Multijet merging techniques

Frank Krauss
(IPPP & CERN)

QCD at the LHC , Trento, September 2010

www.ippp.dur.ac.uk

Outline

- Motivation: Why higher orders?
- Multijet merging
  - Algorithm
  - Results with Herwig++ & truncated showering
  - Results with Sherpa
- Addendum: Powheg and beyond
  - Summary of the method
  - Automation in Sherpa
  - MENLOPS

Motivation: Why higher orders?

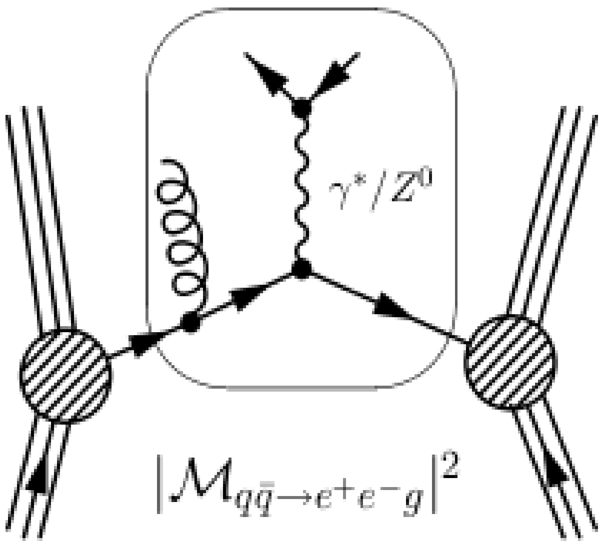
- In standard collinear factorisation, the total cross section is calculated as

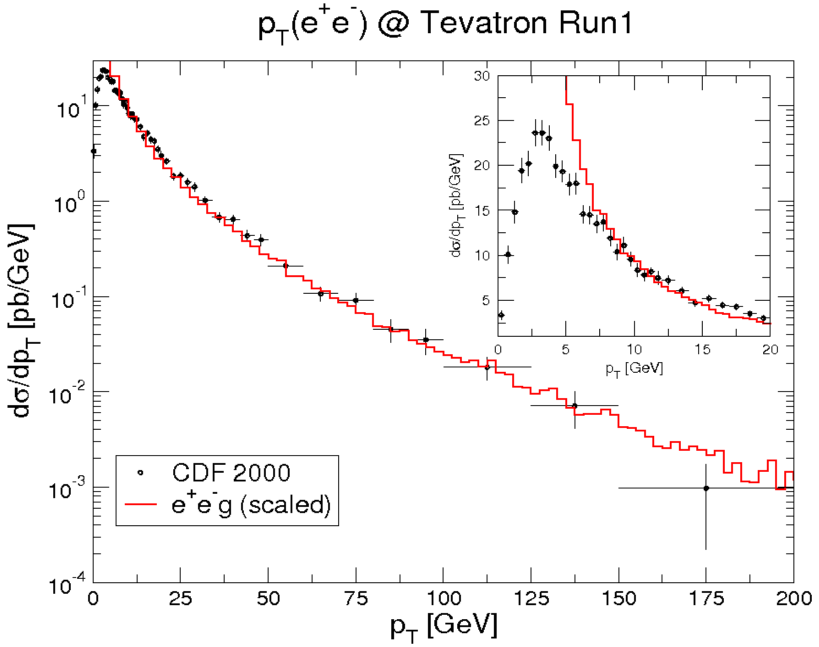
$$\sigma_{pp\to N}(Q^2) = \sum_{a,b} \int \mathrm{d}x_a \mathrm{d}x_b \; g_a(x_a, Q^2) g_b(x_b, Q^2) \, |\mathcal{M}_{ab\to N}|^2 \, \mathrm{d}\Phi_N$$

- Matrix element $|M_{ab\to N}|^2$ encodes fundamental physics
  (quantum interferences, off-shell effects etc.)
- It accounts correctly for high-p_T phenomena
- Possible: systematic improvements through higher orders in pert. expansion
- But: Poor accuracy in soft and collinear regions of phase space
  (Resummation of large logarithms important)
- Only few (<10) partons in final state
- Non-trivial link to hadronisation

Motivation (cont'd)

- Example: Transverse momentrum of Z boson in hadron-hadron collisions

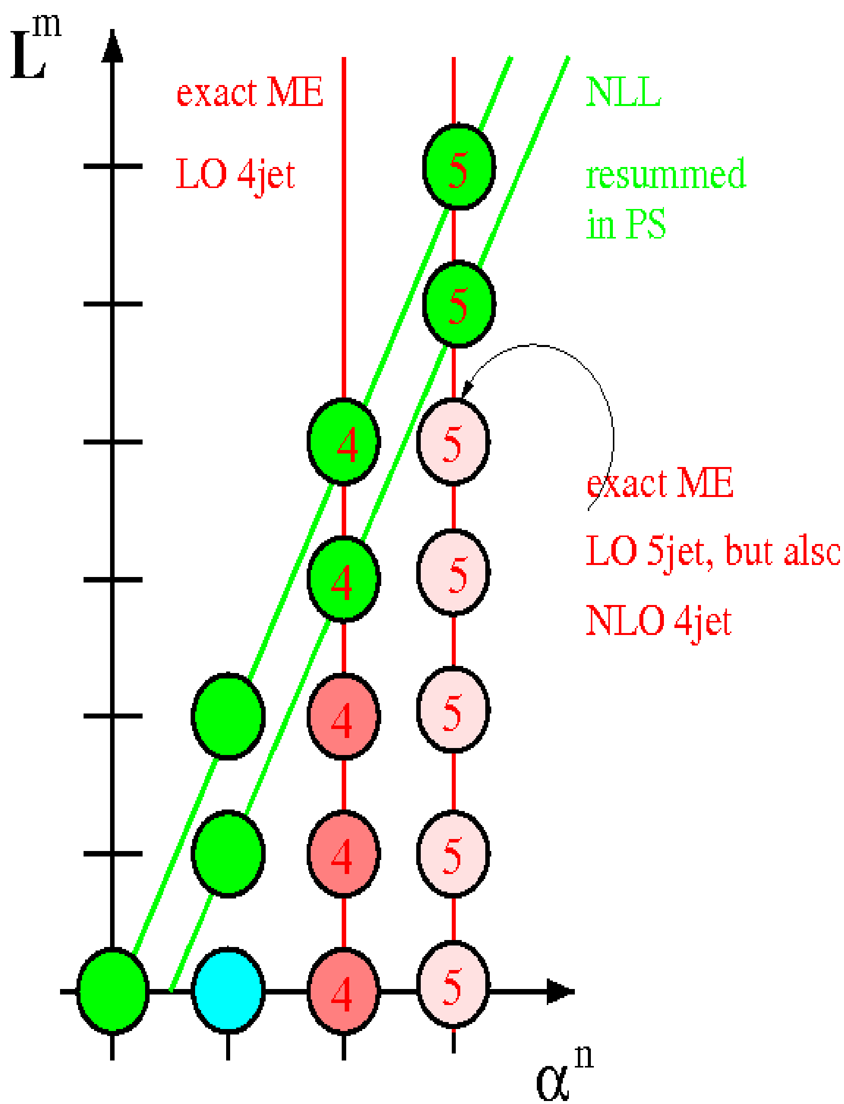




- $|\mathcal{M}_{q\bar{q}\to e^+e^-g}|^2 \sim |\mathcal{M}_{q\bar{q}\to e^+e^-}|^2 \frac{\alpha_S(\mu_R^2)}{p_T^2} \quad \rightsquigarrow \quad \sigma_{pp\to e^+e^-g} \sim \sigma_{pp\to e^+e^-}\,\alpha_S(\mu_R^2) \log\frac{p_T^{\max}}{p_T^{\min}}$
- Integration over p_T yields large logs, related to divergence in ME
- Divergence needs to be regularised and logs resummed
- Note: Universal IR structure → universal log structure → parton shower
  (parton shower as simple trick to achive the log-resummation)
- But: Parton shower approximation only valid in IR region & interplay with factorisation

Motivation (cont'd)

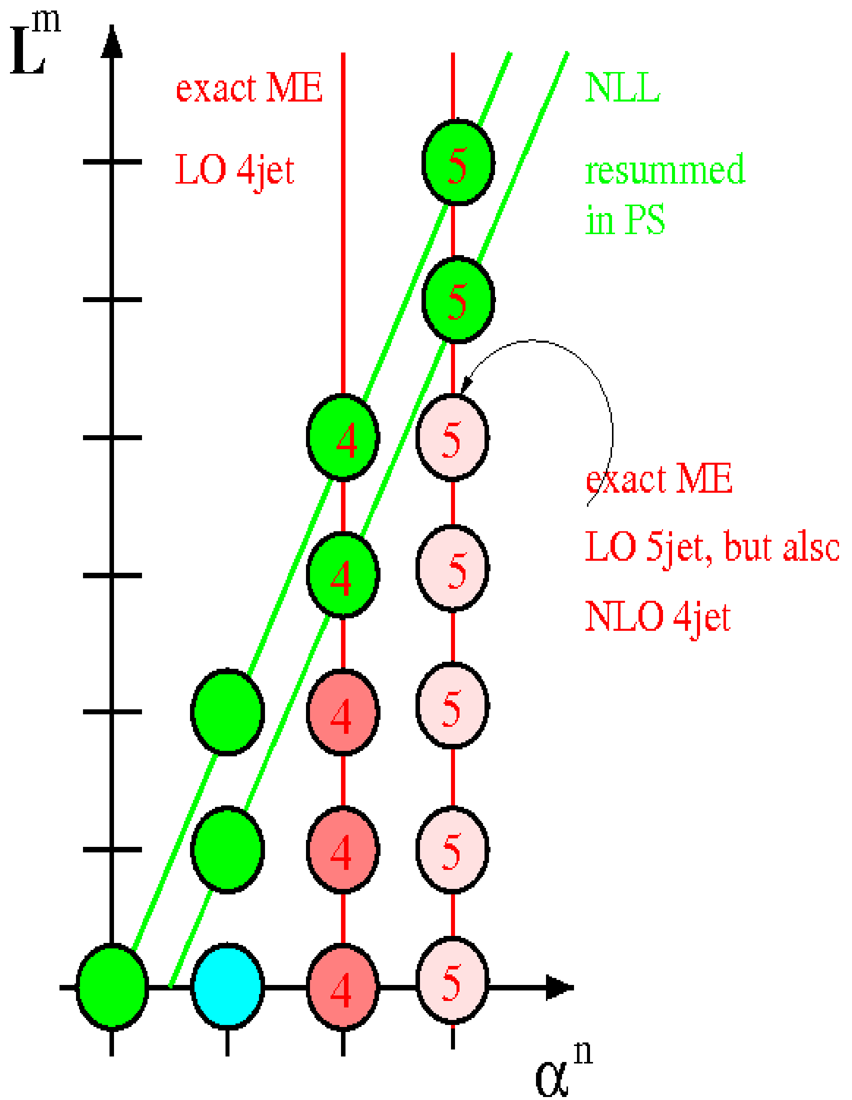
- Parton shower a probabilistic method of resumming the large logs
- It will not account for higher-order corrections to rates
- It will not provide a reliable estimate for hard scattering.
  (and power showers as in Pythia are not very systematic)
- Must combine parton shower with exact higher-order calculations.
- Two methods:
  - Multijet merging and
  - NLO matching



Multijet merging

Multijet merging: General thoughts

- Want to combine without double-counting
  - large logs of parton shower with
  - exact LO matrix elements
  - for towers of jet multiplicities
- Algorithm
  - To avoid the double-counting divide the phase space in two regimes: Jet-evolution and jet-production (this adds a parameter, Q_{cut})
  - Preserve accuracy of parton shower
    - Reweight the matrix elements
    - Veto/truncate the parton shower

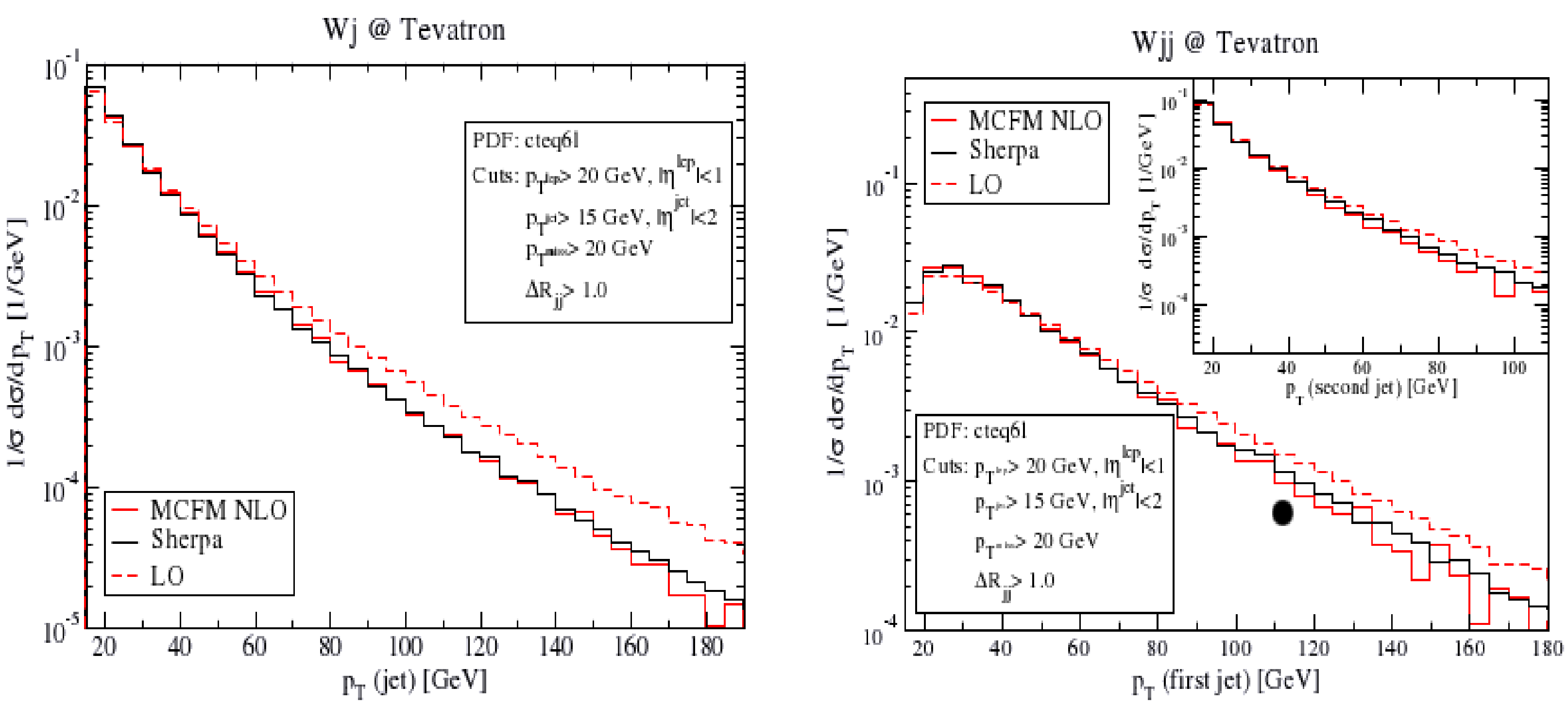


NB: Some ideas why it works

(and to answer one of Joey's questions ...)

- Compare Sherpa on ME level (after reweighting) with MCFM
- Normalised p_T distributions of jets in V+1,2 jets (exclusive)
- Very similar agreement in inclusive distributions

F.K., A.Schälicke, S.Schumann and G.Soff, Phys. Rev. D **70** (2004) 114009: comparison with MCFM; J.Campbell and R.K.Ellis, Phys. Rev. D **65** (2002) 113007



Sherpa = tree-level matrix elements with α_S scales and Sudakov form factors.

Decomposing the Sudakov form factor

- Remember Sudakov form factor, driving the parton shower

$$\Delta_a(\mu^2, t) = \exp\left[-\int\limits_{\mu^2}^{t} \frac{\mathrm{d}t'}{t'} \int\limits_{z}^{\xi_{\max}} \mathrm{d}\xi \frac{1}{2} \sum_{b=q,g} \mathcal{K}_{ba}(\xi, t')\right]$$

- Here $\mathcal{K}_{ba}$ is the splitting kernel and *t* the evolution parameter

  (in Herwig++, *t* is related to an angle, in Sherpa it is the transverse momentum)

- Decompose the splitting kernel (only formally) in an PS and an ME part:

$$K_{ba}(z, t) = K_{ba}(z, t)\Theta[Q_{\mathrm{cut}} - Q_{ba}(z, t)] + K_{ba}(z, t)\Theta[Q_{ba}(z, t) - Q_{\mathrm{cut}}].$$

- Sudakov form factor decomposes into PS and ME part

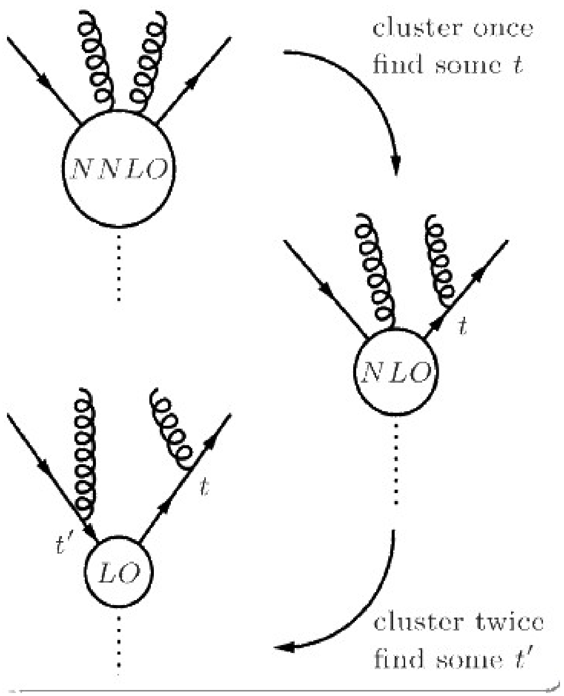
$$\Delta_a(\mu^2, t) = \Delta_a^{PS}(\mu^2, t)\Delta_a^{ME}(\mu^2, t)$$

Reconstructing the PS-history

- In each step, identify most likely splitting according to PS branching prob.
- Phase space measure for this from the parton shower (in Sherpa given by the Catani-Seymour shower)

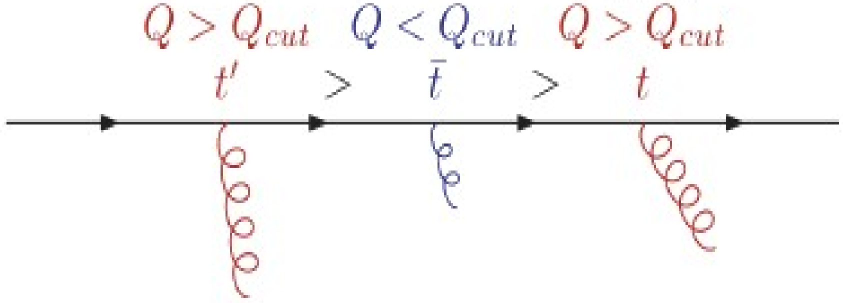
$$Q_{ij}^2 = 2\,p_i p_j \min_{k \neq i,j} \frac{2}{C_{i,j}^k + C_{j,i}^k} \; ; C_{i,j}^k = \begin{cases} \frac{p_i p_k}{(p_i + p_k) p_j} - \frac{m_i^2}{2\,p_i p_j} & \text{if } j = g \\ 1 & \text{else} \end{cases}$$

- For parton recombination use "inverted" PS splitting kinematics
- Continue until core 2→2 process reached
- Unfold the shower, starting at core
- Respect predetermined ME branchings (and insert them into shower while going), but can radiate in between (truncated showering)
- Veto event if shower produces jet in ME region, otherwise veto emission.



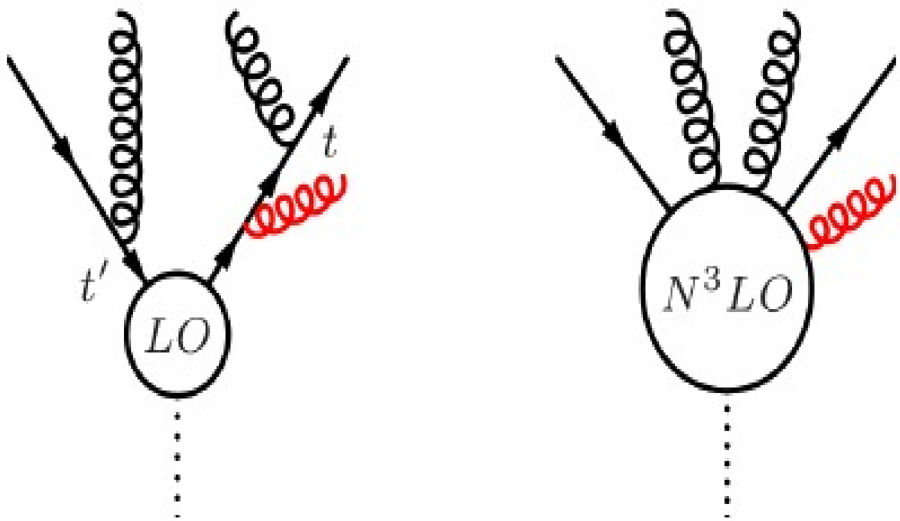
Truncated showering in a sketch

Shower emission below Q_{cut}:



- ⇝ emission accepted
- ⇝ large-angle soft emissions
- ⇝ soft color coherence
- ⇝ approx. in CKKW only

Shower emission above Q_{cut}:

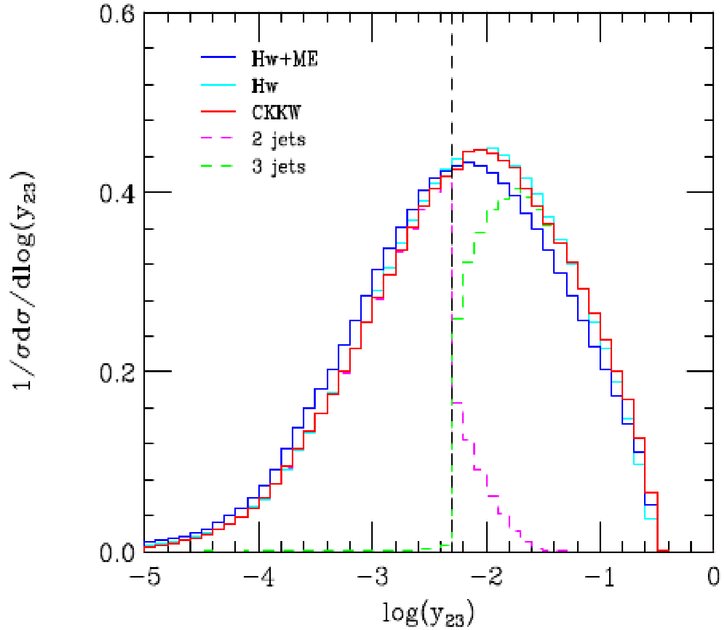


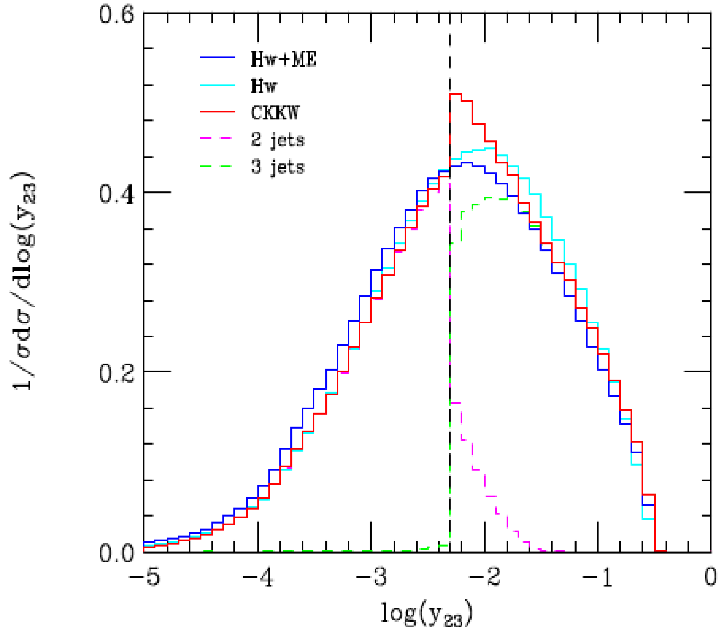
- ⇝ entire event is rejected
- ⇝ Sudakov suppression $\mathcal{P}^{\mathrm{ME}}_{\mathrm{no},a}(t,t')$
- ⇝ to be described by ME instead
- ⇝ σ_{tot} preserved at LO

Truncated showering – example results

(Hamilton, Richardson & Tully, JHEP 0911:038 (2009))

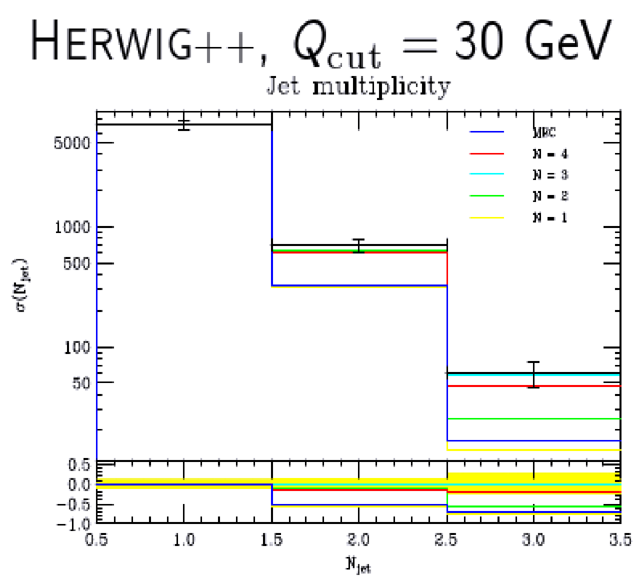
- Particularly important in case of Herwig parton shower:
  - Angular ordering variable allows for soft large-angle emissions to take place before harder, smaller angle emissions.
  - Example results in ee annihilations below
    (left with, right without truncated showering)
- For transverse-momentum ordered parton showers the effect of truncated showering is minor.

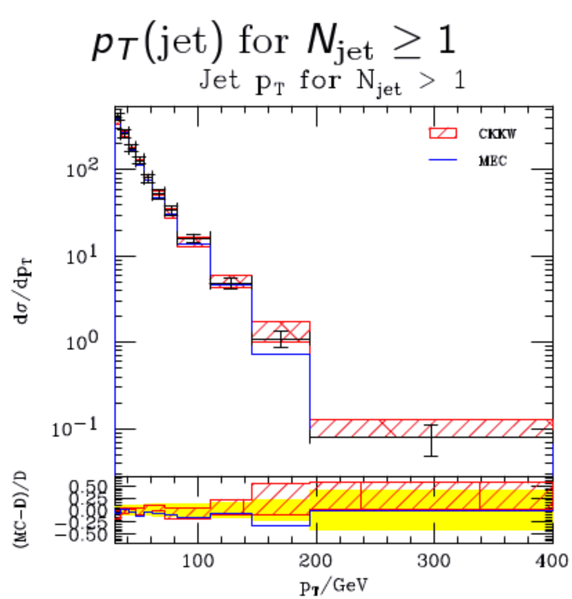


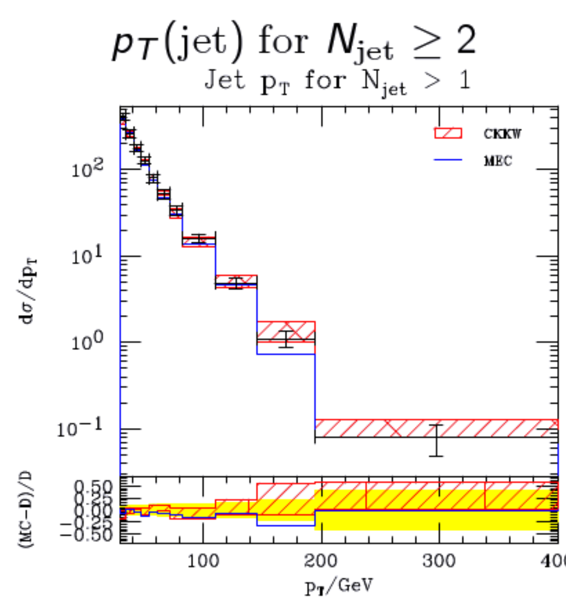


Multijet merging in Herwig++:

- ME+PS for hadronic collisions ongoing in Herwig++ (expect publication soon)
- Must truncate the shower due to mismatch of hardness scale (transverse momentum) and ordering parameter in parton shower (angles) – this resolves the problems apparent in the old Fortran Herwig-based studies by Mrenna and Richardson.
- Sneak-preview results below for DY at Tevatron (CDF)

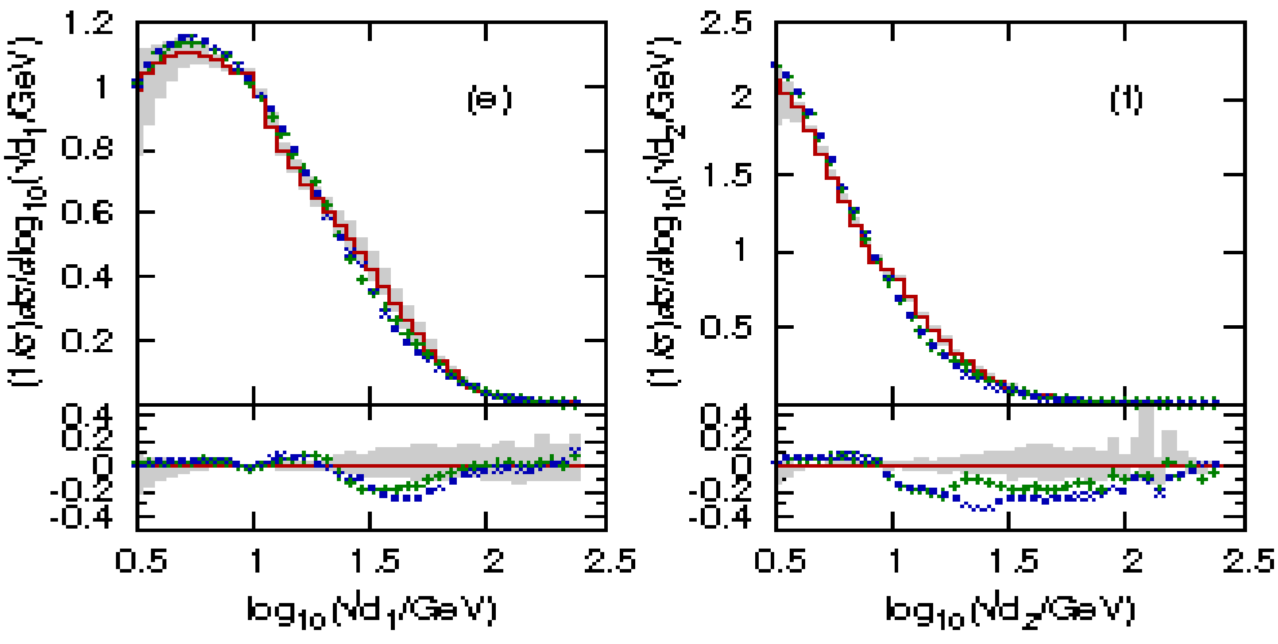


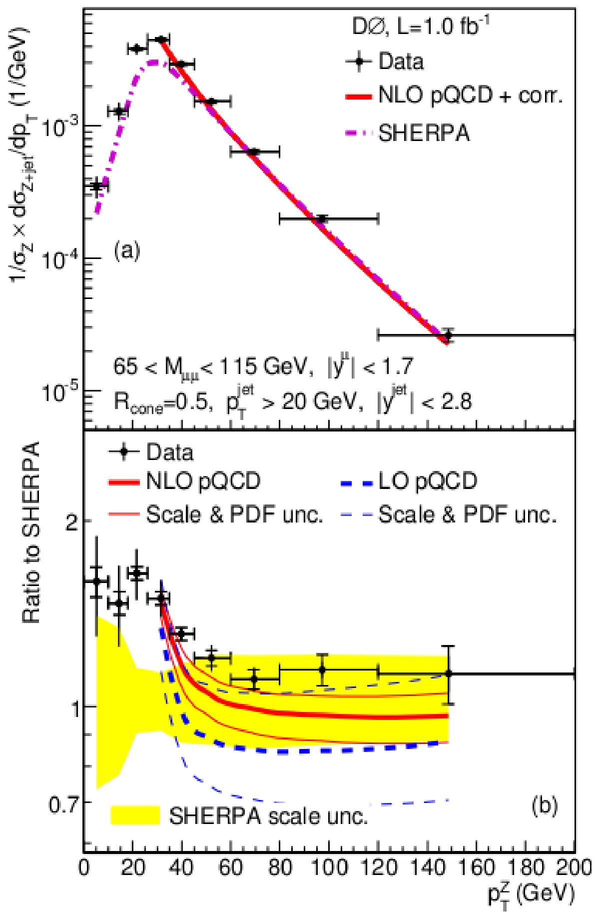




Multijet merging in Sherpa:

- Method has been pioneered for about 10 years in Sherpa
- New algorithm as described before from version 1.2
- Some example results for Sherpa 1.1.3 with old merging algorithm (analytical Sudakov form factors in ME regime)

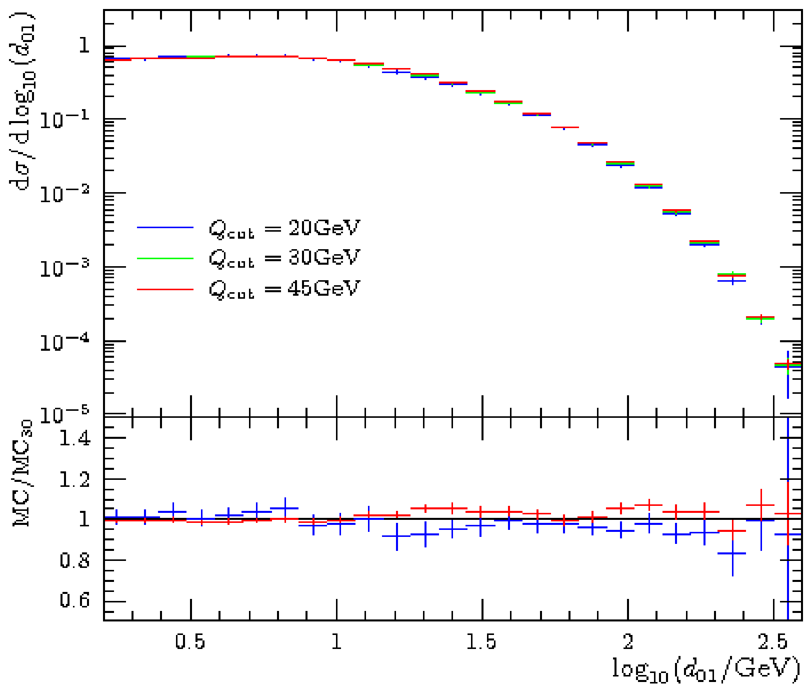


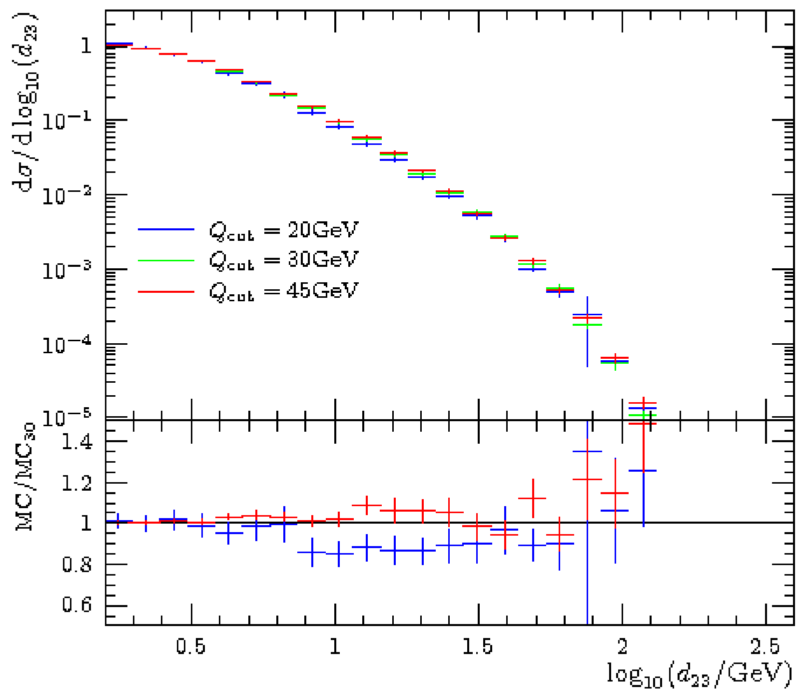


Multijet merging in Sherpa:

(Hoeche, Krauss, Schumann & Siegert, JHEP 0905:053 (2009))

- New improved formalism, greatly reduced merging systematics
  - Shpera 1.2: <15% for Q_{cut} [20, 50] GeV,
  - Old algorithm in Sherpa 1.1 was better than 20% (up to 40% in extreme bins – see previous slide)
- Shown to preserve formal accuracy of parton shower, i.e. shower logarithms are correctly accounted for; (MLM algorithm does not do that)

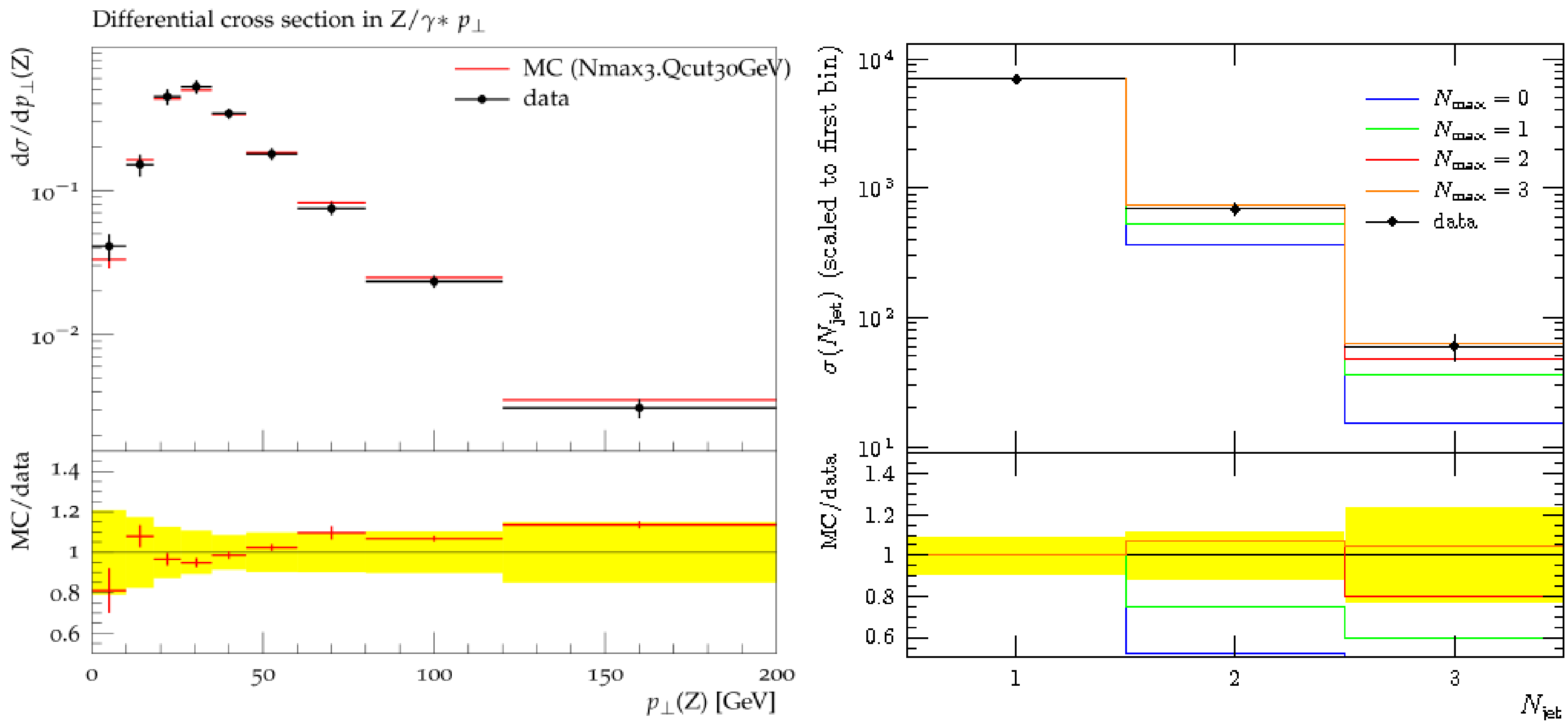




Multijet merging in Sherpa:

(Hoeche, Krauss, Schumann & Siegert, JHEP 0905:053 (2009))

- Typically, Sherpa 1.2 describes Tevatron data very well:

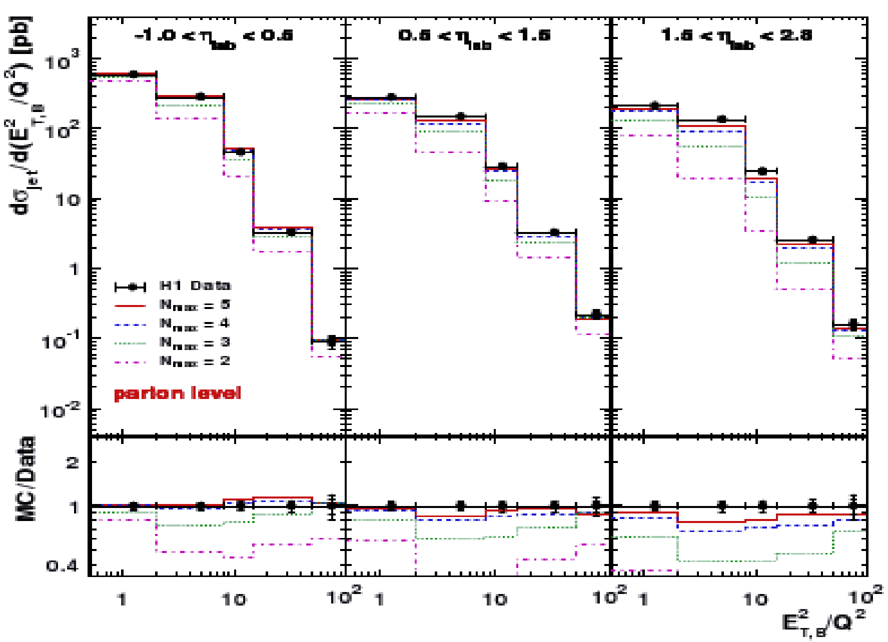


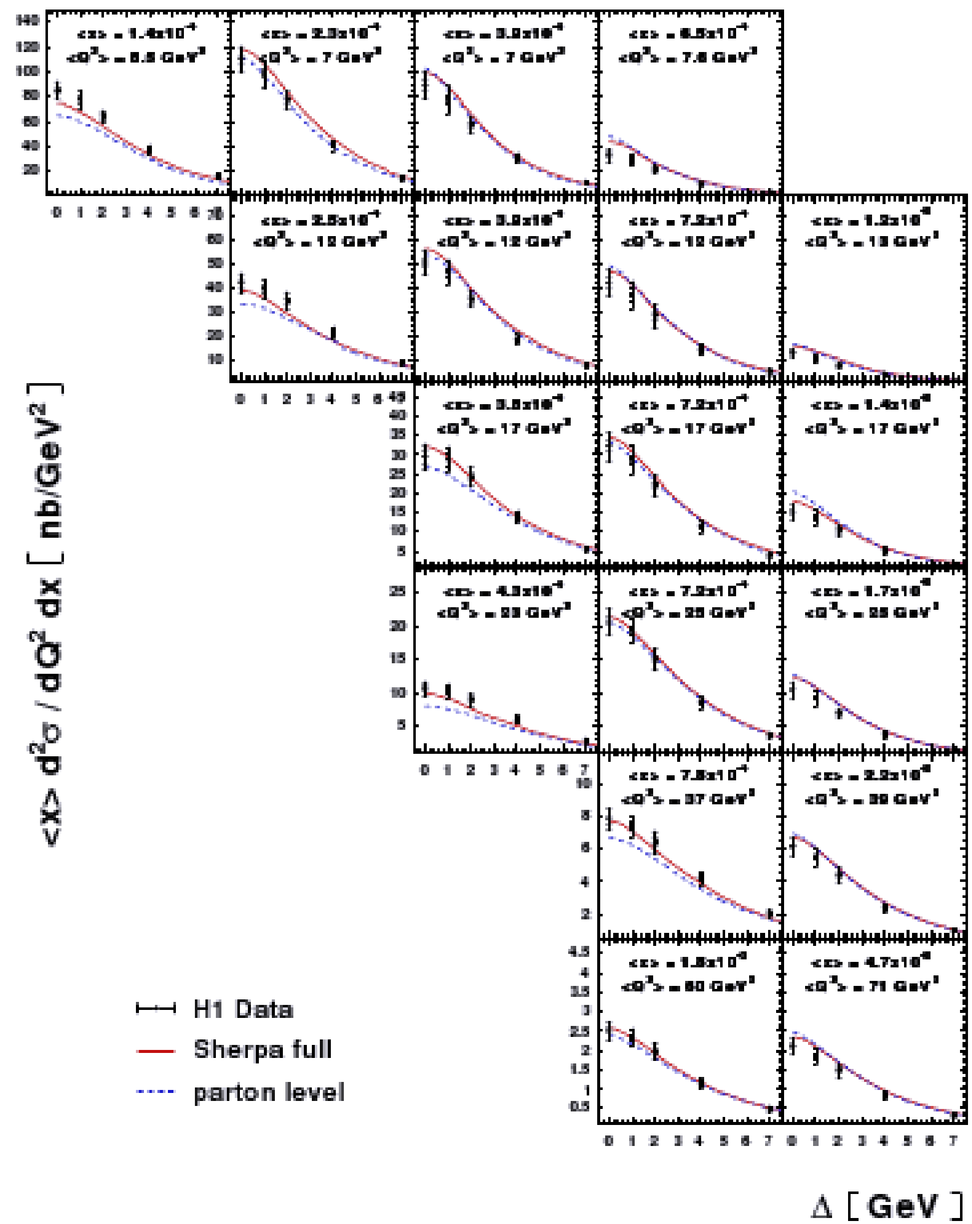
- Similar quality also for other observables and processes like W+jets.

Some results in DIS

(Carli, Gehrmann & Hoeche EPJC 67 (2010) 73)

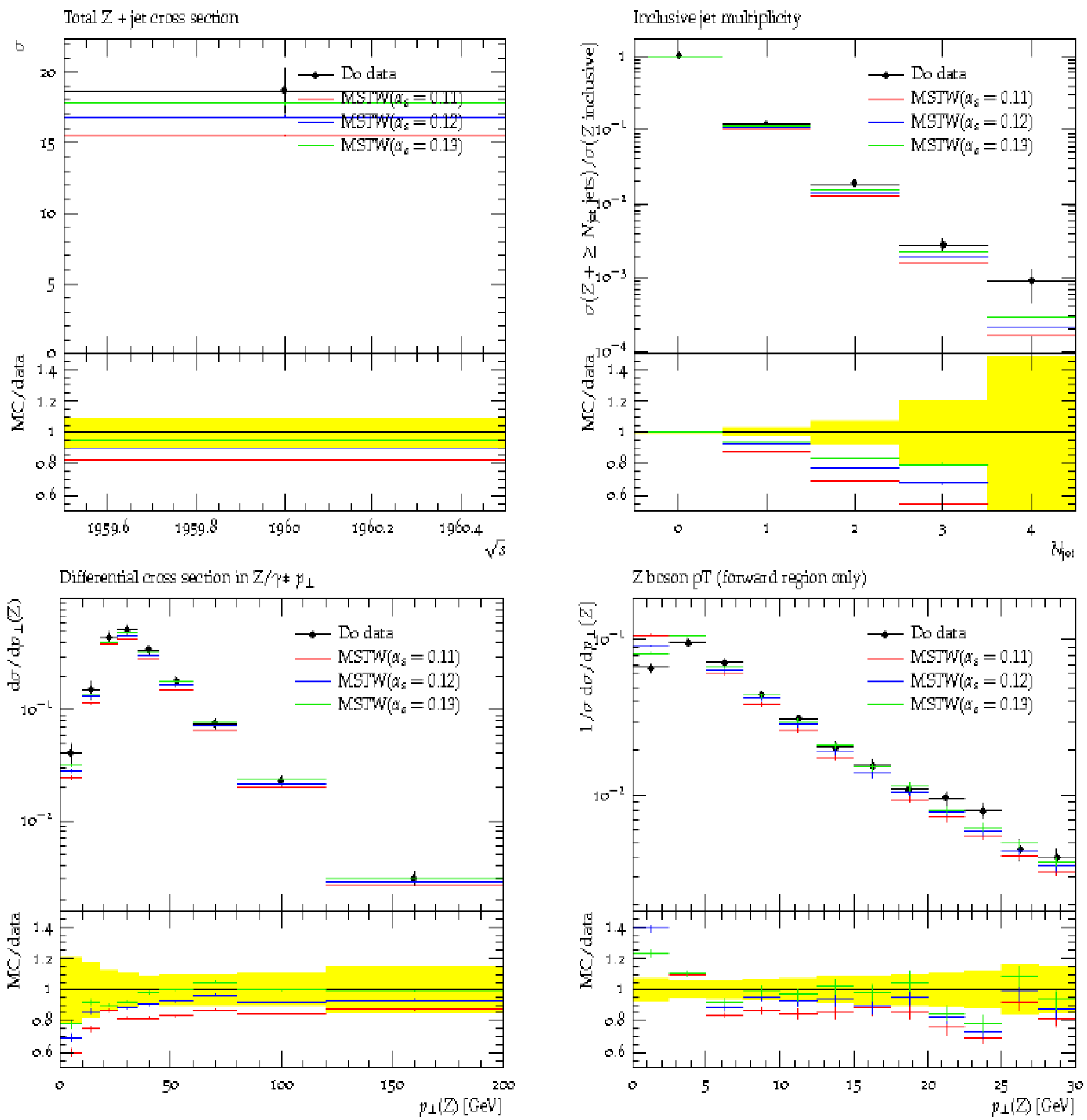
- Inclusive jet and dijet at low-x production, pt-distributions.
- Typically not well described by perturbative Monte Carlo





Merging in Sherpa – Impact of α_s

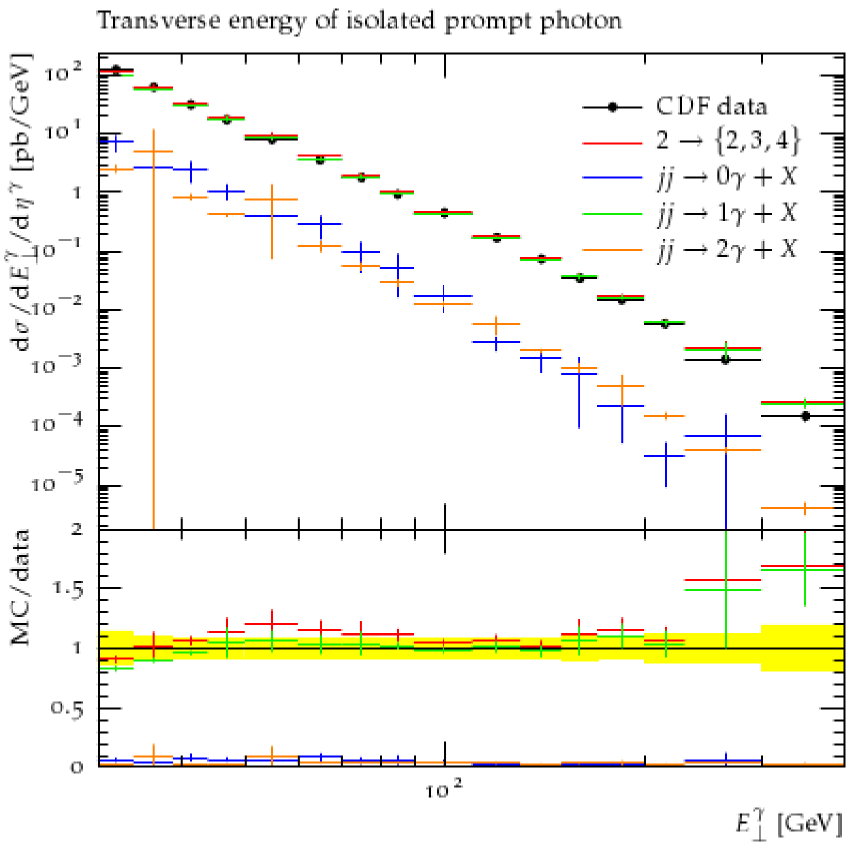
- In contrast to, e.g., Pythia, there is only one strong coupling constant in Sherpa, fixed by the PDF.
- I believe this is the only truly consistent choice, especially in view of treating higher order corrections in multijet merging and NLO matching.
- Therefore: Clear and immediate impact on observables.
- Therefore: Will provide tunes for different PDFs, including those with different values of strong coupling; this will allow for meaningful systematic error estimates

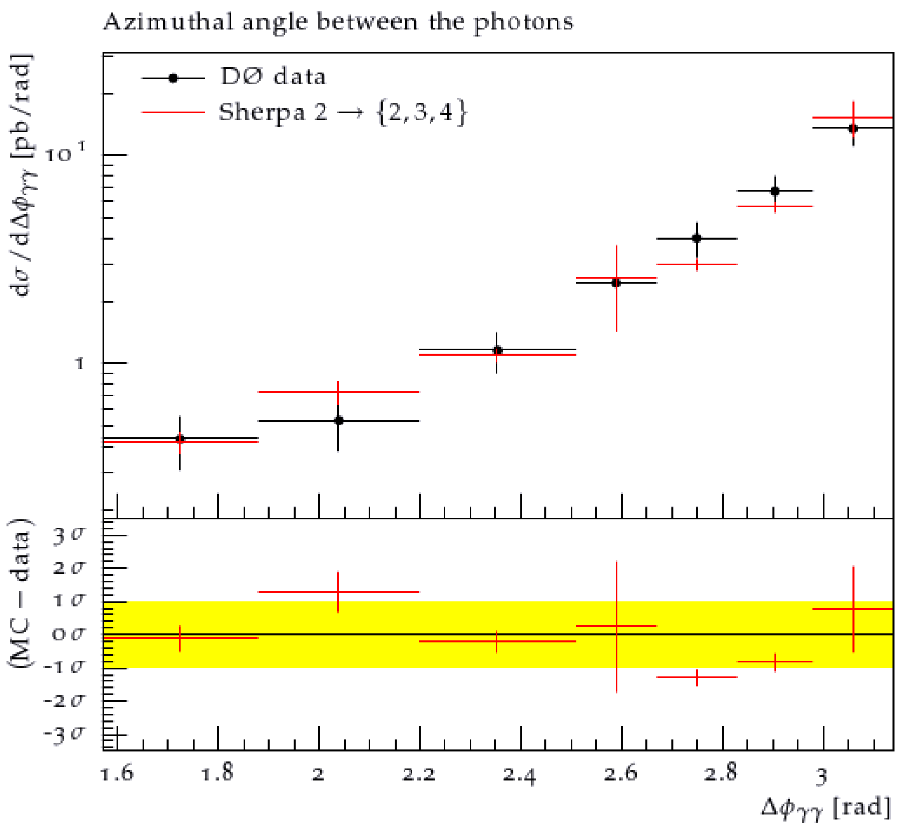


Multijet merging with photons:

(Hoeche, Schumann & Siegert, PRD 81 (2009), 034026)

- Recently extended method also to photons.
- Example results:

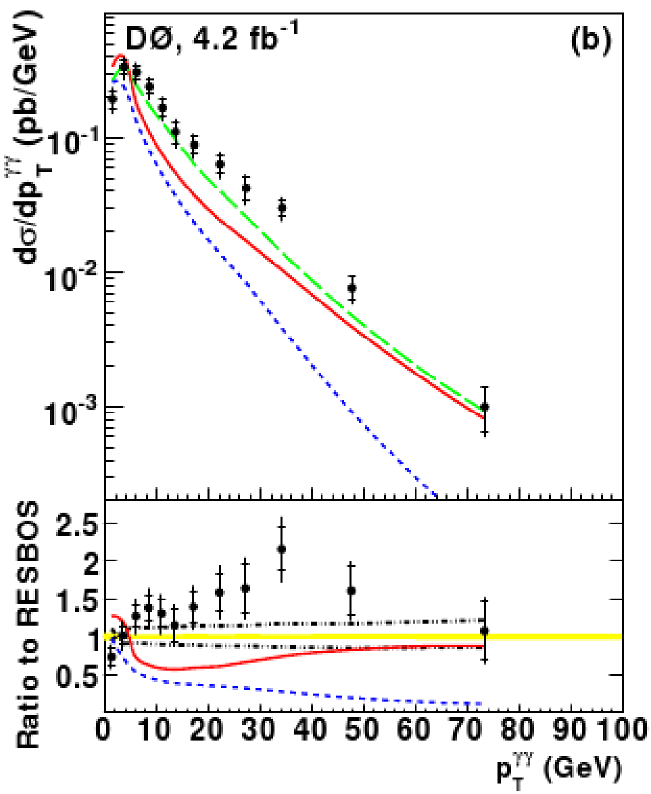


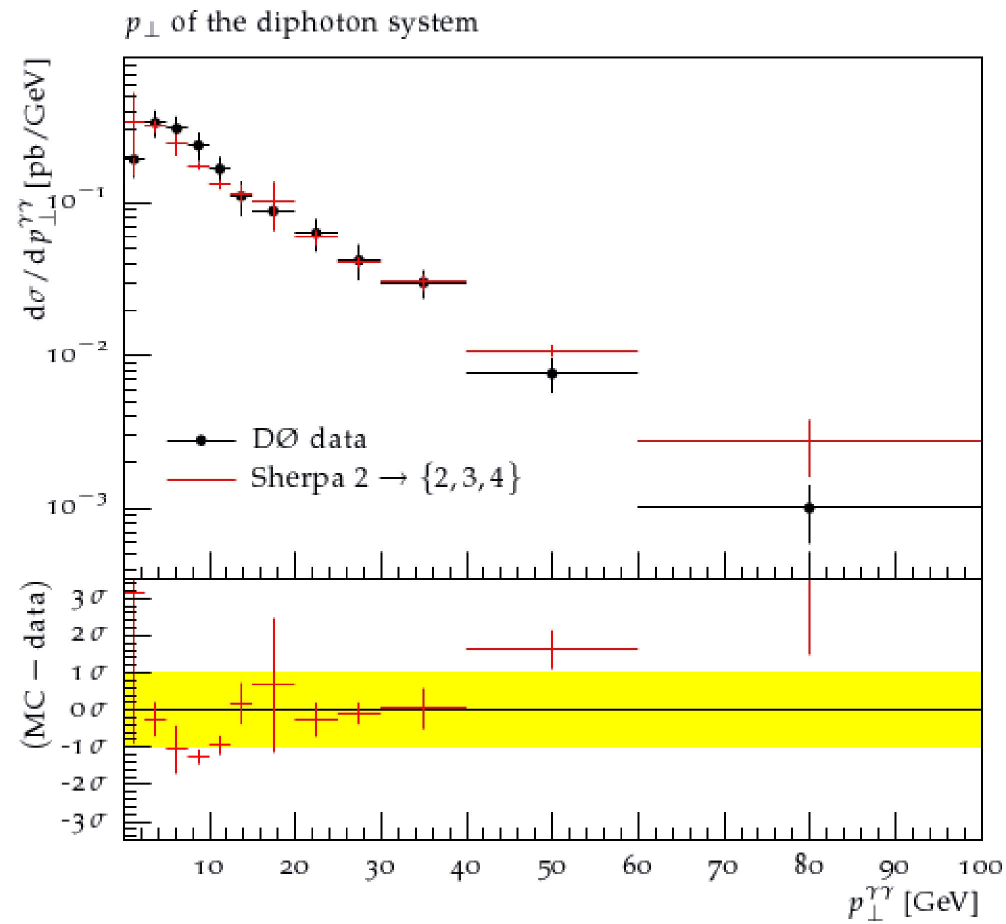


Multijet merging with photons:

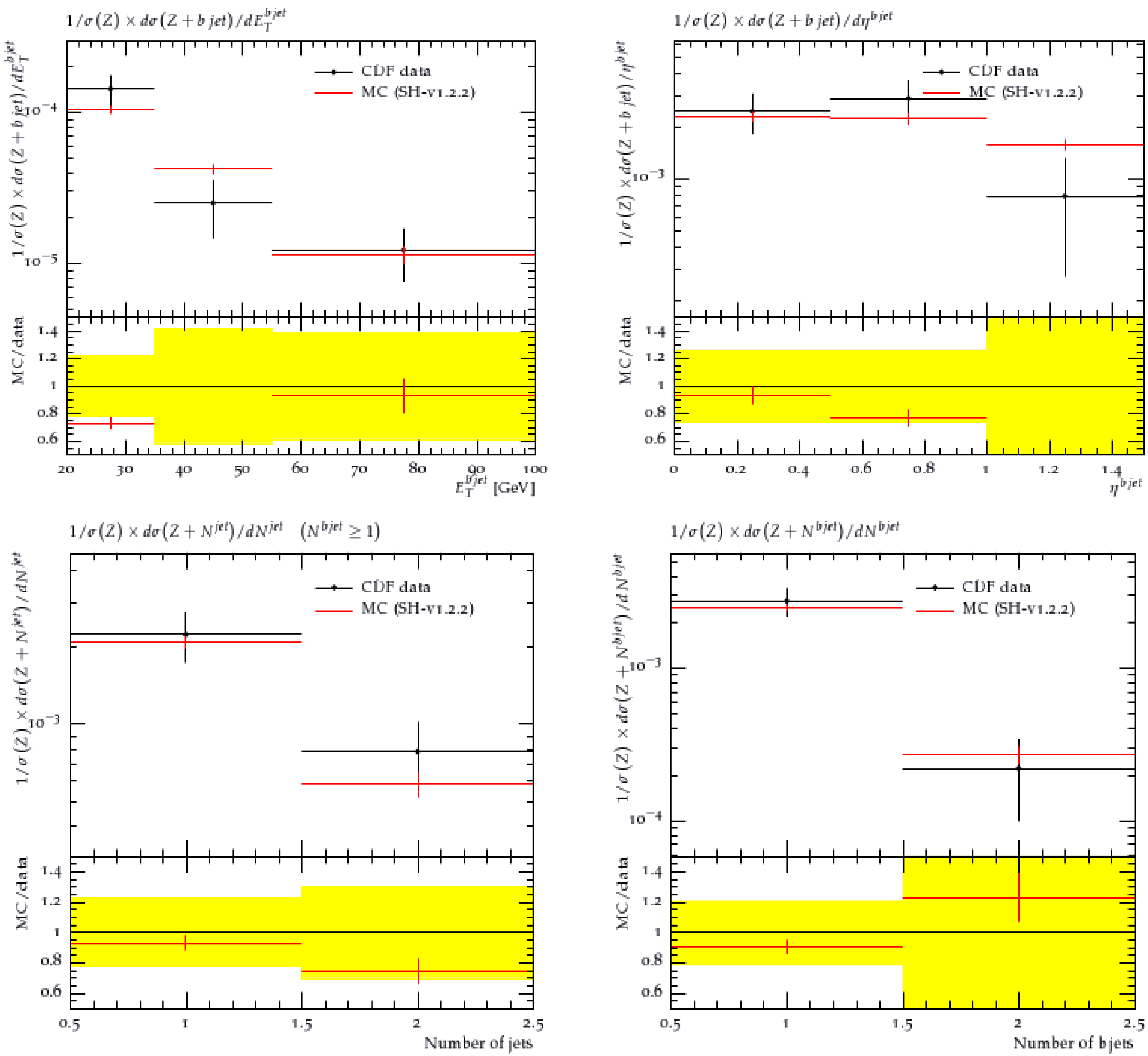
(Hoeche, Schumann & Siegert, PRD 81 (2009), 034026)

- Compare with other codes: DiPhox, ResBos, Pythia
- Sherpa: merged $2 \to \{2,3,4\}$ plus $gg \to \gamma\gamma$ box





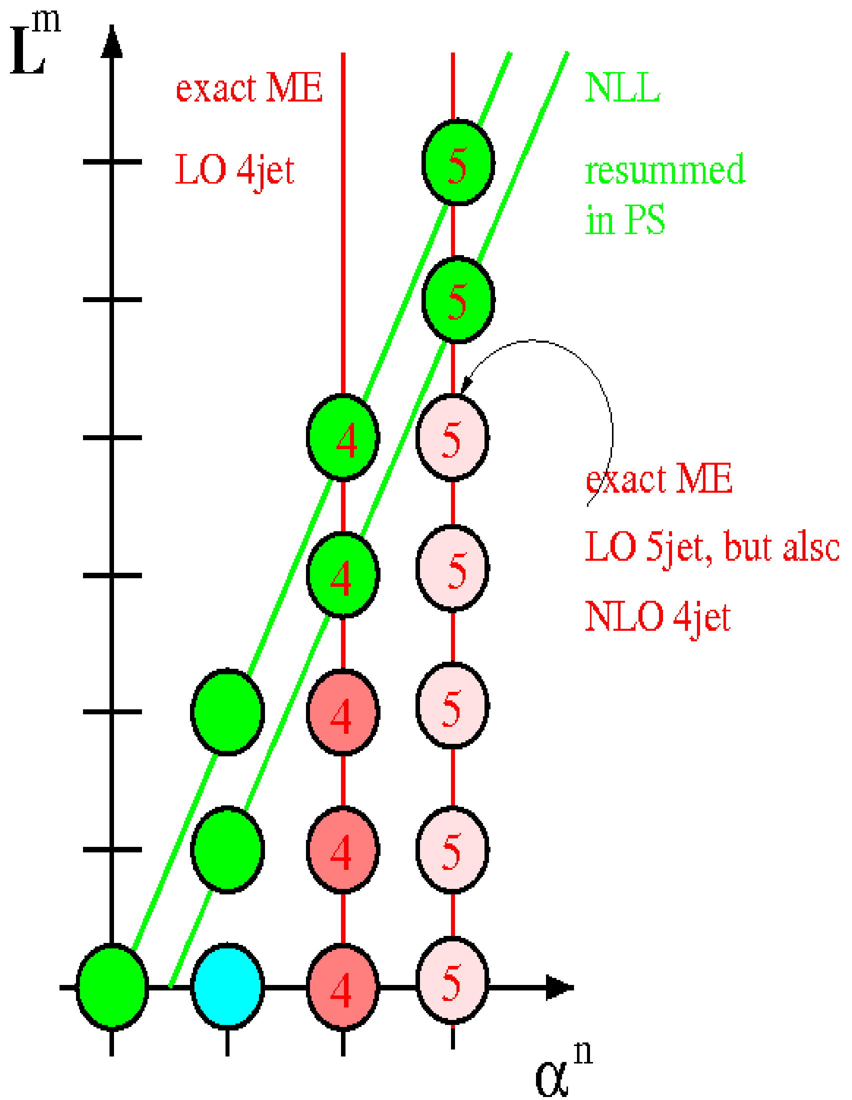
Multijet merging with heavy flavours:



Addendum: NLO matching

NLO matching: General thoughts

- Want to combine without double-counting
  - large logs of parton shower with
  - exact NLO matrix element
  - for one core process
- Algorithm (Powheg)
  - To avoid double-counting of NLO emission correct first (hardest) emission to ME accuracy with R/B
  - This is identical to the good old ME reweighting procedure
  - To retain NLO normalisation start with suitably weighted core configurations (NLO-weighted Born configs)



Powheg method in a nutshell

(Nason, JHEP 0411:040 (2004))

- Want total cross section and first emission correct at $\mathcal{O}(\alpha_S)$
- Master formula:

$$d\sigma_{\rm NLO} = d\Phi_{\mathcal{B}}\bar{\mathcal{B}}(\Phi_{\mathcal{B}})) \left[\bar{\Delta}(p_{\perp,\,\rm min}) + \int\limits_{p_{\perp,\,\rm min}} d\Phi_{\mathcal{R}|\mathcal{B}} \frac{\mathcal{R}(\Phi_{\mathcal{R}})}{\mathcal{B}(\Phi_{\mathcal{R}})} \bar{\Delta}(p_\perp)\right] .$$

- $\mathcal{B}$, $\mathcal{R}$ denote Born and real emission ME, respective phase space $\Phi_{\mathcal{B},\mathcal{R}}$.
- $\Phi_{\mathcal{R}|\mathcal{B}}$ is the phase space for one particle splitting connecting both.
- Since Sudakov form factor $\bar{\Delta}$ reads:

$$\bar{\Delta}(p_\perp) = \exp\left[-\int d\Phi_{\mathcal{R}|\mathcal{B}}\Theta[k_\perp(\Phi_R) - p_\perp]\frac{\mathcal{R}(\Phi_{\mathcal{R}})}{\mathcal{B}(\Phi_{\mathcal{R}})}\right],$$

  the expression in square bracket above $= 1$ (unitarity).
- $\bar{\mathcal{B}}(\Phi_{\mathcal{B}})$ denotes the NLO-weighted differential xsec for Born configuration.

Powheg method: algorithm

- Generate a starting Born-type parton configuration distributed according to

$$\mathrm{d}\Phi_{\mathcal{B}}\bar{\mathcal{B}}(\Phi_B) = \mathrm{d}\Phi_{\mathcal{B}}\left[\mathcal{B}(\Phi_{\mathcal{B}}) + \mathcal{V}(\Phi_{\mathcal{B}}) + \int \mathrm{d}\Phi_{\mathcal{R}|\mathcal{B}}\mathcal{R}(\Phi_{\mathcal{R}})\right]$$

  with $\mathcal{B}$ the Born, $\mathcal{V}$ the virtual, and $\mathcal{R}$ the real emission contribution.

- Generate the hardest emission according to $\bar{\Delta}$, where the usual splitting kernel $\mathcal{K}(t, z)$ is replaced by the ratio $\mathcal{R}(\Phi_{\mathcal{R}})/\mathcal{B}((\Phi_{\mathcal{B}})$:

$$\frac{\mathrm{d}t}{t}\mathrm{d}z\,\mathcal{K}(t, z) \rightarrow \mathrm{d}\Phi_{\mathcal{R}|\mathcal{B}}(t, z)\frac{\mathcal{R}(\Phi_{\mathcal{R}})}{\mathcal{B}(\Phi_{\mathcal{B}})}$$

- Perform a regular truncated shower on the resulting parton configuration.

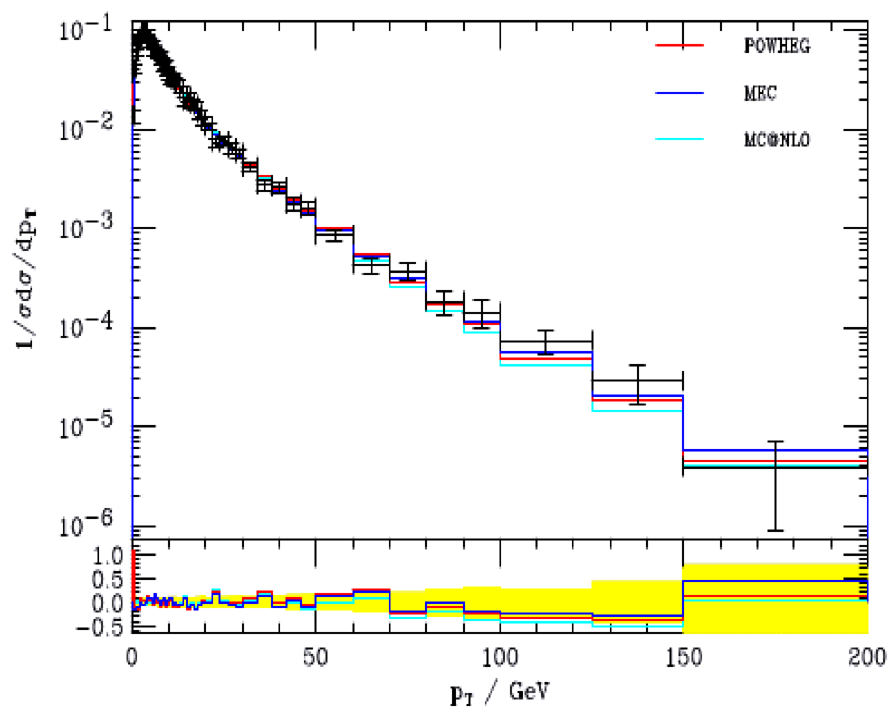
- By now: two publicly available implementations
  - Powheg-Box, sitting on top of a full event generator, communicating through LHA event files (see Paolo's talk for more details)
    available processes: Z, W, ZZ, Zj, single top, tt, H, VBF
  - Herwig++, available processes: Z, W, H, ZH, WH, VBF and VV' in prep.
    (some example results next slides)

Powheg method in Herwig++:

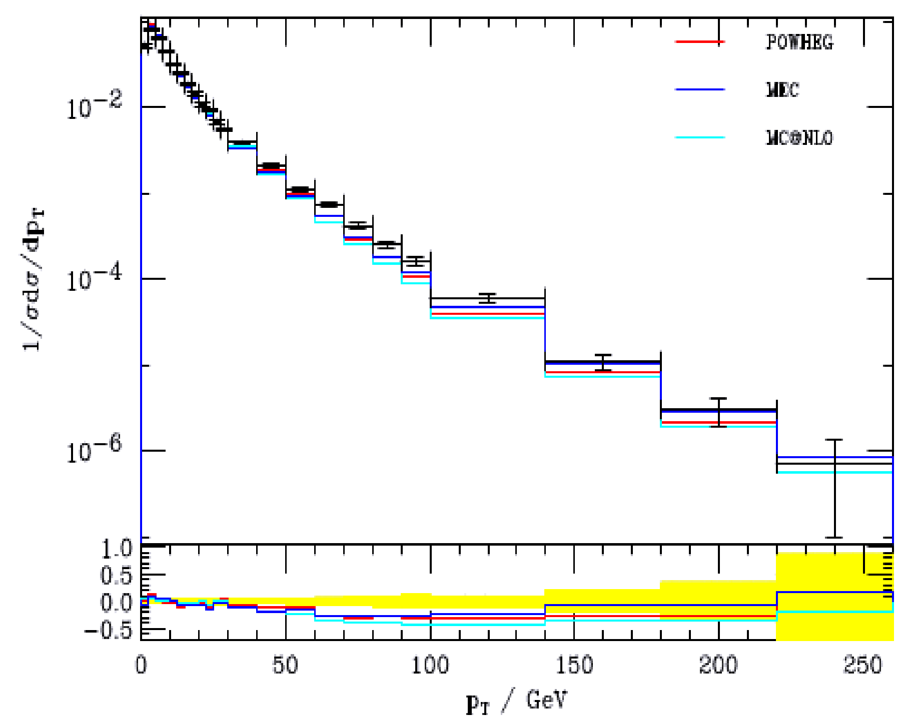
(Hamilton, Richardson & Tully, JHEP 0810:015 (2008))

- To achieve this, need some tricks – will skip this, but feel free to ask.
- Some example results: transverse momentum of Z at Run I and II

POWHEG method as implemented in HERWIG++ vs. MC@NLO and HERWIG++.



CDF Run I

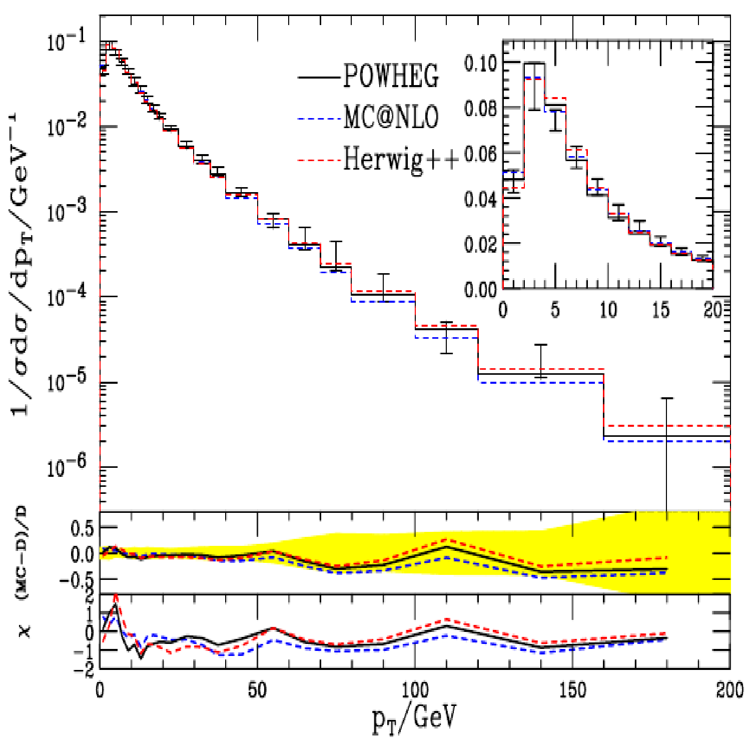


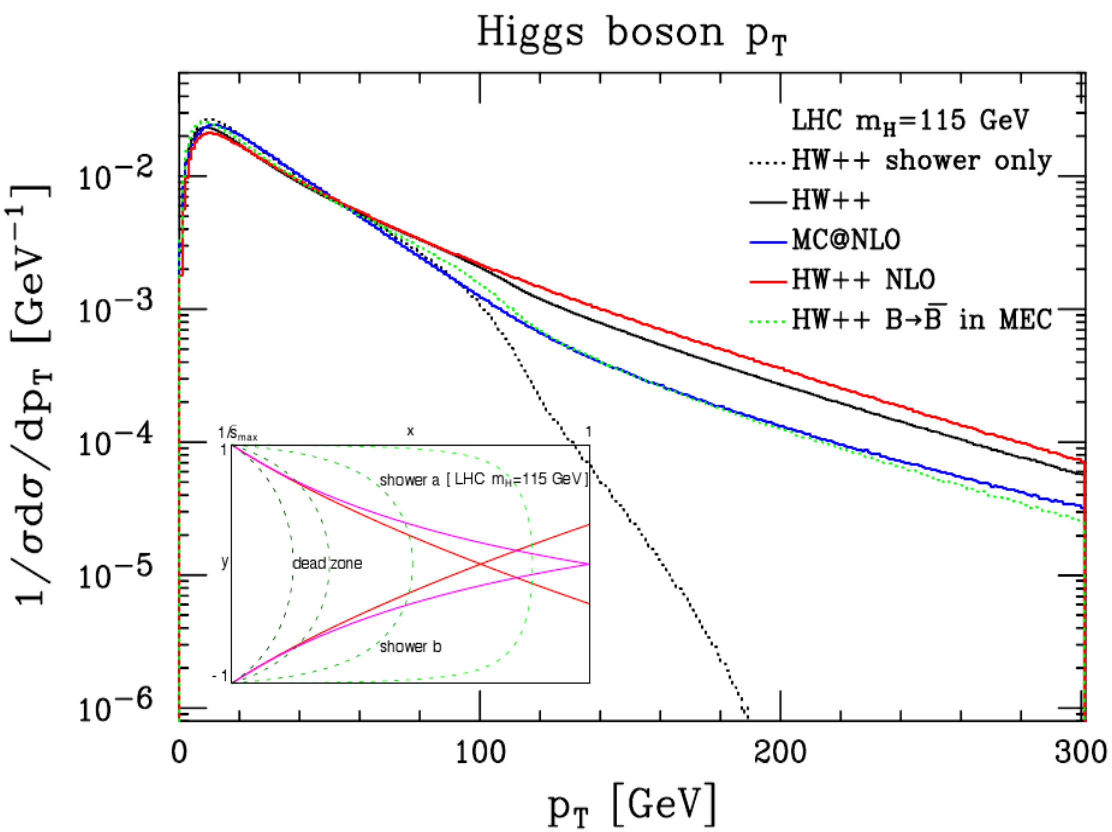
D0 Run II

Powheg method in Herwig++:

(Hamilton, Richardson & Tully, JHEP 0810:015 (2008))
(Hamilton, Richardson & Tully, JHEP 0904:116 (2009))

- To achieve this, need some tricks – will skip this, but feel free to ask.
- Some example results: transverse momentum of W and H

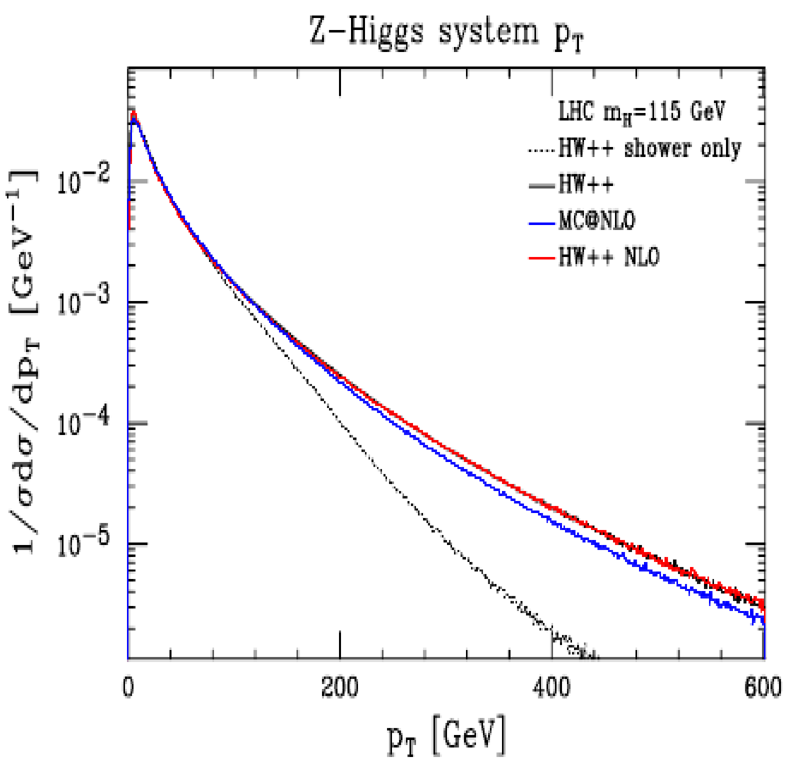


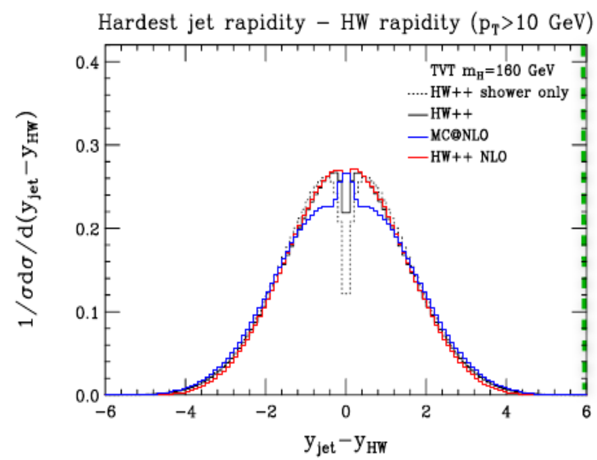


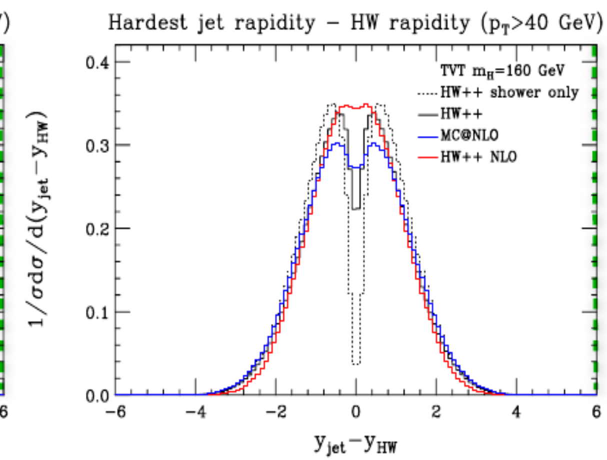
Powheg method in Herwig++:

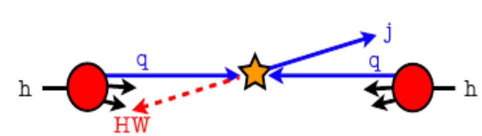
(Hamilton & Richardson, JHEP 0904:116 (2009))

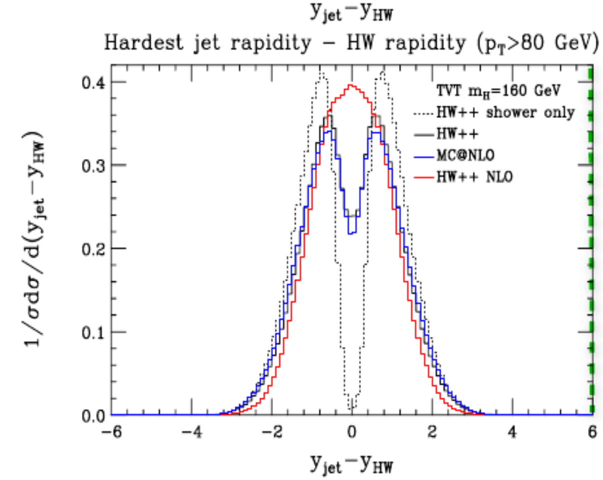
- Powheg method cures some phase-space issues in Herwig++ shower/MC@NLO:
  - Example: ZH production











Aside: Matrix elements in Sherpa

Three kinds of matrix elements:

- Since 1.2.0: Comix - mainly SM, can handle up to 8-10 final state particles

  (implementations for BSM-relevant methods have low priority in Comix.)
- Amegic++ - SM & BSM generator, up to 6 final state particles

  (development stalled, will eventually move to Comix.)
- Specific, hard-coded ME's at LO and NLO.
- Using Comix makes Sherpa even easier to handle:
  - no more libraries written out to be compiled in intermediate step.
- Sherpa/Amegic++ support FeynRules
  - a tool to generate Feynman rules directly from Lagrangians

    (a new standard to propagate BSM models?)
- No support for to read in LHA event-files - considered pointless by Sherpa

  (tools to write out such files may become available some day – ask S.Schumann.)

Aside: Some NLO stuff in Sherpa

- Sherpa was the first code to automate Catani-Seymour subtraction kernels
  - A method to isolate infrared divergences in real-emission part of NLO correction

    (consists of two parts: the actual subtraction term *S*, acting on the real emission bit *R*, and the term to be added to the virtual bit *V*)
- B now extensively used by the Blackhat collaboration, some state-of-the-art results (W+3,4 jets and Z+3 jets) done as Blackhat+Sherpa
- Nice connection to parton shower: Sherpa's new default shower bases on the same phase-space mappings

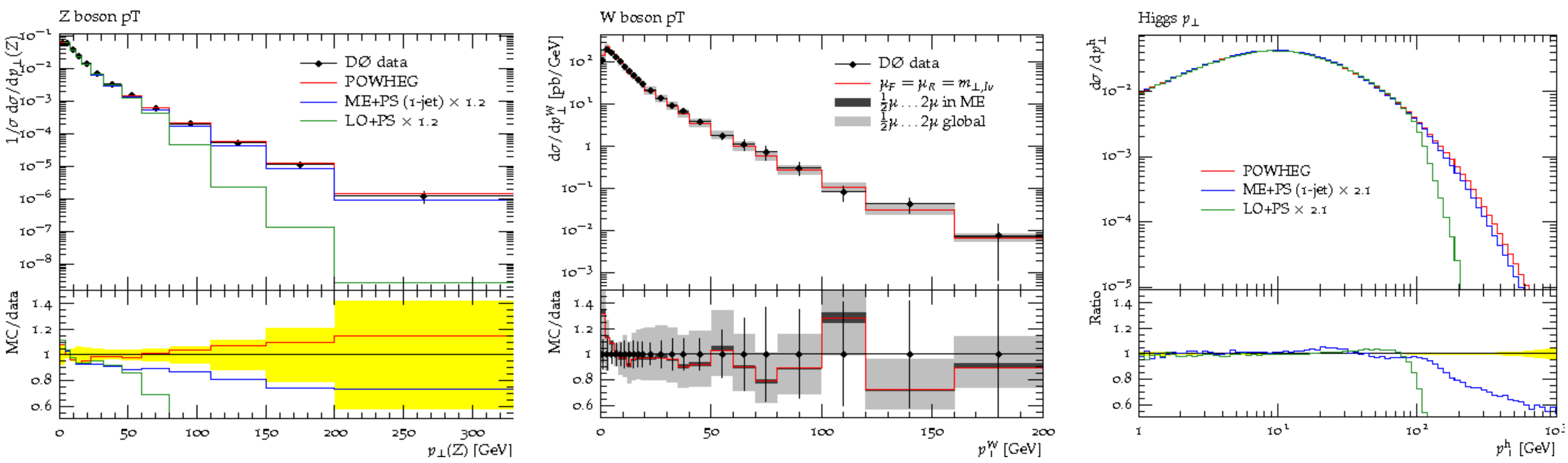
  (just forward, unfolding the extra emission, rather than backward, undoing it in the subtraction procedure)
- This made the automation of the Powheg method in Sherpa quite straightforward

  (Expect a release Sherpa 2.0.α in the next few months)

Automation of Powheg method:

(Hoeche, Krauss, Schoenherr & Siegert, arXiV:1008.5399)

- Method pioneered by Nason et al., implemented in Powheg-box and Herwig++: Aims to include NLO accuracy for total rate and some inclusive observables.
- Automated for simple processes (one colour line: ee, DIS, DY, VV', gg→H) in Sherpa, non-trivial processes to follow very soon.
- Example results (Z-pt, W-pt at Tevatron, Higgs-pt at 14 TeV LHC):



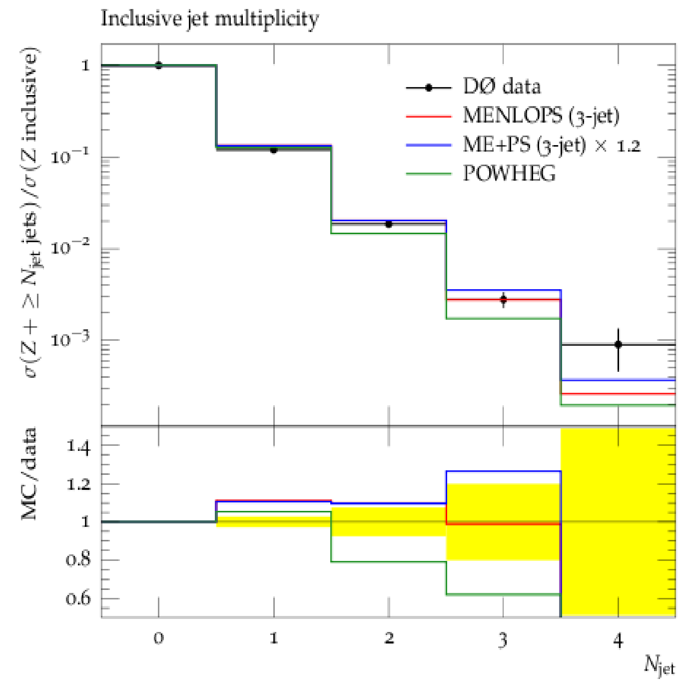
- Performed many checks on implementation, lots of plots in paper.

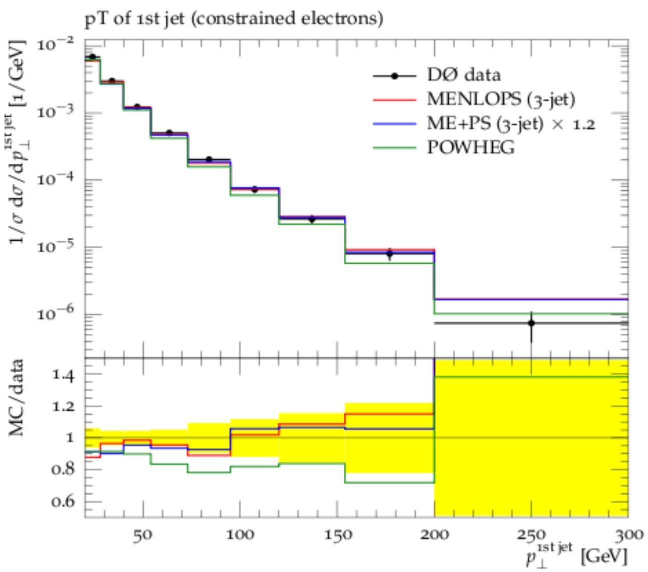
MENLOPS in Sherpa:

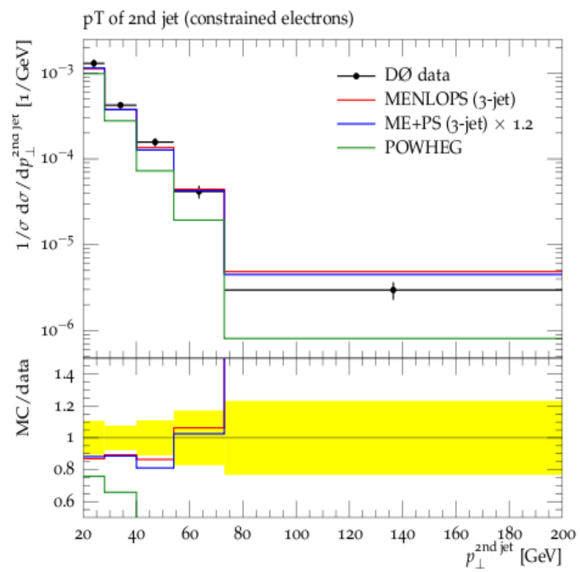
(Hoeche, Krauss, Schoenherr & Siegert, arXiV:1009.1127), also (Hamilton & Nason, JHEP 1006:039 (2010))

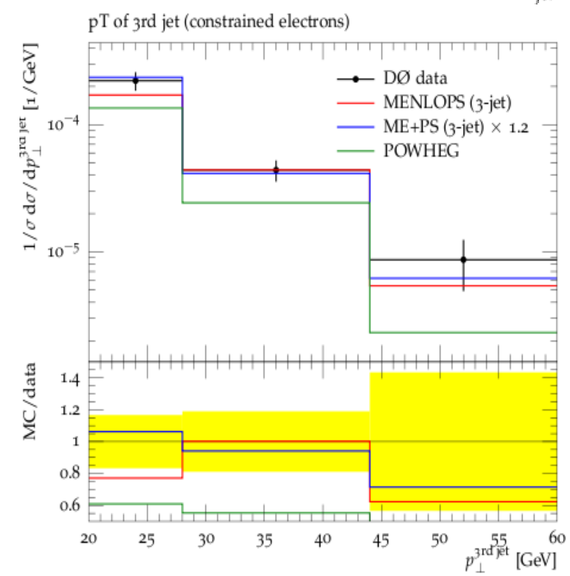
- Based on Powheg, can add multijet-merging on top – **inclusive sample with NLO cross section and ME+PS accuracy in all jet shapes**

(development in parallel with Hamilton and Nason, identical formalism in both, but different implementation: H+N use Powhegbox+Madgraph+Pythia, cf. Paolo's talk)









Comparison of ME+PS, Powheg and MENLOPS

$$\mathrm{d}\sigma^{\mathrm{POWHEG}} = \mathrm{d}\phi_B \bar{\mathcal{B}}(\phi_B) \left[\bar{\Delta}(p_{\perp,\min}) + \int_{p_{\perp,\min}} \mathrm{d}\Phi_{R|B} \frac{\mathcal{R}(\phi_{\mathcal{R}})}{\mathcal{B}(\phi_{\mathcal{B}})} \bar{\Delta}(p_\perp) \right]$$

$$\mathrm{d}\sigma^{\mathrm{MEPS}} = \mathrm{d}\phi_B \mathcal{B}(\phi_B) \left[\Delta(p_{\perp,\min}) + \int_{p_{\perp,\min}}^{Q_{\mathrm{cut}}} \mathrm{d}\Phi_{R|B} \mathcal{K}(\phi_{R|B}) \Delta(p_\perp) + \int_{Q_{\mathrm{cut}}} \mathrm{d}\Phi_{R|B} \frac{\mathcal{R}(\phi_{\mathcal{R}})}{\mathcal{B}(\phi_{\mathcal{B}})} \Delta(p_\perp) \right]$$

- Form of NLO cross sections for both methods nearly identical.
- Most notably NLO vs. LO normalisation ($\bar{B}$ vs. B), but also argument of Sudakov form factors ($\mathcal{R}/\mathcal{B}$ vs. splitting kernels $\mathcal{K}$) in different regions of phase space (jet production vs. jet evolution)

MENLOPS for first emission

- To combine (MENLOPS): Cluster multijet configuration back to a Born-level configuration, where NLO accuracy is implemented, reweight the emerging Born-level expression with the NLO-reweighted term.

$$
\begin{aligned}
\mathrm{d}\sigma^{\mathrm{MENLOPS}} \;=\; & \mathrm{d}\phi_B \bar{\mathcal{B}}(\phi_B) \left[\bar{\Delta}(p_{\perp,\mathrm{min}}) + \int\limits_{p_{\perp,\mathrm{min}}}^{Q_{\mathrm{cut}}} \mathrm{d}\Phi_{R|B} \frac{\mathcal{R}(\phi_{\mathcal{R}})}{\mathcal{B}(\phi_{\mathcal{B}})} \bar{\Delta}(p_\perp) \right. \\
& \left. + \int\limits_{Q_{\mathrm{cut}}} \mathrm{d}\Phi_{R|B} \frac{\mathcal{R}(\phi_{\mathcal{R}})}{\mathcal{B}(\phi_{\mathcal{B}})} \Delta(p_\perp) \right]
\end{aligned}
$$

Conclusions and Outlook:

- Merging and matching methods extremely powerful tools, but some care needs to be taken:
  - Note that codes such as Alpgen, Helac, and Madgraph inherit α_s and to some extent scale choices from underlying event generator – may lead to inconsistencies and deserves careful studies (see talks during V+jet workshop at IPPP on this topic)
- Especially event generators Herwig++ and Sherpa made huge progress towards embedding HO accuracy in a systematic way, first steps also in Pythia8 (W+jets, by Lonnblad & Prestel)
- Automation of PowHeg method for trivial processes in Sherpa successful, work on non-trivial processes started
- MENLOPS method will become the standard for simulations, incorporating the best of both worlds (merging and matching).
- Systematic multijet merging at NLO for hadronic collisions in reach ....

Sherpa status:

- Construction started in late 1990's, mainly through diploma and PhD theses.
- Up to date: 7 PhD theses and 12 diploma theses finished, in total around 50 man-years of code development and physics improvement.
- First release in 2002: Proof-of-concept version Sherpa 1.0.ά, by now 1.2.3 in prep.
- Structure very modular, essentially bottom-up as design principle.
- From the beginning: Focus on perturbative aspects, especially multi-jet merging.
- By now: Two independent matrix element generators, a new shower (old Pythia-like v.o. shower to be disbanded in next release, 1.3), an independent version of cluster hadronization, an independent implementation of Pythia's old UE model, elaborate hadron decays, and QED final state radiation.
- Only recently: own fragmentation model, hadron decays, and QED radiation, interface to Pythia 6 kept for systematic checks and backward compatibility.
- Current main focus: NLO accuracy in multijet merging.
- In addition: New, independent Minimum Bias and Underlying Event model under active development.