High multiplicity processes with BlackHat and Sherpa

Daniel Maître
IPPP, Durham

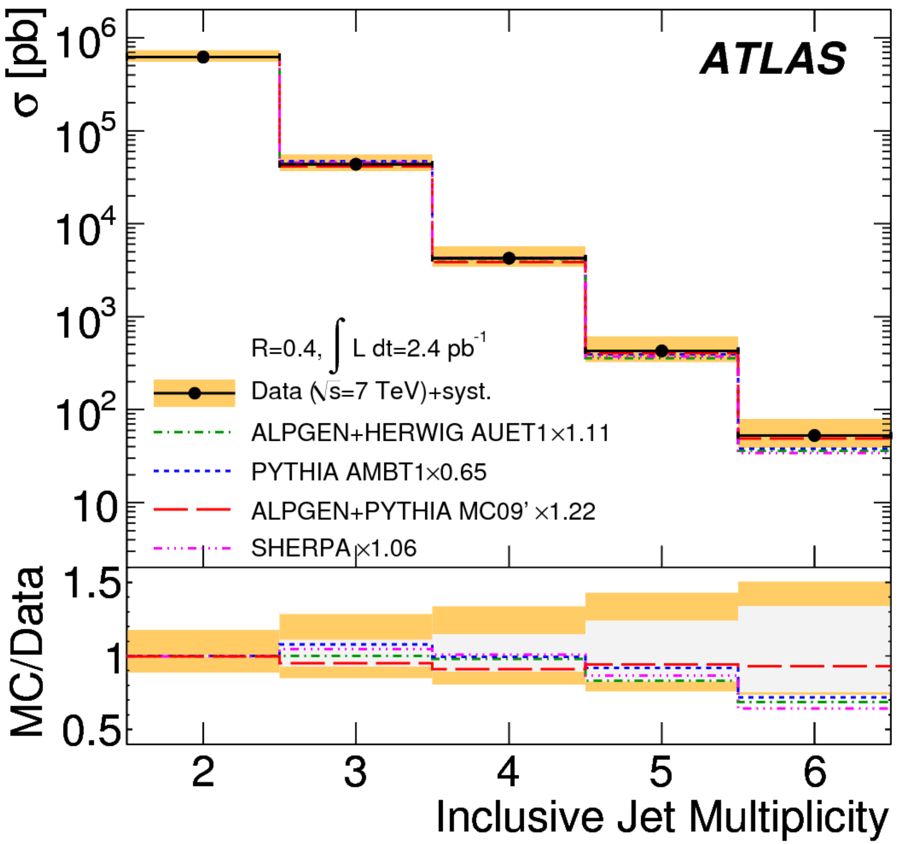
In collaboration with
Zvi Bern, L. Dixon, F. Febres Cordero, S. Höche, H. Ita,
D. Kosower, K. Ozeren, N. A. Lo Presti

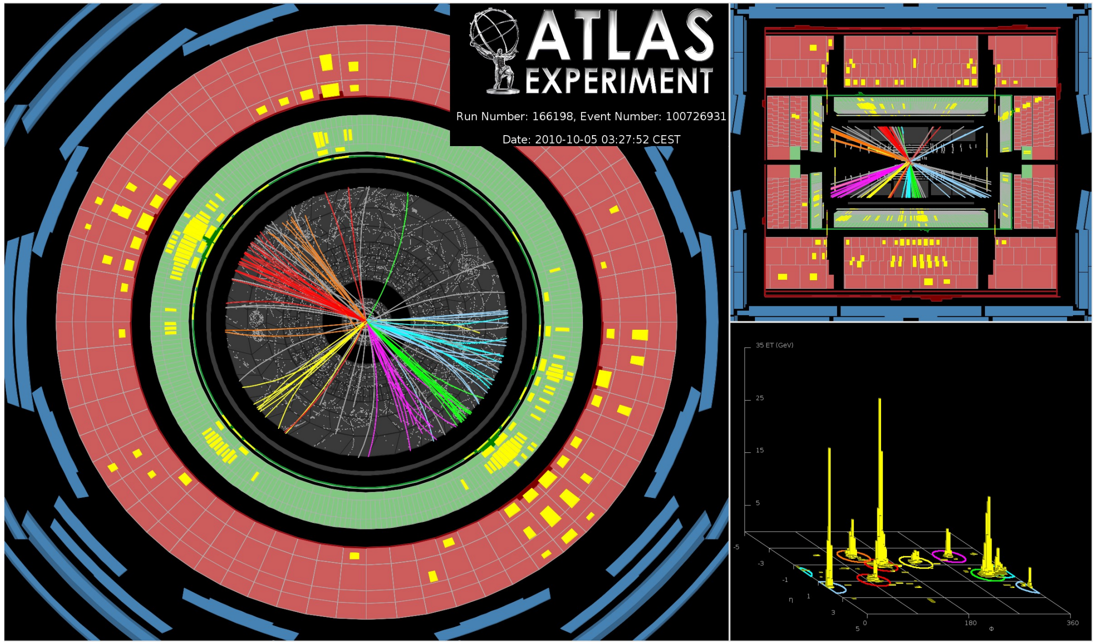
Outline

- BlackHat+Sherpa n-Tuple files and library
- Exclusive sums
- Universal behaviour in W+jets production

High multiplicities

- At the LHC events with many jets are common place



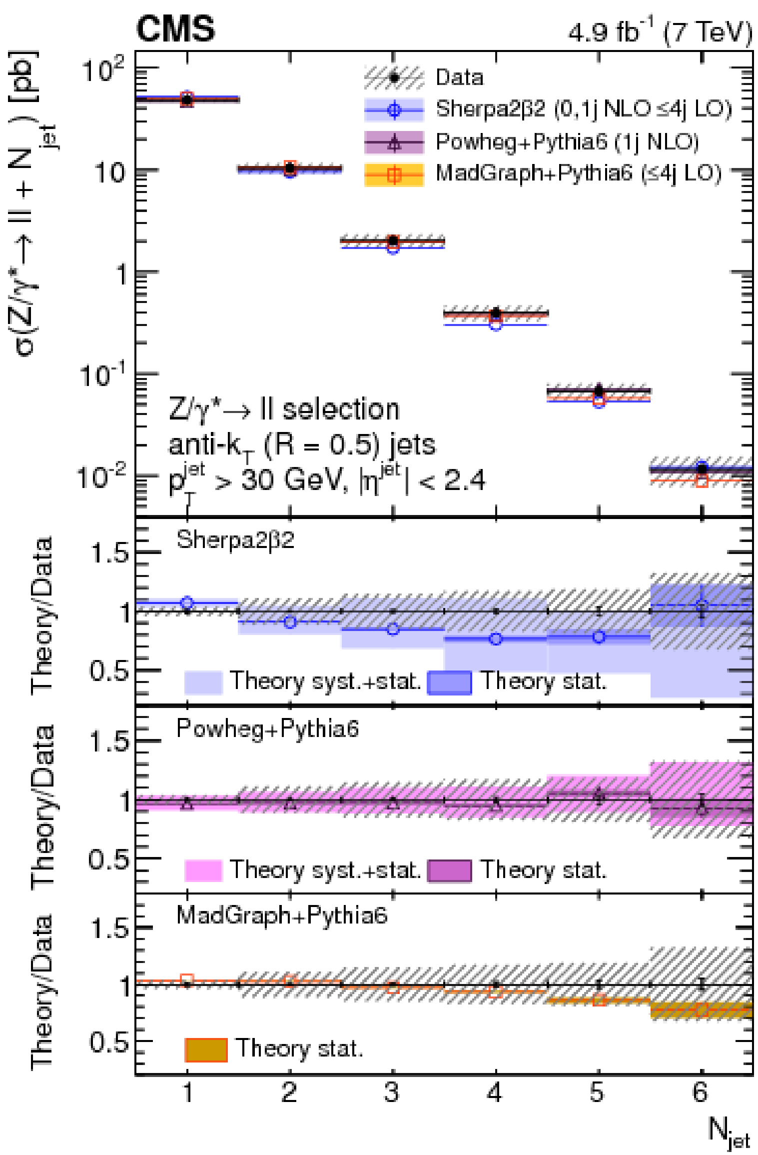


8-jets event at ATLAS

ArXiv:1202.3548

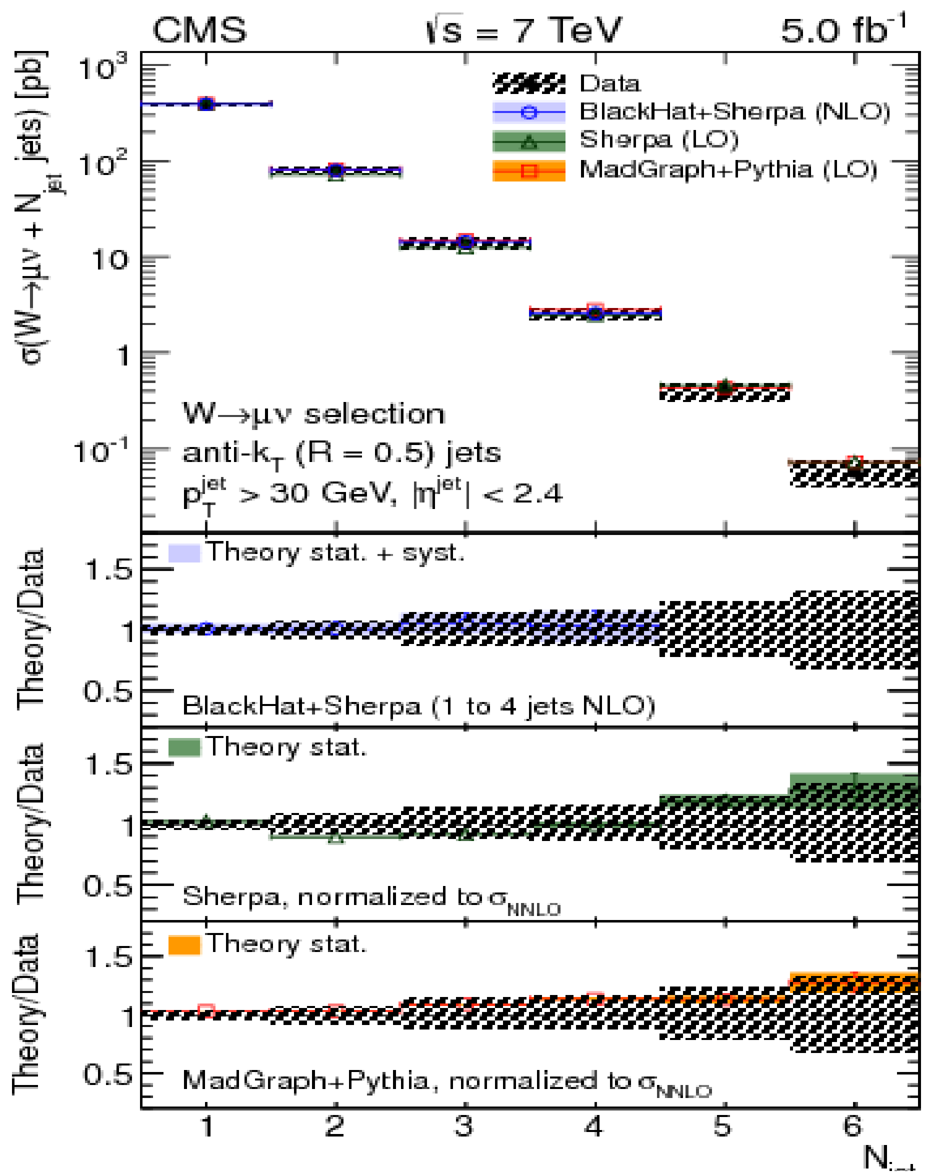
High multiplicities

- Z+jets



arXiv:1408.3104

- W+jets



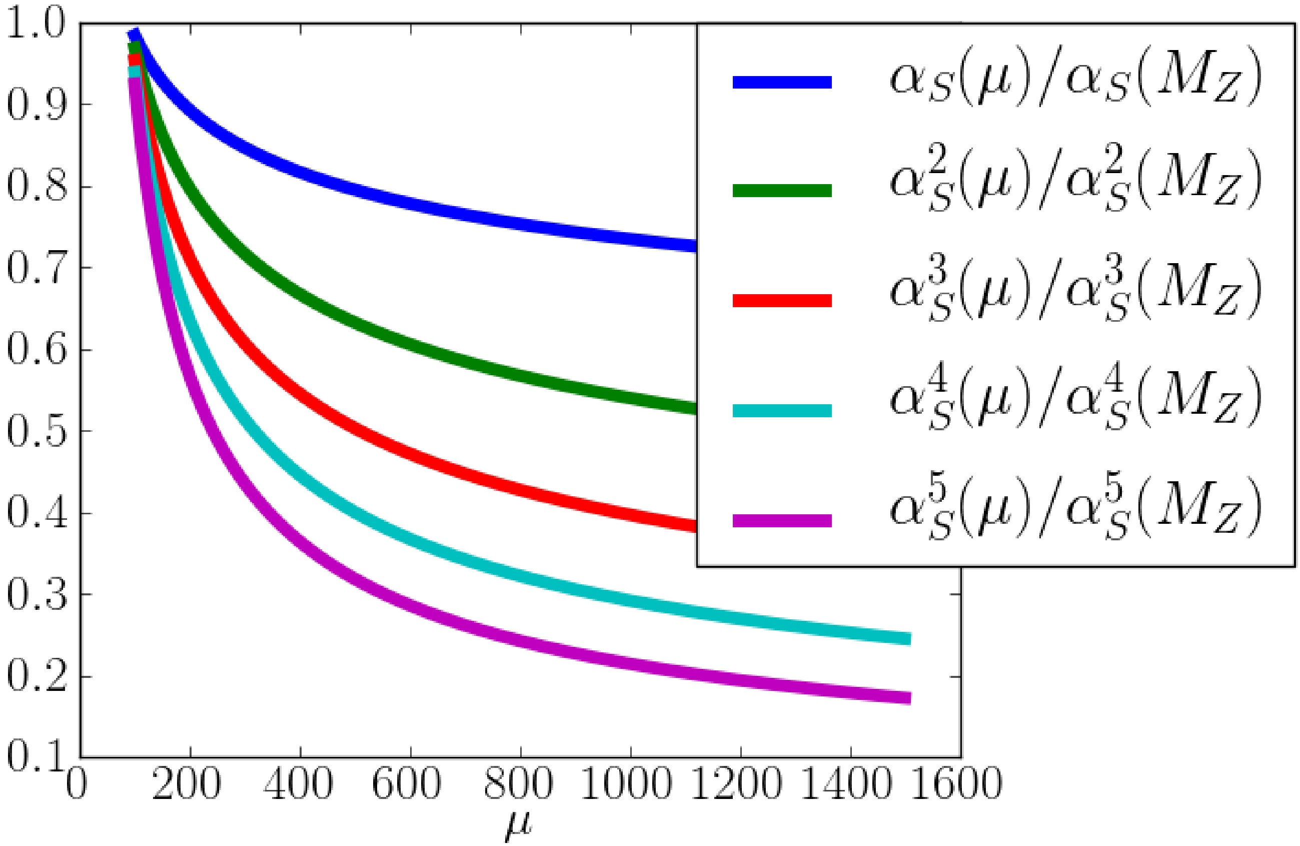
arXiv:1406.7533

Precise predictions

- Precise predictions are needed
  - Signal
  - Background
  - Also for data-driven methods
    - Extrapolation from control to signal region
    - Transfer of information from one process to another
- NLO improves
  - Absolute normalisation
  - Shapes of distributions
  - Scale dependence

NLO scale dependence

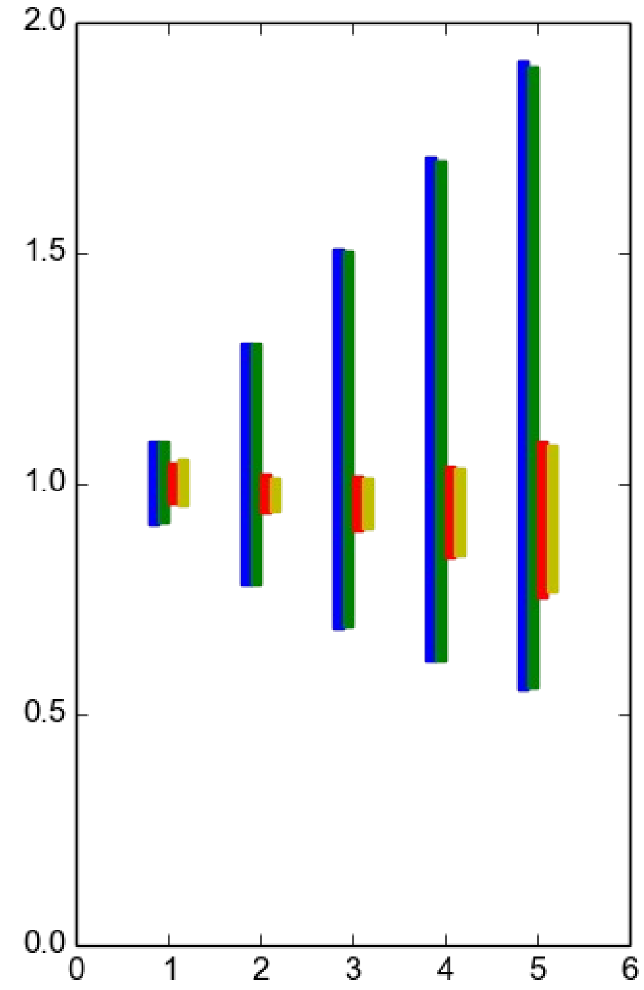
- The theory prediction depends on two unphysical scales, the renormalisation and factorisation scale
- The strong coupling constant is a function of the renormalisation scale



Scale variation

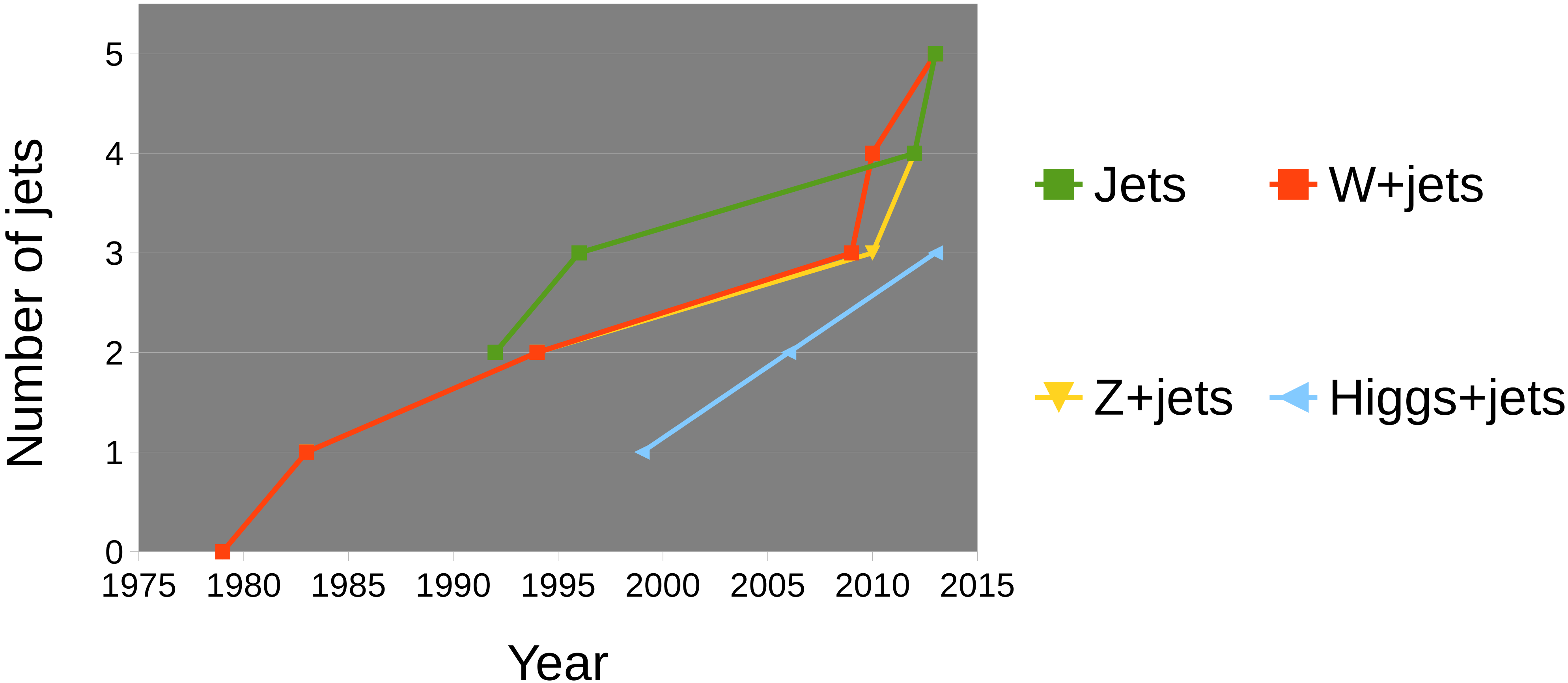
- Scale variation is largely reduced at NLO
- More and more important as multiplicity increases
- W+n jets cross sections:

Jets	W^- LO	W^- NLO	W^+ LO	W^+ NLO
1	$284.0(0.1)^{+26.2}_{-24.6}$	$351.2(0.9)^{+16.8}_{-14.0}$	$416.8(0.6)^{+38.0}_{-35.5}$	$516(3)^{+29.}_{-23}$
2	$83.76(0.09)^{+25.45}_{-18.20}$	$83.5(0.3)^{+1.6}_{-5.2}$	$130.0(0.1)^{+39.3}_{-28.1}$	$125.1(0.8)^{+1.8}_{-7.4}$
3	$21.03(0.03)^{+10.66}_{-6.55}$	$18.3(0.1)^{+0.3}_{-1.8}$	$34.72(0.05)^{+17.44}_{-10.75}$	$29.5(0.2)^{+0.4}_{-2.8}$
4	$4.93(0.02)^{+3.49}_{-1.90}$	$3.87(0.06)^{+0.14}_{-0.62}$	$8.65(0.01)^{+6.06}_{-3.31}$	$6.63(0.07)^{+0.21}_{-1.03}$
5	$1.076(0.003)^{+0.985}_{-0.480}$	$0.77(0.02)^{+0.07}_{-0.19}$	$2.005(0.006)^{+1.815}_{-0.888}$	$1.45(0.04)^{+0.12}_{-0.34}$



Recent progress

- Number of jets in addition to the vector boson



- Has been called 'NLO revolution'

BlackHat+Sherpa

- BlackHat is a C++ library for virtual one-loop matrix elements
- Recent calculations with Sherpa:
  - W/Z+4 jets
  - 2,3,4 jets
  - Z/gamma ratios with up to 3 jets
  - W+5 jets
  - diphoton+2 jets

n-Tuple files [arXiv:1310.7439]

- High multiplicity NLO calculations are computationally intensive
- It would not be possible to rerun a high multiplicity W+jet calculation every time a new interesting observable comes up
- Matrix elements are expensive, while
  - Jet clustering
  - Observables
  - PDF evaluation

  are relatively cheap
- Store each matrix element, PS point and the information necessary to change the factorisation and renormalisation scales in large files we call n-Tuple files
- We use ROOT file as storage

n-Tuple files

- Goodies
  - One can change the analysis cuts, add observables
  - Scale variation
  - PDF errors (otherwise extremely expensive)
  - Easy communication between theorists and experimenters
  - No need for specific know-how of the tool which produced the NLO calculation
  - Easier to “endorse” an event file than a program
- Price to pay
  - Large files
  - Generation cuts need to be loose enough to accommodate many analysis → efficiency cost

n-Tuple availability

- The n-Tuple files are available
  - On the grid
  - On castor at CERN
- For a range of processes

Process	Pathname	Energy	Jet cut
W+ + 1,2,3,4 jets	Wp<n>j	7TeV	25GeV
W+ + 1,2,3 jets	Wp<n>j	8TeV	20GeV
W- + 1,2,3,4 jets	Wm<n>j	7TeV	25GeV
W- + 1,2,3 jets	Wm<n>j	8TeV	20GeV
Z/gamma* + 1,2 jets	Zee<n>j	7TeV	25GeV
Z/gamma* + 3,4 jets	Zee<n>j	7TeV	20GeV
Z/gamma* + 1,2,3 jets	Zee<n>j	8TeV	20GeV
2,3,4 jets	PureQCD<n>j	7TeV,8TeV	40GeV

From http://blackhat.hepforge.org/trac/wiki/Availability

nTupleReader library

- We provide a C++ library to facilitate the use of the n-Tuple files
- Allows:
  - Change of factorisation and renormalisation scales
  - Change of pdf (from LHAPDF set), including error sets
- Has a Python interface
- Template for a customised implementation
- Available on hepforge

nTupleReader library

- Example

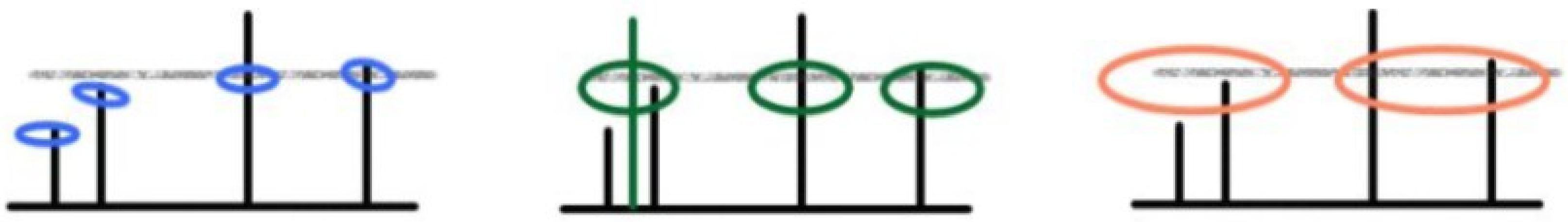
```
import nTupleReader as NR
r=NR.nTupleReader()
r.addFile('sample.root')

r.setPDF("CT10nlo.LHgrid")
r.setPDFmember(12)

while r.nextEntry():
    # compute new scales
    RenScale = ....
    FacScale = ....
    newWeight=r.computeWeight(FacScale,RenScale)
    // use this weight in the analysis
    ...
```

Pitfalls

- There can be negative weights
- For the real part the matrix elements and the subtraction terms are highly anti-correlated (by construction)
- Some common operations have to be modified to take this into account:
  - statistical error calculation
  - Rebinning, cumulative distributions

NLO event files

- Issues with NLO event files
  - Only fixed number of radii are supported



  - Negative weights
  - Highly anti-correlated weights (by construction!) affect
    - Integration error estimate
    - Rebining and cumulative distributions

NLO events

- Standard formula for integration error:

$$\sigma^2 = \langle(\omega - \langle\omega\rangle)^2\rangle = \langle\omega^2\rangle - (\langle\omega\rangle)^2$$
$$\simeq \frac{\sum \omega_i^2}{N} - \left(\frac{\sum \omega_i}{N}\right)^2 = \frac{1}{N}\left(\sum \omega_i^2 - \frac{\left(\sum \omega_i\right)^2}{N}\right)$$

- If we have only pairs of large anti-correlated weights

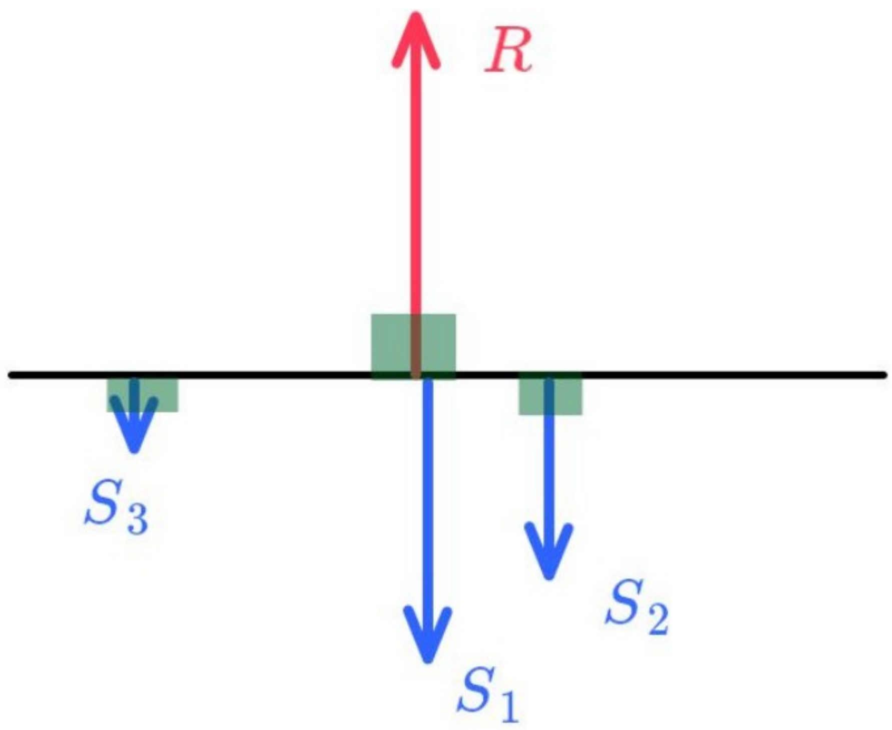
$$\omega_+ \simeq L\ ,\ \omega_- \simeq -L\ , \qquad \omega_+ - \omega_- \simeq s$$
$$\sigma^2 \simeq \frac{1}{N}\left(NL^2 + \frac{(Ns)^2}{N}\right) = L^2 - s^2 \simeq L^2$$

- This is far too large, one should use

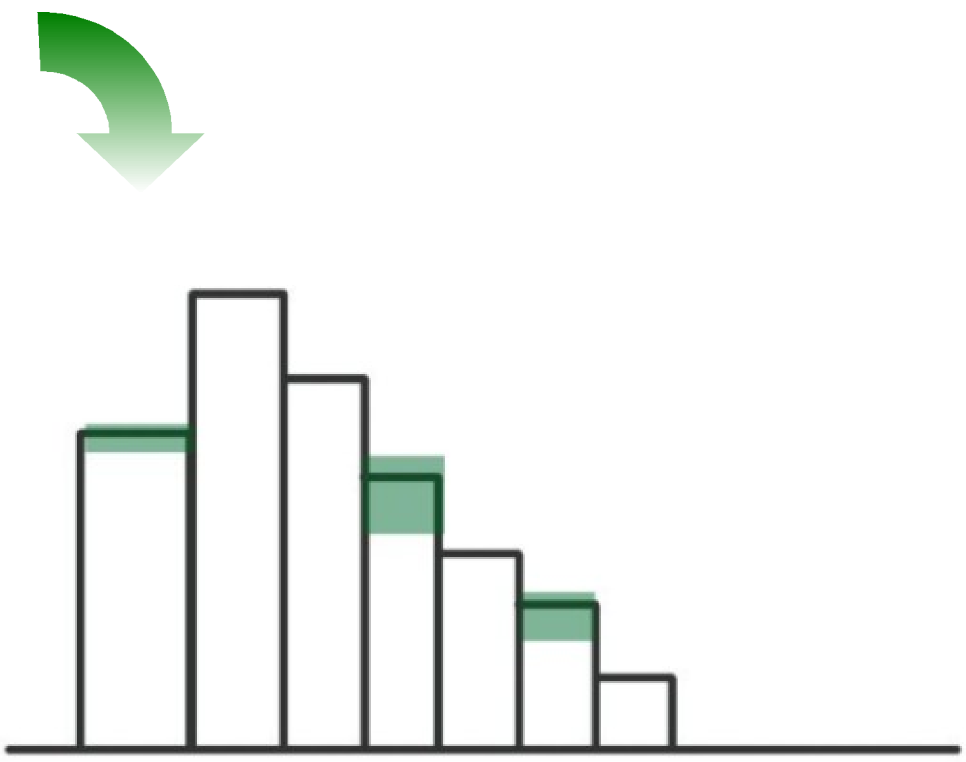
instead $$\sigma^2 \simeq \frac{1}{N}\left(\sum(\omega_{i,+} + \omega_{i,-})^2 - \frac{\left(\sum \omega_i\right)^2}{N}\right)$$

NLO events

- First bin all contribution from the event in auxiliary histogram



- Then transfer to the true histogram



fastNLO

- A direct analysis using n-Tuple files can still be too slow for some applications
  - PDF4LHC-type error require several PDF sets with errors
  - alphas determination
  - PDF fitting
- We can use the n-Tuple files to create fastNLO tables for very quick evaluation
  - Very fast
  - Need one table for each measured distribution
- Similar to the APPLgrid and other projects

fastNLO

- Replace PDF functions with

  where the the extrapolation kernels *E(x)* have the properties

$$f_a(x) \simeq \sum_i f_a(x_i) E(x)$$

- The cross section can be written as

$$\sum_i E_i(x) = 1 \ , \qquad E_i(x_j) = \begin{cases} 1 & \text{if} \quad i = j \\ 0 & \text{otherwise} \end{cases}$$

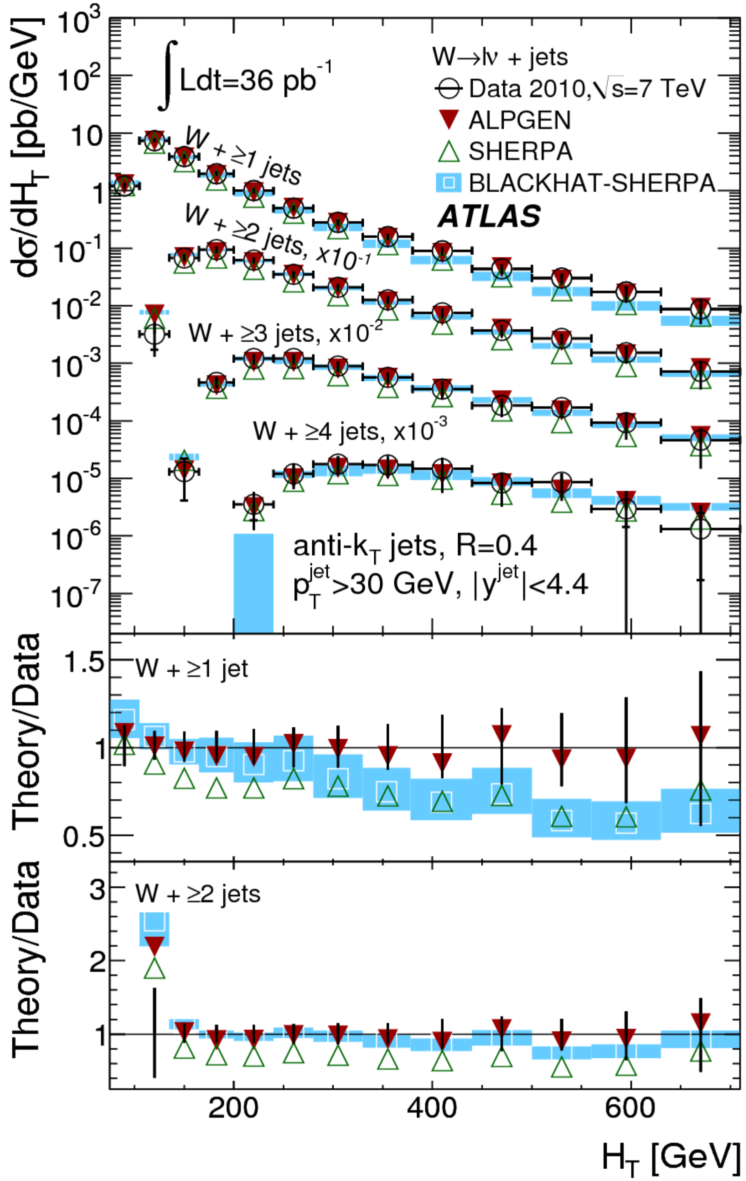
$$\sigma = \sum_{a,b,i,j} f_a(x_i) f_b(x_j) \underbrace{\int dx_1\, dx_2\, E_i(x_1) E_j(x_2) \int d\phi\, \hat{\sigma}(x_1, x_2; \phi)}_{c_{a,b;i,j}}$$

Can compute this with n-Tuples!

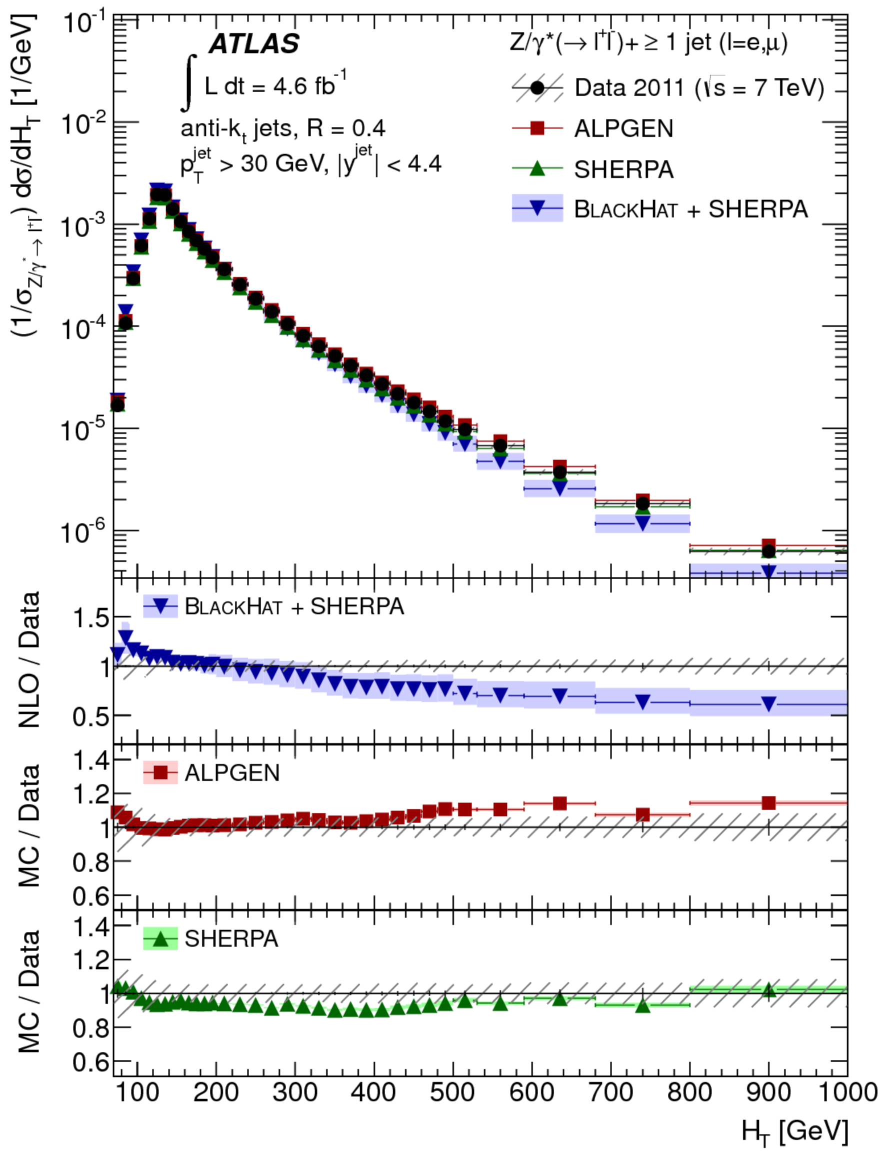
Speed vs Generality

HT distribution

- W+jet



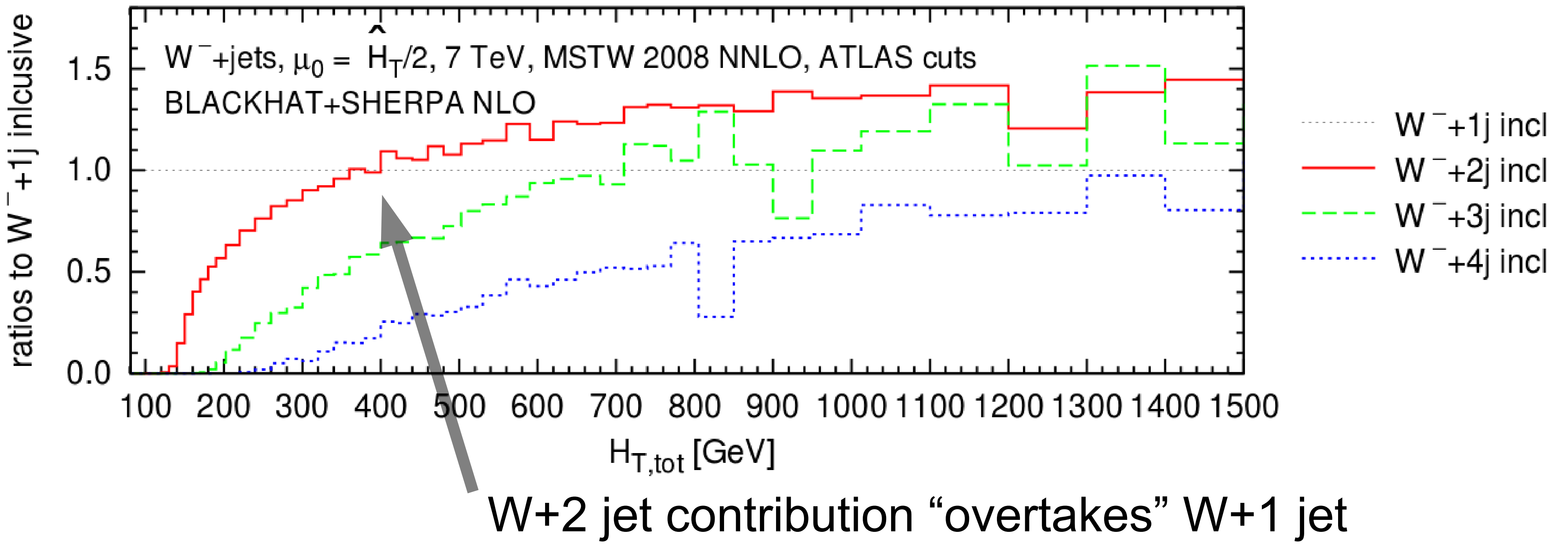
- Z+jets



*HT is basically the sum of all transverse energies of the particles in the event

HT distribution

- Already at quite moderate HT the higher multiplicity samples have sizeable contributions

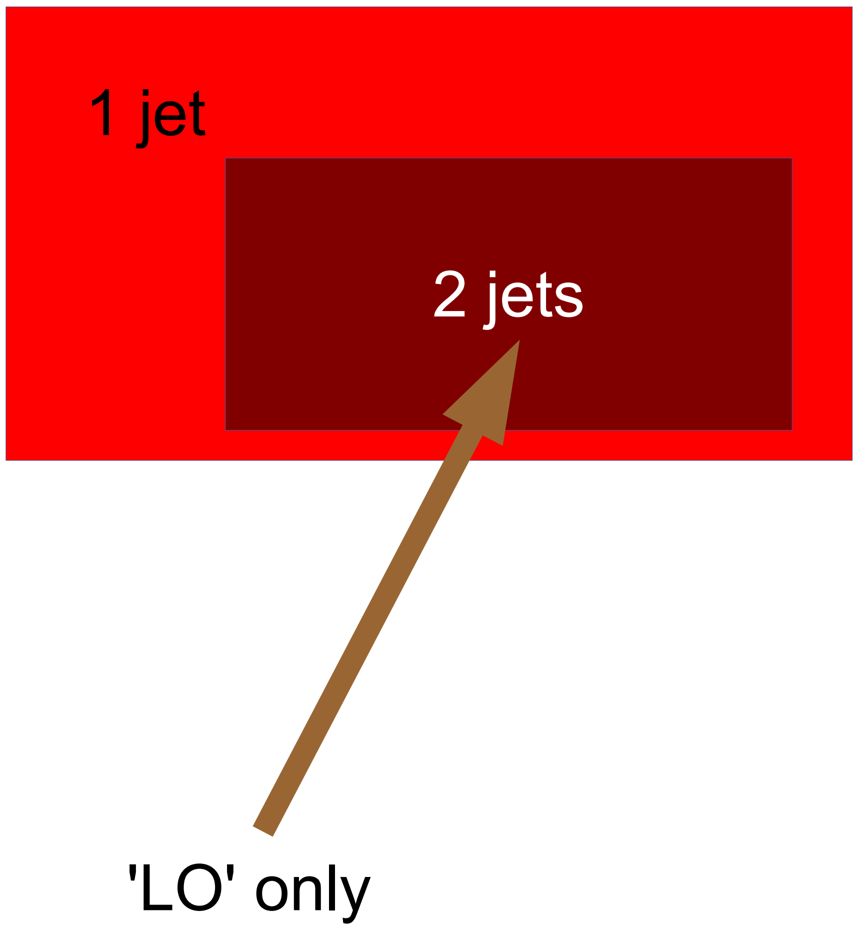


Exclusive sums

- Combine NLO event samples of different multiplicity
- Justified (if at all) for observables where higher multiplicities are important
- Avoid double counting by restricting the samples to a fixed multiplicity
- Formally not better than a NLO calculation
- No systematic study of uncertainties/stability
- Very crude method

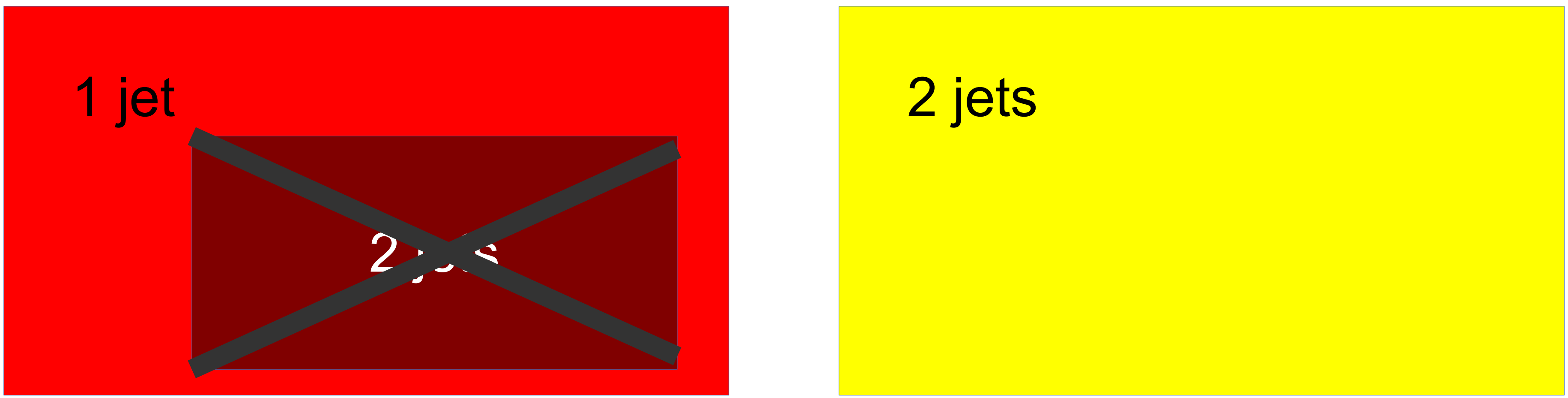
Exclusive sums

- W+1 jet at NLO

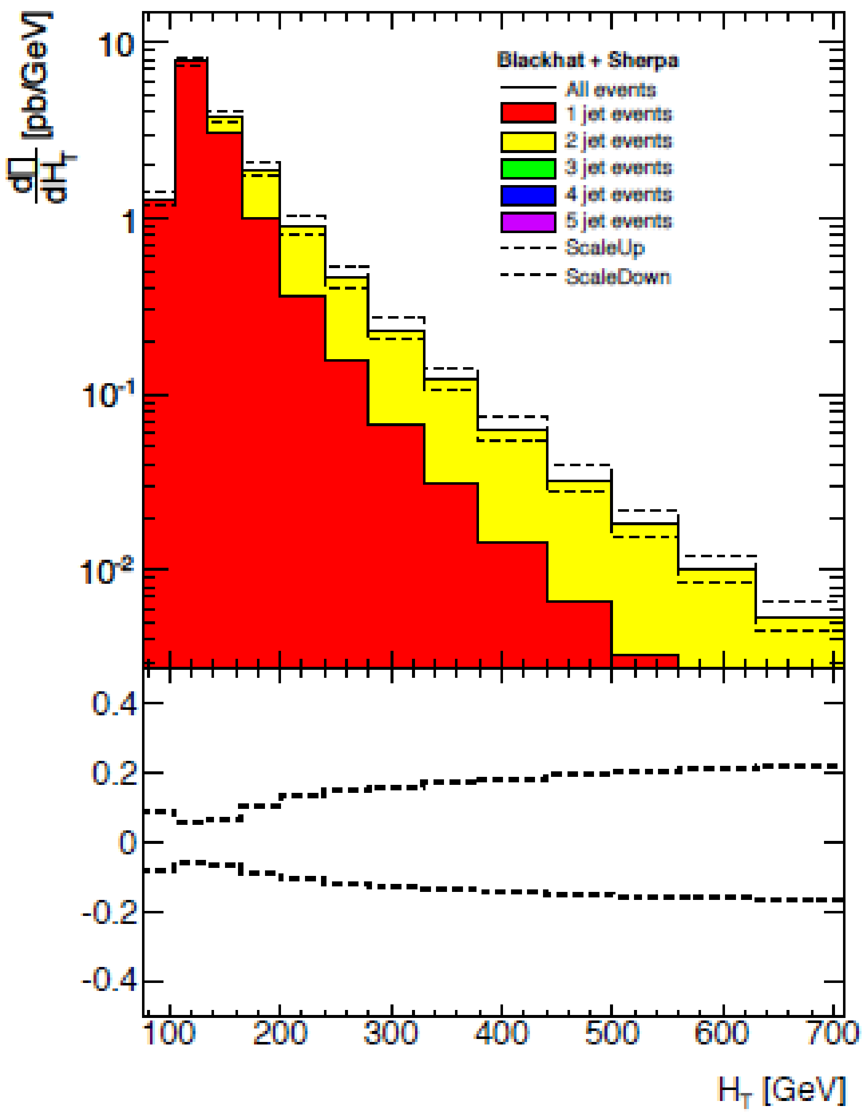


Exclusive sums

- W+1 jet at NLO W+2 jets at NLO



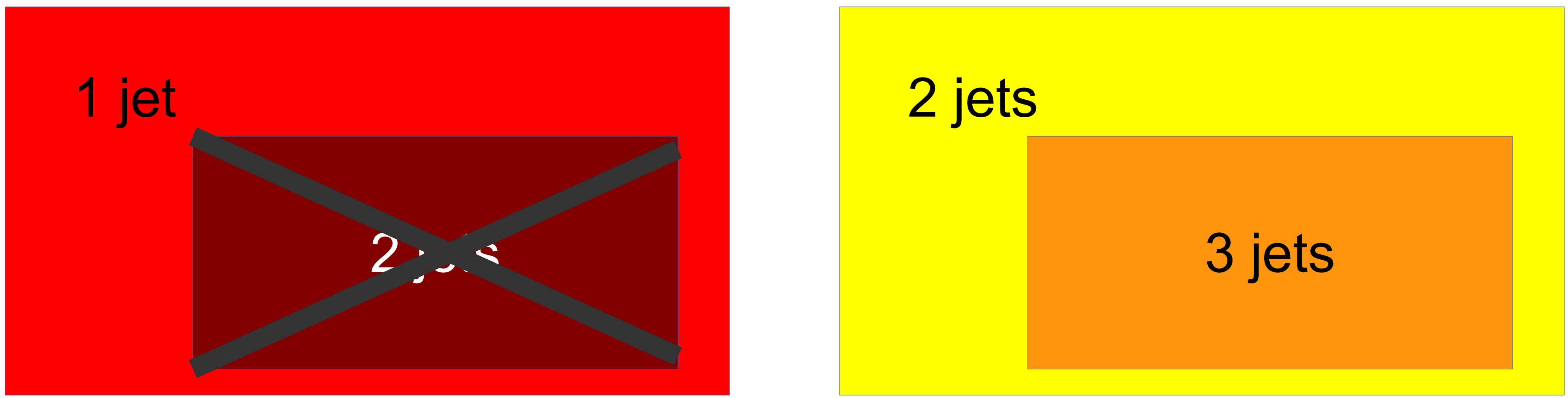
W+1 jet



Blackhat + Sherpa
All events
1 jet events
2 jet events
3 jet events
4 jet events
5 jet events
ScaleUp
ScaleDown
$\frac{d\Pi}{dH_T}$ [pb/GeV]
10
1
10^{-1}
10^{-2}
0.4
0.2
0
-0.2
-0.4
100
200
300
400
500
600
700
H_T [GeV]

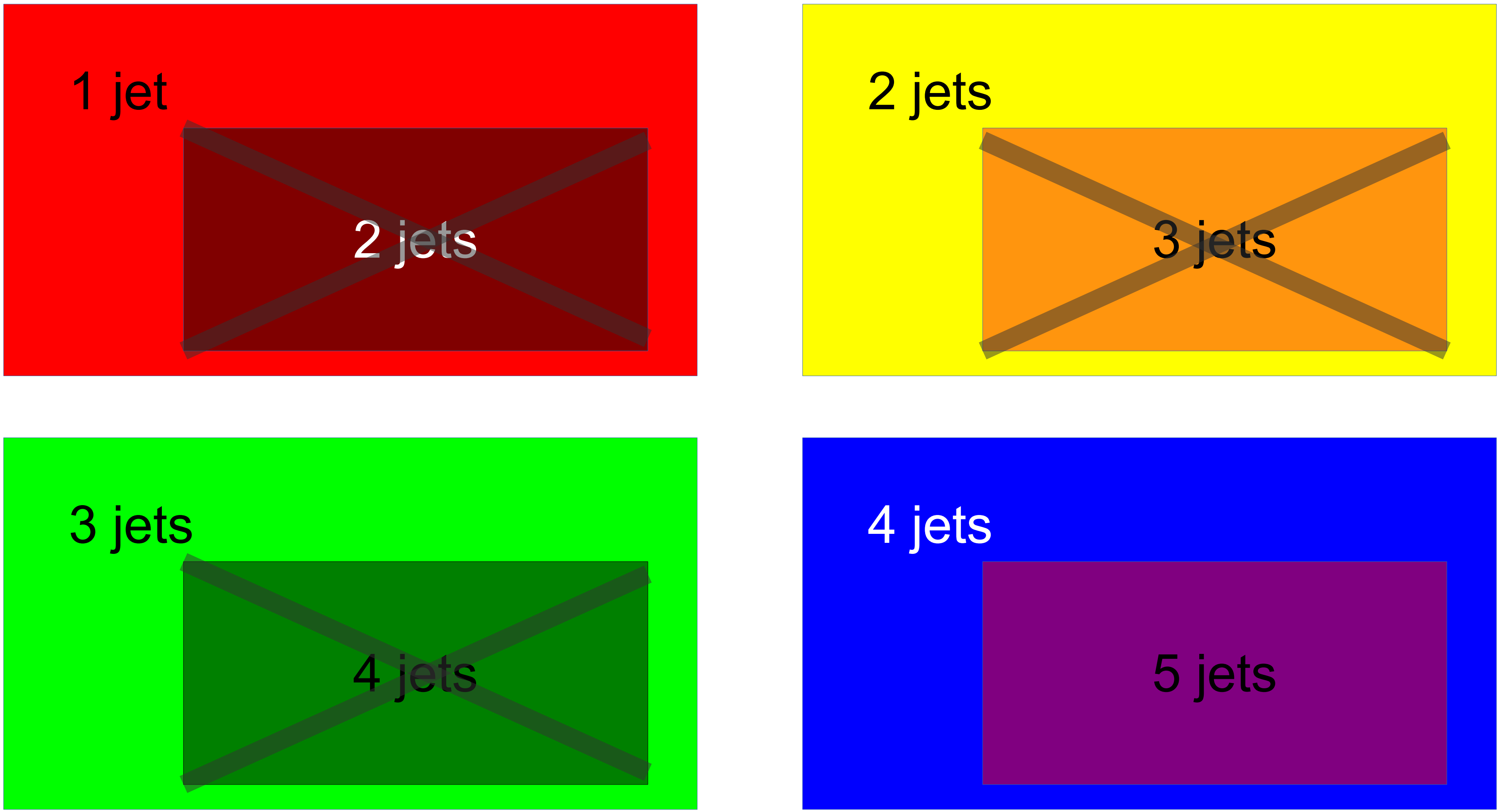
Exclusive sums

- W+1 jet at NLO W+2 jets at NLO



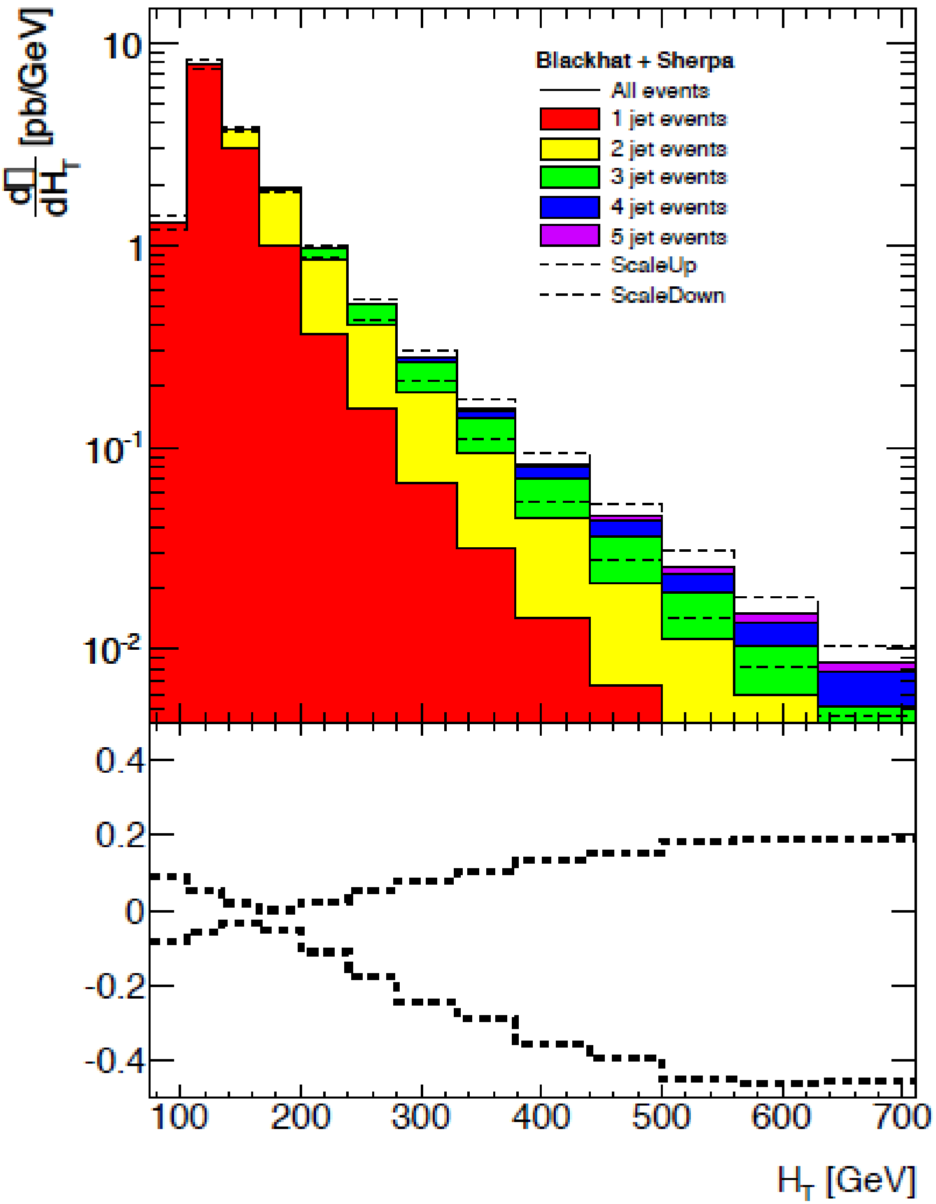
Exclusive sums

- W+1 jet at NLO W+2 jets at NLO



Exclusive sums

- Scale variation much larger than at NLO
- Need to be investigated more precisely
- Combination can be made 'official' using LoopSim [Rubin,Salam,Sapeta] (under investigation)
- Better : 'ME+PS'-type merging

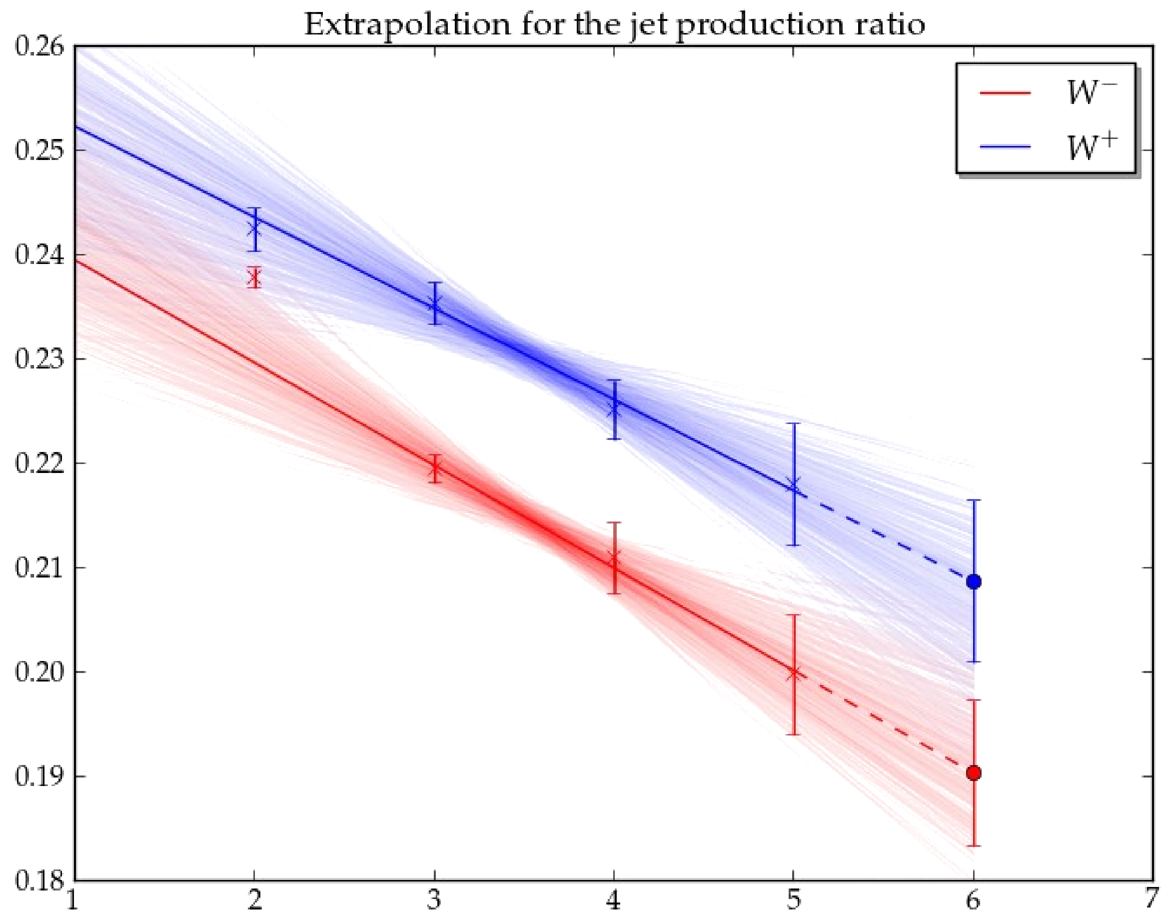


Towards higher multiplicities?

- We have a lot of prediction for high multiplicity processes at NLO
- We can try to find 'universal' properties/features
- Usually need to discard 0-jet and 1-jet because new partonic channels open
- Usually these features are more easily seen in ratios between multiplicities

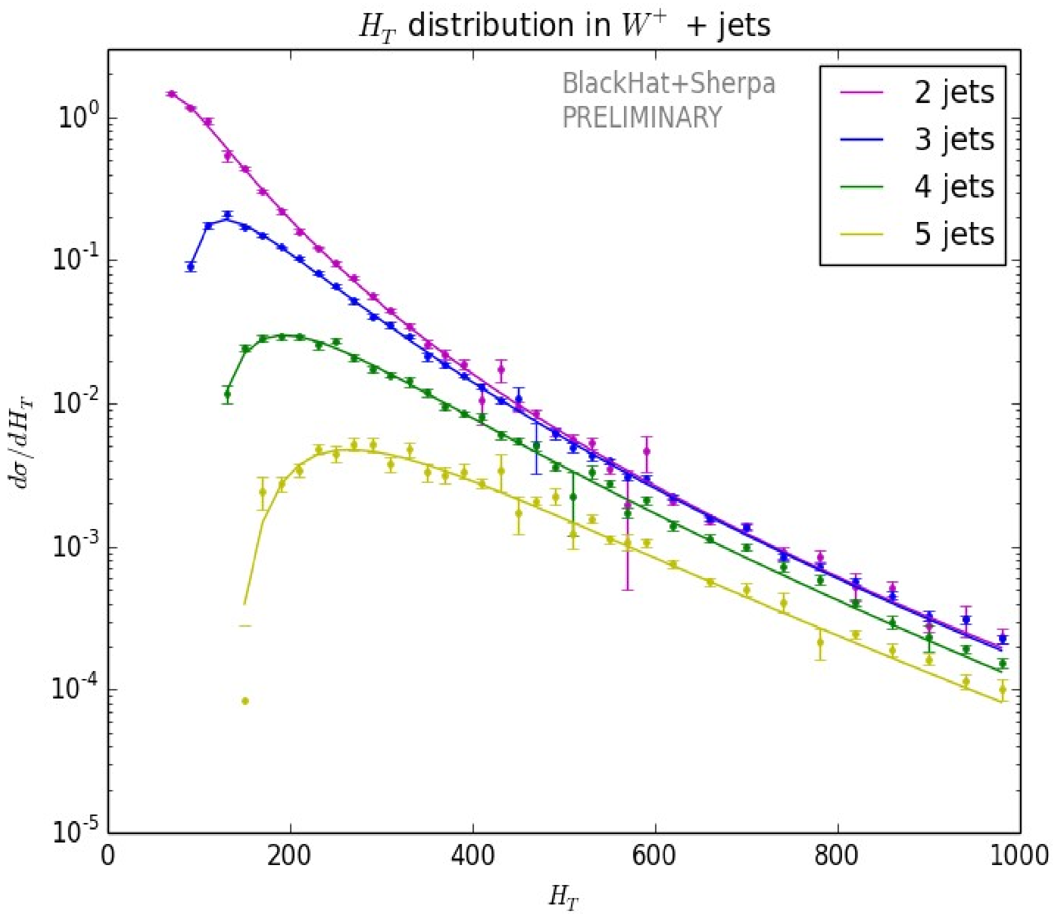
Extrapolation for ratios

- Ratio V+n jets/(V+n-1 jets)
- Consistent with straight line for n>2
- Use extrapolation for 6 jets:
- W^- : 0.15 ± 0.01 pb
- W^+ : 0.30 ± 0.03 pb
- Consistent with extrapolation of charge asymmetry
- Error estimates through Monte Carlo method



Distributions

- What about distributions?
- Look at sum of transverse energies of the jets (HT)
- Cannot extrapolate the value of each bin separately
  - Statistical errors are too large
  - Different thresholds
  - Different peak positions



Distributions

- Instead find a parametrisation and extrapolate the parameters of the parametrisation
- Ansatz for the HT distribution:

$$\frac{d\sigma_{V+n}}{d\,H_T} = \left(\frac{N_C \alpha_s}{2\pi}\right)^n f(H_T)\mathcal{N}_n \ln^{\tau_n} \rho_{H,n} \left(1 - H_T/H_T^{\max}\right)^{\gamma_n}$$

$$\rho_{H,n} = H_T/(np_T^{min})$$

HT Distribution

- Instead find a parametrisation and extrapolate the parameters of the parametrisation
- Ansatz for the HT distribution:

$$\frac{d\sigma_{V+n}}{d\,H_T} = \left(\frac{N_C \alpha_s}{2\pi}\right)^n f(H_T)\mathcal{N}_n \ln^{\tau_n} \rho_{H,n} \left(1 - H_T/H_T^{\max}\right)^{\gamma_n}$$

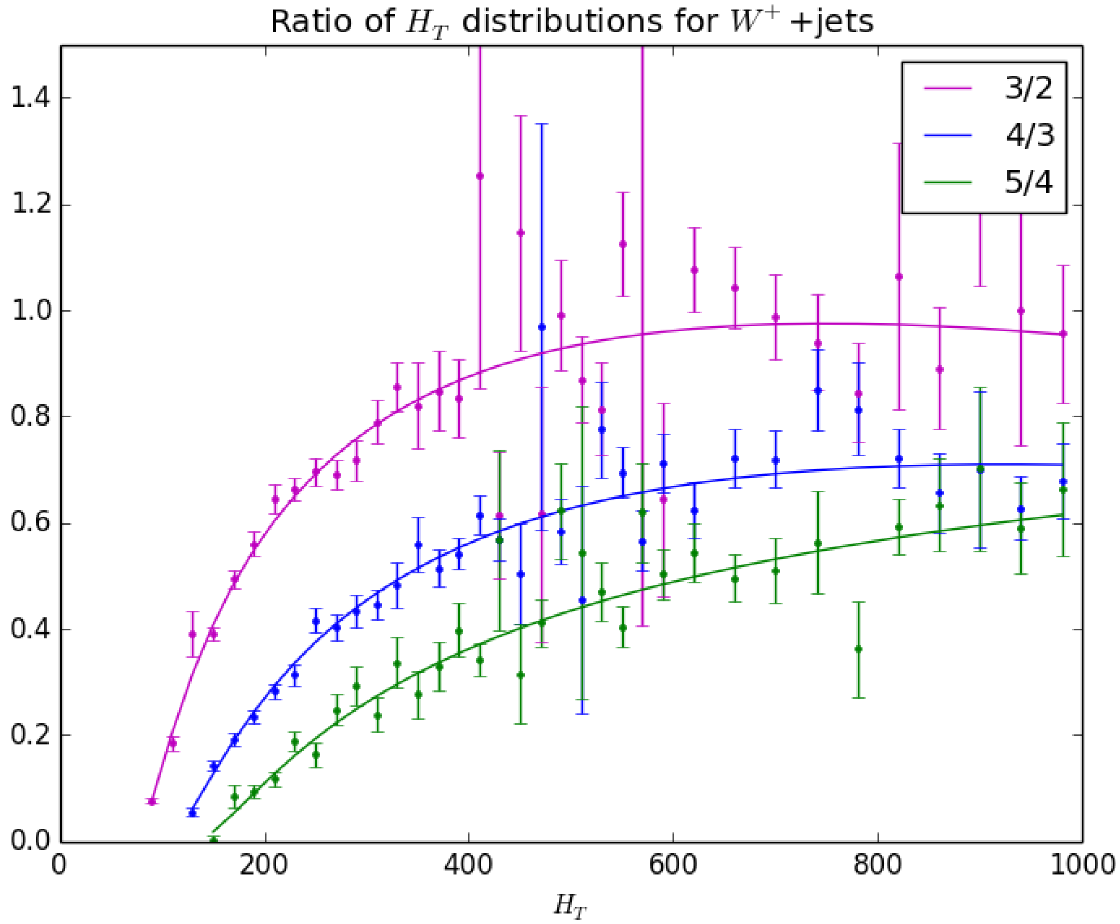
Independent of *n* ($f(H_T)$)

parameters ($\mathcal{N}_n$, τ_n, γ_n)

$$\rho_{H,n} = H_T/(np_T^{min})$$

HT distribution

- Fit ratios to get the parameters



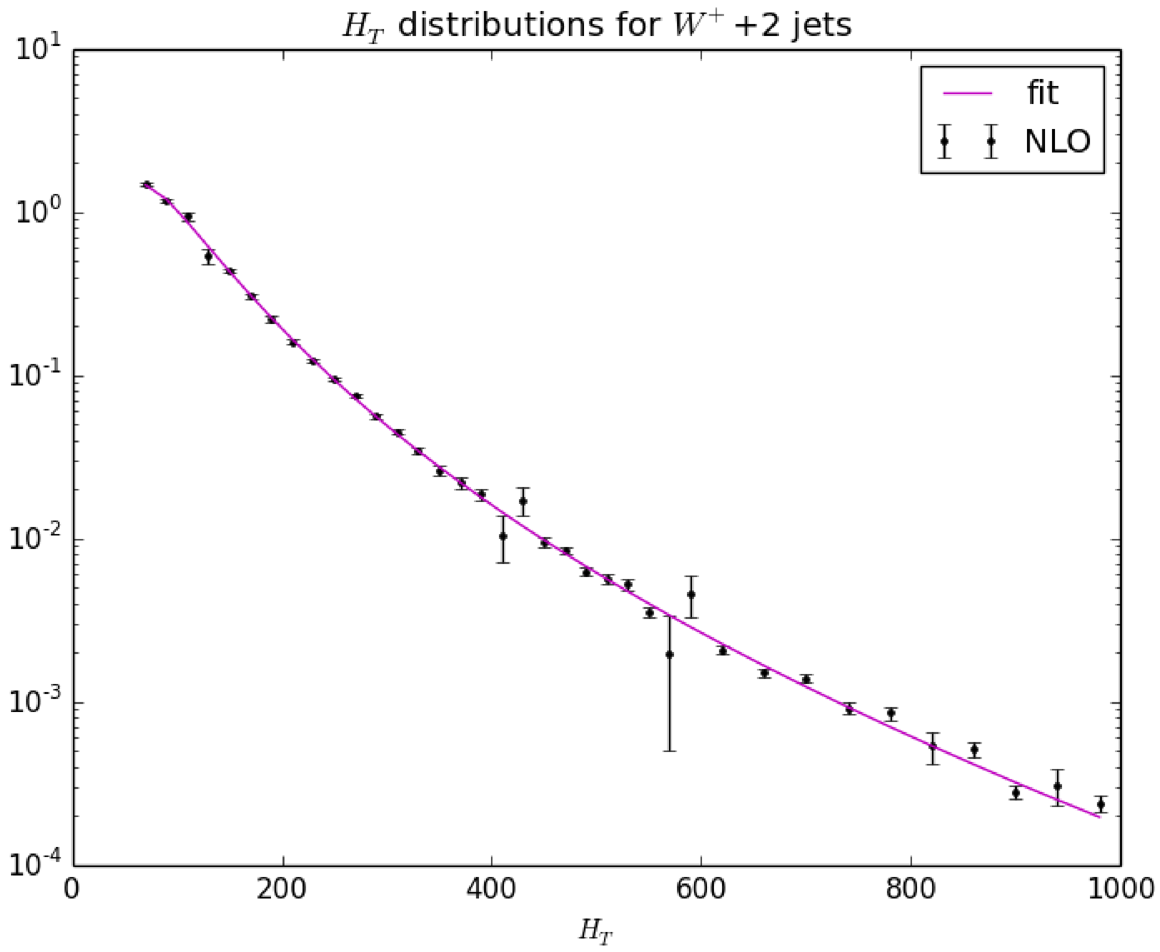
HT distribution

- With the parameters one can extract *f(H)* from a distribution
- But it is more convenient to have an analytical form for it
- We can use the following form

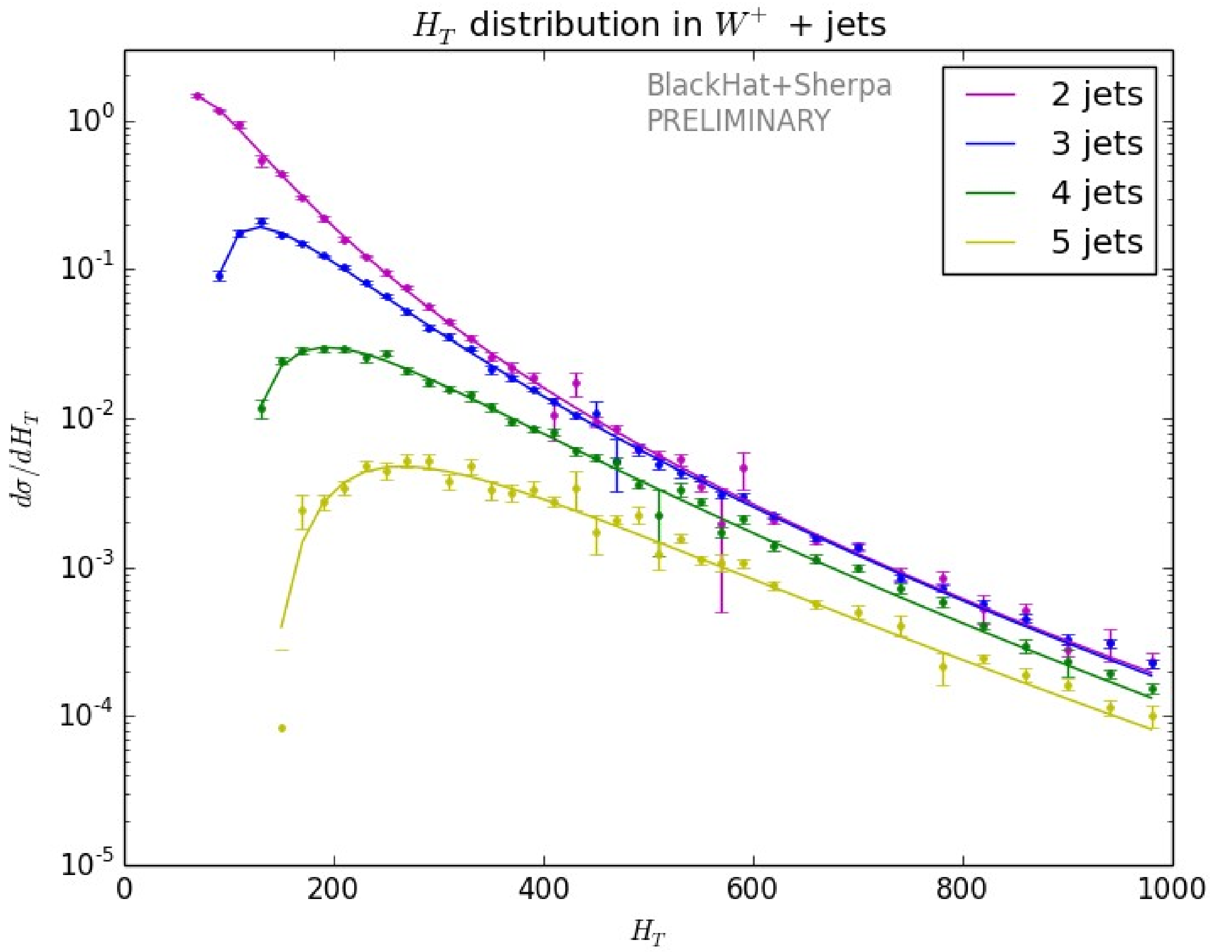
$$f(H) = c \ln^r(H/10) \left(\frac{H}{2p_T^{\min}} \right)^{\omega_2} e^{-h_* H},$$

and extract the parameters from the W+2 jets distribution

HT distribution



HT distribution



Distributions

- Extrapolated HT distribution
- Uncertainty bands are estimated using a MC method