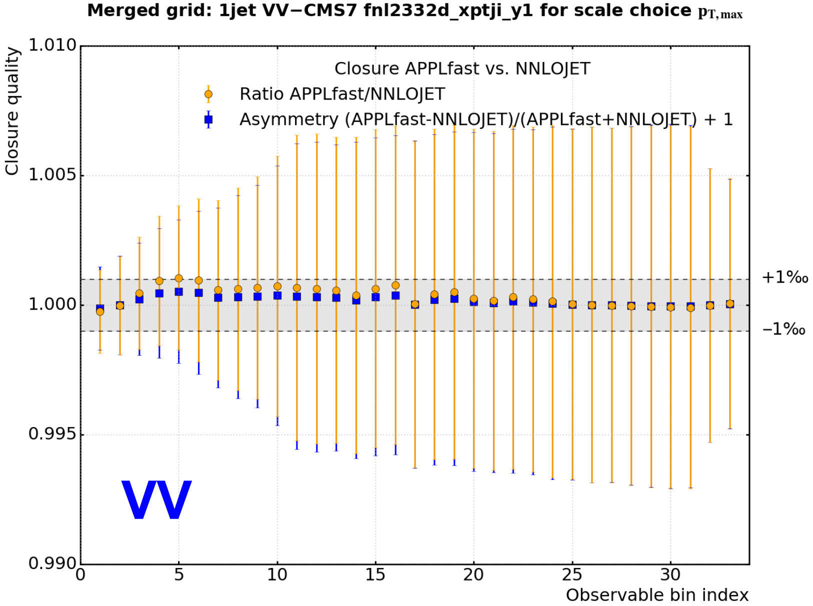
+1%

±1‰

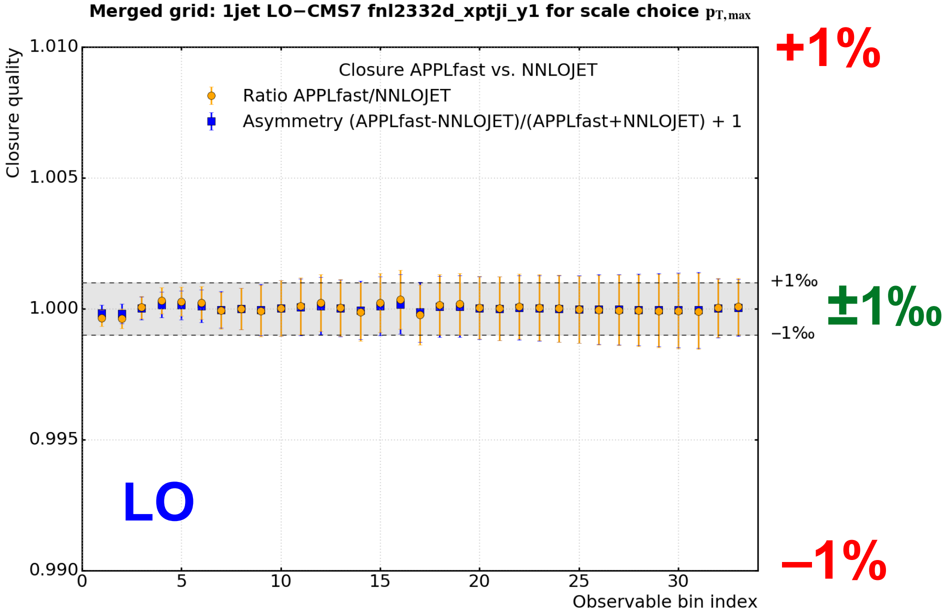
–1%


Merged grid: 1jet RRb−CMS7 fnl2332d_xptji_y1 for scale choice $p_{\mathrm{T,max}}$


Merged grid: 1jet RV−CMS7 fnl2332d_xptji_y1 for scale choice $p_{\mathrm{T,max}}$

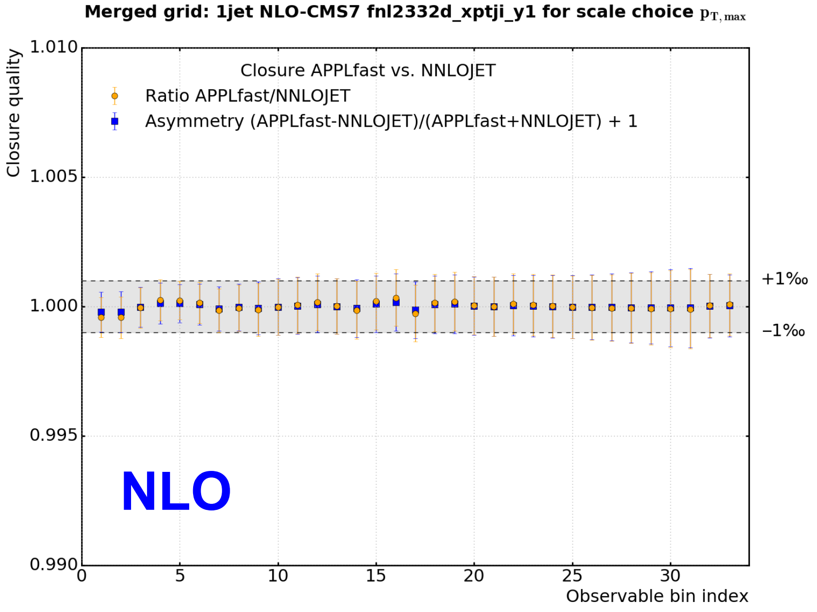

Merged grid: 1jet VV−CMS7 fnl2332d_xptji_y1 for scale choice $p_{\mathrm{T,max}}$

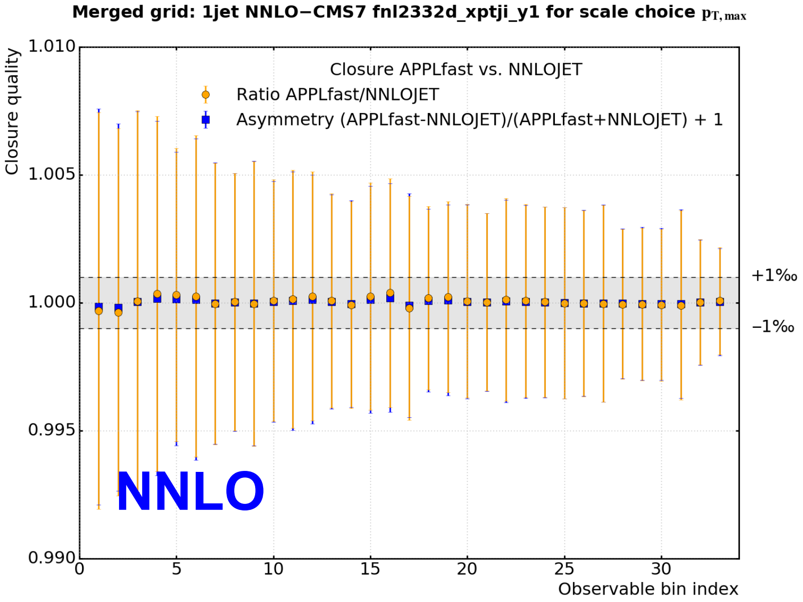
inclusive jet pt – combined grid



closure **APPLfast/NNLOJET**

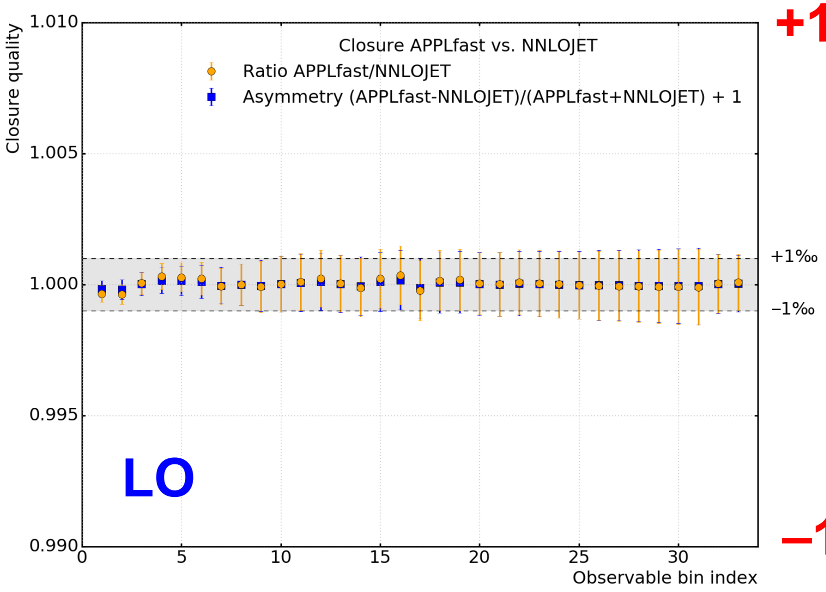
error bars: statistical uncertainty estimate from NNLOJET





inclusive jet pt – combined grid

Merged grid: 1jet LO−CMS7 fnl2332d_xptji_y1 for scale choice $p_{T,\max}$



+1%

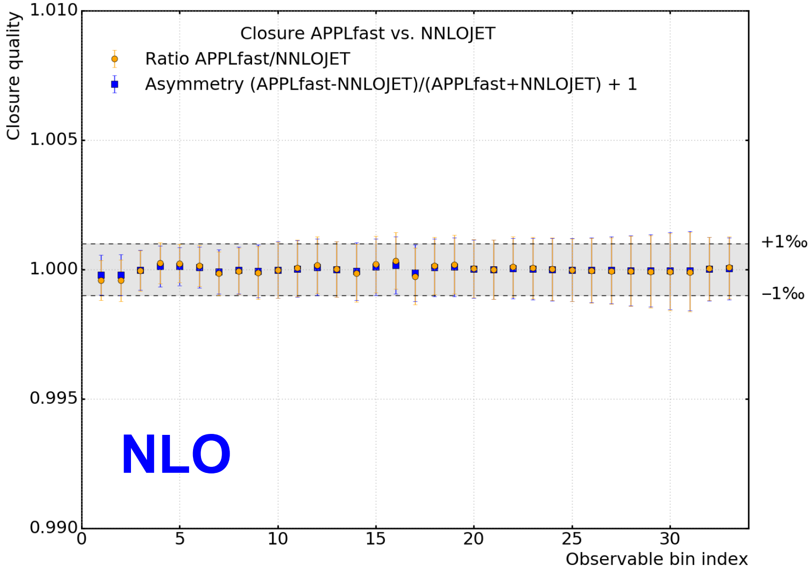
±1‰

–1%

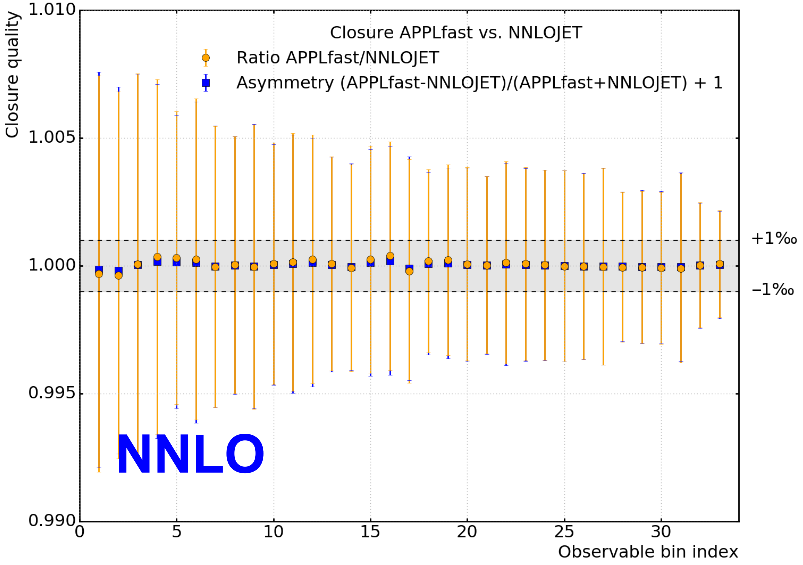
closure **APPLfast/NNLOJET**

error bars: statistical uncertainty estimate from NNLOJET

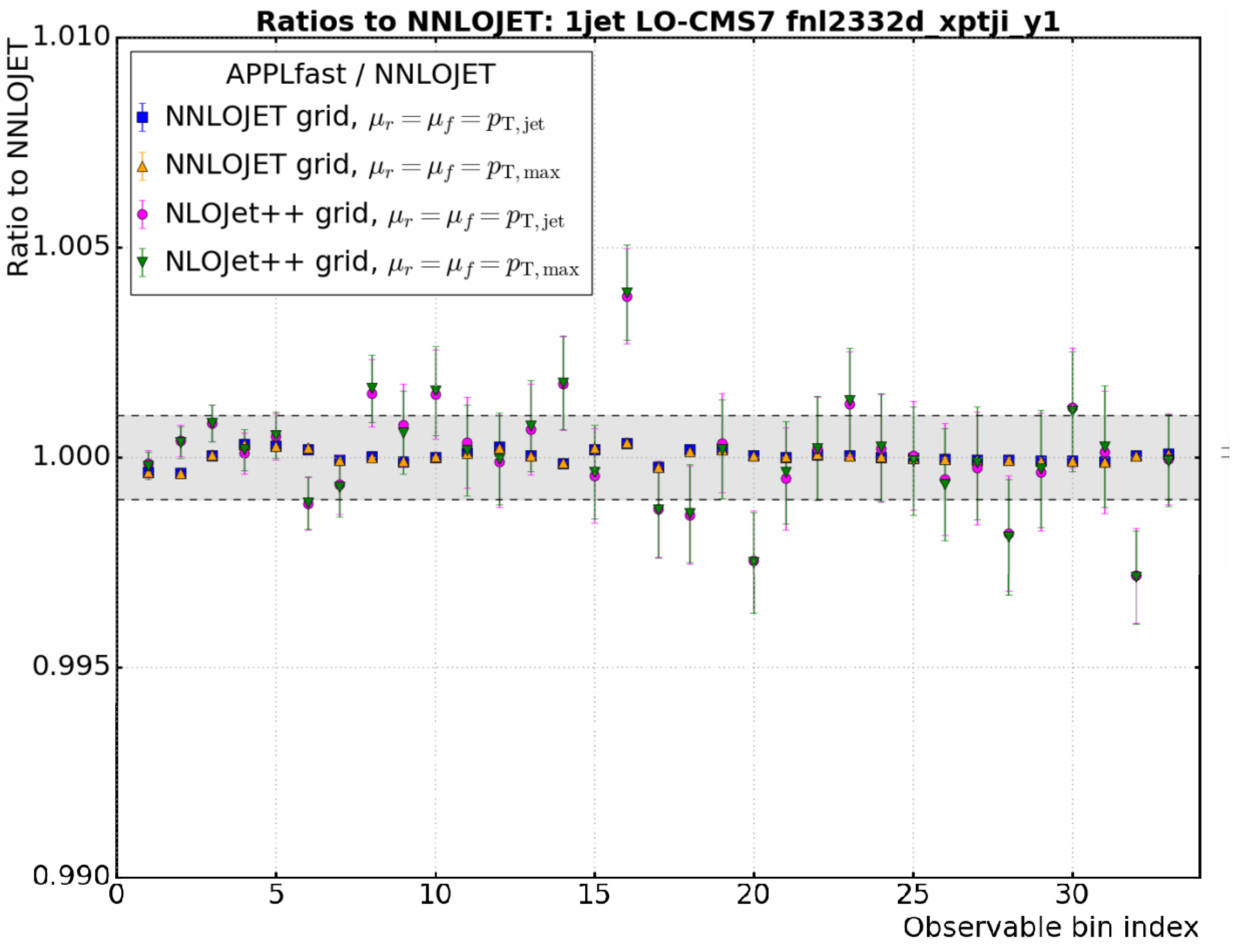
Merged grid: 1jet NLO-CMS7 fnl2332d_xptji_y1 for scale choice $p_{T,\max}$



Merged grid: 1jet NNLO−CMS7 fnl2332d_xptji_y1 for scale choice $p_{T,\max}$



cross check with NLOJet++ at LO

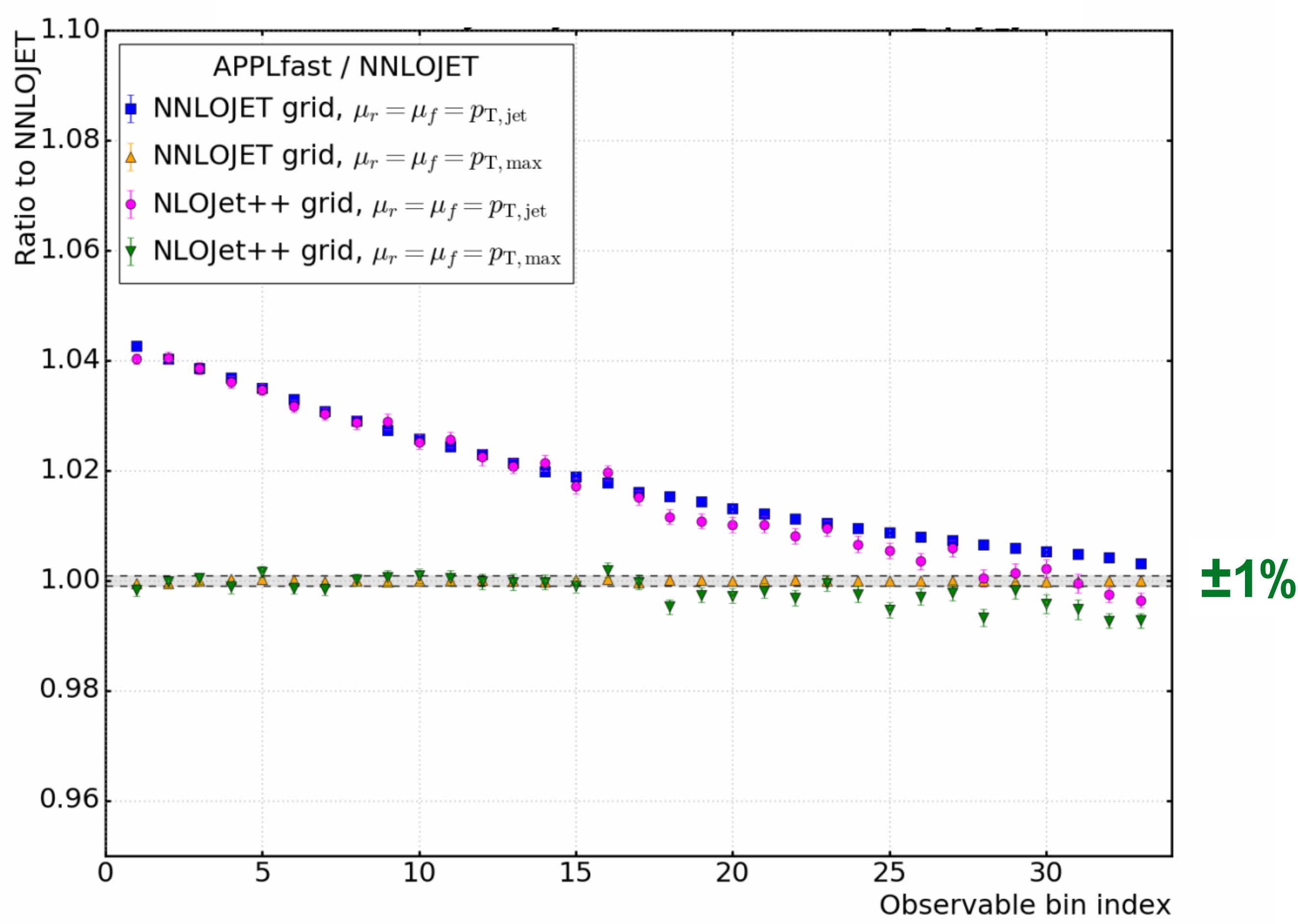


±1‰

ratio always to NNLOJET with scale ptmax

error bars: stat. uncertainty estimate from NNLOJET and NLOJet++

cross check with NLOJet++ at NLO



±1‰

ratio always to NNLOJET with scale ptmax

error bars: stat. uncertainty estimate from NNLOJET and NLOJet++

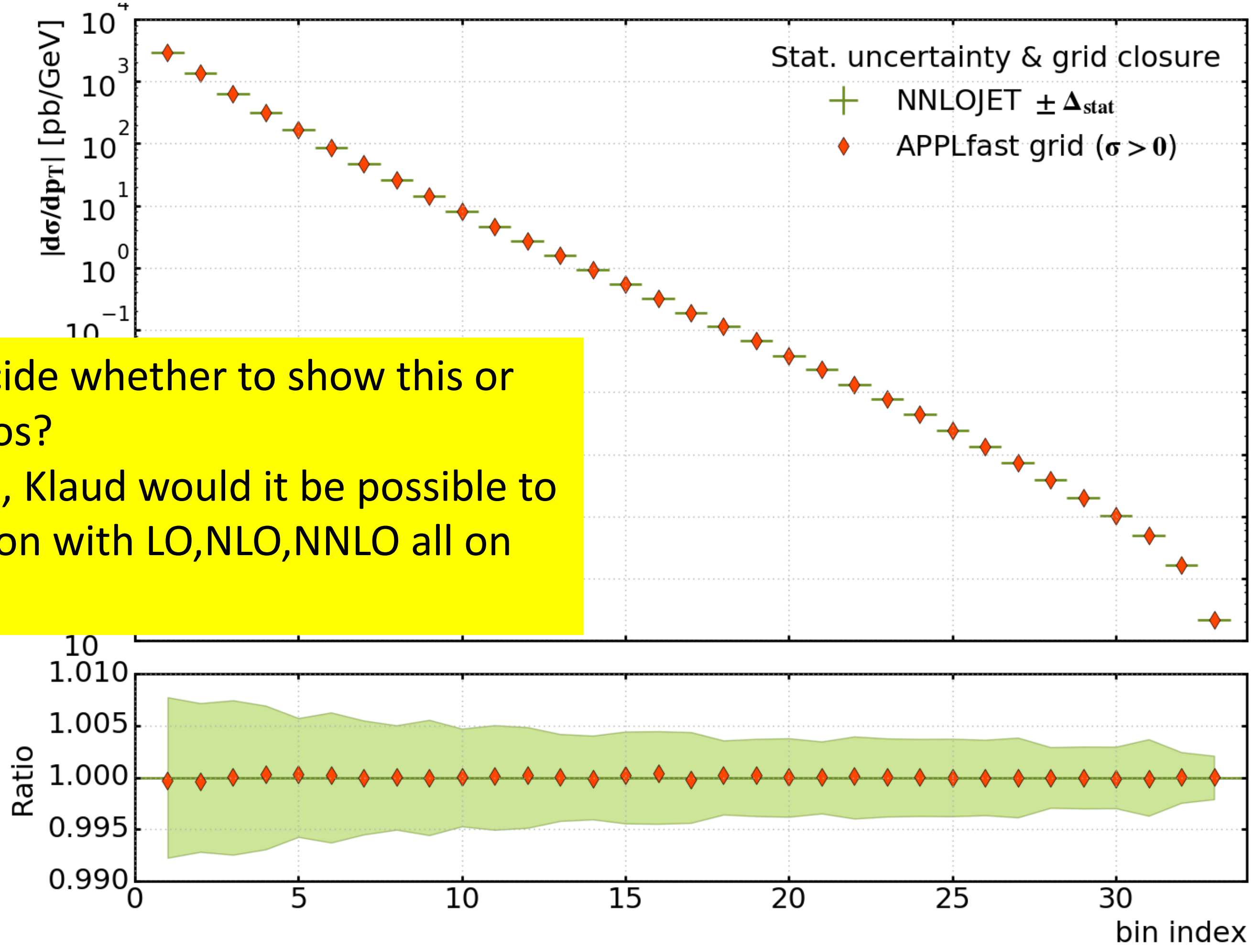
step 7: results

DISCLAIMER – what follows is not to be yet used;

work still in progress, but shows that 'full chain of steps' in place

Any preference on what exact wording I use here to say : DO NOT USE

cross section comparison



Need to decide whether to show this or just the ratios?
If we do use, Klaud would it be possible to have a version with LO,NLO,NNLO all on one?

NNLOJET vs APPLfast reproduction (scale ptmax)

outlook

NNLOJET provides NNLO QCD calculations in common interface

pp: Z & W incl., Z+jet, W+jet, jet+dijet, H incl., H+jet; **ep:** jet+dijet; **ee:** 3jets

APPLfast interface (NNLO-bridge) is working

numerous adaptions implemented by all sides for optimal performance

large scale productions tested for pp Z+jet, pp inclusive jet and DIS jet

DIS NNLO pdfs and αs published (H1 coll., Eur.Phys.J C77 (2017) 791)

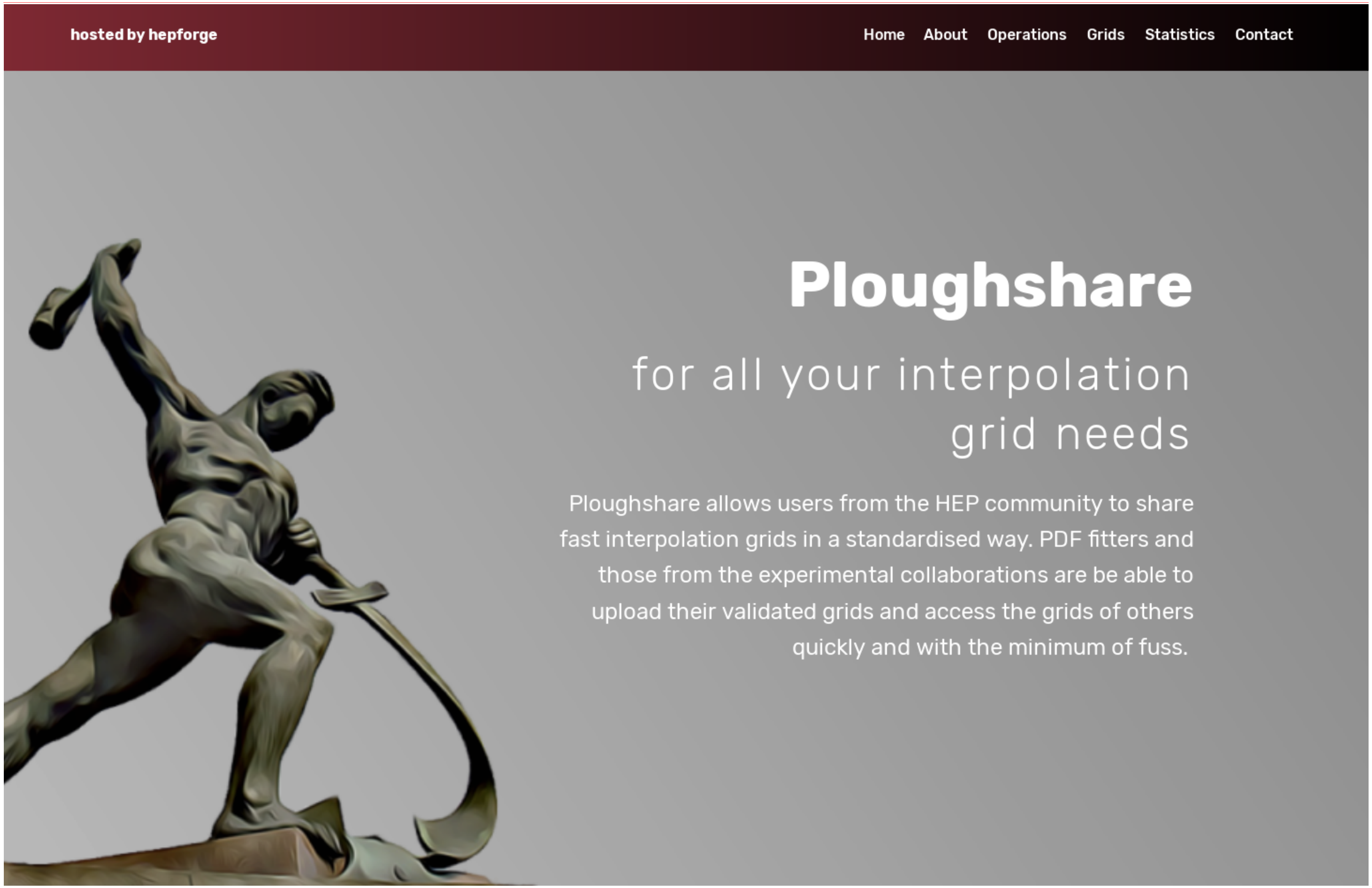
final combination prescription for NNLOJET results received last Nov.

removes fluctuations from incomplete cancellations, weighted interpolation table merging implemented

production for CMS pp inclusive jets at 7 TeV recently finished

many new NNLO interpolation grids planned

grid distribution – Ploughshare

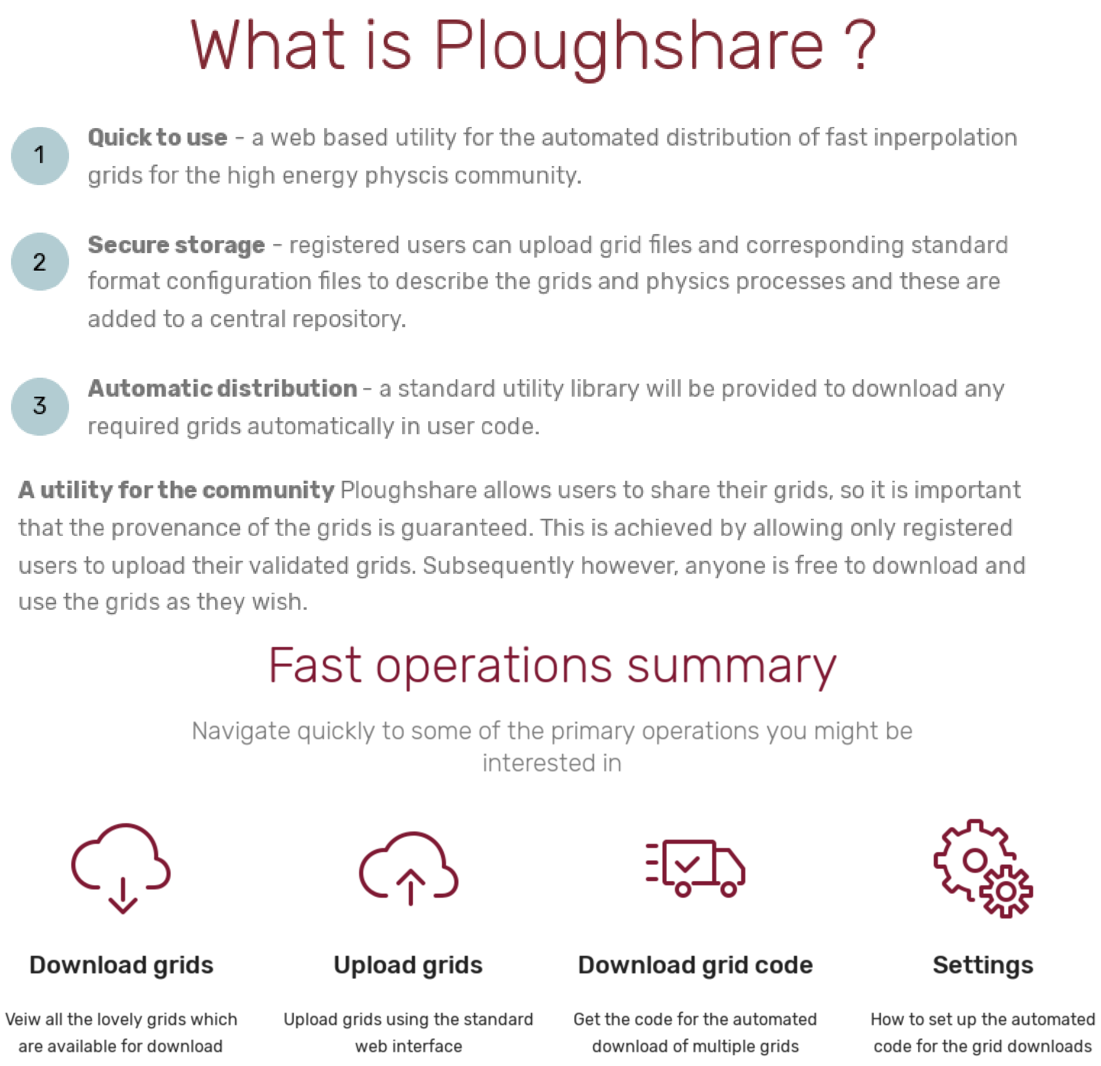


Ploughshare

for all your interpolation grid needs

Ploughshare allows users from the HEP community to share fast interpolation grids in a standardised way. PDF fitters and those from the experimental collaborations are be able to upload their validated grids and access the grids of others quickly and with the minimum of fuss.

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 **HepForge** package; registered users can upload grids with documentation
- automated job treats upload
  - adds to appropriate location in file system
  - generates relevant lists and display web pages
- provides user interface for **automated download** with a simple line of code
- expression of interest from many stakeholders…
- proof of concept ready… **HELP WELCOME!**

summary

APPLfast interface (NNLO–Bridge) and interpolation is working

large scale productions tested for pp Z+jet, pp inclusive jet and DIS jet

combination of grids with weights à la NNLOJET implemented

addressing last issues, checking on possible remaining outliers in grids; finalising validation

starting to produce a series of APPLgrid and/or fastNLO tables for various processes with publically available data

final grids will be made available via a common repository on HEPFORGE; open for contributions from the community

extras