



Progress in CTEQ modified LO pdf's + some other stuff*

J. Huston

Michigan State University

and

IPPP Durham

*apologies for incompleteness as I lost a lot in a disk crash this last week



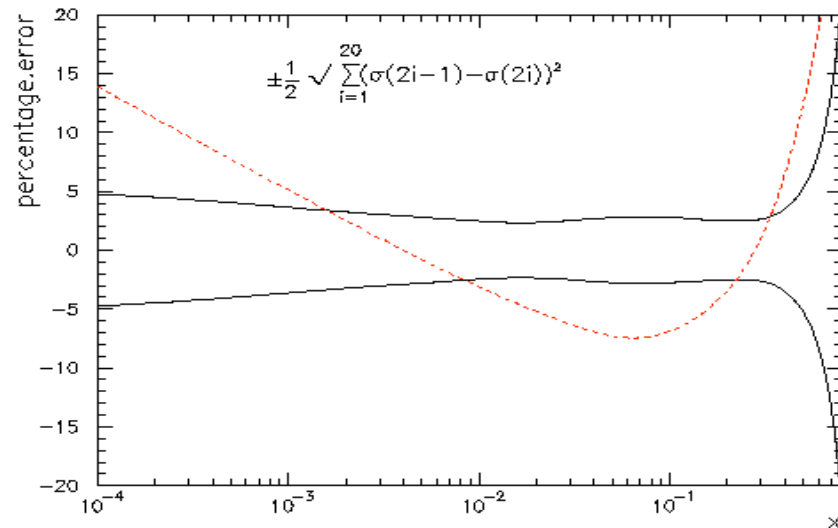
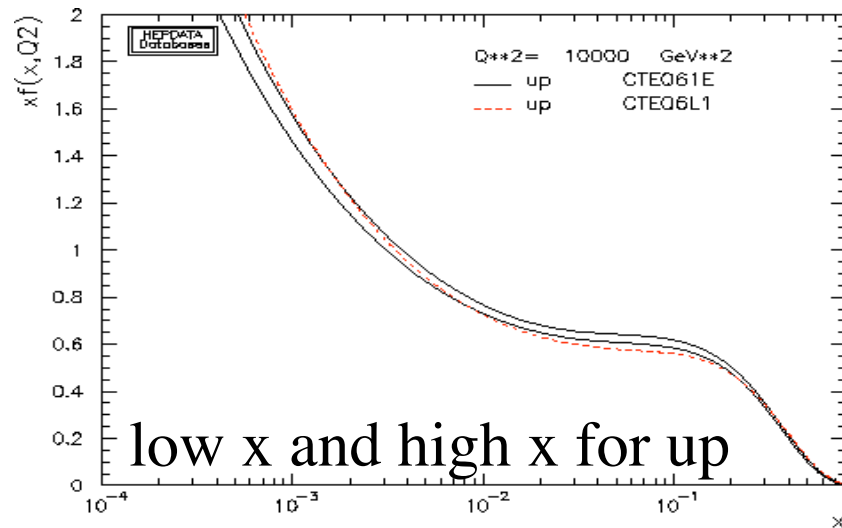
-
- H-L Lai, S Mrenna, P Nadolsky, J Pumplin, D Stump, WK Tung, CP Yuan
 - Michigan
 - State
 - Taiwan
 - Washington



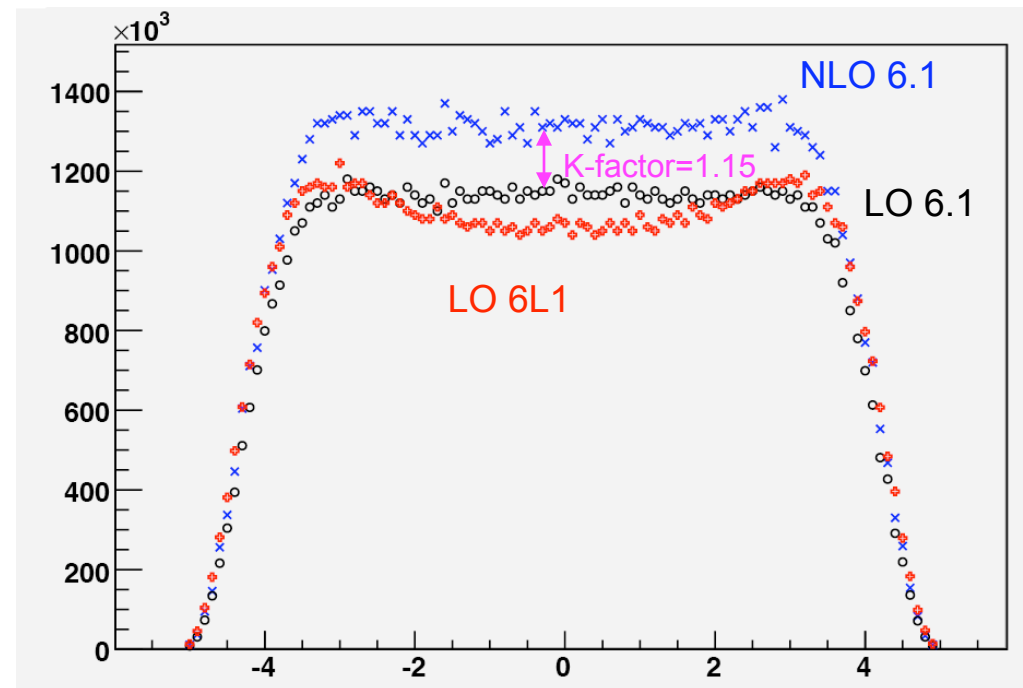
- What about pdf's for parton shower Monte Carlos?
 - ◆ standard has been to use LO pdf's, most commonly CTEQ5L/CTEQ6L, in Pythia, Herwig, Sherpa, ALPGEN/Madgraph+...
- ...but
 - ◆ LO pdf's can create cross sections/acceptances that differ in both shape and normalization from NLO due to influence of HERA data
 - ◆ ...and are often outside NLO error bands
 - ◆ experimenters use the NLO error pdf's in combination with the central LO pdf even with this mis-match
 - ▲ causes an error in pdf re-weighting
 - ◆ predictions for inclusive observables from LO matrix elements for many of the collider processes that we want to calculate are not so different from those from NLO matrix elements (aside from a reasonably constant K-factor)
- ...but
 - ◆ we *like* the low x behavior of LO pdf's and rely upon them for our models of the underlying event at the Tevatron and its extrapolation to the LHC
- thus, the need for modified LO pdf's

CTEQ

Where are the differences between LO and NLO partons?



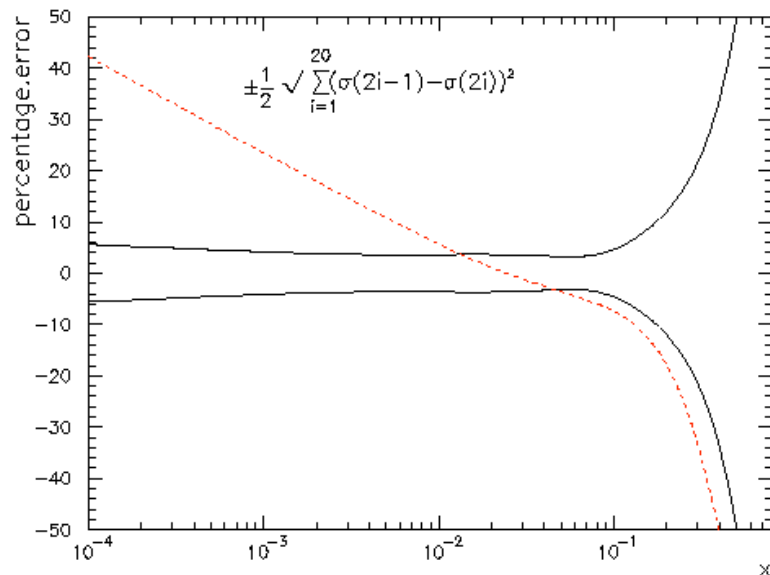
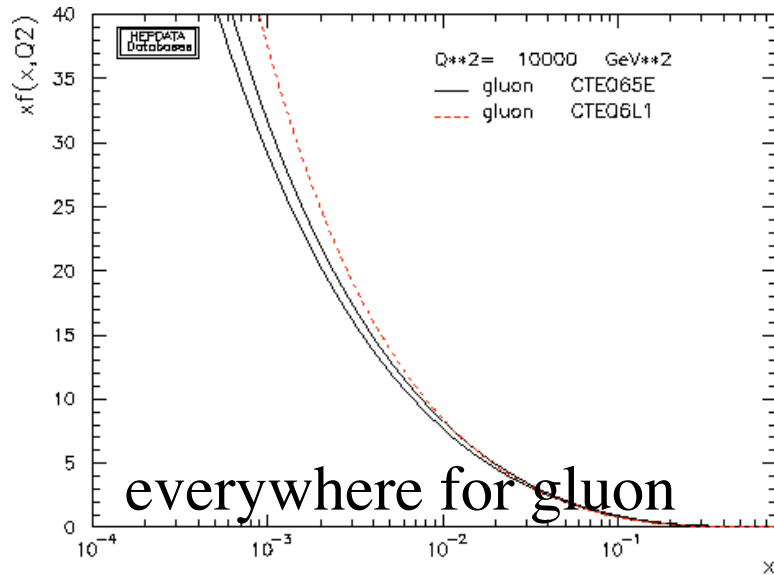
W^+ rapidity distribution at LHC



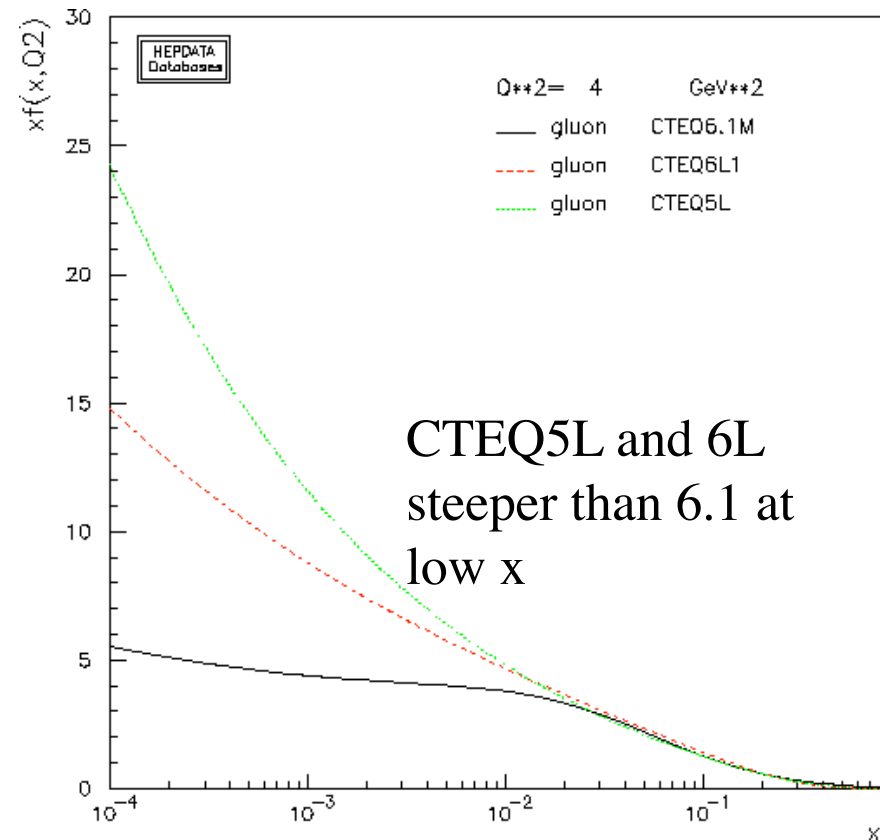
y_{W^+}

For example, the shape of the W^+ rapidity distribution is significantly different than the NLO result if the LO pdf is used, but very similar if the NLO pdf is used.

Where are the differences?



• at low Q



CTEQ talking points

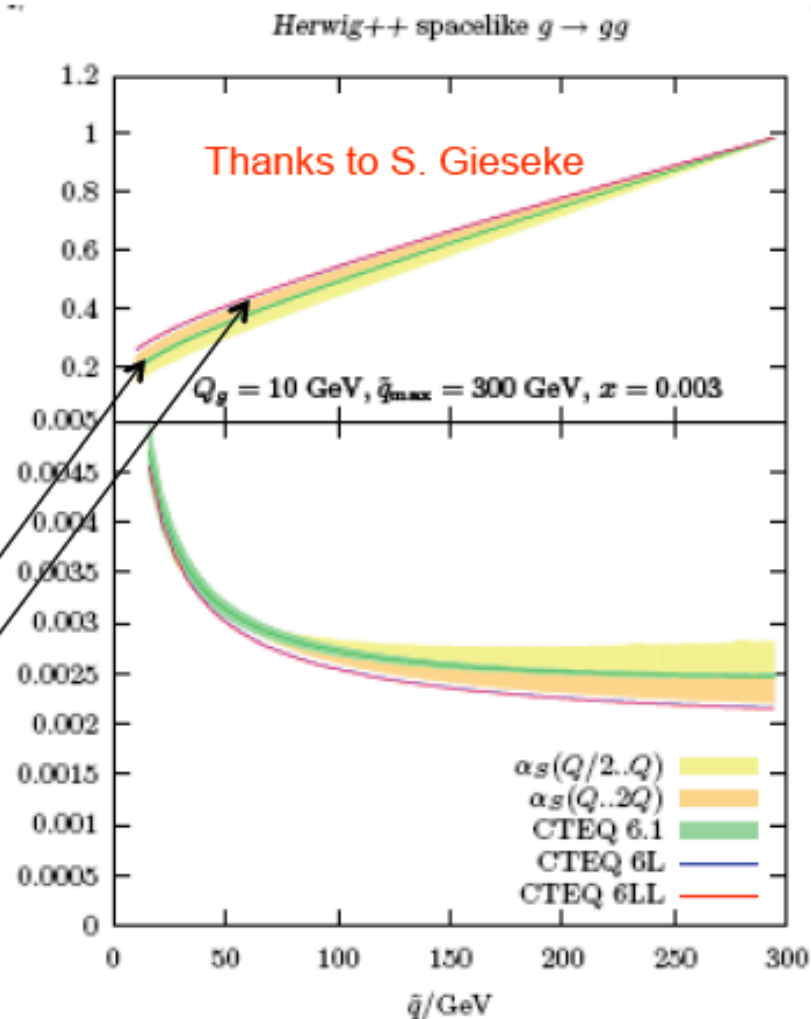


- LO* pdf's should behave as LO as $x \rightarrow 0$; as close to NLO as possible as $x \rightarrow 1$
- LO* pdf's should be universal, i.e. results should be reasonable run on any platform with nominal physics scales
- It should be possible to produce error pdf's with
 - ◆ similar Sudakov form factors
 - ◆ similar UE
 - ◆ so pdf re-weighting makes sense
- LO* pdf's should describe underlying event at Tevatron with a tune similar to CTEQ6L (for convenience) and extrapolate to a *reasonable* UE at the LHC

Sudakov form factors



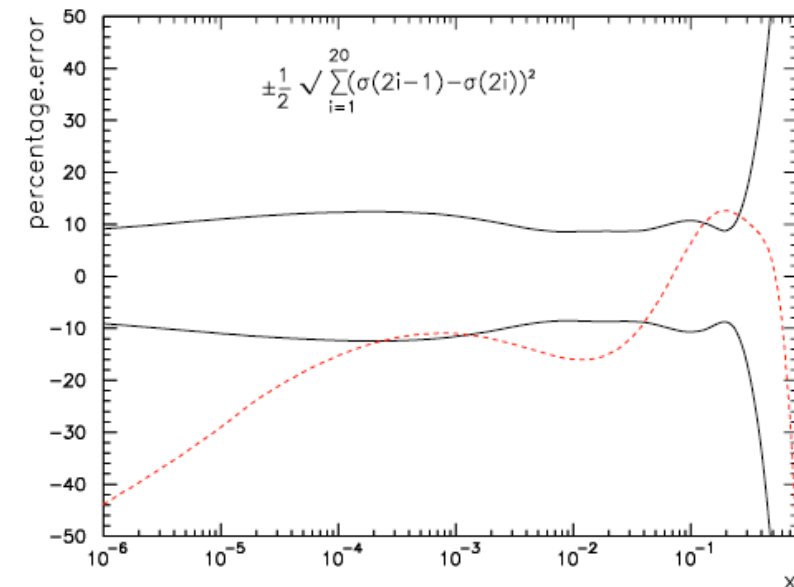
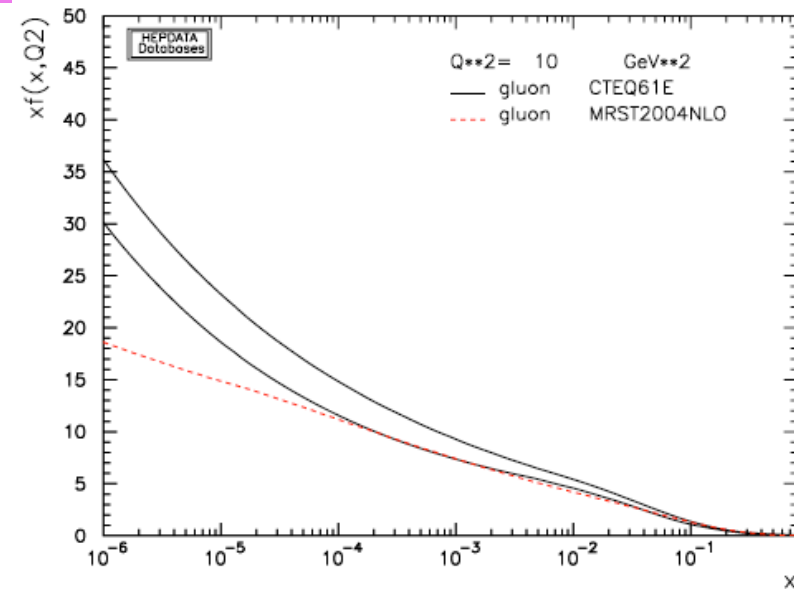
- Sudakov form factors form the basis for parton showers
- Typically at both the Tevatron and LHC, MC events are generated with a LO pdf and then pdf uncertainty is evaluated by performing a pdf re-weighting using the NLO error pdf's
- Works if Sudakov is the same for the LO pdf and the NLO error pdf's
- NLO pdf error band very small
- LO Sudakov outside this error band, so ISR not correct for re-weighted events generated using LO pdf
- Need to generate MC events and to evaluate pdf's with same order



Underlying event



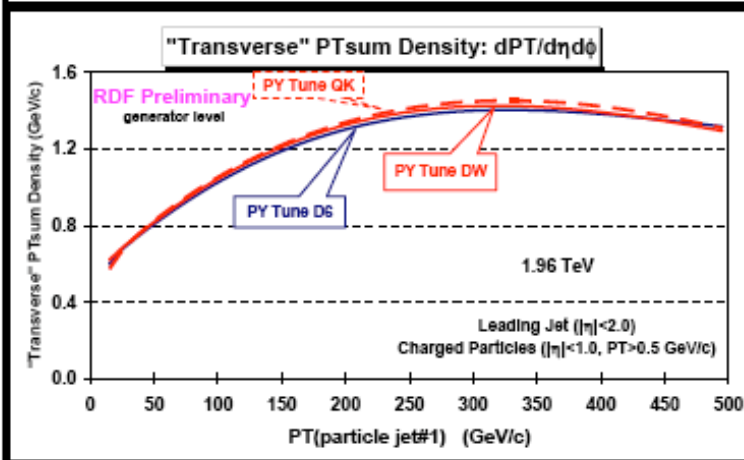
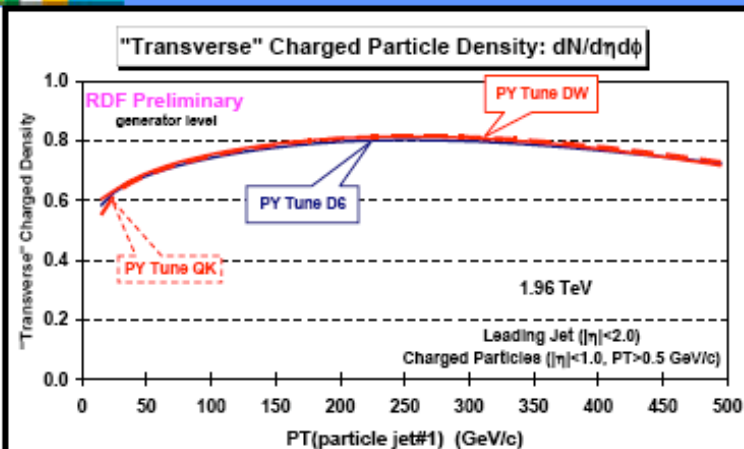
- hep-ph/0803.3633
(Mike et al)
demonstrated that
UE variation within
NLO error set of
CTEQ6.1 was small
(LO gluon behavior
similar for all 41
pdf's)
- May not be true for
larger variation of low
x gluon





- Tune A (and derivatives) obtained with CTEQ5L but 6L works just as well

New PYTHIA 6.2 Tunes



	1.96 TeV		14 TeV	
	$P_{T0}(\text{MPI})$ GeV	$\sigma(\text{MPI})$ mb	$P_{T0}(\text{MPI})$ GeV	$\sigma(\text{MPI})$ mb
Tune DW	1.9409	351.7	3.1730	549.2
Tune DWT	1.9409	351.7	2.6091	829.1
ATLAS	2.0046	324.5	2.7457	768.0
Tune D6	1.8387	306.3	3.0059	546.1
Tune D6T	1.8387	306.3	2.5184	786.5
Tune QK	1.9409	259.5	3.1730	422.0
Tune QKT	1.9409	259.5	2.6091	588.0

- ➔ Average charged particle density and PTsum density in the "transverse" region ($p_T > 0.5$ GeV/c, $|\eta| < 1$) versus $P_T(\text{jet\#1})$ at 1.96 TeV for **PY Tune DW**, **Tune D6**, and **Tune QK**.

CTEQ techniques



- Include in LO* fit (weighted) pseudo-data for characteristic LHC processes produced using CTEQ6.6 NLO pdf's with NLO matrix elements (using MCFM), along with full CTEQ6.6 dataset (2885 points)

- ◆ low mass bB
 - ▲ fix low x gluon for UE
- ◆ tT over full mass range
 - ▲ higher x gluon
- ◆ W^+, W^-, Z^0 rapidity distributions
 - ▲ quark distributions
- ◆ $gg \rightarrow H$ (120 GeV) rapidity distribution

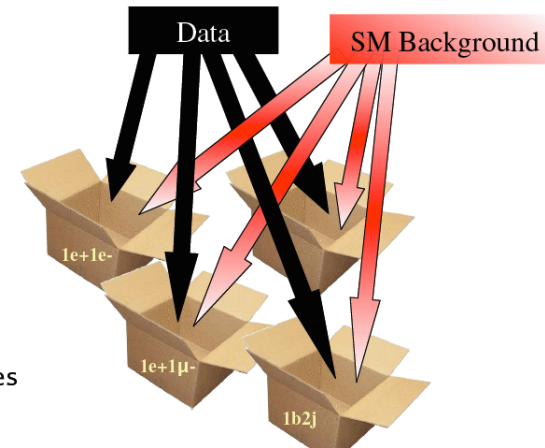
Choices

- Use of 2-loop or 1-loop α_s
 - ◆ MC preference for 2-loop?
- Fixed momentum sum rule, or not
 - ◆ re-arrange momentum within proton and/or add extra momentum
 - ◆ extra momentum appreciated by some of pseudo-data sets but not others and may lose some useful correlations
- Fix pseudo-data normalizations to K-factors expected from higher order corrections, or let float
- Scale variation within reasonable range for fine-tuning of agreement with pseudo-data
 - ◆ for example, let vector boson scale vary from $0.5 m_B$ to $2.0 m_B$
- May provide pdf's with several of these options for user
- That point has not yet been reached on the decision tree



- CTEQ global fitting package with CTEQ6.6 global data set plus NLO pseudo-data
- MCFM
- VISTA
 - a tool developed (by Steve Mrenna and Bruce Knuteson) to perform a global analysis of collider data with an eye to discovering new phenomena
 - but which turns out also to be useful to debug predictions of the standard model at the Tevatron and LHC
 - and to examine the effects of parton showering and of UE

Vista: partitioning into final states



Look in many final states

Implement the Standard Model

Final State	Chi2	data	blk	
1b3j1pmiss_sumPt400+ [73]	9.0	451	374.5 +- 18	(pyth_jj_200 =
2b1e+2j [-]	8.0	15	6.5 +- 1.9	(tstep0z = 2.3
2j_sumPt0-400 [161]	6.0	69704	67013.6 +- 1171.2	(pyth_jj_018 =
2j2mu+1pmiss [-]	-5.0	2	12.2 +- 3	(mad_mu+mu-jj
1b2e+2j [-]	5.0	9	3.9 +- 1.5	(mrenna_e+e-jj
1j1p1pmiss [5]	4.0	2591	2470.1 +- 37.7	(pyth_pj_045 =
2j1mu+1ph [-]	4.0	11	11.2 +- 2.2	(mrenna_mu+mu-
1e+1j1mu+ [-]	4.0	13	6.8 +- 2.1	(zstep51 = 3.4
1e+2j1ph [-]	4.0	31	20.9 +- 2.7	(mad_aa_jj = 6.
3j2mu+ [-]	4.0	34	23.2 +- 2.7	(mrenna_mu+mu-
2b2j1pmiss_sumPt400+ [-]	-3.0	17	30.4 +- 4.2	(pyth_jj_200 =
1b2j_sumPt400+ [229]	-3.0	4669	4518.6 +- 72.7	(pyth_jj_200 =
4j_sumPt0-400 [253]	-3.0	2611	2736.9 +- 42.3	(pyth_jj_040 =
2b1j1p1pmiss [-]	3.0	6	2.7 +- 1.5	(pyth_jj_200 =
1b1j1mu+ [-]	3.0	67	53.8 +- 4.3	(pyth_jj_018 =
1j1ph [277]	3.0	31738	31149.8 +- 352.1	(pyth_pj_045 =
1e+1mu+ [-]	3.0	66	53.5 +- 3.2	(zstep51 = 36.6
4j1mu+ [-]	3.0	73	61.3 +- 2.6	(pyth_jj_040 =
5j [269]	3.0	448	906 +- 14.5	(pyth_jj_040 =
1b5j [-]	3.0	8	8.9 +- 1.7	(pyth_jj_060 =
1b1j1pmiss_sumPt0-400 [-]	2.0	120	104 +- 7.2	(pyth_jj_040 =
2j1pmiss_sumPt0-400 [37]	2.0	2381	2281.2 +- 73.9	(pyth_jj_018 =

...

Have a measure of agreement



Sometimes it is useful to define a K-factor (NLO/LO). Note the value of the K-factor depends critically on its definition. K-factors at LHC (mostly) similar to those at Tevatron.

CHS

Table 1. K -factors for various processes at the Tevatron and the LHC, calculated using a selection of input parameters. In all cases, the CTEQ6M PDF set is used at NLO. \mathcal{K} uses the CTEQ6L1 set at leading order, whilst \mathcal{K}' uses the same set, CTEQ6M, as at NLO. Jets satisfy the requirements $p_T > 15$ GeV and $|\eta| < 2.5$ (5.0) at the Tevatron (LHC). In the $W + 2$ jet process the jets are separated by $\Delta R > 0.52$, whilst the weak boson fusion (WBF) calculations are performed for a Higgs of mass 120 GeV.

Process	Typical scales		Tevatron K-factor			LHC K-factor		
	μ_0	μ_1	$\mathcal{K}(\mu_0)$	$\mathcal{K}(\mu_1)$	$\mathcal{K}'(\mu_0)$	$\mathcal{K}(\mu_0)$	$\mathcal{K}(\mu_1)$	$\mathcal{K}'(\mu_0)$
W	m_W	$2m_W$	1.33	1.31	1.21	1.15	1.05	1.15
$W + 1$ jet	m_W	$\langle p_T^{\text{jet}} \rangle$	1.42	1.20	1.43	1.21	1.32	1.42
$W + 2$ jets	m_W	$\langle p_T^{\text{jet}} \rangle$	1.16	0.91	1.29	0.89	0.88	1.10
$t\bar{t}$	m_t	$2m_t$	1.08	1.31	1.24	1.40	1.59	1.48
$b\bar{b}$	m_b	$2m_b$	1.20	1.21	2.10	0.98	0.84	2.51
Higgs via WBF	m_H	$\langle p_T^{\text{jet}} \rangle$	1.07	0.97	1.07	1.23	1.34	1.09

K-factors may differ from unity because of new subprocesses/ contributions at higher order and/or differences between LO and NLO pdf's

Higgs + 1 jet

1.42

Higgs + 2 jets

1.15

tT + 1 jet

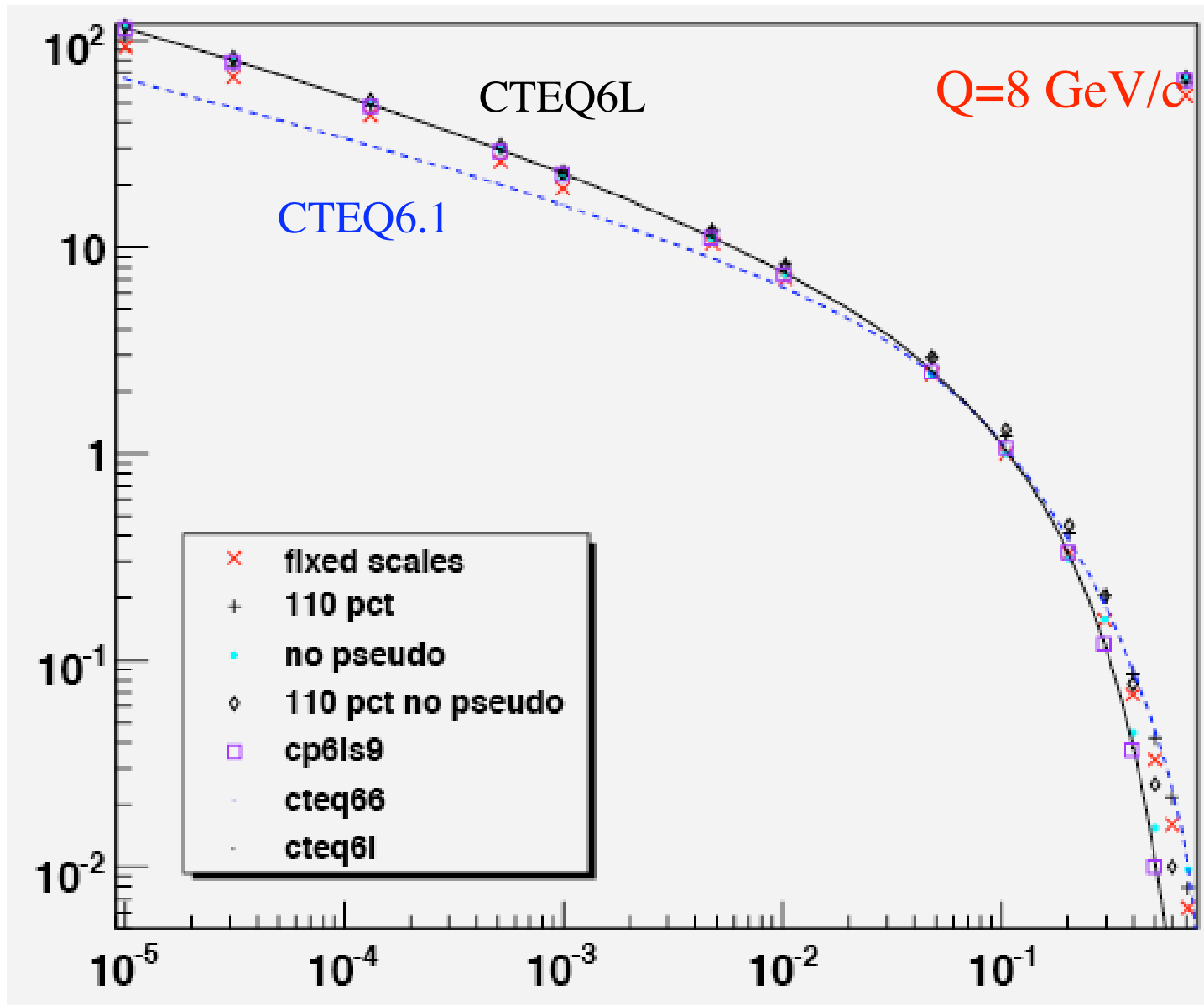
1.191.371.260.971.291.10

Some observations



- Pseudo-data has conflicts with global data set
 - ◆ that's the motivation of the modified pdf's
- Requiring better fit to pseudo-data increases chisquare of LO fit to global data set (although this is not the primary concern; the fit to the pseudo-data is)
 - ◆ χ^2 improves with α_s free in fit
 - ◆ χ^2 improves with momentum sum rule free
 - ▲ prefers more momentum, smaller α_s
 - ▲ normalization of pseudo-data (needed K-factor) gets closer to 1
 - ▲ still some conflicts with DIS data that don't prefer more momentum
 - ◆ χ^2 typically improves if K-factors can vary from values given in previous slide

Example: gluon distribution

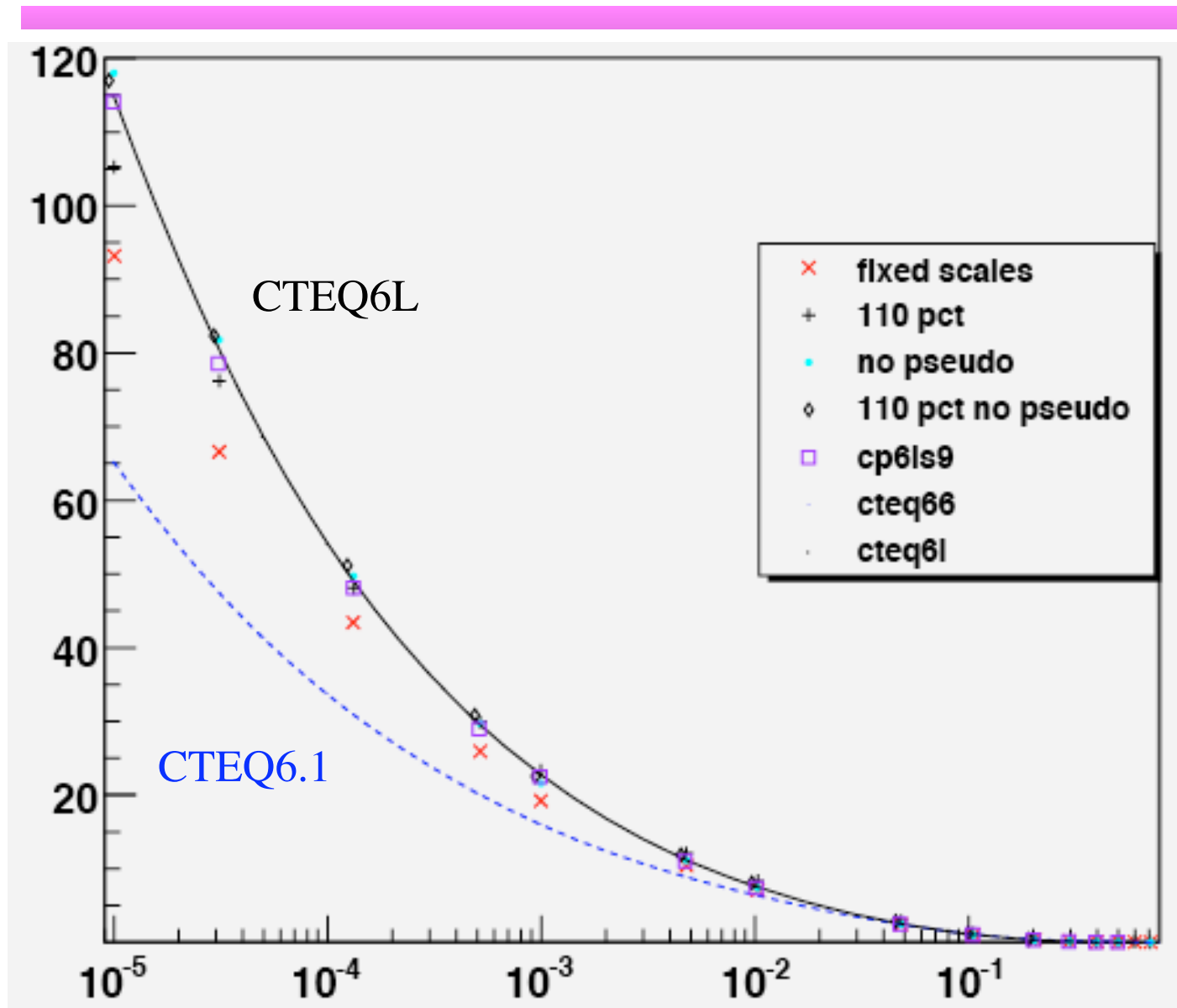


- Candidate pdf titled *fixed scales* tries to fit pseudo-data
- Larger than CTEQ6L at high x , but smaller at low x
- With *110%* momentum in proton, gluon is larger at high x
- Including the pseudo-data in the fit increases the high x gluon even more



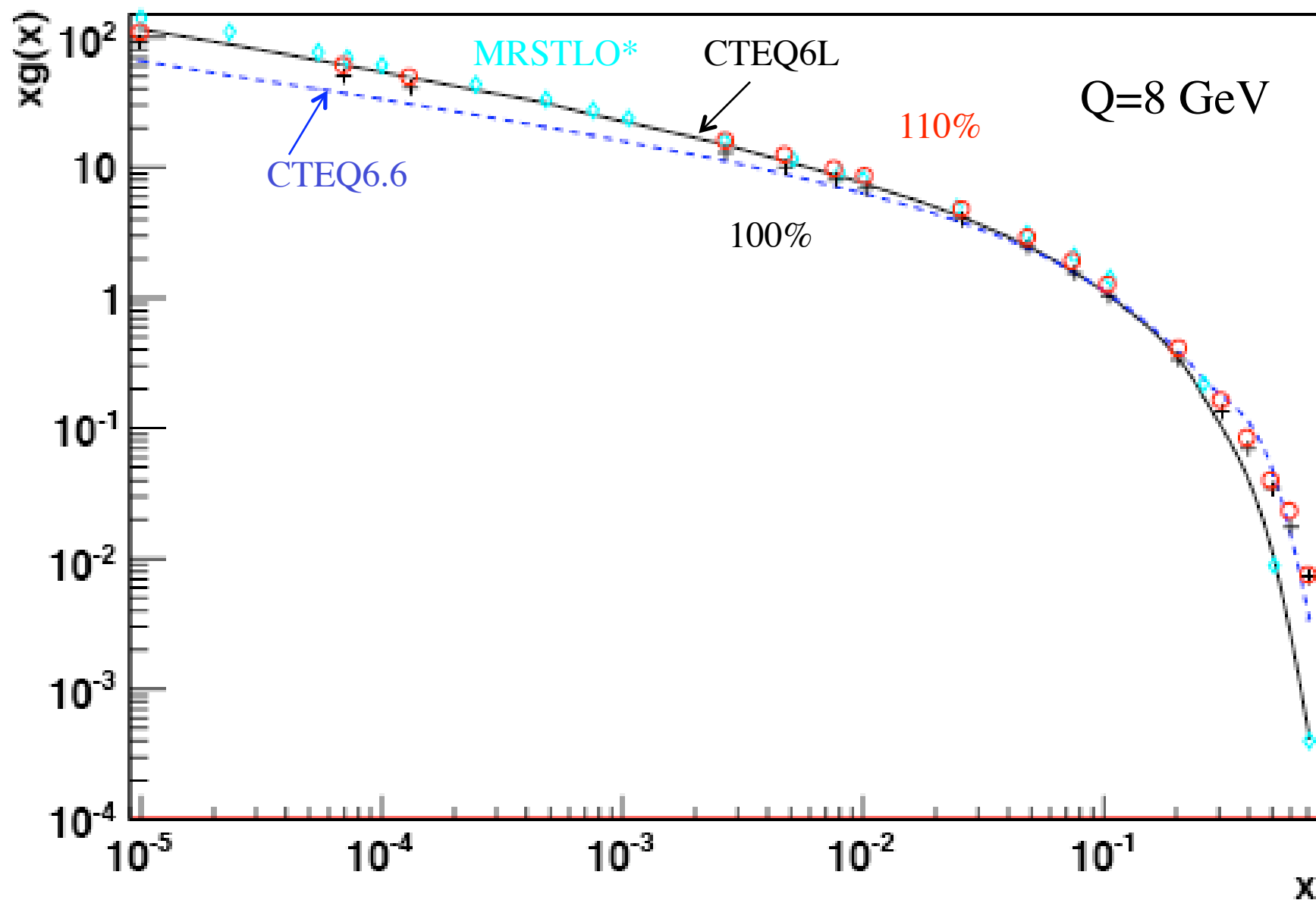
CTEQ

Low x behavior (closeup)



CTEQ

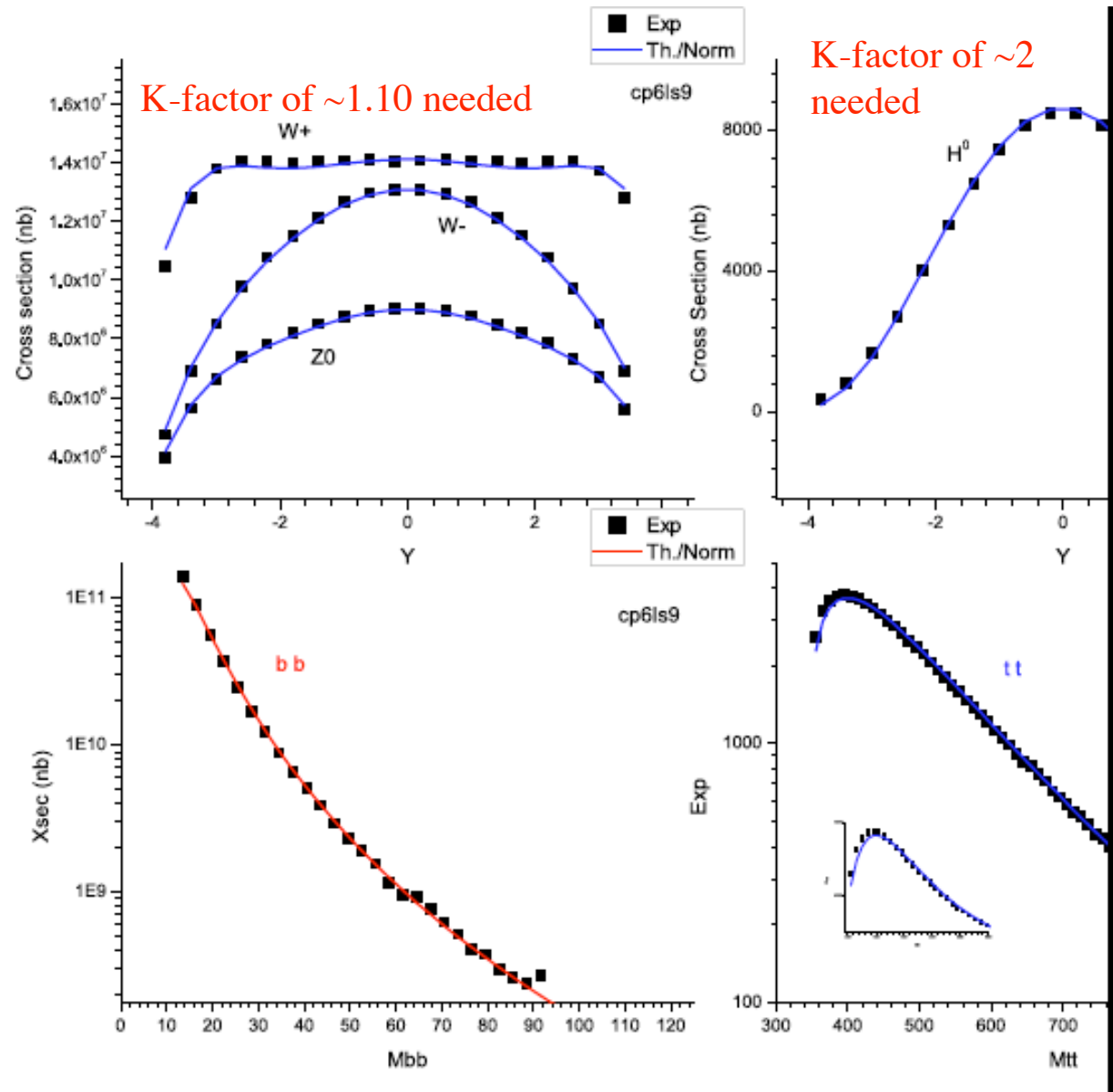
Comparison



Some comparisons to a candidate LO* pdf



- Additional normalization still needed, but somewhat smaller than with pure LO pdf
- Shapes for pseudo-data fit better (by construction)



Error pdf's



- In order to be truly useful, there should be accompanying error pdf's of a similar character as the LO* pdf's
 - ◆ so at the least, experimenters will not mix the NLO error pdf's with a central LO pdf
 - ▲ but maybe not so bad as far as gluon radiation is concerned if same α_s used
 - ▲ would still be a problem for UE if low x gluons are different
- But error pdf's imply a level of precision that is inherent to NLO
 - ◆ at NLO, we can construct an orthonormal set of eigenvectors accompanying a level of precision corresponding to a given change of $\Delta\chi^2$ in the global fit
 - ◆ that level of $\Delta\chi^2$, that variation, less well defined for LO fits

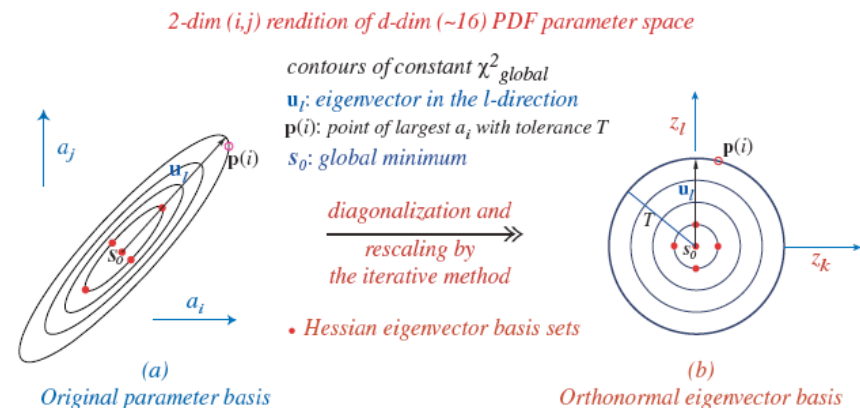


Figure 28. A schematic representation of the transformation from the pdf parameter basis to the orthonormal eigenvector basis.

- We are currently working on several ways of implementing this at LO*, but we have not finished stuffing the sausage casings yet

CTEQ4LHC/FROOT



- Collate/create cross section predictions for LHC
 - ◆ processes such as W/Z/ Higgs(both SM and BSM)/ diboson/tT/single top/photons/ jets...
 - ◆ at LO, NLO, NNLO (where available)
 - ▲ new: W/Z production to NNLO QCD and NLO EW
 - ◆ pdf uncertainty, scale uncertainty, correlations
 - ◆ impacts of resummation (q_T and threshold)
- As prelude towards comparison with actual data
- Using programs such as:
 - ◆ MCFM
 - ◆ ResBos
 - ◆ Pythia/Herwig/Sherpa
 - ◆ ... private codes with CTEQ
- First on webpage and later as a report

Primary goal: have all theorists write out parton level output into ROOT ntuples

Secondary goal: make libraries of prediction ntuples available

- FROOT: a simple interface for writing Monte-Carlo events into a ROOT ntuple file
- Written by Pavel Nadolsky (nadolsky@pa.msu.edu)
- CONTENTS
- =====
- froot.c -- the C file with FROOT functions
- taste_froot.f -- a sample Fortran program writing 3 events into a ROOT ntuple
- taste_froot0.c -- an alternative top-level C wrapper (see the compilation notes below)
- Makefile

CTEQ

PDF Uncertainties and FROOT



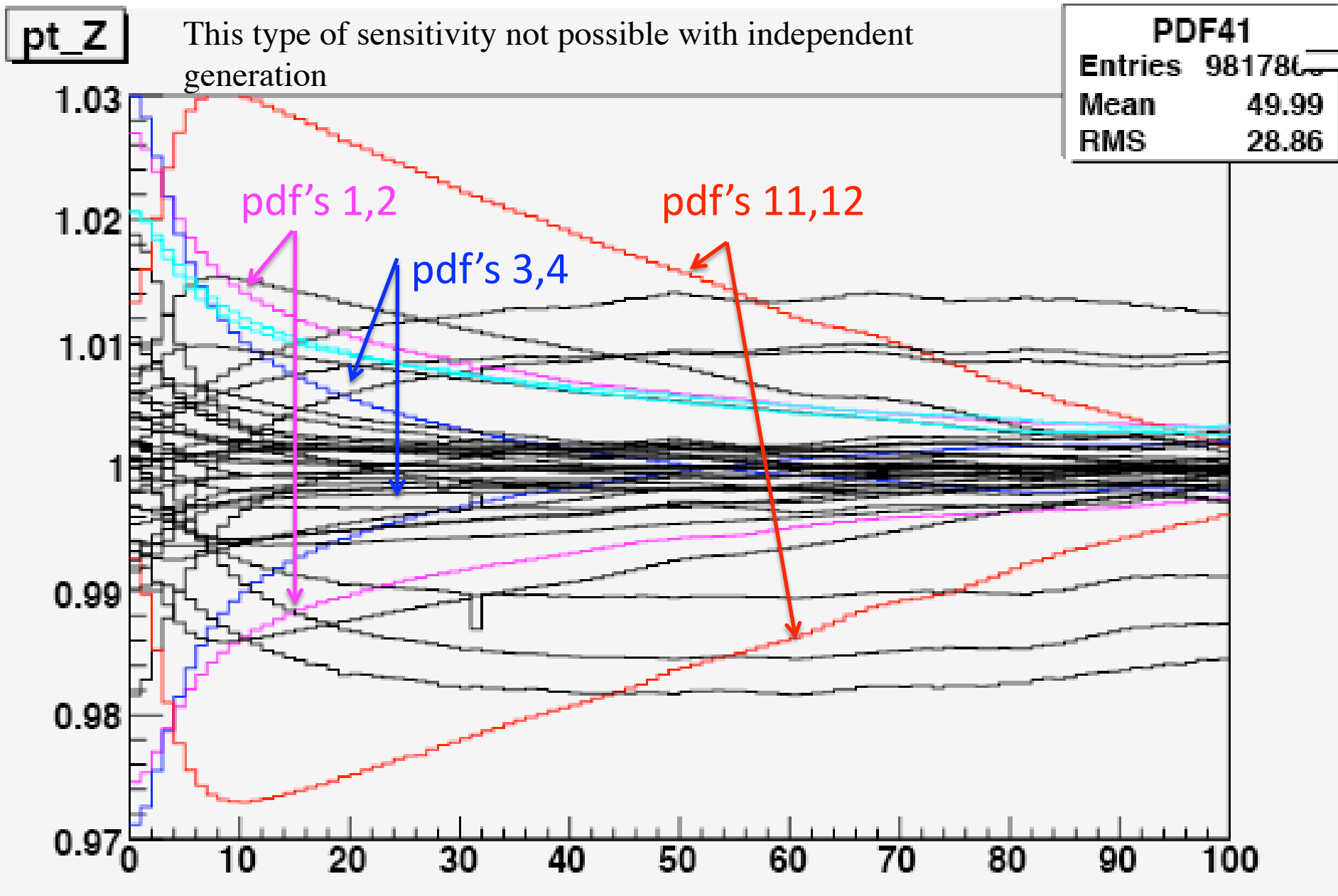
Z production in ResBos

new way, all pdf weights stored in ntuple, events generated once

old way -> independent ntuple for each pdf

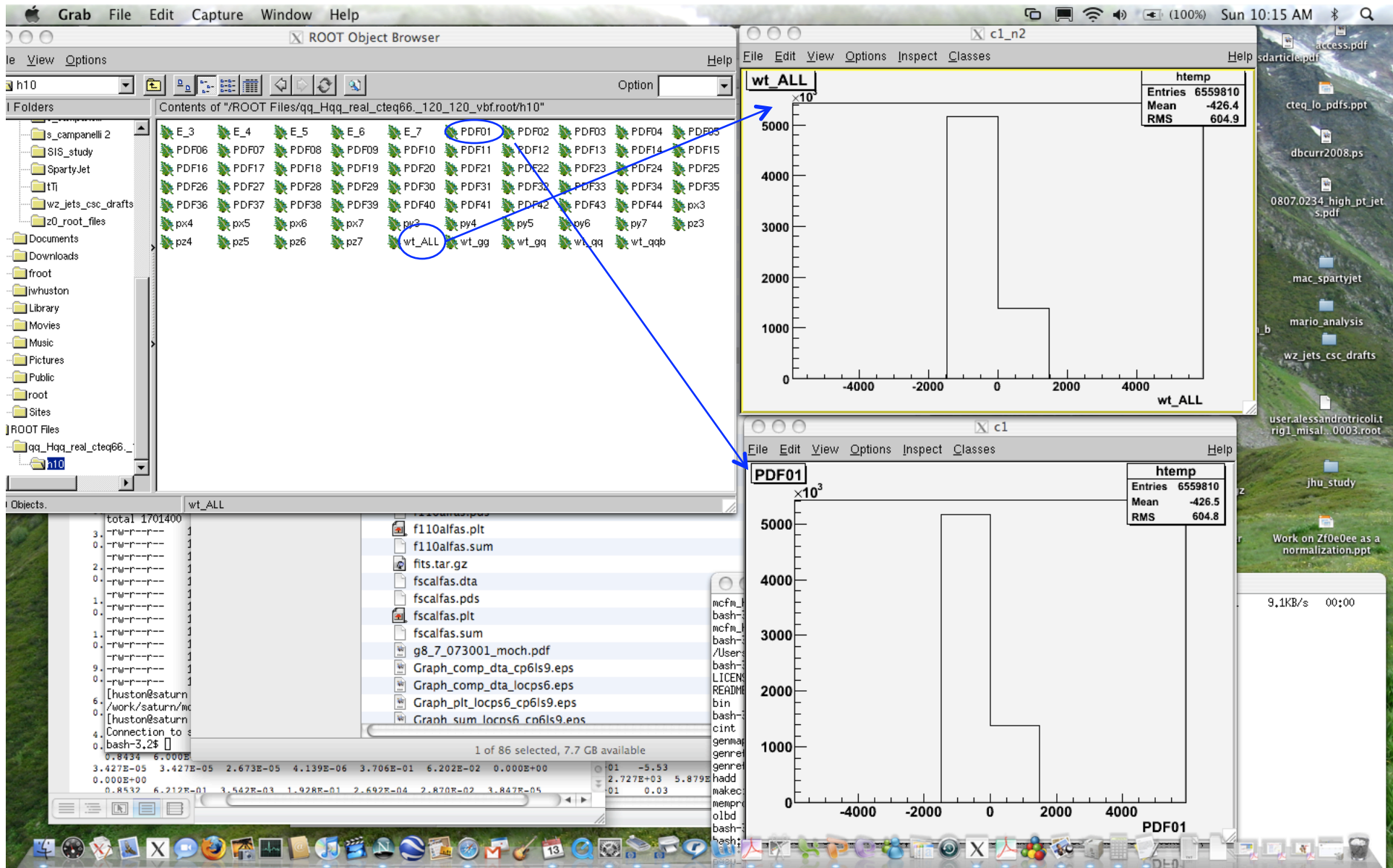
CTEQ

Ratio of Z p_T distributions to that from CTEQ6.6



CTEQ

MC FM 5.3 will be FROOT-able



CTEQ



EXTRA

CTEQ

SpartyJet



J. Huston, K. Geerlings
Michigan State University

P-A. Delsart, LAPP

Sparty



What is SpartyJet?

- “a framework intended to allow for the easy use of multiple jet algorithms in collider analyses”
 - Fast** to run, no need for heavy framework
 - Easy** to use, basic operation is very simple
 - Flexible**
 - ROOT-script or standalone execution
 - “on-the-fly” execution for event-by-event results
 - many different input types
 - different algorithms
 - output format

JetBuilder

- basically a frontend to handle most of the details of running SpartyJet
- not necessary, but makes running SpartyJet **much** simpler
- Allows options that are not otherwise accessible
 - text output
 - add minimum bias events

```
gSystem->Load("libTree.so");
gSystem->Load("libAlgoCore.so");
gSystem->Load("libCDFJet");
std::string input("data/1_Clusters.dat");
JetBuilder builder;
builder.configure_input(InputMaker* &input);
builder.add_default_alg(new cdf::JetClusFinder("myJetClu"));
builder.set_default_cut(0.1*input.getGeV());
builder.configure_output("SpartyJet_Tree", "data/output/simple.root");
builder.process_events(10);
```

```
FILE f("/home/deisart/spartyjet/vwithSIScone/example/data/small.root");
TTree *tree = (TTree*) f.Get("CollectionTree");
atlas::CBNTInput input;
input.init(tree);
JetAlgorithm *alg = new JetAlgorithm("MidPointJets");
JetPSelectorTool *selec = new JetPSelectorTool(1*GeV);
MidPoint *midpoint = new MidPoint("TOTOT");
alg->addTool((JetTool*)midpoint);
alg->addTool((JetTool*)selec);
alg->init();
NtupleMaker ntp;
ntp.addJetVar("MidPointJets");
ntp.init("jetTree", "out.root");
Jet::jet_list_t jets;
Jet::jet_list_t outjets;
input->fillInput(2, jets);
alg->execute(injets, outjets);
ntp.set_data("MidPointJets", outjets);
ntp.fillJets();
clear_jetlist(injets);
clear_jetlist(outjets);
input->fillInput(5, jets);
alg->execute(injets, outjets);
ntp.set_data("MidPointJets", outjets);
ntp.fillJets();
ntp.finalize();
```

without JetBuilder

“on-the-fly” method

- no input data file, no output data file
- from other C++ programs, call a variant of `jets = SpartyJet::getjets(JetTool*, data)`
- Currently supported data types:

```
Jet::jet_list_t& SpartyJet::getjets( JetTool* tool
                                   Jet::jet_list_t& inputJets);

std::vector<TLorentzVector>& SpartyJet::getjets( JetTool* tool
                                                std::vector<TLorentzVector>& input);

std::vector<TLorentzVector>& SpartyJet::getjets( JetTool* tool
                                                std::vector<TLorentzVector>& input,
                                                std::vector<std::vector<int>>& constituents);

std::vector<SpartyJet::simplejet> SpartyJet::getjets( JetTool* tool
                                                    std::vector<simplejet>& input);
```

Available Algorithms

- CDF
- JetClu
 - MidPoint (with optional second pass)
- D0
- D0RunIICone
- (from Lars Sonnenschein)
- ATLAS
- Cone
 - FastKt
- FastJet (from Gavin Salam and Matteo Cacciari)
- FastKt
 - Seedless Infrared Safe Cone (SIScone)
- Pythia 8
- CellJet

all algorithms are fully
parameterizable

reconstruct
individual
jets with
new
parameters
in context
of
analysis

Gui interface



canvas_group
num cols num rows
1 2
Reset Clear

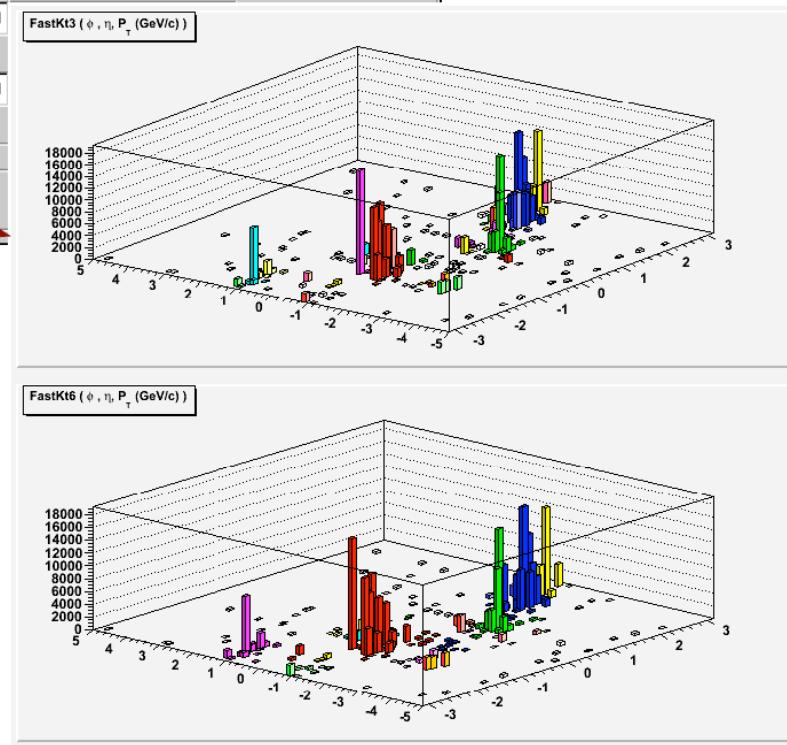
Event by Event plots
Algorithms
☒ FastKt3 ☒ FastKt6
Previous Event 0 Next Event

Operations
☐ Print jet quantities
☐ SnowMass potential
☐ DvsZ plot
☒ Lego plot
☐ 2D view
Draw
Options

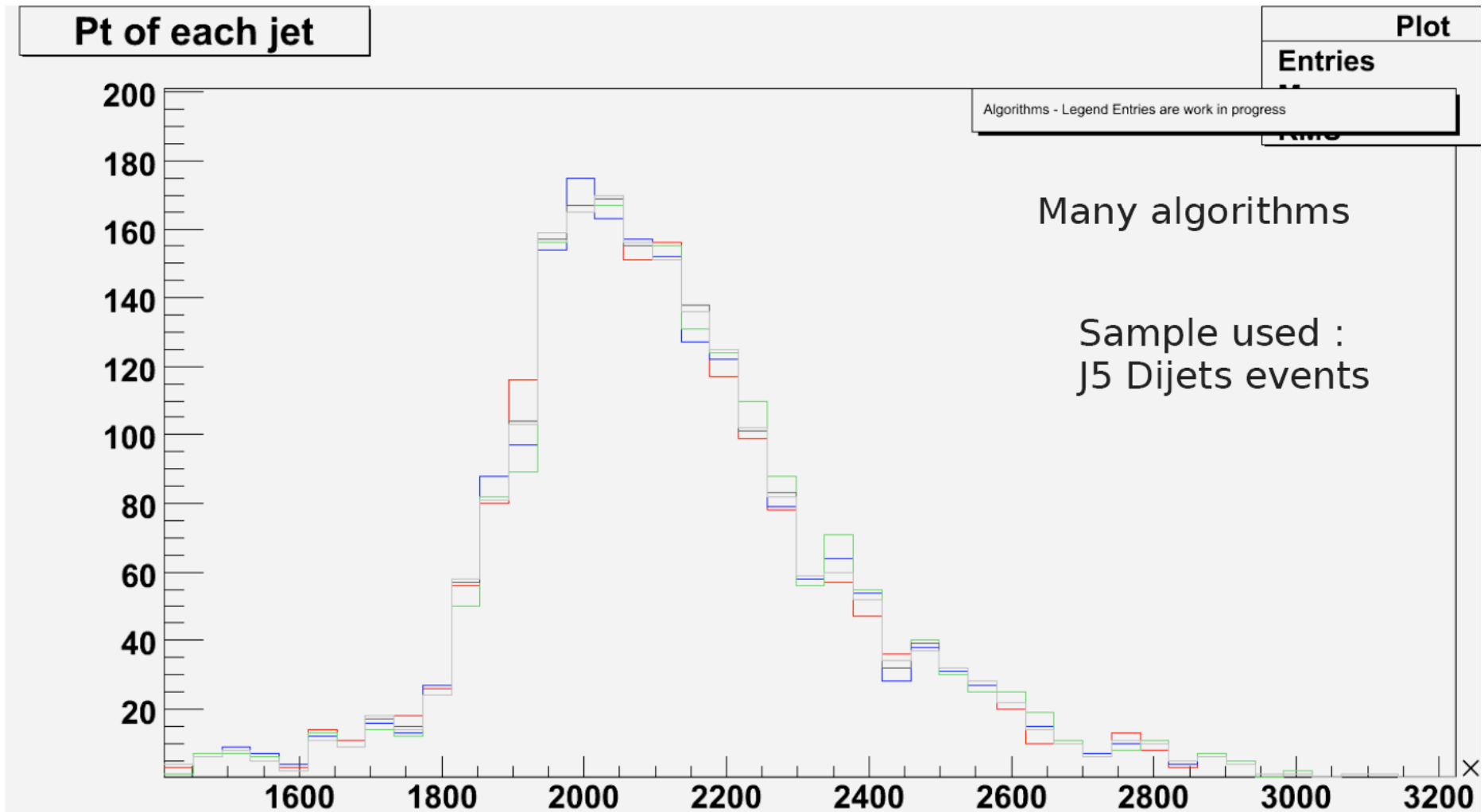
Run plots
Algorithms
☐ FastKt3 ☐ FastKt6
☒ e ☐ pt
☐ eta
plotted var
selection
first evt
min

On-The-Fly_group
Add Algorithm

- 0** Number of plots in canvas
- 1** Event-by-event gui
What event
What algorithm to draw
What to plot
- 2** All sample plots
~interface to TTree::Draw

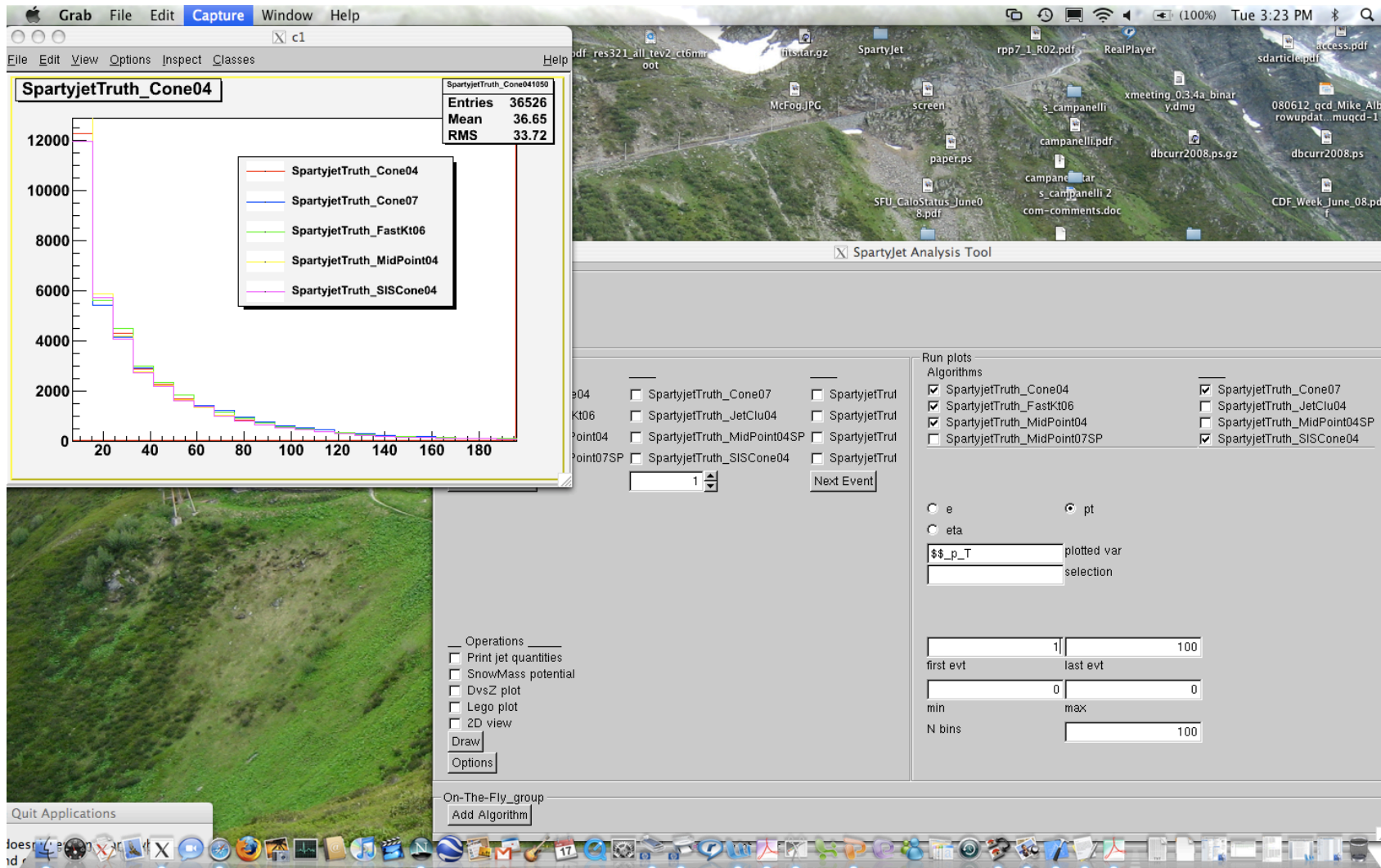


2:Interactive plots



CTEQ

Laptop running





Hard interactions of quarks and gluons: a primer for LHC physics

J M Campbell¹, J W Huston² and W J Stirling³

¹ Department of Physics and Astronomy, University of Glasgow, Glasgow G12 8QQ, UK

² Department of Physics and Astronomy, Michigan State University, East Lansing, MI 48840, USA

³ Institute for Particle Physics Phenomenology, University of Durham, Durham DH1 3LE, UK

E-mail: j.campbell@physics.gla.ac.uk, huston@msu.edu and w.j.stirling@durham.ac.uk

Received 14 July 2006, in final form 6 November 2006

Published 19 December 2006

Online at stacks.iop.org/RoPP/70/89

Abstract

In this paper, we will develop the perturbative framework for the calculation of hard-scattering processes. We will undertake to provide both a reasonably rigorous development of the formalism of hard-scattering of quarks and gluons as well as an intuitive understanding of the physics behind the scattering. We will emphasize the role of logarithmic corrections as well as power counting in α_s in order to understand the behaviour of hard-scattering processes. We will include ‘rules of thumb’ as well as ‘official recommendations’, and where possible will seek to dispel some myths. We will also discuss the impact of soft processes on the measurements of hard-scattering processes. Experiences that have been gained at the Fermilab Tevatron will be recounted and, where appropriate, extrapolated to the LHC.

(Some figures in this article are in colour only in the electronic version)

Review

Jets in hadron–hadron collisions

S.D. Ellis^{a,*}, J. Huston^b, K. Hatakeyama^c, P. Loch^d, M. Tönnemann^e

^a University of Washington, Seattle, WA 98195, United States

^b Michigan State University, East Lansing, MI 48824, United States

^c Rockefeller University, New York, NY 10021, United States

^d University of Arizona, Tucson, AZ 85721, United States

^e Max Planck Institute für Physik, Munich, Germany

arXiv:07122447 Dec 14, 2007

Abstract

In this article, we review some of the complexities of jet algorithms and of the resultant comparisons of data to theory. We review the extensive experience with jet measurements at the Tevatron, the extrapolation of this acquired wisdom to the LHC and the differences between the Tevatron and LHC environments. We also describe a framework (SpartyJet) for the convenient comparison of results using different jet algorithms.

© 2007 Elsevier B.V. All rights reserved.

Keywords: Jet; Jet algorithm; LHC; Tevatron; Perturbative QCD; SpartyJet

Contents

1. Introduction.....	485
2. Factorization.....	486
3. Jets: Parton level vs experiment	490
3.1. Iterative cone algorithm	490
3.1.1. Definitions	490
3.1.2. R_{sep} , seeds and IR-sensitivity	495
3.1.3. Seedless and midpoint algorithms.....	498
3.1.4. Merging	499
3.1.5. Summary	499