Sherpa Tutorial

Stefan Höche [a]
Université Catholique de Louvain
Technische Universität Dresden

a) for the Sherpa collaboration: T. Fischer, T.Gleisberg, SH, F. Krauss, T. Laubrich
A. Schälicke, S. Schumann, F. Siegert, J. Winter

Overview

Part I: Introduction to Sherpa

- Status of development
- Physics modules
  - Selected results

Part II: Hands-on examples

- Installation guide
- Examples:
  - Z+jets production @ Tevatron
  - Diboson production
  - Decay chains
  - ...

Status of Sherpa

Scope of the project:

- Provide a multi-purpose tool, capable of simulating
  - SM backgrounds as well as
  - new physics scenarios (e.g. MSSM, ADD)
    at ee, $\gamma\gamma$ and hadron colliders (others to come)

Special emphasis:

- Account for multi-jet production through tree level MEs
- Combine ME and PS using CKKW prescription
  to obtain inclusive event samples

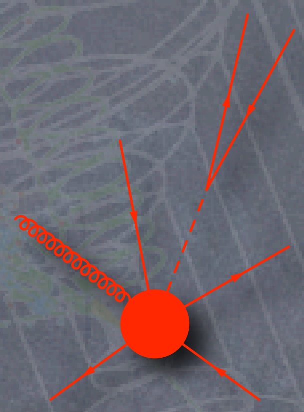
Where to find us:

- http://www.sherpa-mc.de
  for downloads, manual, bug reports ...
- T. Gleisberg, SH, F. Krauss, A. Schälicke, S. Schumann
  and J. Winter JHEP 0402:056,2004

The Sherpa framework

Key features

- Automatic ME generation via AMEGIC++



TECHNISCHE UNIVERSITÄT DRESDEN

The Sherpa framework

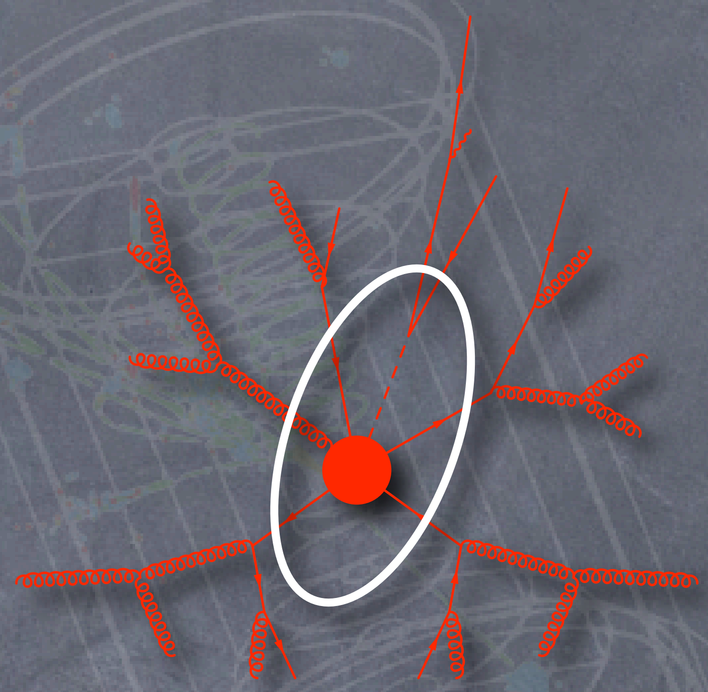
Key features

- Automatic ME generation via AMEGIC++
- Generation of QCD/QED radiation via APACIC++ (PYHTIA-like parton shower)

The Sherpa framework

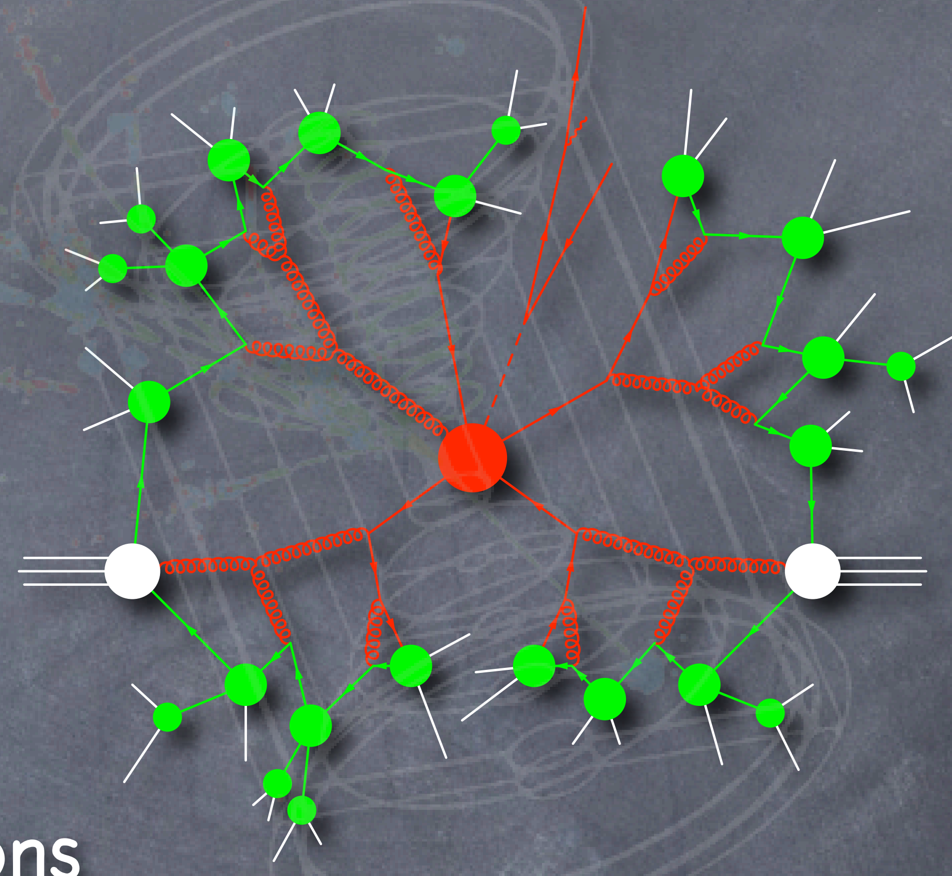
Key features

- Automatic ME generation via AMEGIC++
- Generation of QCD/QED radiation via APACIC++ (PYHTIA-like parton shower)
- Merging of ME and PS according to CKKW



The Sherpa framework

Key features

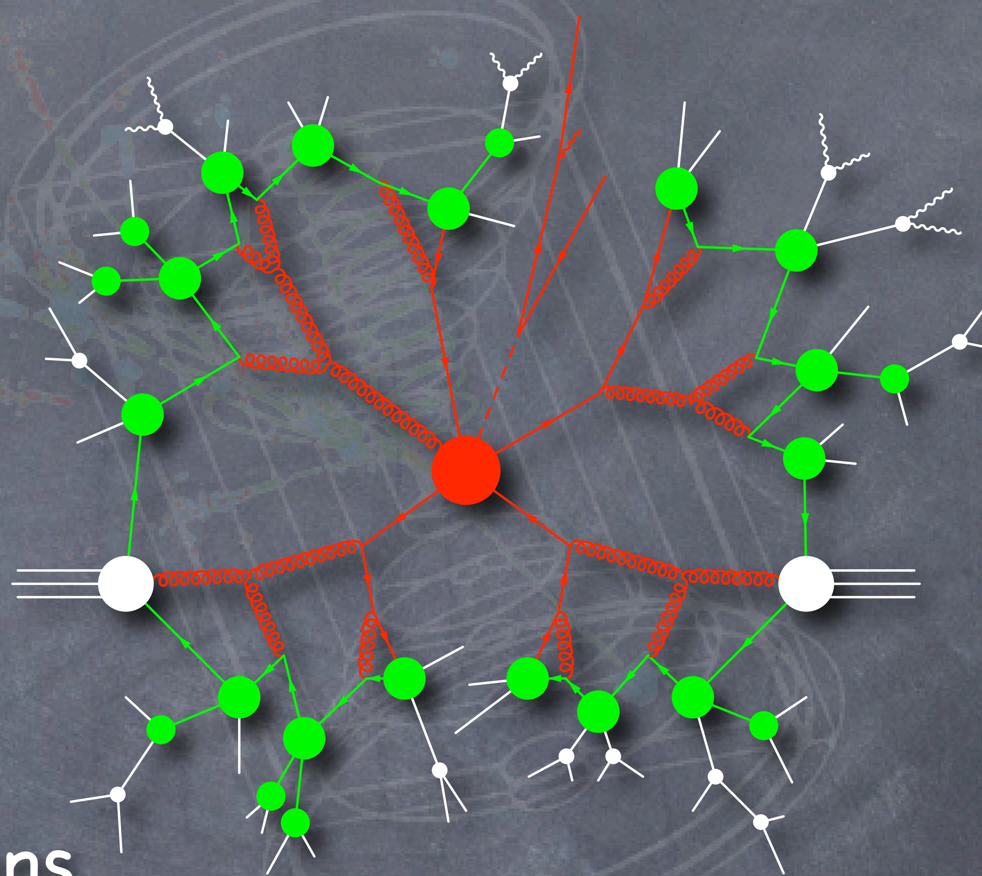


- Automatic ME generation via AMEGIC++
- Generation of QCD/QED radiation via APACIC++ (PYHTIA-like parton shower)
- Merging of ME and PS according to CKKW
- Simulation of Multiple Interactions acc. to T. Sjöstrand and M. van Zijl, Phys. Rev. D36 (1987)
- Cluster fragmentation in preparation
  J. Winter, F.Krauss, G. Soff Eur.Phys.J.C36:381-395,2004
  Currently string fragmentation via PYTHIA

The Sherpa framework

Key features

- Automatic ME generation via AMEGIC++
- Generation of QCD/QED radiation via APACIC++ (PYHTIA-like parton shower)
- Merging of ME and PS according to CKKW
- Simulation of Multiple Interactions acc. to T. Sjöstrand and M. van Zijl, Phys. Rev. D36 (1987)
- Cluster fragmentation in preparation
  J. Winter, F.Krauss, G. Soff Eur.Phys.J.C36:381-395,2004
  Currently string fragmentation via PYTHIA
- Own hadron decay framework and τ decay library



Sherpa itself is the framework for steering the generator

Matrix Elements: AMEGIC++

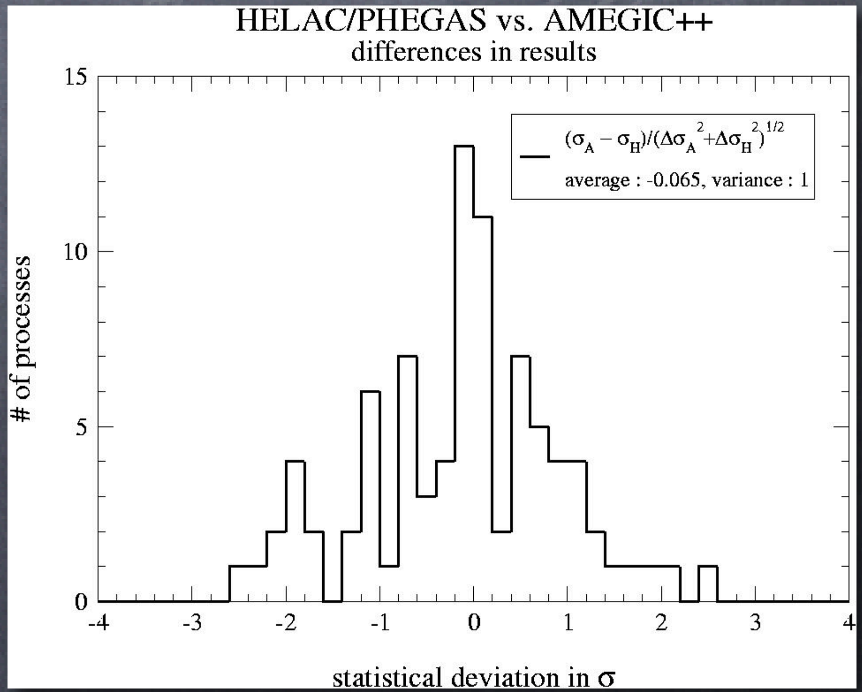
R.Kuhn, F.Krauss, G.Soff, JHEP 0202:044,2002

Sherpas built in ME Generator provides

- Fully automated calculation of (polarised) cross sections in the SM, MSSM and ADD model
- Performance comparable to that of dedicated codes
- Study of signal and backgrounds in one framework
- Expandability (new physics models)

- Extensively tested e.g. in $e^+e^- \to 6f$ vs. HELAC/PHEGAS
- Recent comparison of arbitrary 2→2 SUSY processes vs. WHIZARD/O'Mega & SMadGraph

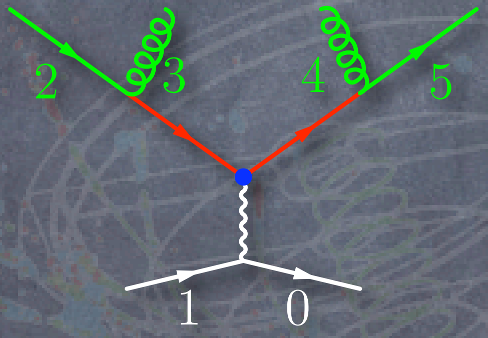
K.Hagiwara, W. Kilian, F.Krauss, T.Ohl, T.Plehn, D.Rainwater, J.Reuter, S.Schumann hep-ph/0512260



Matrix Elements: AMEGIC++

AMEGIC++ is a generator-generator:

- Given initial and final state, AMEGIC++ constructs diagrams
- Translates diagrams into helicity amplitudes
- Generates phase space mappings for each diagram
  (to be used in multi-channel integration)



$$\leftrightarrow \quad D_{\rm iso}(23,45) \otimes P_0(23) \otimes P_0(45) \otimes D_{iso}(2,3) \otimes D_{iso}(4,5)$$

- C++ code representing all the above stored to disk ...

New features:

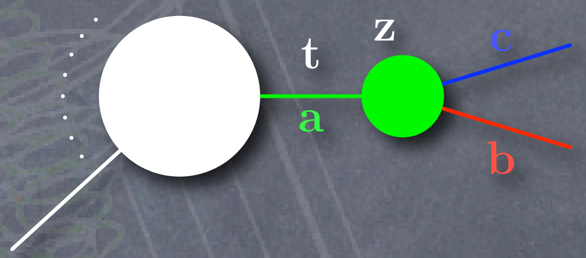
- MSSM spectra can be read from SLHA input files
- Specific decay modes of particles can be enforced
  e.g. $t \to W^+ \to b l^+ \nu_l$ or $\tilde{e}_R \to e^- \tilde{\chi}_1^0$

Parton Showers: APACIC++

R.Kuhn, F.Krauss, G.Ivanyi, G.Soff, CPC 134 (2001) 223

F.Krauss, A. Schaelicke, G.Soff, hep-ph/0503087

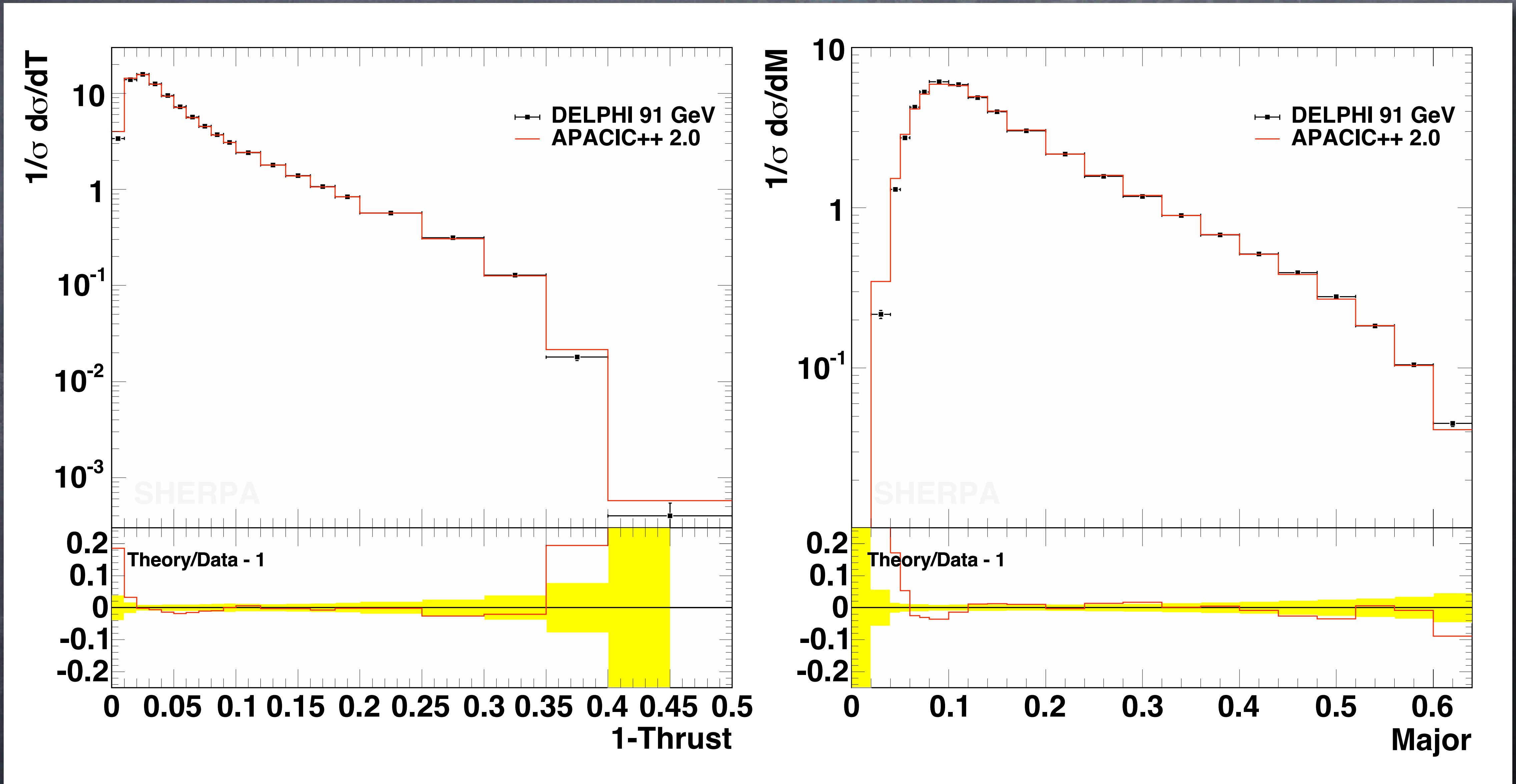
Features of Sherpas parton shower:



- Virtuality ordered parton cascade, colour coherence imposed by angular veto
- Final and intial state radiation in e^+e^- and hadron collisions (no DIS-like situations so far)
- Extensively tested (see next slides)
- Algorithm similar to old PYTHIA shower
- 2nd key ingredient of CKKW implementation in Sherpa (1st is AMEGIC++)

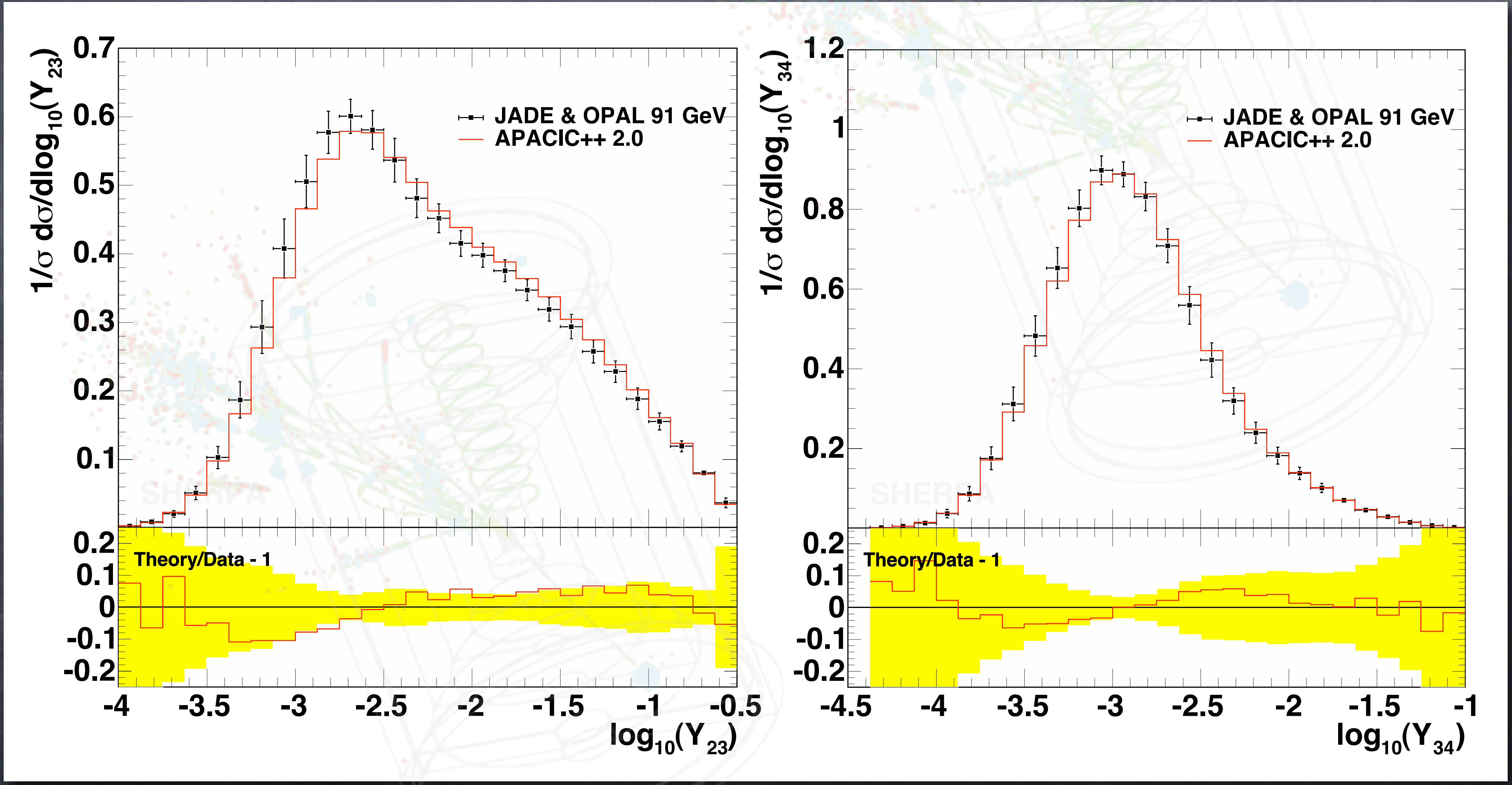
Parton Showers: APACIC++

- Event shapes in e^+e^- annihilation at $E_{cms} = 91$ GeV (LEP)



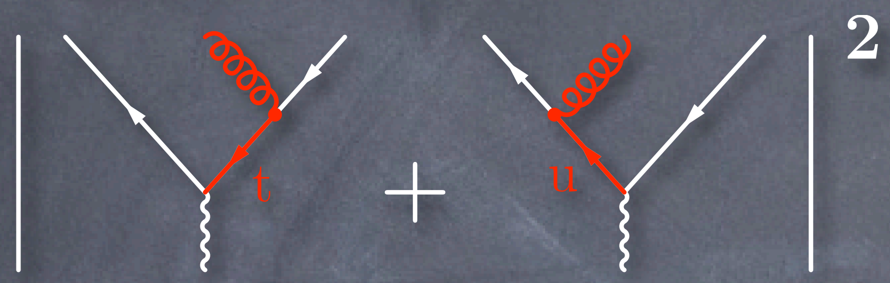
Parton Showers: APACIC++

- Diff. jet rates in e^+e^- annihilation at E_{cms} = 91 GeV (LEP)



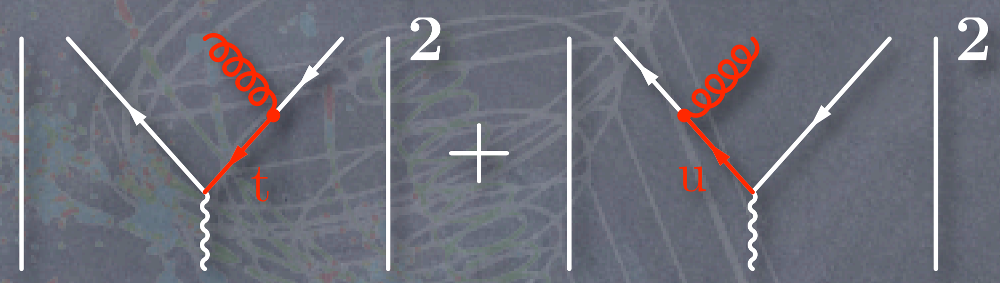
ME vs. PS

Matrix Elements



- Exact to fixed order in running coupling α
- Include all quantum interferences
- Calculable only for low FS multiplicity (n≲6-8)

Parton Showers



$$\mathrm{d}\sigma_{n+1} = \mathrm{d}\sigma_n \otimes \sum_{b\in q,g} \frac{\mathrm{d}t}{t}\mathrm{d}z\,\frac{\alpha_s(t,z)}{2\pi}P_{a\to b}(z)$$

- Resum (next-to) leading logarithms to all orders
- Interference effects e.g. through angular ordering

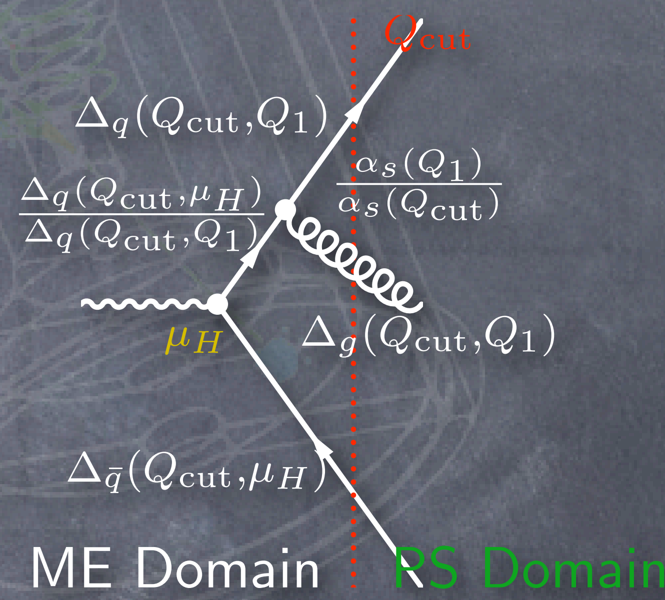
→ Desirable to combine both approaches to have
- Good description of hard/wide-angle emissions (ME)
- Correct intrajet evolution (PS)

→ Must prevent double counting through CKKW

TECHNISCHE UNIVERSITÄT DRESDEN

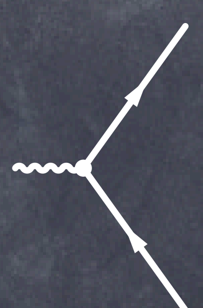
Combining ME & PS: CKKW

- Define jet resolution parameter Q_{cut} (Q-jet measure)
  ➡ divide phase space into regions of jet production (n-jet ME) & jet evolution (PS)
- Select jet multiplicity and kinematics according to σ 'above' Q_{cut}
- K_T cluster backwards (construct PS tree) and identify core process
- Reweight ME to get exclusive samples at resolution scale Q_{cut}
- Start PS at scale μ_{hard}, reject all emissions above Q_{cut}



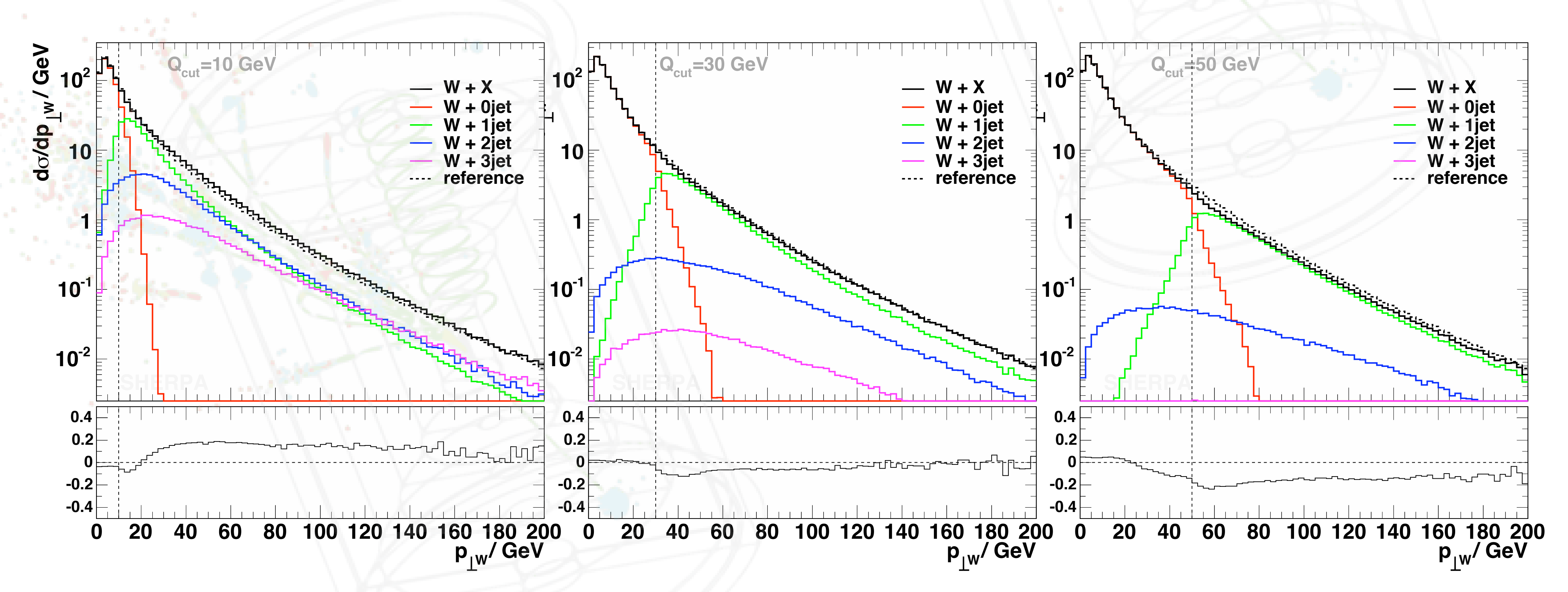
➡ This yields the correct jet rates !
e.g. 2-jet rate in 2-jet event at scale q

$$R_2(q^2) = \left(\Delta(Q_{\rm cut}, \mu_{\rm hard}) \, \frac{\Delta(q, \mu_{\rm hard})}{\Delta(Q_{\rm cut}, \mu_{\rm hard})} \right)^2$$



Consistency checks

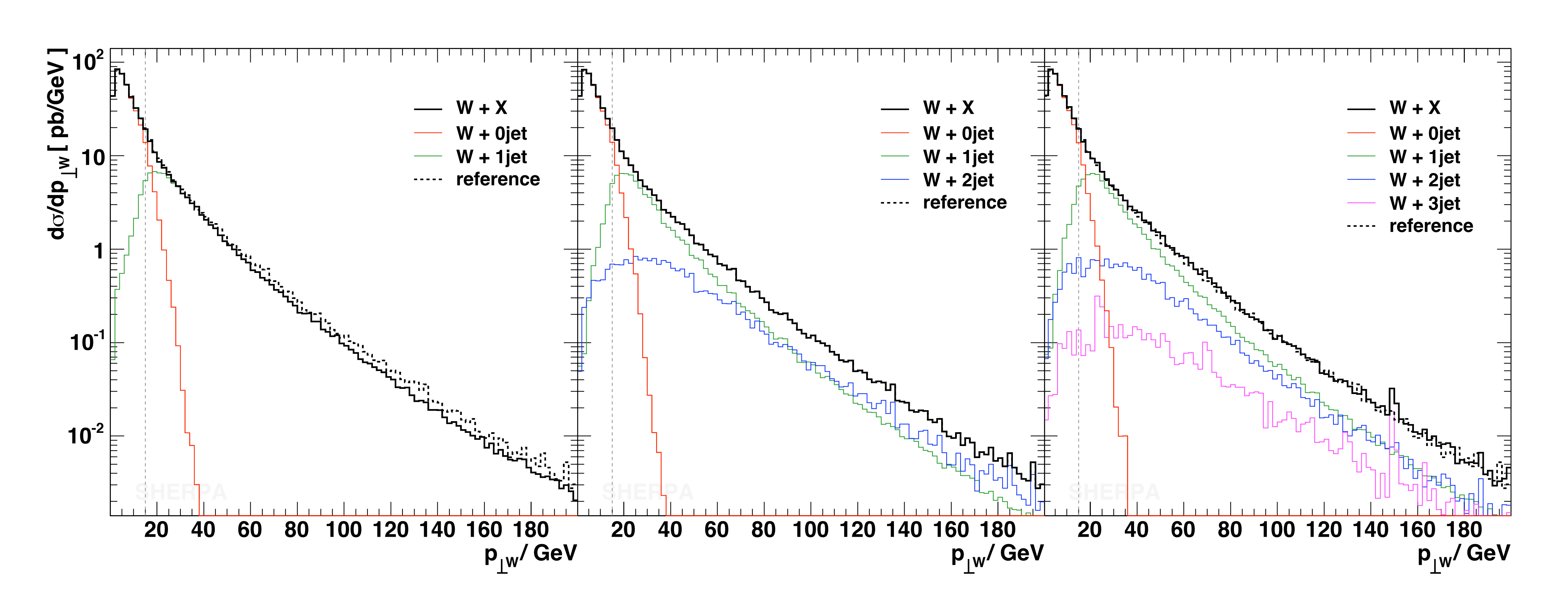
- W+jets production at Tevatron Run II
  Stability tests of the procedure
  F. Krauss, A.Schälicke, S. Schumann,
  Phys.Rev.D70(2004)114009, Phys.Rev.D72(2005)054017
  - Variation of phase space separation cut Q_{cut}



- Global K-factor

Consistency checks

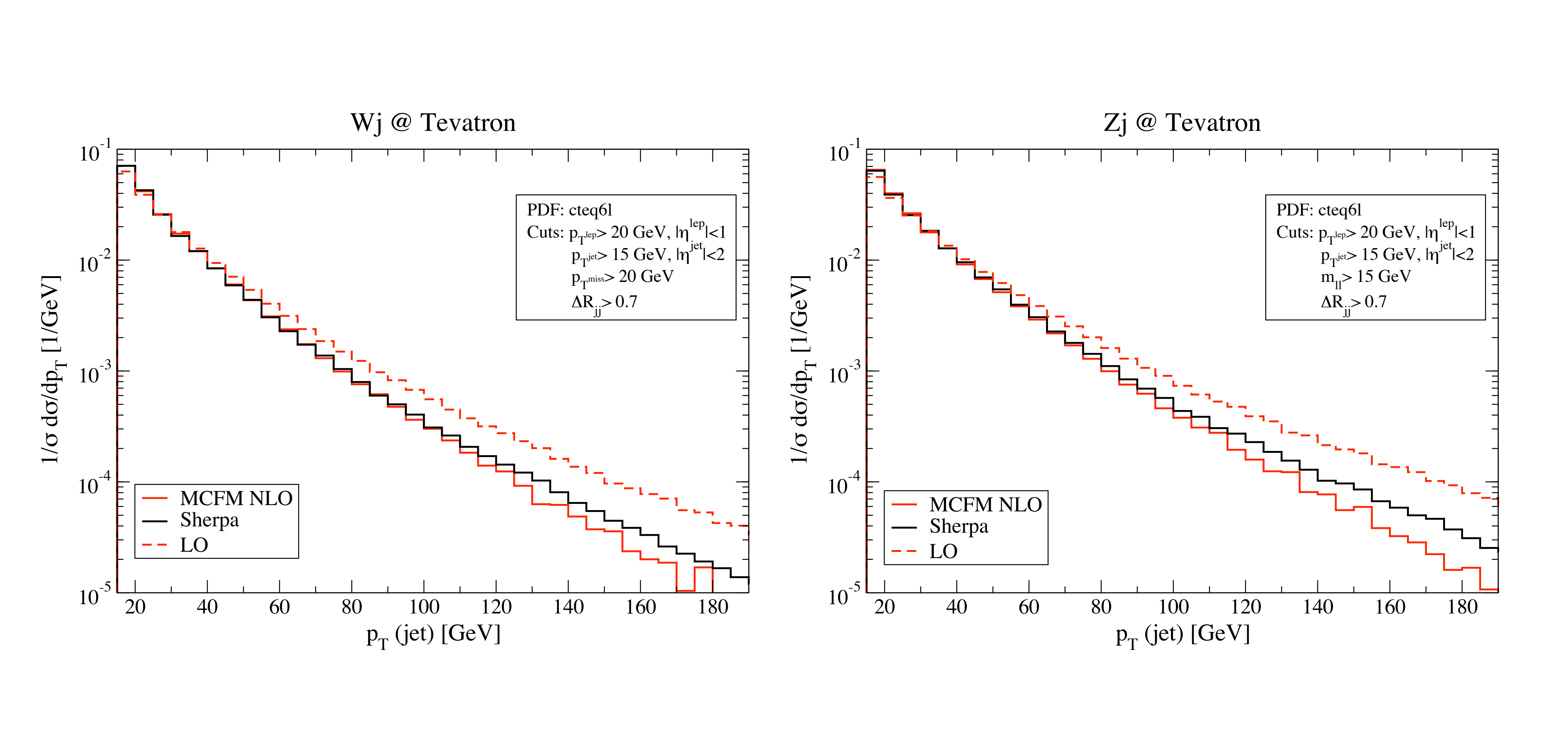
- W+jets production at Tevatron Run II
  Stability tests of the procedure
  F. Krauss, A.Schälicke, S. Schumann,
  Phys.Rev.D70(2004)114009, Phys.Rev.D72(2005)054017
  - Variation of maximum jet multiplicity



- Global K-factor

V+jets production @ Tevatron

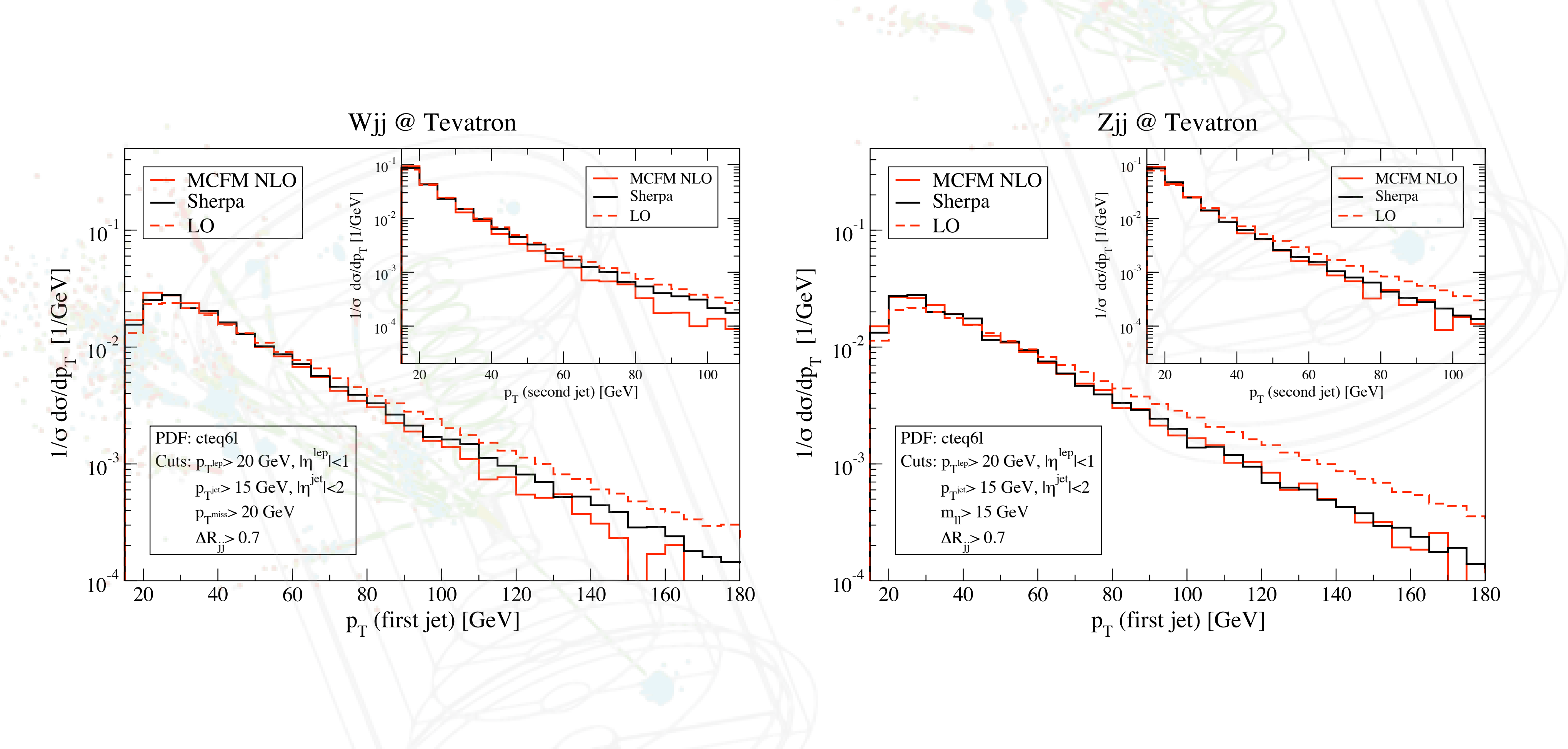
- Jet p_T in W- and Z+1jet events
  Sherpa vs. MCFM



- Global K-factor

V+jets production @ Tevatron

- Jet p_T in W- and Z+2jet events
  Sherpa vs. MCFM

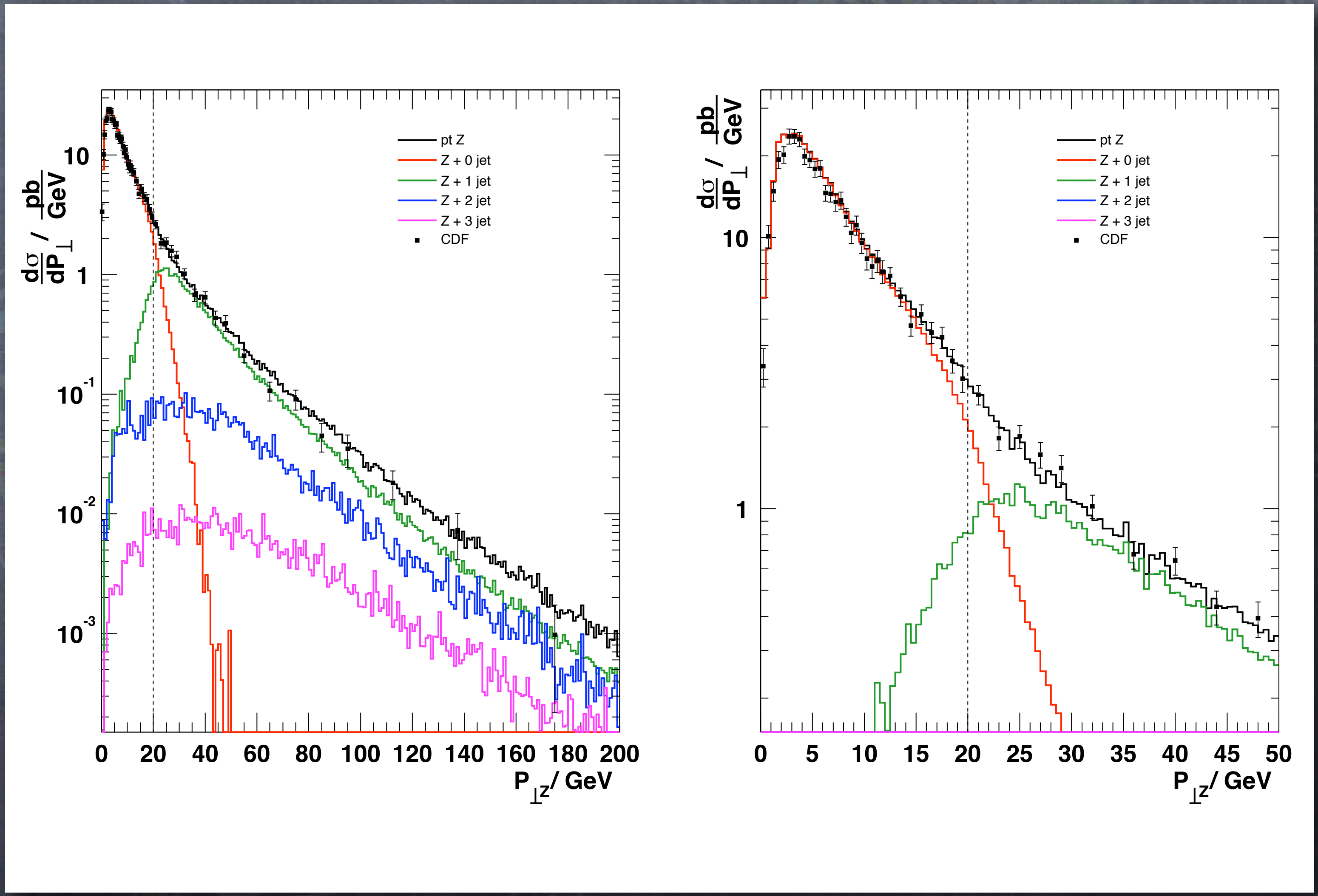


- Global K-factor

V+jets production @ Tevatron

A.Schälicke, F. Krauss JHEP 0507:018,2005

- $p_{T,Z}$ measured at CDF
  (Phys. Lett. B513 (2001) 292)

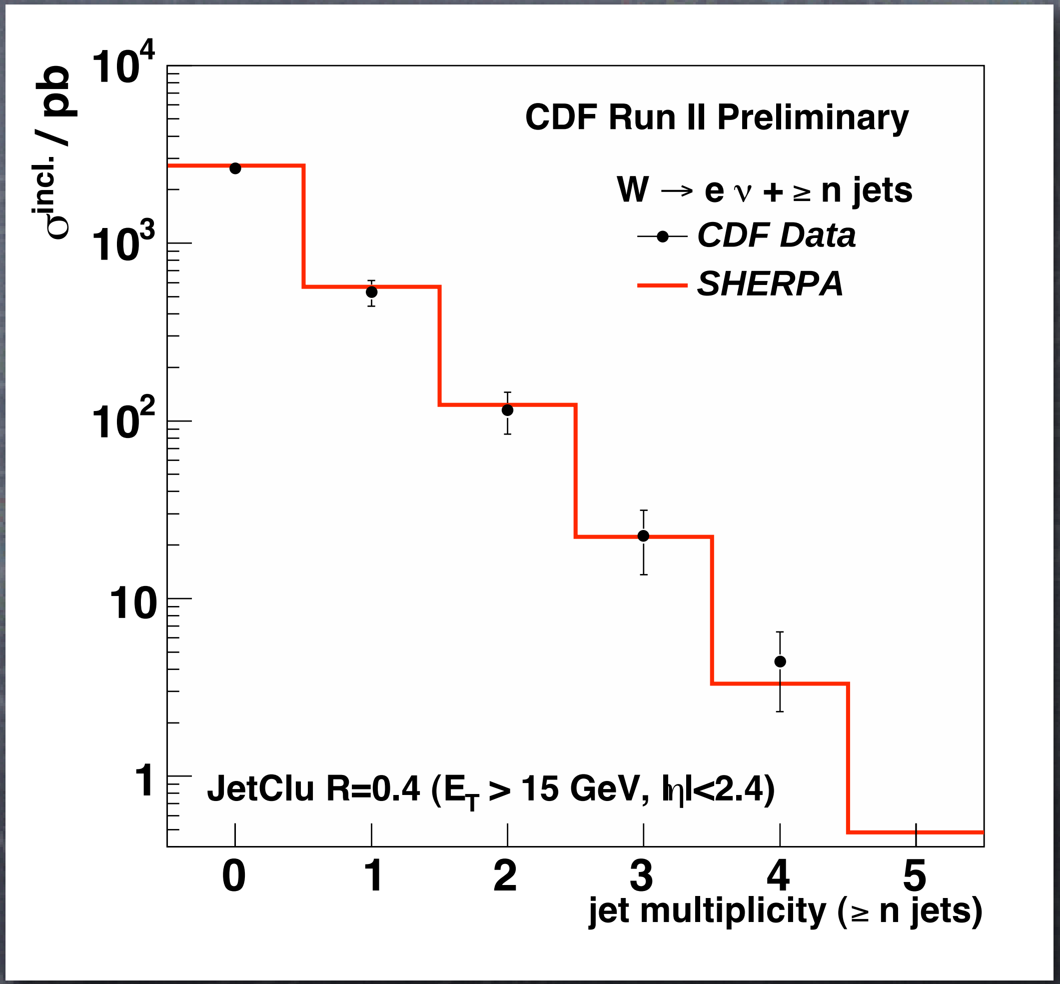


- Global K-factor

V+jets production @ Tevatron

A.Schälicke, F. Krauss JHEP 0507:018,2005

- Inclusive jet cross sections, CDF (hep-ex/0405067)

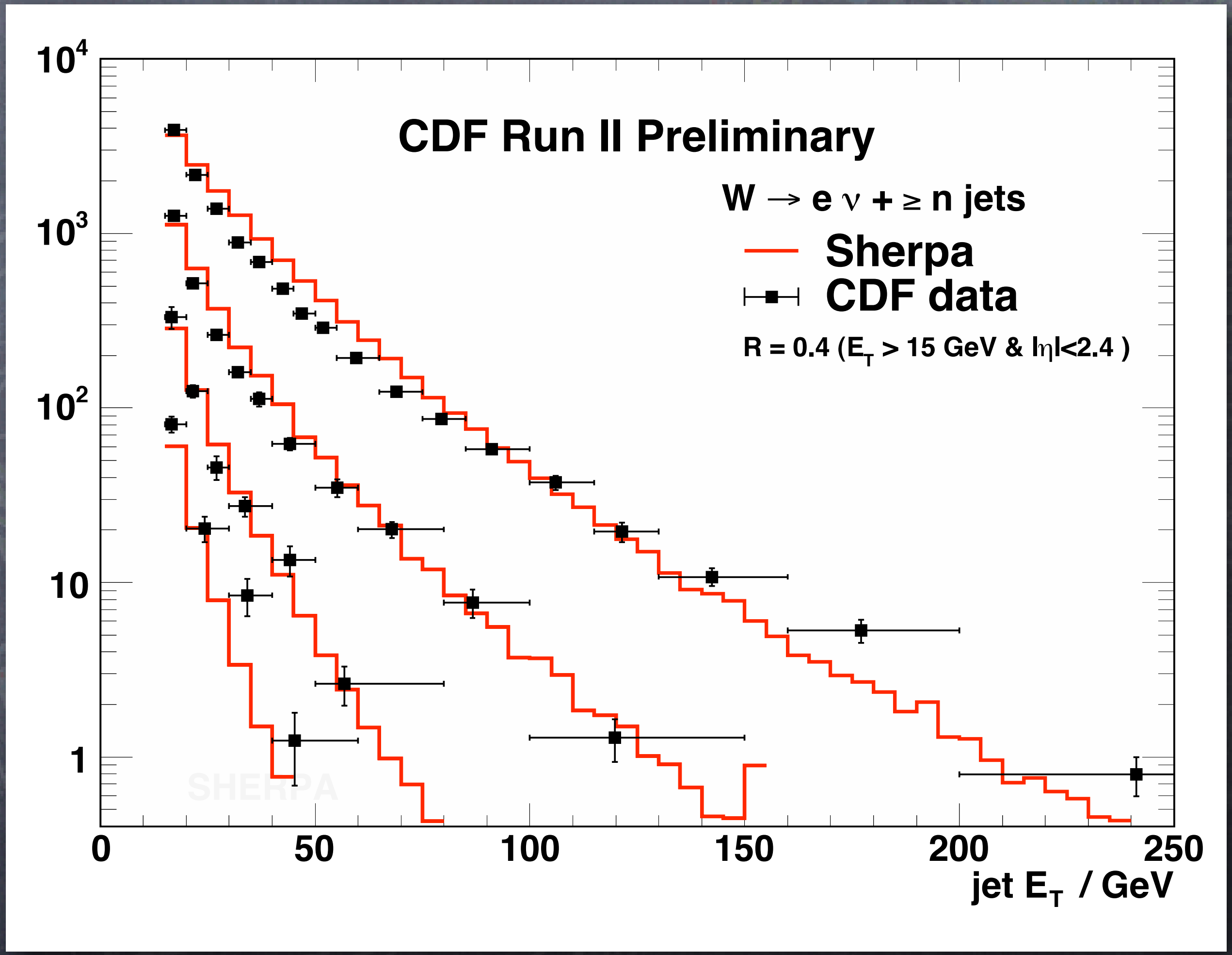


- Global K-factor

V+jets production @ Tevatron

A.Schälicke, F. Krauss JHEP 0507:018,2005

- jet-p_T, measured at CDF
  (hep-ex/0405067)

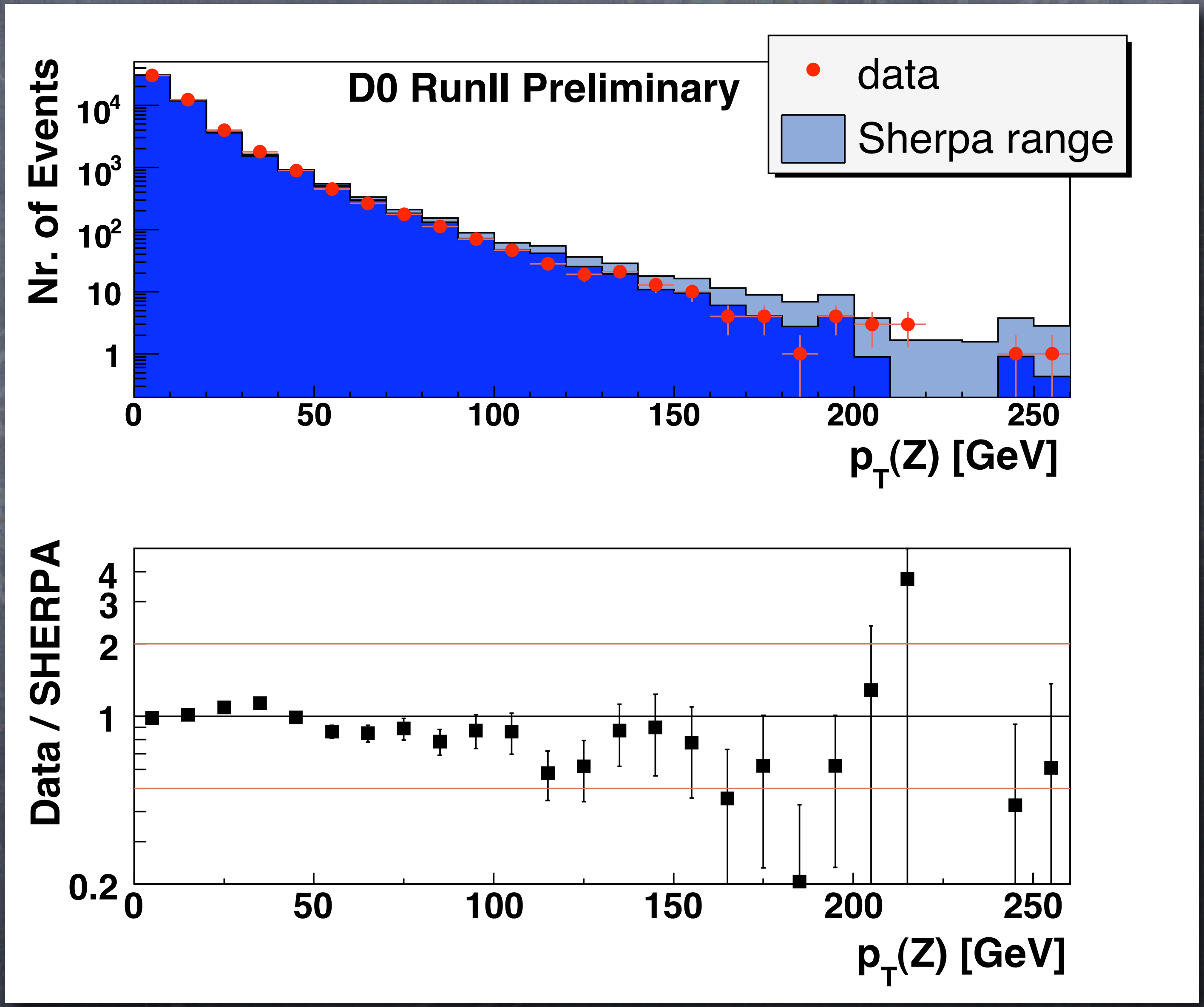


- Global K-factor

V+jets production @ Tevatron

The DØ Collaboration, DØ Note 5066-CONF

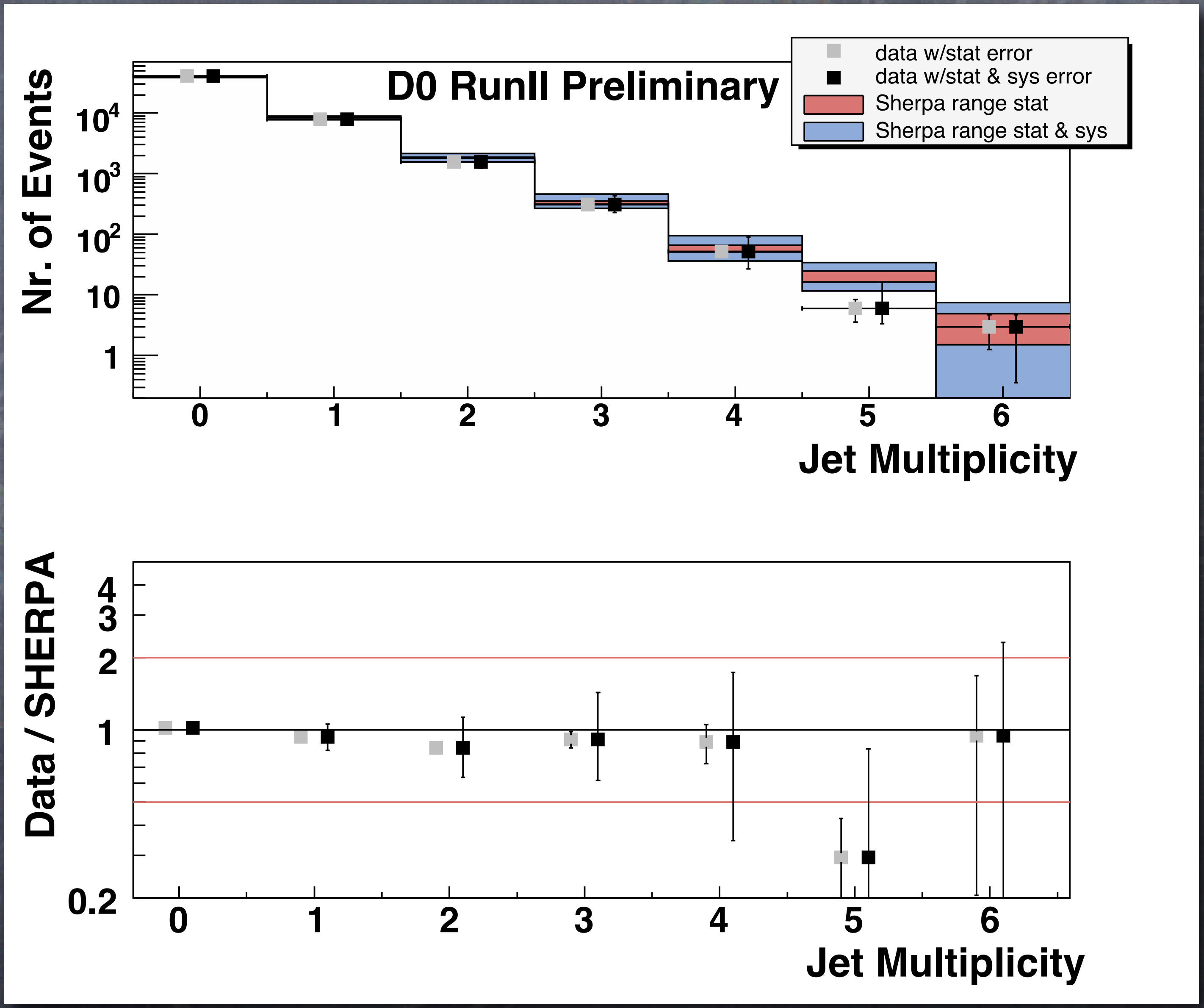
- Z-p_T, measured at DØ



V+jets production @ Tevatron

The DØ Collaboration, DØ Note 5066-CONF

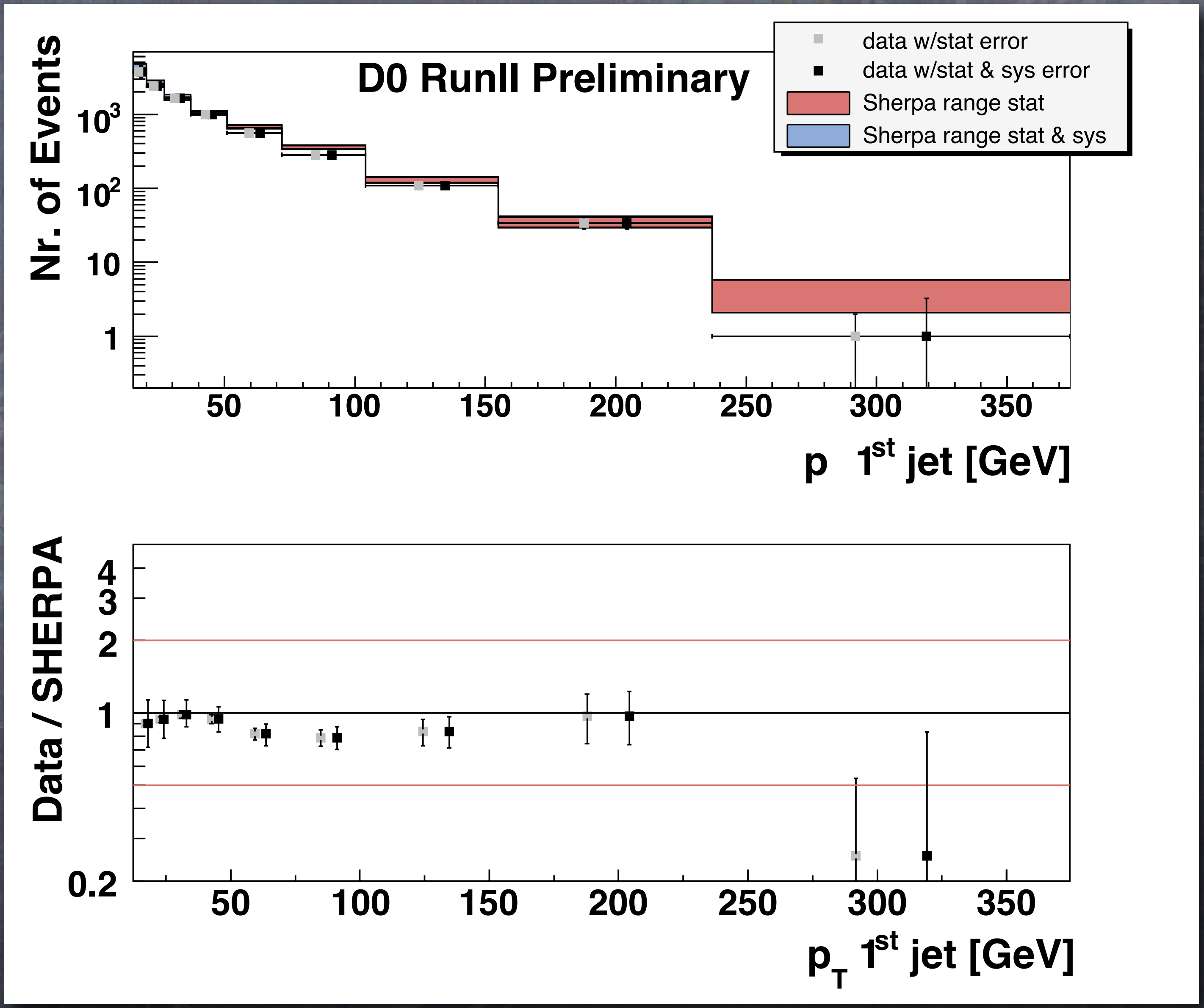
- jet multiplicity, measured at DØ



V+jets production @ Tevatron

The DØ Collaboration, DØ Note 5066-CONF

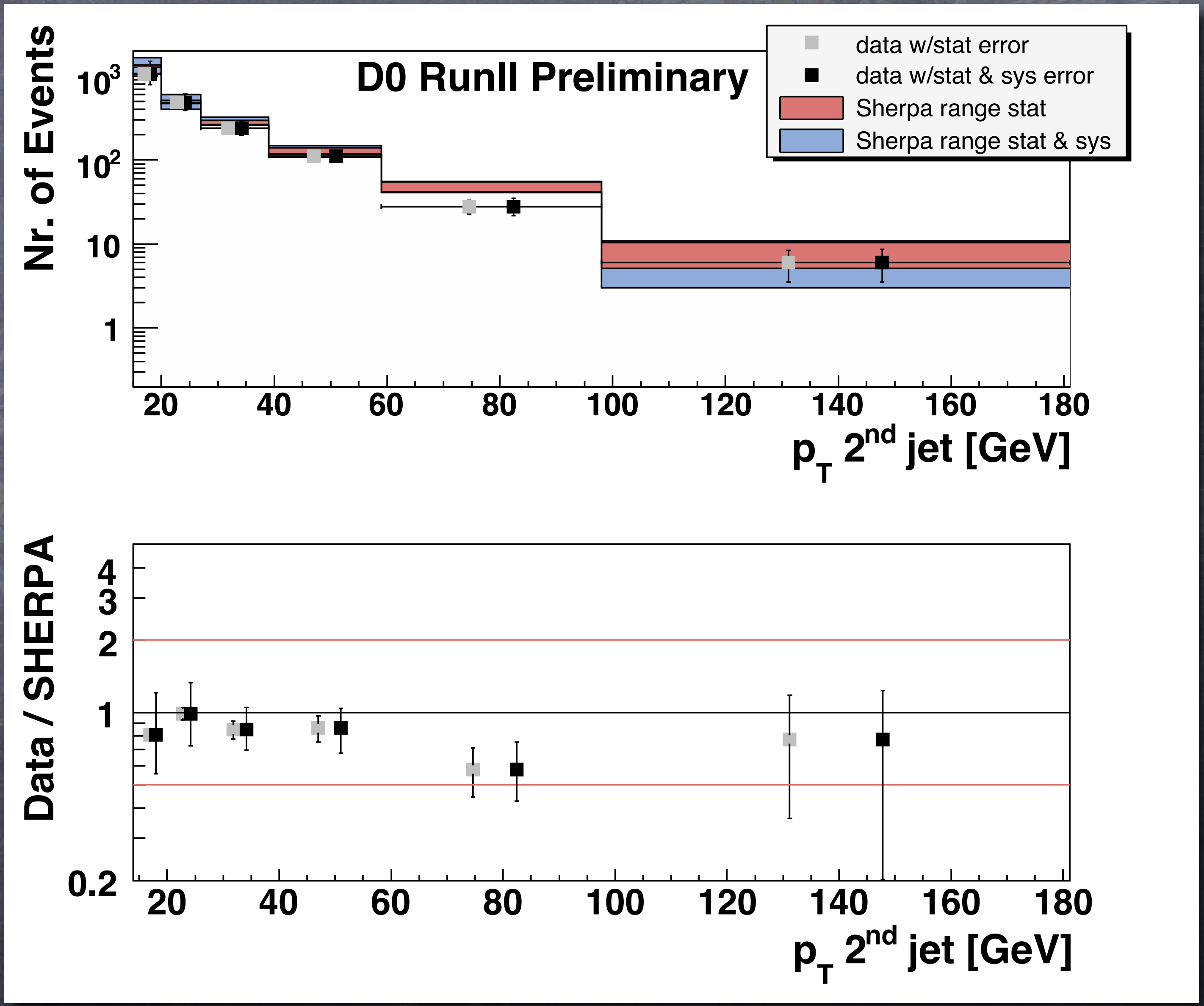
- $p_{T,\ jet\ 1}$, measured at DØ



V+jets production @ Tevatron

The DØ Collaboration, DØ Note 5066-CONF

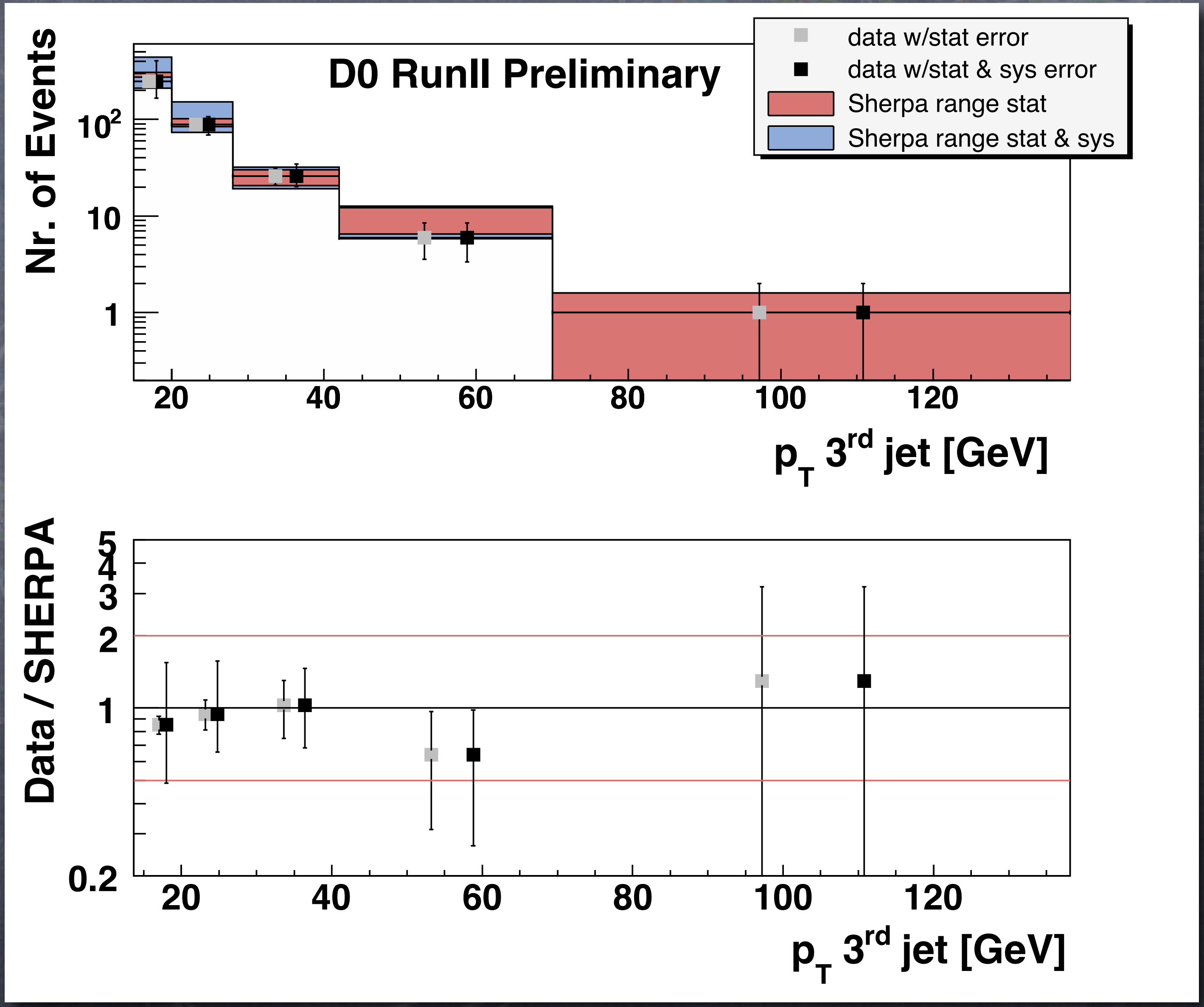
- $p_{T,\ jet\ 2}$, measured at DØ



V+jets production @ Tevatron

The DØ Collaboration, DØ Note 5066-CONF

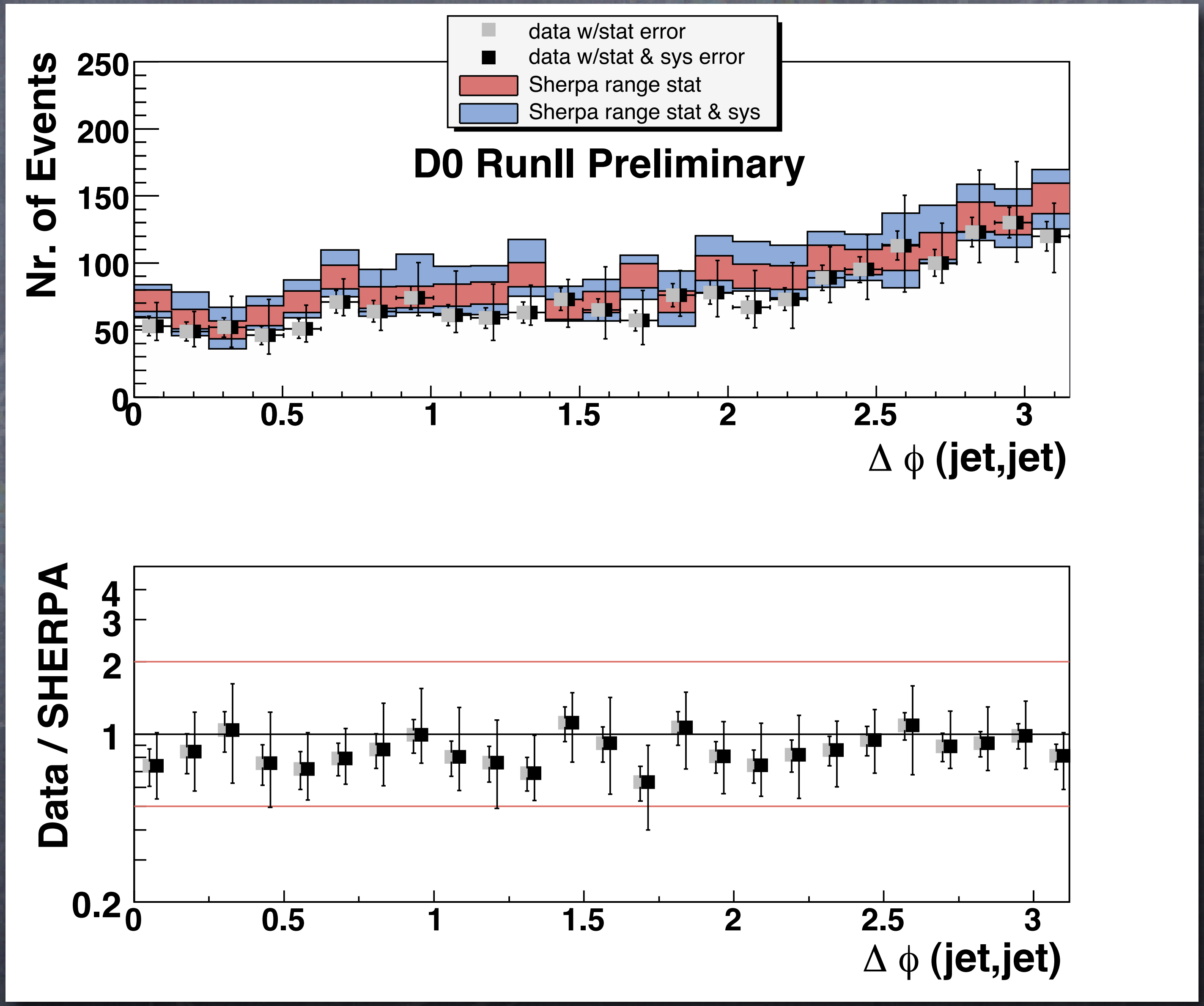
- $p_{T,\ jet\ 3}$, measured at DØ



V+jets production @ Tevatron

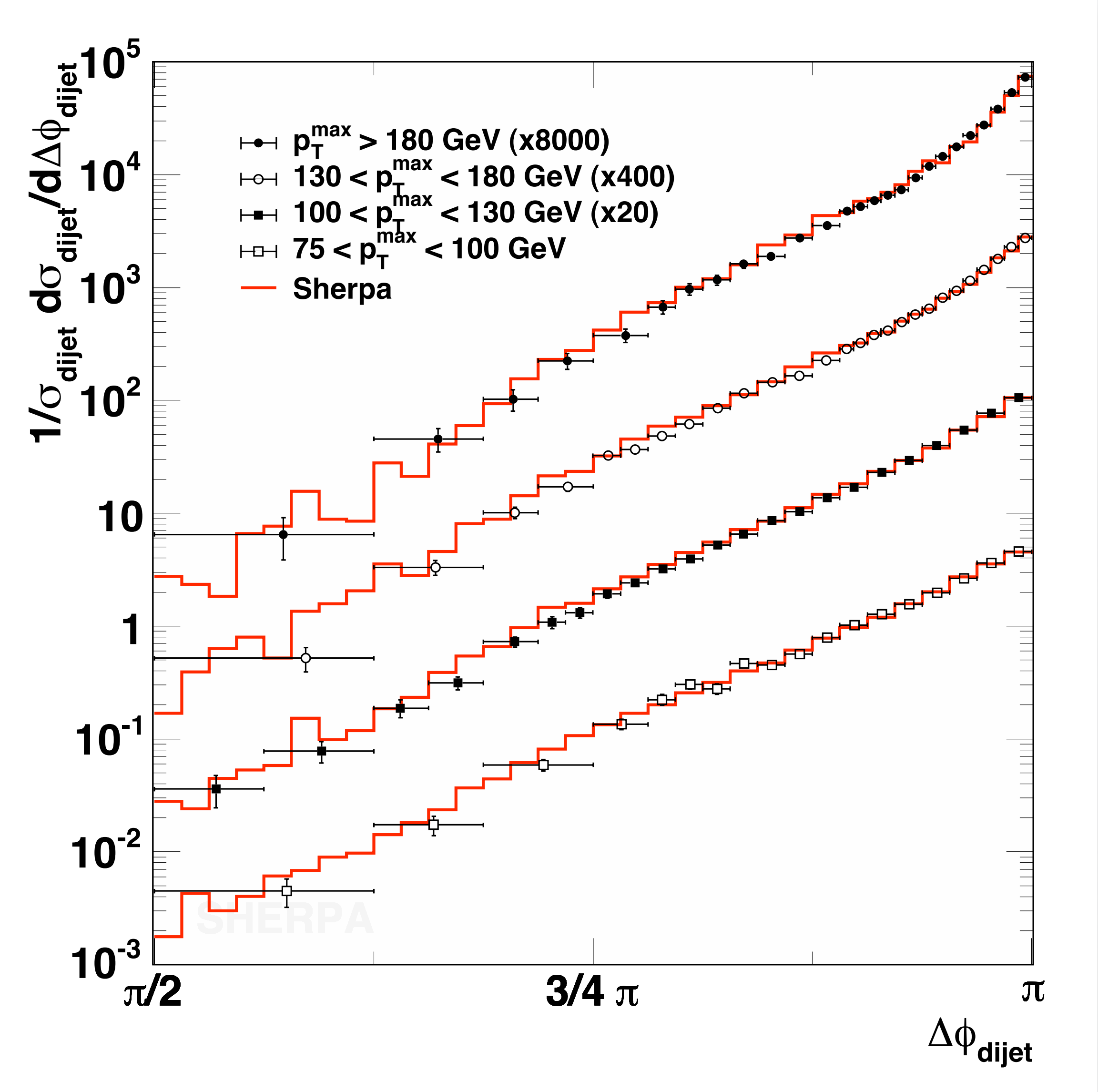
The DØ Collaboration, DØ Note 5066-CONF

- $\Delta\phi_{12}$, measured at DØ



Jet production @ Tevatron

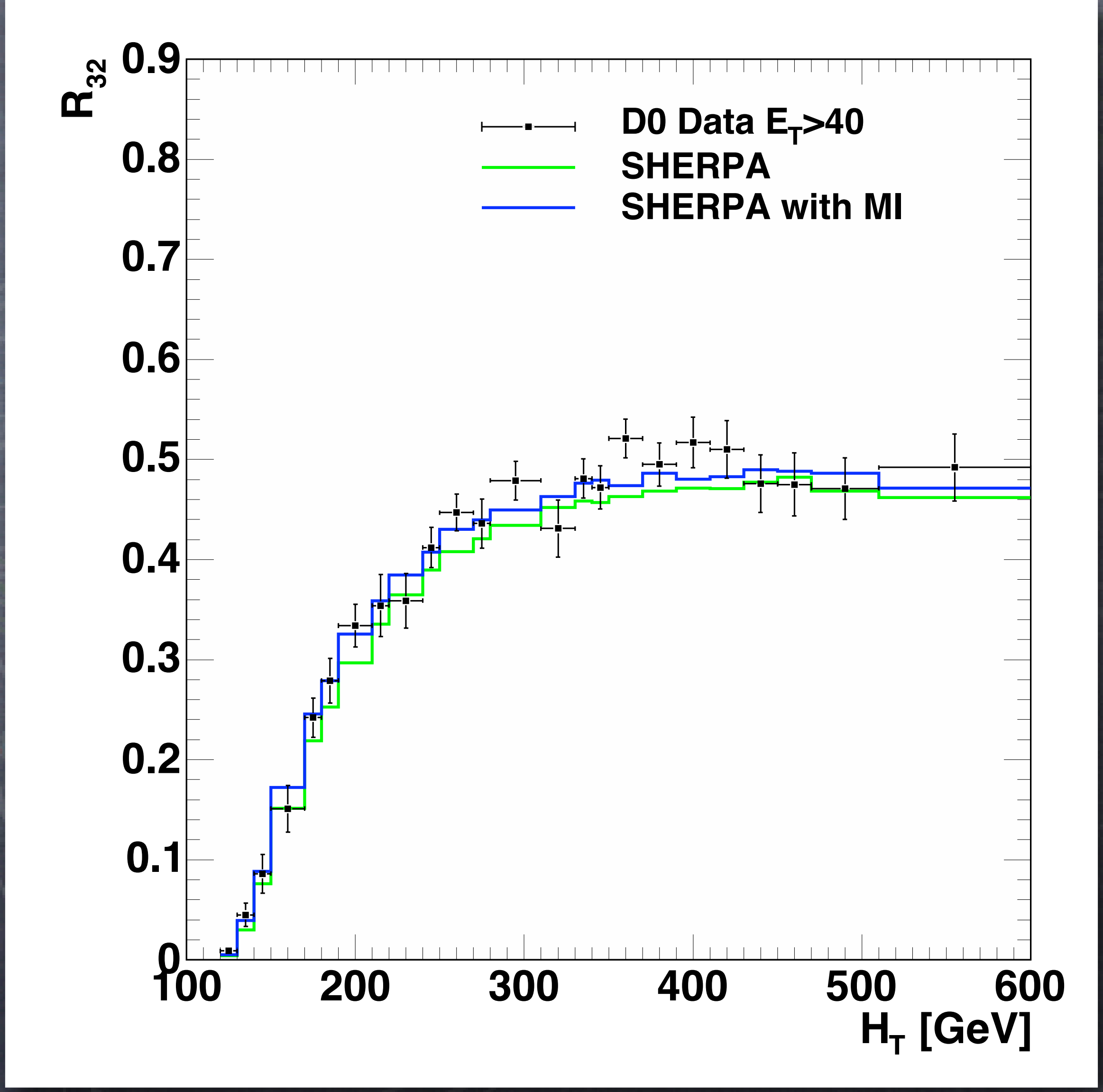
- Azimuthal dijet decorrelation in $p_{T,\,max}$ bins



Jet production @ Tevatron

- Jet rate ratio

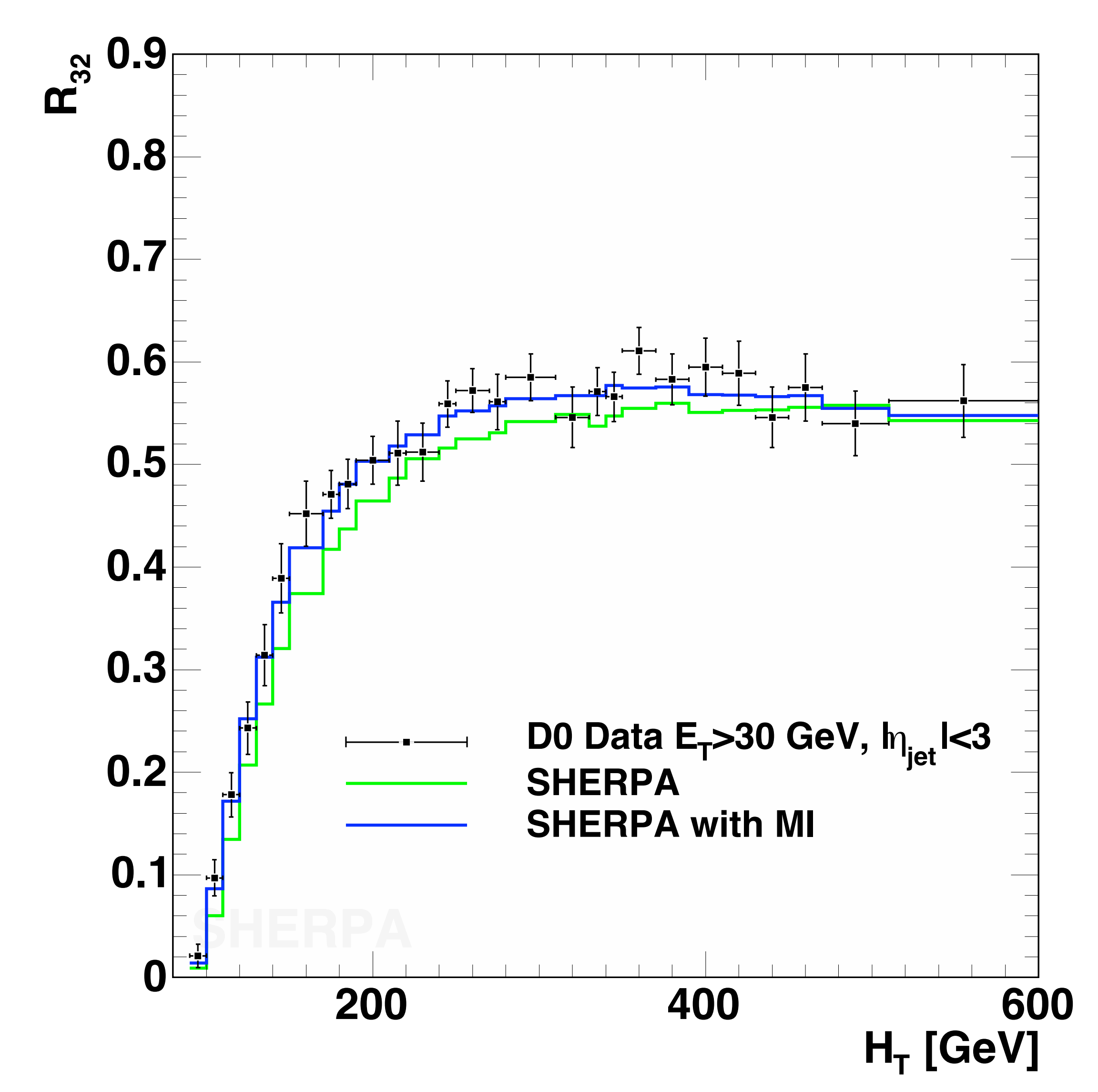
$$R_{32} = \frac{R_3}{R_2}$$



Jet production @ Tevatron

- Jet rate ratio

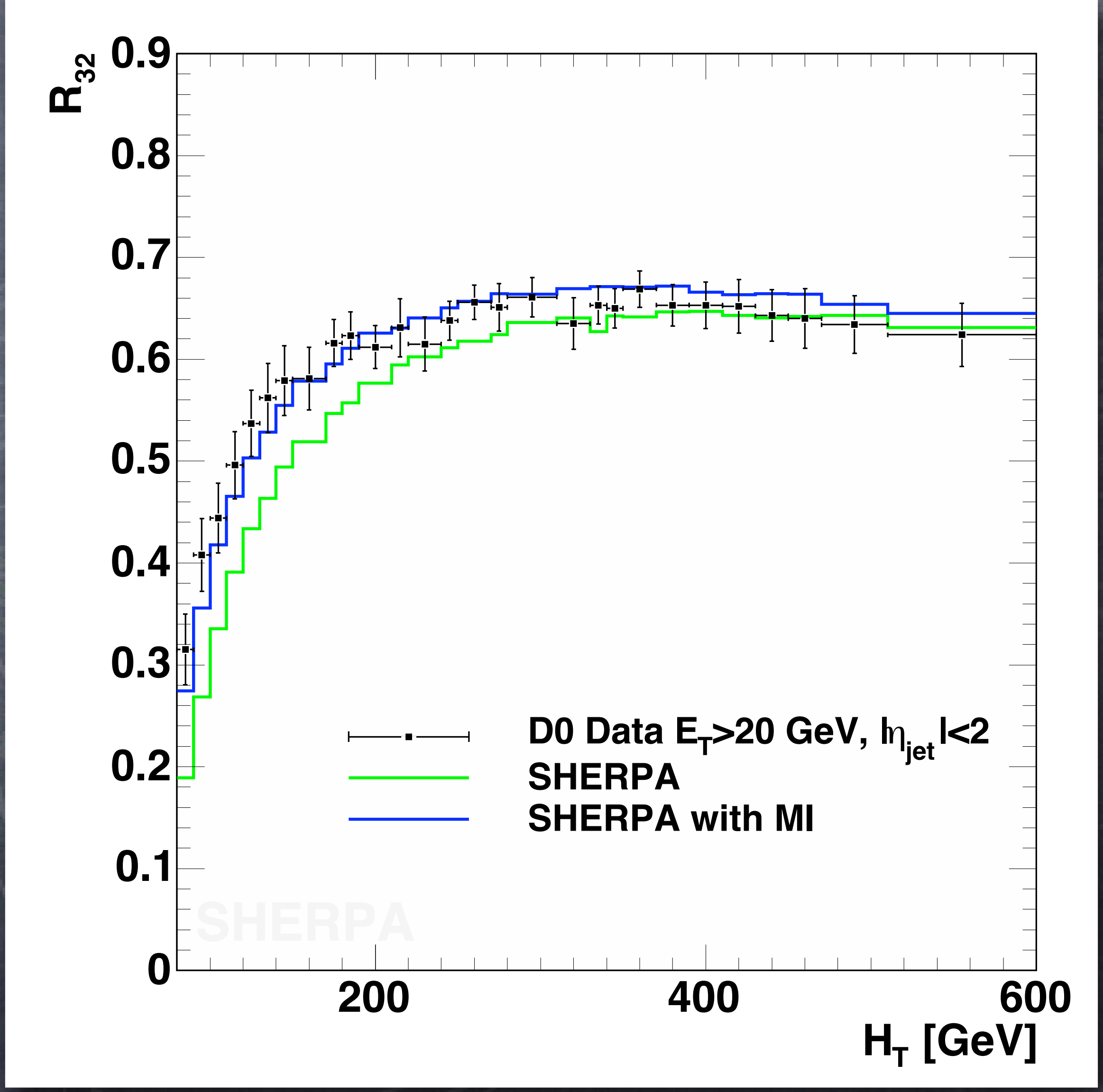
$$R_{32} = \frac{R_3}{R_2}$$



Jet production @ Tevatron

- Jet rate ratio

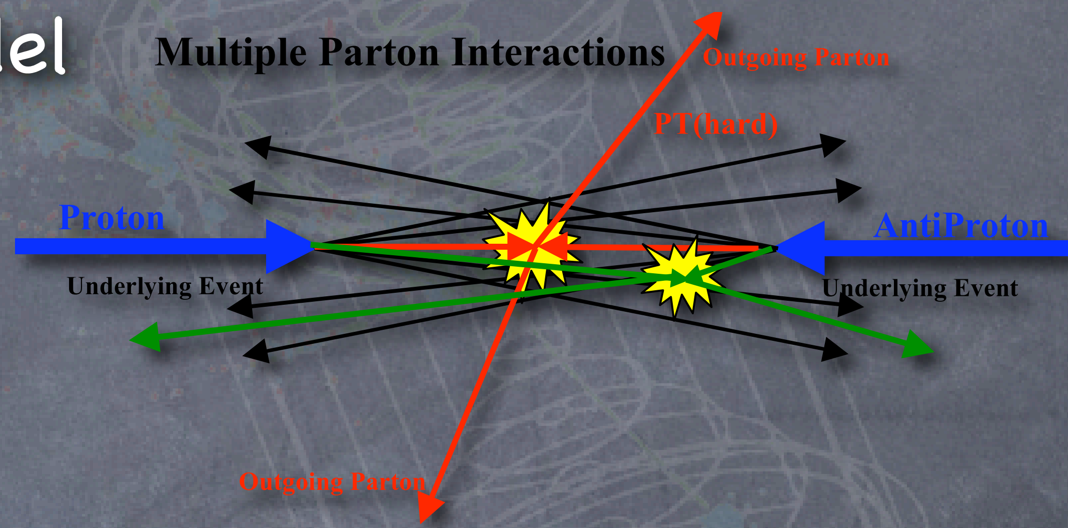
$$R_{32} = \frac{R_3}{R_2}$$



Multiple Interactions: AMISIC++

Features of Sherpas multiple interaction module

- Built according to PYTHIA model
  T. Sjöstrand & M. van Zijl,
  Phys. Rev. D36 (1987)
- Parton showers attached
  to the secondary interactions

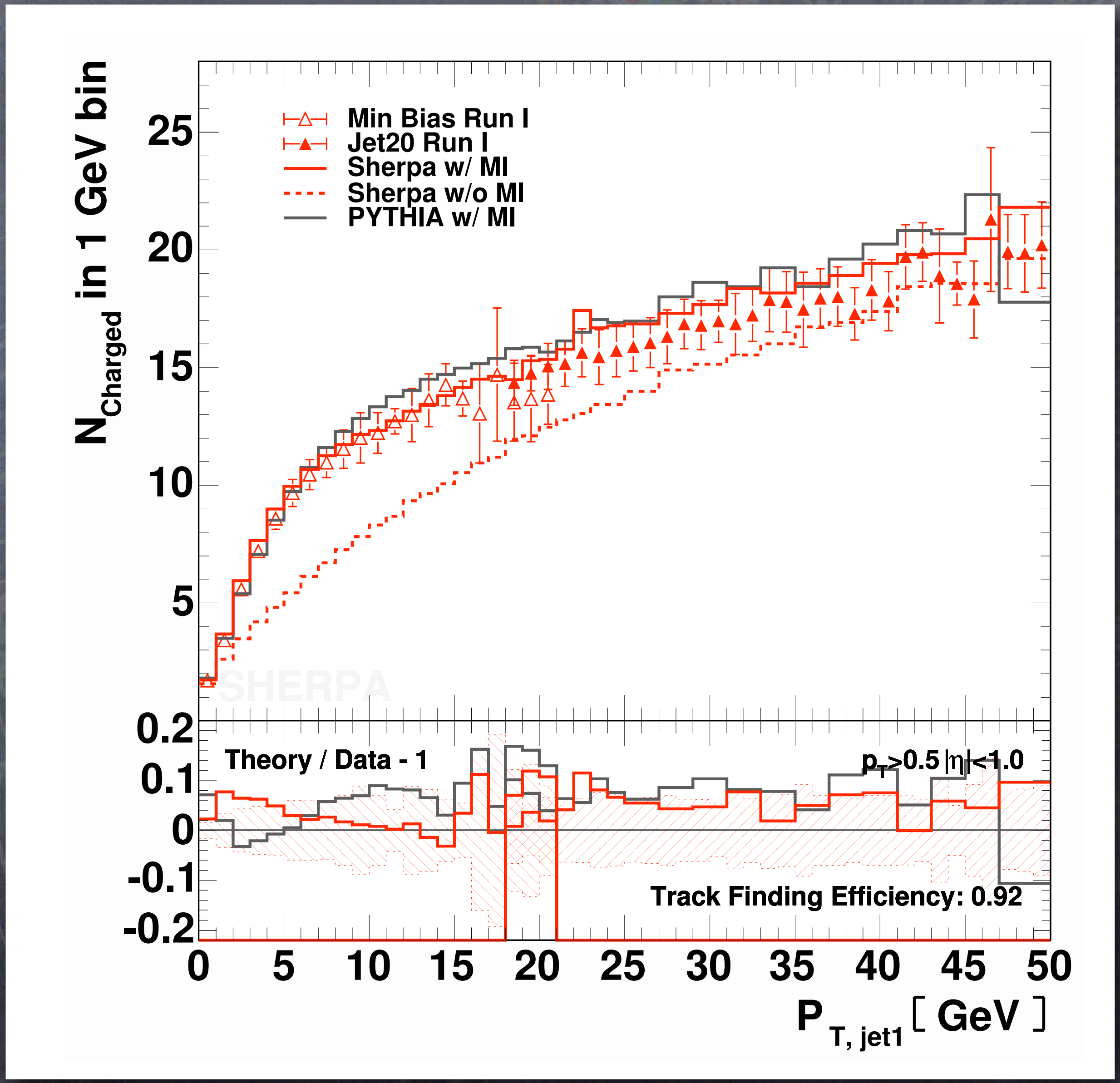


AMISIC++ set up to work with the CKKW matching

- Hard processes with FS multiplicity different from 2 require unique definition of starting scale for evolution μ_{MI}
- Sherpa algorithm (works for arbirary n-jet ME):
  - employ K_T algorithm to define 2→2 core process
  - set $\mu_{\rm MI}$ to $p_\perp$ of QCD partons from this process

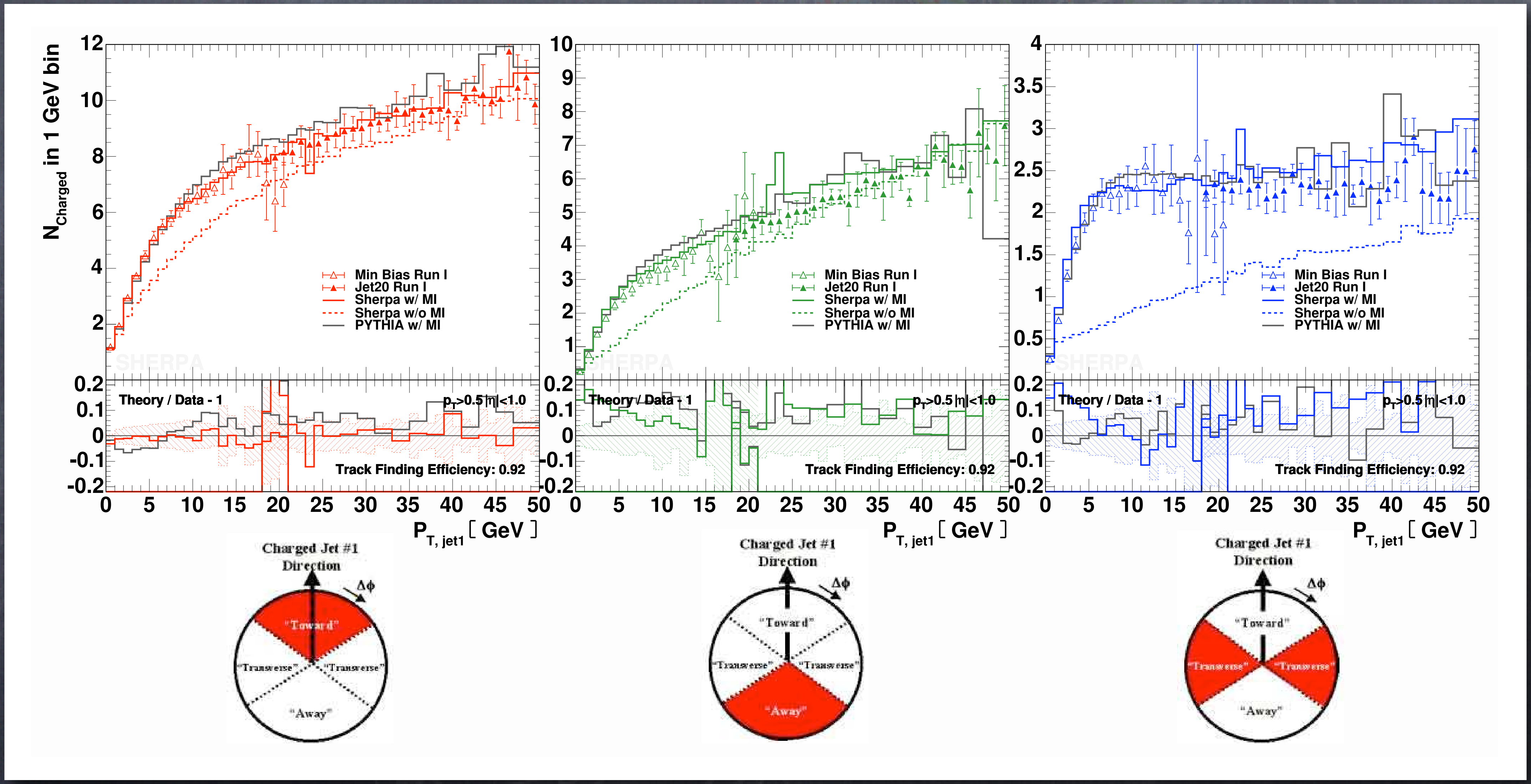
Underlying Events @ Tevatron

- $N_{charged}$ vs. $p_{T,\ jet1}$ in CTC



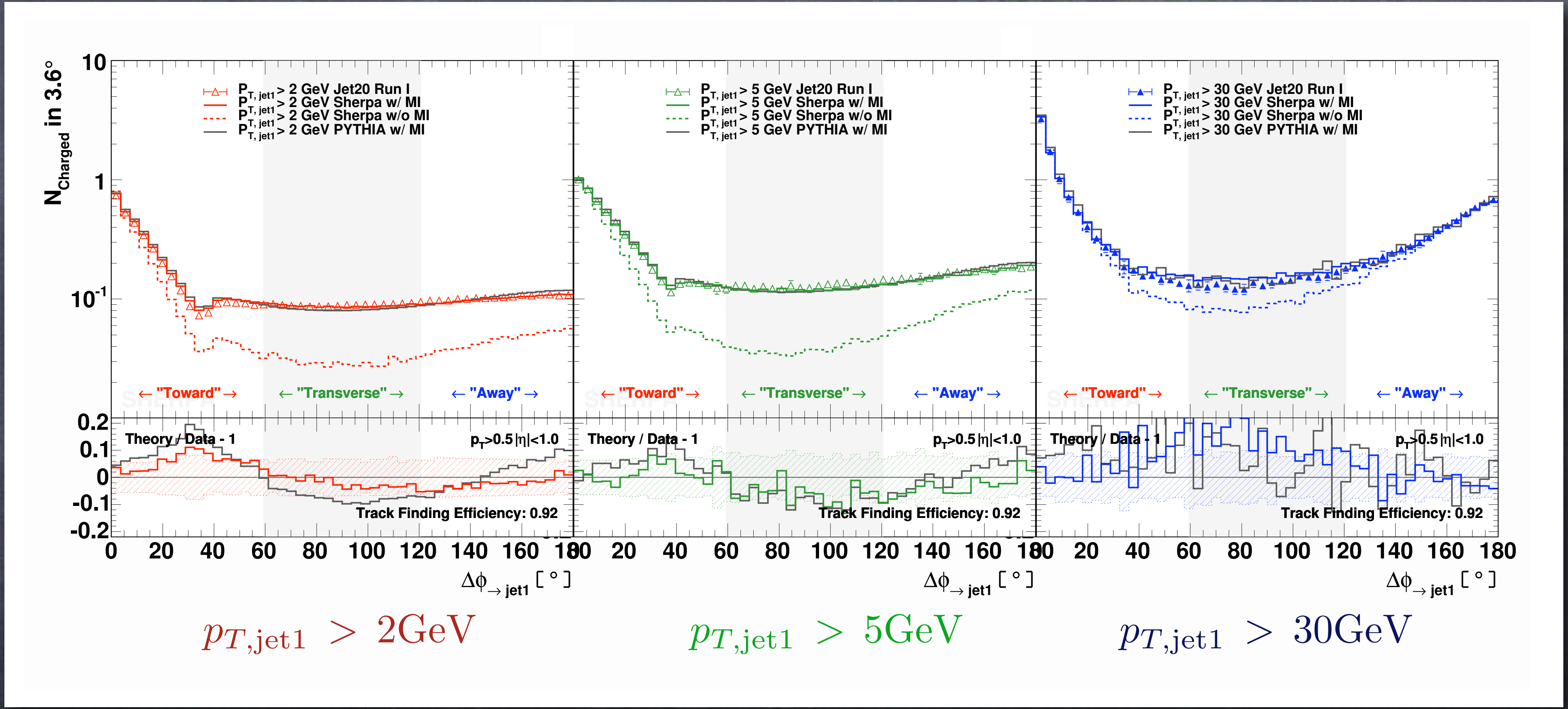
Underlying Events @ Tevatron

- $N_{charged}$ vs. $p_{T,\,jet1}$ in CTC
  in different regions w.r.t. leading jet



Underlying Events @ Tevatron

- $N_{charged}$ vs. $\Delta\varphi_{jet1}$ in CTC for different p_T of leading jet



Hadron and τ Decays: HADRONS++

Features of Sherpas new hadron decay module

- Full flexibility, all information read from parameter files
  (branching ratios, decay channels, form factors, integrators)
- Easy to extend with specific decay models

HADRONS++ extensively tested in τ decays

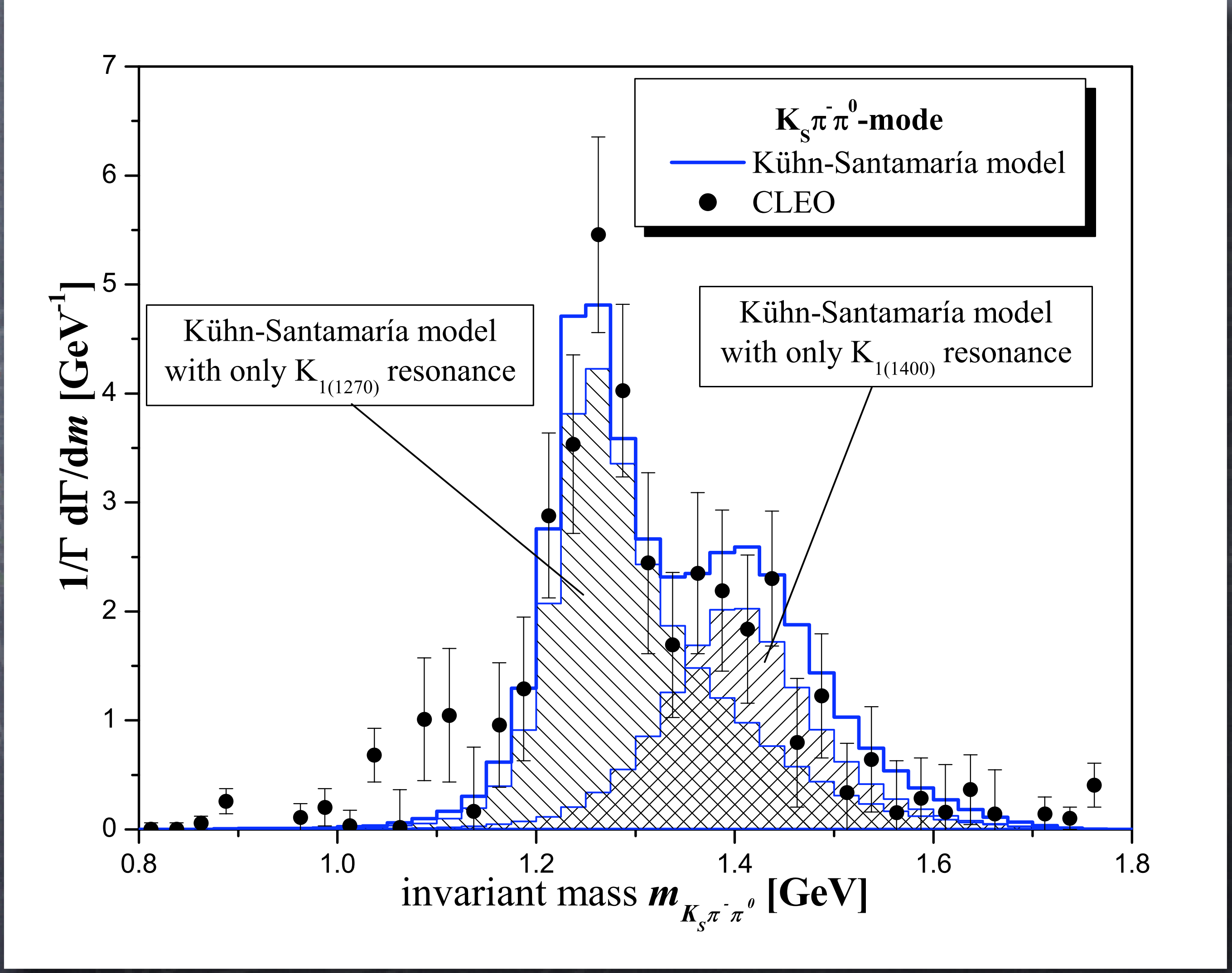
- Decay kinematics chosen according to
  Kühn-Santamaria model or resonance chiral theory
  (some rare decays to 5/6 h according to PS)

So far only few hadron decays

- Decays according to phase space for
  $\pi,\ \eta,\ \eta',\ \rho,\ K,\ K^*,\ \phi,\ \omega(782),\ a_2(1320),\ f_2(1270),\ f_2'(1525)$
- Currently extended to handle B / D decays

Hadron and τ Decays: HADRONS++

- Invariant mass spectrum in $\tau \rightarrow \mathbf{K}^{-}\pi^{+}\pi^{-}\nu_{\tau}$ (CLEO-CONF-94-23)



Conclusions and Outlook

New features:

- Revised SUSY sector, including SLHA interface
- τ and first hadron decays
- Decay chain treatment

To do list:

- Finalise alternative underlying event model
- Include & tune cluster fragmentation model
- Extend hadron decay package, special emphasis on Bs
- ...

Sherpa is a powerful tool to describe present-day Tevatron data and to study the extrapolation to LHC energies

Now the hands-on part ...

Basic Installation

Get your Sherpa tarball

- Download Sherpa α-1.0.7 from our website
  http://www.sherpa-mc.de
  Please download also the manual
- Unpack the distribution using
  tar –xzvf Sherpa-1.0.7.tar.gz

Compile the code

- PLEASE: Employ the installation script that comes with the distribution !
- It's easy: TOOLS/makeinstall –t

  To display more options of the script
  (e.g. incorporation of ROOT/CLHEP in the code)
  run TOOLS/makeinstall –h

All this has been done for you in advance ...

Setup

A Sherpa setup consists of several parameter files (plain ASCII)

- Analysis.dat
- Beam.dat
- Fragmentation.dat
- Hadron.dat
- ISR.dat
- Integration.dat
- Lund.dat
- ME.dat
- MI.dat
- Model.dat
- Particle.dat
- Processes.dat
- Run.dat
- Selector.dat
- Shower.dat

How to run such a setup ...

- Case 1: Run locally in the setup directory.
  (Convenient to add the binary path to your PATH)
  - Change to your setup: cd setup_dir/
  - Execute Sherpa: Sherpa

  Process-specific code is generated, run stops ...

  THIS IS NORMAL !!!
  - Compile the process specific code: ./makelibs
  - Execute Sherpa again: Sherpa

Setup

How to run a setup ...

- Case 2: Run from outside the setup directory.
  (e.g. from the Sherpa binary path, using ./Sherpa)
  - Execute Sherpa with path information:
    ./Sherpa PATH=setup_dir/
    Process-specific code is generated, run stops ...
    THIS IS NORMAL AGAIN !!!
  - Go to the setup directory and compile code:
    cd setup_dir/ && ./makelibs
  - Return and execute Sherpa again

Advanced usage

- For time-consuming processes,
  you might want to reuse integration results:
  - Create corresponding directory: mkdir res_dir/
  - Use it: ./Sherpa PATH=setup_dir/ RESULTS_DIRECTORY=res_dir/

Setup

Be aware that changes to the physics parameters can render your process-specific code, generated by AMEGIC++ and your integration results useless, unreliable or even wrong !

In this case, remove the code ... rm -rf setup_dir/Process/
and the integration results ... rm -rf res_dir/*
and start afresh ...

In any case, you don't know what to do, drop us an email

info@sherpa-mc.de