LUND UNIVERSITY

MC4LHC readiness
29 March – 1 April 2010
CERN, Geneva

PYTHIA 8 Status

Torbjörn Sjöstrand

Department of Astronomy and Theoretical Physics
Lund University

+ Several improvements relative to PYTHIA 6

– Difficulties with MB/UE tuning

Ambition

- Meet experimental request for C++ code.
- Housecleaning $\Rightarrow$ more homogeneous.
- More user-friendly (e.g. settings names).
- Better match to software frameworks (e.g. card files).
- More space for growth.
- Better interfaces to external standards.

Reality

- Work begun autumn 2004.
- 3 years at CERN $\Rightarrow$ good progress.
- First release autumn 2007.
- Since then: slower progress, requests lagging behind.
- Slow adoption (rightfully so?).

Team members

Stefan Ask
Richard Corke
Stephen Mrenna
Peter Skands

Contributors

Bertrand Bellenot
Lisa Carloni
Hendrik Hoeth
Tomas Kasemets
Mikhail Kirsanov
Ben Lloyd
Marc Montull
Sparsh Navin
MSTW, CTEQ, H1: PDFs
DELPHI, LHCb: D/B BRs
+ several bug reports & fixes

PYTHIA Physics (part I)

Hard processes:

- Built-in library of many leading-order processes.
  Standard Model: almost all $2 \to 1$ and $2 \to 2$, a few $2 \to 3$.
  Beyond the SM: a bit of each.
- External input via Les Houches Accord and Les Houches Event Files from MadGraph, CompHep, AlpGen, ...
- Resonance decays, often but not always with angular correlations.

Showers:

- Transverse-momentum-ordered ISR & FSR.
- Includes $\mathrm{q} \to \mathrm{qg}$, $\mathrm{g} \to \mathrm{gg}$, $\mathrm{g} \to \mathrm{q\overline{q}}$, $\mathrm{f} \to \mathrm{f}\gamma$, $\gamma \to \mathrm{f\overline{f}}$ (f = fermion).
- ISR by backwards evolution.
- Dipole-style approach to recoils (*partly new*).
- Matching to ME's for first (=hardest) emission in many processes, especially gluon emission in resonance decays.

PYTHIA Physics (part II)

Underlying events and minimum-bias events:

- Multiple parton–parton interactions,
  with dampening of cross-section in $p_\perp \to 0$ limit,
  impact-parameter dependence, and tailormade PDF's.
- Combined evolution MI + ISR + FSR downwards in $p_\perp$ (*partly new*).
- Beam remnants colour-connected to interacting systems,
  and detailed modelling of flavour and momentum structure.

Hadronization:

- String fragmentation ("the Lund Model").
- Particle decays, usually isotropic.
- Link to external decay packages, say for τ (TAUOLA) or B (EVTGEN).
- Optional Bose-Einstein effects.

Utilities:

- Four-vectors, random numbers, parton densities, . . .
- Event study routines: sphericity, thrust, jet finding.
- Simple built-in histogramming package (line-printer mode).

Key differences between PYTHIA 6.4 and 8.1

Old features definitely removed include, among others:

- independent fragmentation
- mass-ordered showers

Features omitted so far include, among others:

- ep, γp and $\gamma\gamma$ beam configurations
- several processes, especially Technicolor, partly SUSY

New features, not found in 6.4 ($\star$ = see below):

- interleaved $p_{\perp}$-ordered MI + ISR + FSR evolution
- richer mix of underlying-event processes (γ, J/ψ, DY, ...)

$\star$ possibility for two selected hard interactions in same event

$\star$ allow rescattering in MI framework

$\star$ hard scattering in diffractive systems

$\star$ several new processes, within and beyond SM

- possibility to use one PDF set for hard process and another for rest

$\star$ up-to-date decay data and LO PDF sets

New processes

SUSY expanded to main processes, but still not all complete and tested.

- Complete, with non-minimal flavour violation and/or CP violation:
  $\tilde{\mathrm{q}}\tilde{\mathrm{q}}^*,\ \tilde{\mathrm{q}}\tilde{\mathrm{q}},\ \tilde{\mathrm{q}}\chi^{0,\pm},\ \chi^{0,\pm}\chi^{0,\mp}$
- In flavour-diagonal, CP-conserving limit: $\tilde{\mathrm{g}}\tilde{\mathrm{g}},\ \tilde{\mathrm{g}}\tilde{\mathrm{q}}$
- Still missing: $\tilde{\mathrm{g}}\chi^{0,\pm},\ \tilde{\ell}^{+}\tilde{\ell}^{-},\ \tilde{\ell}^{\pm}\tilde{\nu},\ \tilde{\nu}\tilde{\nu}$

Process control: all, 12(+) subgroups, specific pair of outgoing flavours.

Randall-Sundrum extra dimensions, massive decaying G^*, since before.
Now also Large Extra Dimensions, invisible variable-mass G:

- monojets: $\mathrm{Gg},\ \mathrm{Gq},\ \mathrm{G}\gamma,\ \mathrm{GZ}^0$
- virtual G exchange: $\gamma\gamma$, $\ell^+\ell^-$

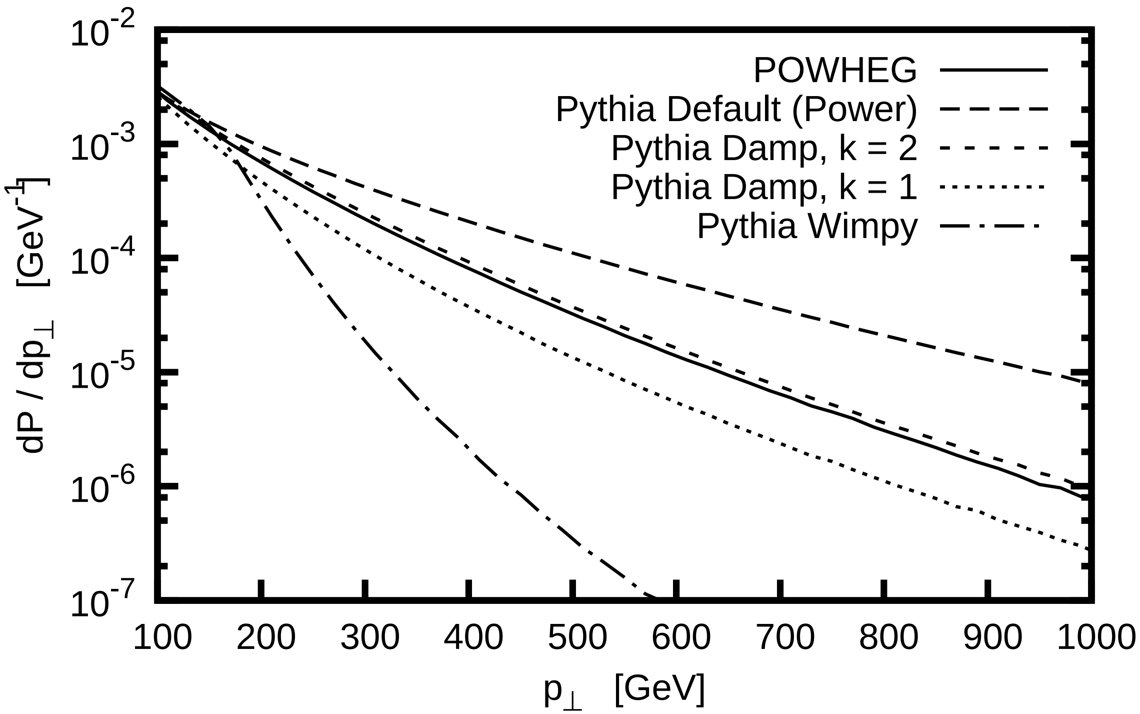
Same processes reused for Unparticle production,
only with different process names and parameter space.

Hidden Valley scenario with radiation into hidden sector (coming in 8.140):

- copies F_v of all SM f in fundamental representation of HV
- pair production $q\,\overline{q}, g\,g \to Q_v\,\overline{Q}_v$, $f\,\overline{f} \to \gamma^*/Z^0 \to F_v\,\overline{F}_v$
- HV particle q_v such that $F_v \to f\,q_v$
- gauge bosons g_v (or γ_v for non-self-interacting)
- interleaved showering $F_v \to F_v\,g_v/g$, $q_v \to q_v\,g_v$, $\mathrm{g}_v \to g_v\,g_v$.

Shower matching to MEs

Aim: provide better default shower behaviour at large $p_\perp$,
to bridge gap between "power" and "wimpy" showers.



$\mathrm{t\bar{t}}$ production

$$M^2 = m_{\perp\mathrm{t}}^2 = m_\mathrm{t}^2 + p_{\perp\mathrm{t}}^2$$

$$\frac{\mathrm{d}P_{\mathrm{ISR}}}{\mathrm{d}p_\perp^2} \propto \frac{1}{p_\perp^2}\frac{k^2M^2}{k^2M^2+p_\perp^2}$$ for coloured final state

No dampening for uncoloured final state ($\mathrm{W^+W^-}$, ..., SUSY).
R. Corke & TS, arXiv:1003.2384 [hep-ph] (coming in 8.140)
(+ improved interfacing to POWHEG, ...)

VINCIA: towards NLO showers

Simple shower formalism based on 2 $\rightarrow$ 3 antenna factorization for arbitrary evolution variables, recoil maps, radiation kernels, etc.

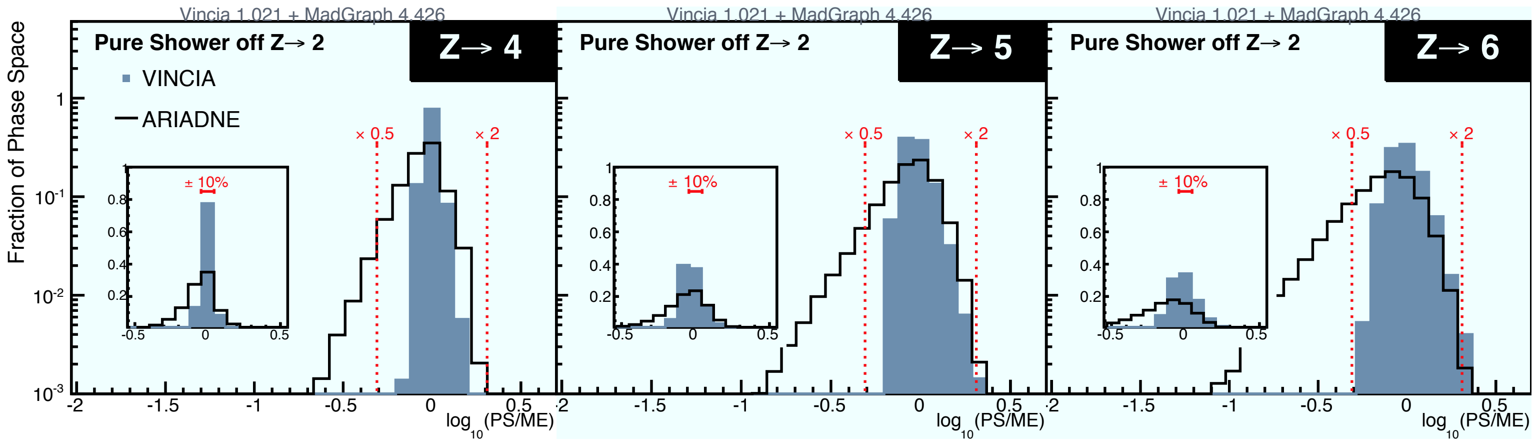
Matching = cancel dependence on free parameters to given order

+ *Exponentiate matching* = Use subleading logs in ME to improve resummation instead of destroying it (currently no "matching scale" needed before $\alpha_s^3 \times$ Born)

+ *Improve Shower* = No dead zones, Markov Ordering (+ partial NLL matching)

Tree-Level expansion vs MadGraph (flat phase-space scan, full color):

(Narrow distributions = good logarithmic precision)



Pure shower

$<$VINCIA$>$ =

$|M_4|^2 \pm$ 8%

$|M_5|^2 \pm$ 20%

$|M_6|^2 \pm$ 25%

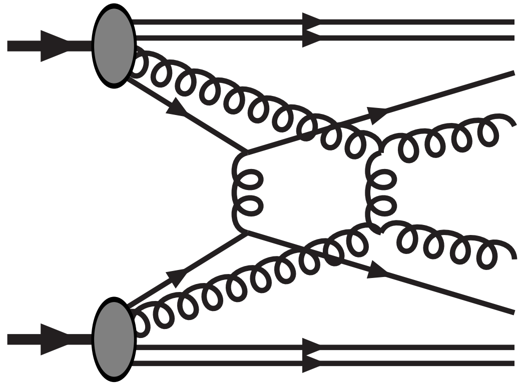
(+ can do matching, up through $Z \rightarrow 6$ is already in)

Current Version = 1.021: massless FSR (first public release Sep 2009).

- Short Term: Long writeup (shower + tree-matching + LEP Pheno study)
- This Summer: Massive quarks (with M. Ritzmann, A. Gehrmann-de-Ridder)
- Long Term: Initial-State Radiation and multijet 1-loop matching.

A second hard process

Multiple interactions key aspect
of PYTHIA since $>$ 20 years.
Central to obtain agreement with data:
Tune A, Professor, Perugia, ...



Before 8.1 no chance to select character of second interaction.
Now free choice of first process (including LHA/LHEF)
and second process combined from list:

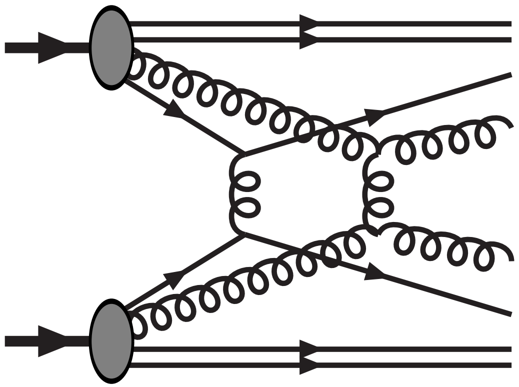
- TwoJets (with TwoBJets as subsample)
- PhotonAndJet, TwoPhotons
- Charmonium, Bottomonium (colour octet framework)
- SingleGmZ, SingleW, GmZAndJet, WAndJet
- TopPair, SingleTop

Can be expanded among existing processes as need arises.

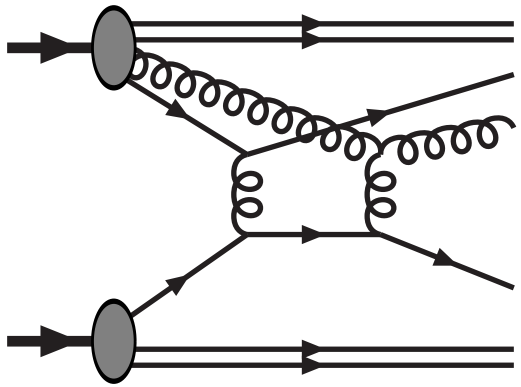
By default same phase space cuts as for "first" hard process
$\Longrightarrow$ second can be harder than first.
However, possible to set $\widehat{m}$ and $\widehat{p_\perp}$ range separately.

Rescattering

Often assume that MI =



. . . but should also include



Same order in α_s, $\sim$ same propagators, but

- one PDF weight less $\Rightarrow$ smaller σ
- one jet less $\Rightarrow$ QCD radiation background $2 \to 3$ larger than $2 \to 4$

$\Rightarrow$ will be tough to find direct evidence.

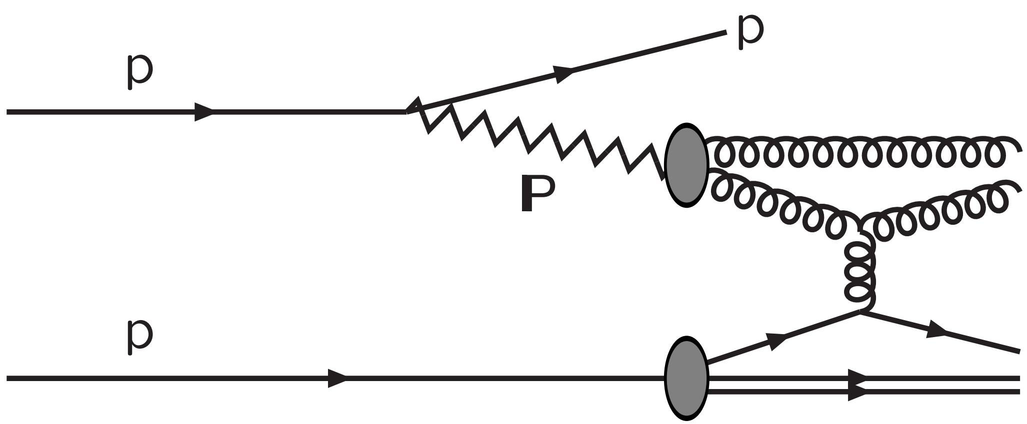
Rescattering grows with number of “previous” scatterings:

	Tevatron		LHC	
	Min Bias	QCD Jets	Min Bias	QCD Jets
Normal scattering	2.81	5.09	5.19	12.19
Single rescatterings	0.41	1.32	1.03	4.10
Double rescatterings	0.01	0.04	0.03	0.15

R. Corke & TS, JHEP 01 (2010) 035 [arXiv:0911.1909]

Diffraction

Ingelman-Schlein: Pomeron as hadron with partonic content
Diffractive event = (Pomeron flux) $\times$ ($\mathbb{P}$p collision)



Used e.g. in
POMPYT
POMWIG
PHOJET

1) σ_{SD} and σ_{DD} taken from existing parametrization or set by user.

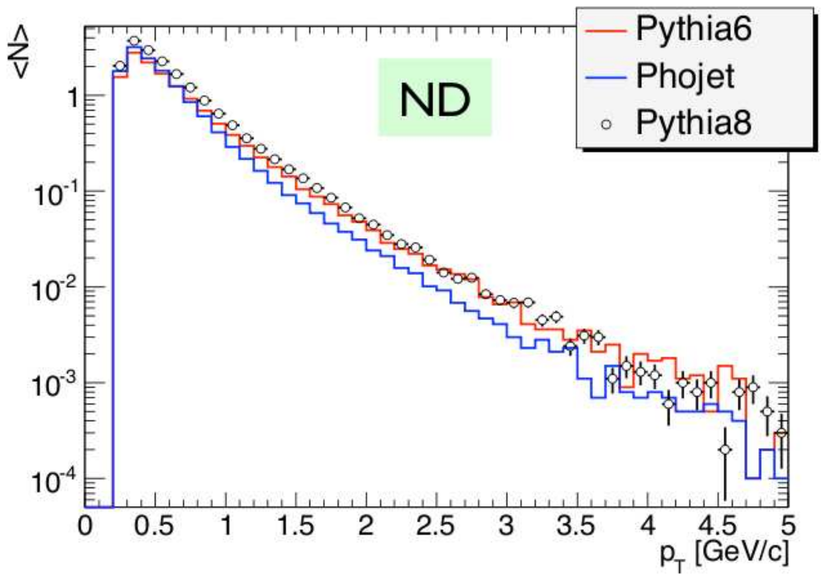
2) Shape of Pomeron distribution inside a proton, $f_{\mathbb{P}/\mathrm{p}}(x_{\mathbb{P}}, t)$ gives diffractive mass spectrum and scattering $p_{\perp}$ of proton.

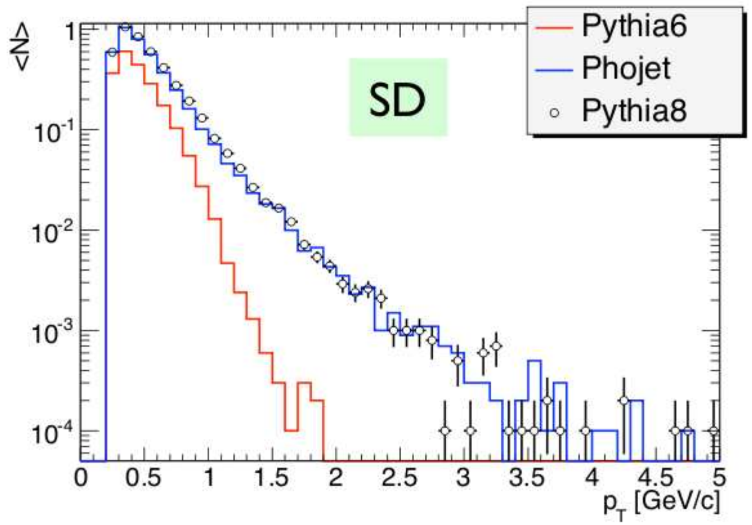
3) At low masses retain old framework, with longitudinal string(s). Above 10 GeV begin smooth transition to $\mathbb{P}$p handled with full pp machinery: multiple interactions, parton showers, beam remnants, ...

4) Choice between 5 Pomeron PDFs.
Free parameter $\sigma_{\mathbb{P}\mathrm{p}}$ needed to fix $\langle n_{\mathrm{interactions}} \rangle = \sigma_{\mathrm{jet}}/\sigma_{\mathbb{P}\mathrm{p}}$.

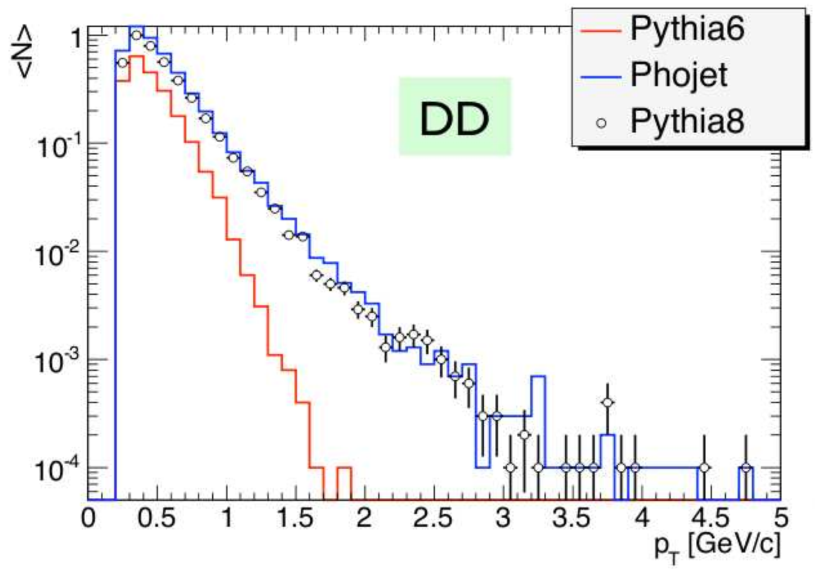
5) Framework needs testing and tuning, e.g. of $\sigma_{\mathbb{P}\mathrm{p}}$.

p_T Distributions ($\sqrt{s}$=0.9 TeV)





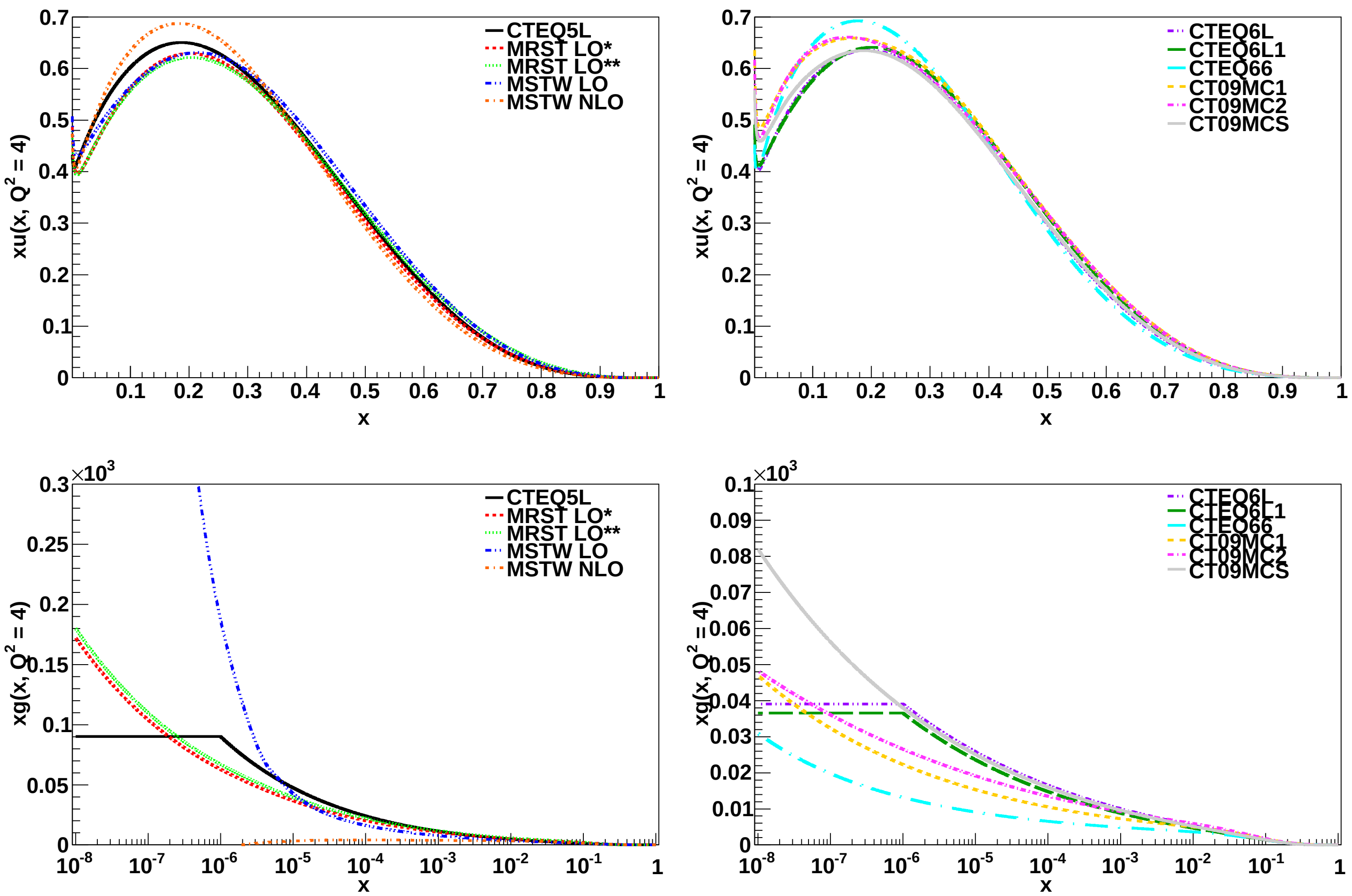
- Softer p_T spectrum in **Pythia6** due to lack of high mass diffraction
- **Pythia8** and **Phojet** agree quite well



Beate Heinemann, MB/UE Working Group, March 1 (also Sparsh Navin)

New PDFs

u quark and gluon at $Q^2 = 4\ \mathrm{GeV}^2$ (for MB/UE physics!):



Old PDFs:
GRV 94L
CTEQ5L

New PDFs:
MRST LO*
MRST LO**
MSTW LO
MSTW NLO
CTEQ 6L
CTEQ 6L1
CTEQ 66
CT09MC1
CT09MC2
CT09MCS

NLO: beware!

Native implementation: faster, checked to work

Tomas Kasemets, arXiv:1002.4376 [hep-ph]

Interface to experimental framework (CMS list)

- Can be used as a C++ shared library.
- Accepts input from LHEF.
- Provides output in HepMC format (when linked to HepMC).
- Memory of running program $\sim$ 10 MB.
- All relevant parameters can be set via input files . . .

- . . . but if allowed to send in pointers to derived classes then
  - ⋆ external runtime LHA event generation;
  - ⋆ external random number generator;
  - ⋆ external particle decays (EVTGEN, TAUOLA, . . .);
  - ⋆ external processes, resonances and PDFs;
  - ⋆ external user hooks to reweight cross sections and more;
  - ⋆ external beam shape (momentum and primary vertex);
  - ⋆ external parton showers (VINCIA, . . .).

Generally good opinions on user-friendliness,
and on documentation — if it was known (responsibles take notice!) —
but no complete update of PYTHIA 6.4 manual in near future

PYTHIA 8 Index

Program Overview

Setup Run Tasks

PYTHIA 8

Welcome to PYTHIA - The Lund Monte Carlo!

PYTHIA 8 is the successor to PYTHIA 6, rewritten from scratch in C++. With the release of PYTHIA 8.1 it now becomes the official "current" PYTHIA version, although PYTHIA 6.4 will be supported in parallel with it for some time to come. Specifically, the new version has not yet been enough tested and tuned for it to have reached the same level of reliability as the older one. This testing will only happen if people begin to work with the program, however, which is why we encourage a gradual transition to the new version, starting now. There are some new physics features in PYTHIA 8.1, that would make use of it more attractive, but also some topics still missing, where 6.4 would have to be used. Further, many obsolete features will not be carried over, so for some backwards compatibility studies again 6.4 would be the choice.

Documentation

On these webpages you will find the up-to-date manual for PYTHIA 8.1. Use the left-hand index to navigate this documentation of program elements, especially of all possible program settings. All parameters are provided with sensible default values, however, so you need only change those of relevance to your particular study, such as choice of beams, processes and phase space cuts. The pages also contain a fairly extensive survey of all methods available to the user, e.g. to study the produced events. What is lacking on these webpages is an overview, on the one hand, and an in-depth physics description, on the other.

The overview can be found in the attached PDF file
A Brief Introduction to PYTHIA 8.1
T. Sjöstrand, S. Mrenna and P. Skands, Comput. Phys. Comm. 178 (2008) 852 [arXiv:0710.3820].
You are strongly recommended to read this summary when you start out to learn how to use PYTHIA 8.1. Note that some details have changed since the 8.100 version described there.

For the physics description we refer to the complete
PYTHIA 6.4 Physics and Manual
T. Sjöstrand, S. Mrenna and P. Skands, JHEP05 (2006) 026,
which in detail describes the physics (largely) implemented also in PYTHIA 8, and also provides a more extensive bibliography than found here.

When you use PYTHIA 8.1, you should therefore cite both, e.g. like
T. Sjöstrand, S. Mrenna and P. Skands, JHEP05 (2006) 026, Comput. Phys. Comm. 178 (2008) 852.

Furthermore, a separate
PYTHIA 8 Worksheet,
also an attached PDF file, offers a practical introduction to using the generator. It has been developed for and used at a few summer schools, with minor variations, but is also suited for self-study.

Timelike Showers

The PYTHIA algorithm for timelike final-state showers is based on the recent article [Sjo05], where a transverse-momentum-ordered evolution scheme is introduced. This algorithm is influenced by the previous mass-ordered algorithm in PYTHIA [Ben87] and by the dipole-emission formulation in Ariadne [Gus86]. From the mass-ordered algorithm it inherits a merging procedure for first-order gluon-emission matrix elements in essentially all two-body decays in the standard model and its minimal supersymmetric extension [Nor01].

The normal user is not expected to call `TimeShower` directly, but only have it called from `Pythia`. Some of the parameters below, in particular `TimeShower:alphaSvalue`, would be of interest for a tuning exercise, however.

Main variables

Often the maximum scale of the FSR shower evolution is understood from the context. For instance, in a resonace decay half the resonance mass sets an absolute upper limit. For a hard process in a hadronic collision the choice is not as unique. Here the factorization scale has been chosen as the maximum evolution scale. This would be the *pT* for a *2 -> 2* process, supplemented by mass terms for massive outgoing particles. Some small amount of freedom is offered by

TimeShower:pTmaxFudge 1.0 (`default = 1.0`; `minimum = 0.25`; `maximum = 2.0`)
While the above rules would imply that *pT_max = pT_factorization*, `pTmaxFudge` introduced a multiplicative factor *f* such that instead *pT_max = f * pT_factorization*. Only applies to the hardest interaction in an event, cf. below. It is strongly suggested that *f = 1*, but variations around this default can be useful to test this assumption.
Note: Scales for resonance decays are not affected, but can be set separately by user hooks.

TimeShower:pTmaxFudgeMI 1.0 (`default = 1.0`; `minimum = 0.25`; `maximum = 2.0`)
A multiplicative factor *f* such that *pT_max = f * pT_factorization*, as above, but here for the non-hardest interactions (when multiple interactions are allowed).

The amount of QCD radiation in the shower is determined by

TimeShower:alphaSvalue 0.1383 (`default = 0.1383`; `minimum = 0.06`; `maximum = 0.25`)
The *alpha_strong* value at scale *M_Z^2*. The default value corresponds to a crude tuning to LEP data, to be improved.

The actual value is then regulated by the running to the scale *pT^2*, at which the shower evaluates *alpha_strong*

TimeShower:alphaSorder (`default = 1`; `minimum = 0`; `maximum = 2`)
Order at which alpha_strong runs,
- ○ **0** : zeroth order, i.e. alpha_strong is kept fixed.
- ◉ **1** : first order, which is the normal value.
- ○ **2** : second order. Since other parts of the code do not go to second order there is no strong reason to use this option, but there is also nothing wrong with it.

PYTHIA 8 Index

Program Overview

Frontpage
Program Flow
Settings Scheme
Particle Data Scheme
Program Files
Program Classes
Program Methods
Sample Main Programs

Setup Run Tasks

Save Settings
Main-Program Settings
Beam Parameters
Random-Number Seed
PDF Selection
Master Switches
Process Selection
-- QCD
-- Electroweak
-- Onia
-- Top
-- Fourth Generation
-- Higgs
-- SUSY
-- New Gauge Bosons
-- Left-Right Symmetry
-- Leptoquark
-- Compositeness
-- Extra Dimensions
A Second Hard Process
Phase Space Cuts

Sample Main Programs

Descriptions of available classes, methods and settings are all very good and useful. Ultimately they are necessary for you to be able to fine-tune your runs to the task at hand. To get going, however, nothing helps like having explicit examples to study. This is what is provided in the `examples` subdirectory, along with instructions how they should be run:

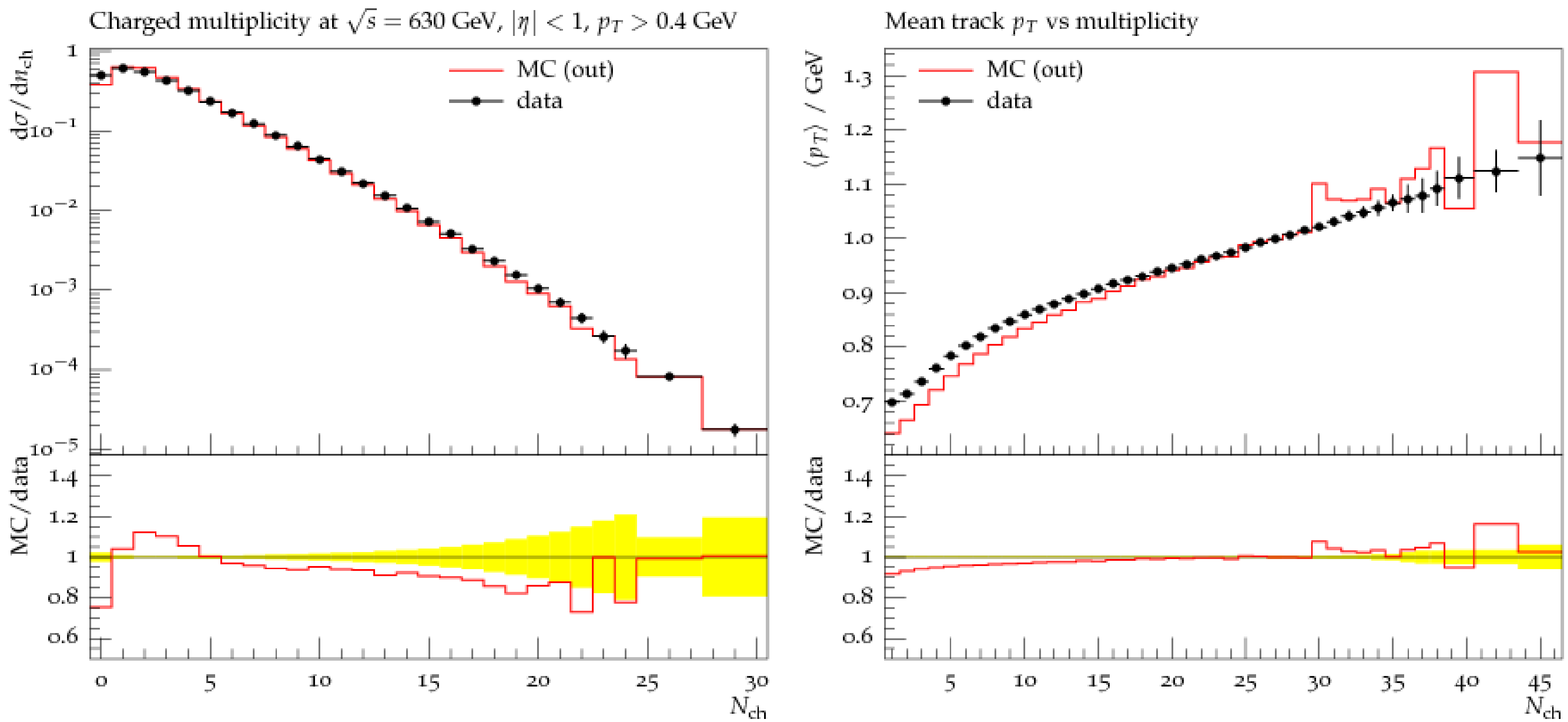
- `main01.cc` : a simple study of the charged multiplicity for jet events at the LHC. (Brief example fitting on one slide.)
- `main02.cc` : a simple study of the *pT* spectrum of Z bosons at the Tevatron. (Brief example fitting on one slide.)
- `main03.cc` : a simple single-particle analysis of jet events, where input is set by `main03.cmnd` "cards file".
- `main04.cc` : a simple study of several different kinds of events, with the choice to be made in the `main04.cmnd` "cards file".
- `main05.cc` : generation of QCD jet events at the LHC, with jet analysis using the `CellJet` cone-jet finder.
- `main06.cc` : tests of cross sections for elastic and diffractive topologies, using `main06.cmnd` to pick process.
- `main07.cc` : tests of cross sections for minimum-bias events, using `main07.cmnd` to pick options.
- `main08.cc` : generation of the QCD jet cross section by splitting the run into subruns, each in its own *pT* bin, and adding the results properly reweighted. Two options, with limits set either in the main program or by subrun specification in the `main08.cmnd` file.
- `main09.cc` : generation of LEP1 hadronic events, i.e. *e^+e^- -> gamma*/Z^0 -> q qbar*, with charged multiplicity, sphericity, thrust and jet analysis.
- `main10.cc` : illustration how userHooks can be used interact directly with the event-generation process.
- `main11.cc` : generation of two predetermined hard interactions in each event.
- `main12.cc` : a study of top events, fed in from the Les Houches Event File `ttbar.lhe`, here generated by PYTHIA 6.4. This file currently only contains 100 events so as not to make the distributed PYTHIA package too big, and so serves mainly as a demonstration of the principles involved.
- `main13.cc` : a more sophisticated variant of `main12.cc`, where two Les Houches Event Files (`ttbar.lhe` and `ttbar2.lhe`) successively are used as input. Also illustrating some other aspects, like the capability to mix in internally generated events.
- `main14.cc` : a systematic comparison of several cross section values with their corresponding values in PYTHIA 6.4, the latter available as a table in the code.
- `main15.cc` : loop over several tries, either to redo B decays only or to redo the complete hadronization chain of an event. Since much of the generation process is only made once this is a way to increase efficiency.
- `main16.cc` : put all user analysis code into a class of its own, separate from the main program; provide the "cards file" name as a command-line argument.
- `main17.cc` : collect the Pythia calls in a wrapper class, thereby simplifying the main program; provide the "cards file" name as a command-line argument.

Done

Tuning

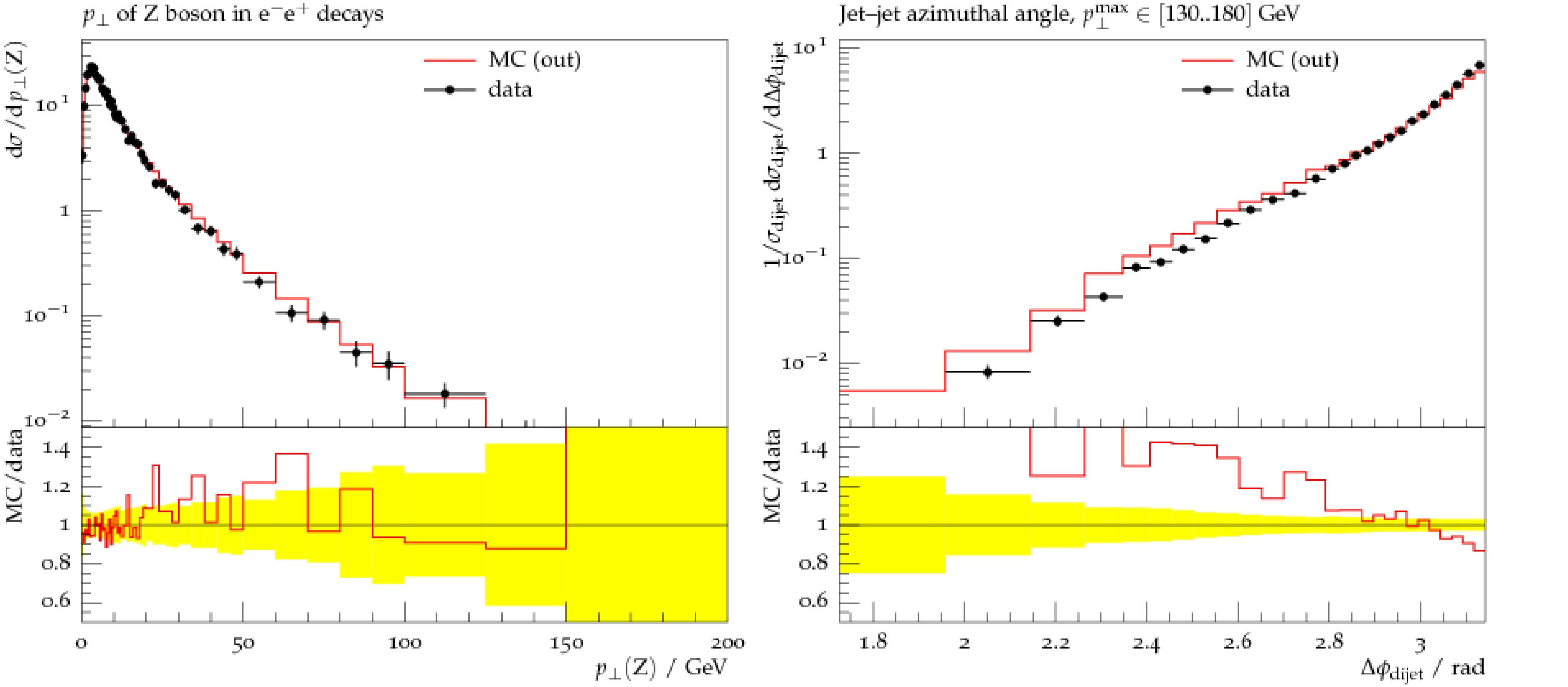
Tuning to e^+e^- closely related to $p_\perp$-ordered PYTHIA 6.4;
Rivet+Professor (H. Hoeth) $\Rightarrow$ FSR & hadronization OK (?)

First tuning to MB data by P. Skands:



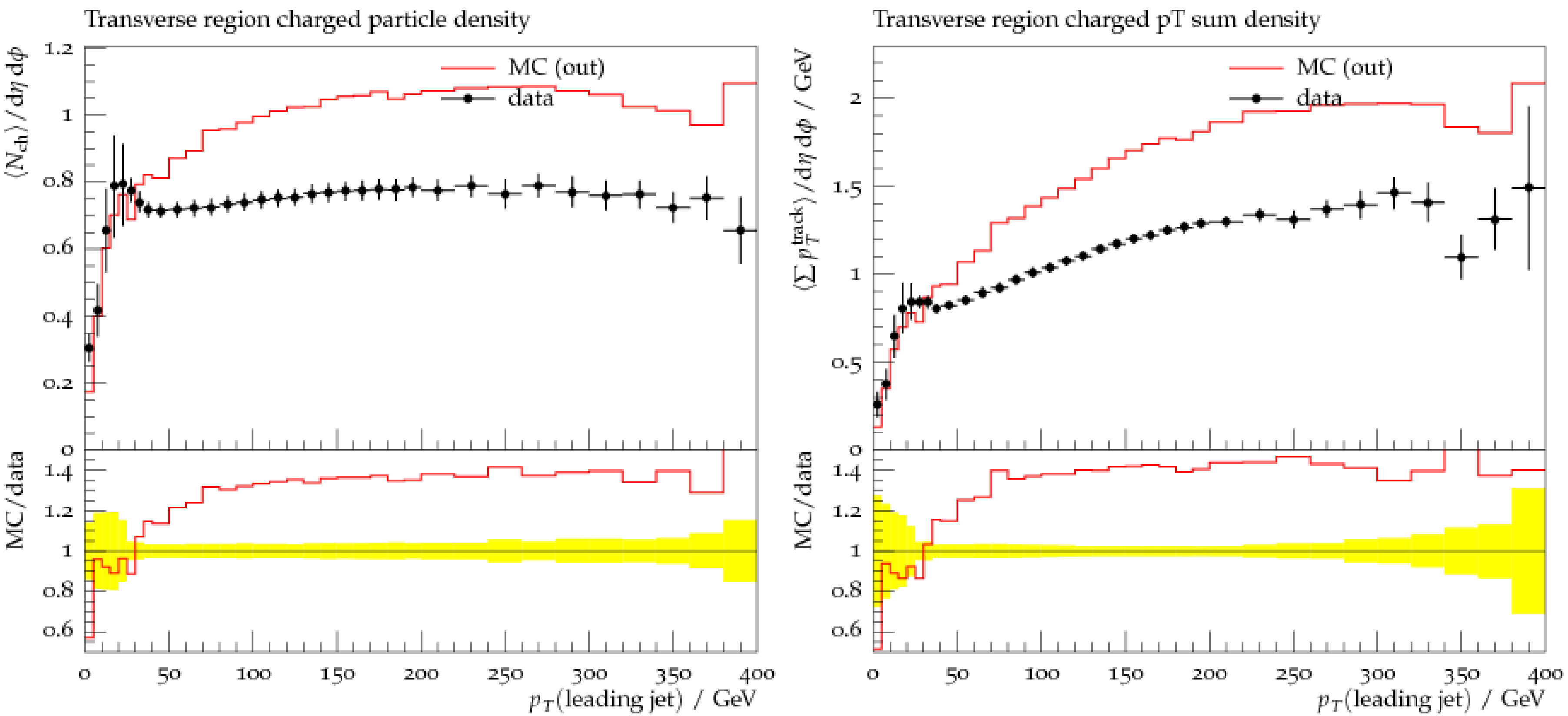
$\Rightarrow$ MPI & colour reconnection OK (?)

Also works for hard-physics distributions:



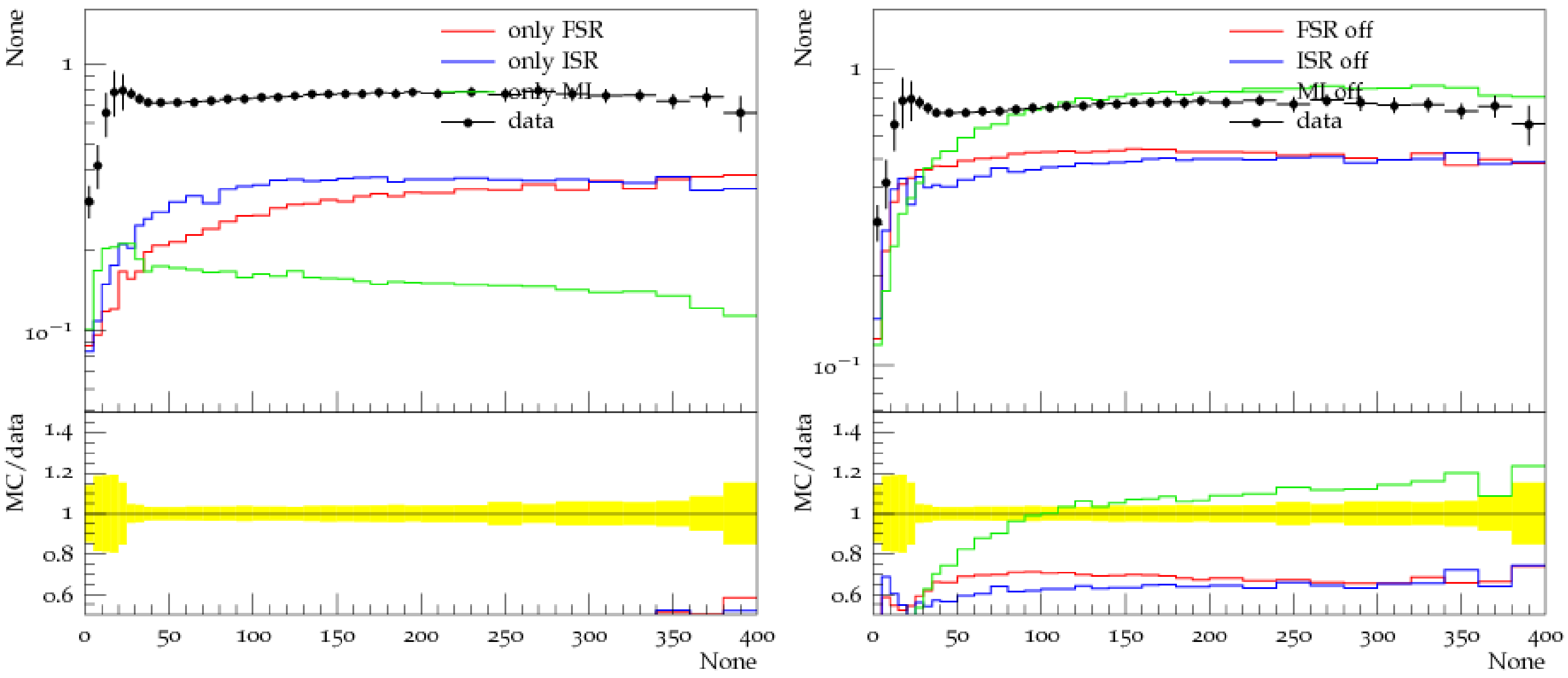
$\Rightarrow$ ISR OK (?)

But Rivet+Professor (H. Hoeth) shows it fails miserably for UE
(Rick Field's transverse flow as function of jet $p_\perp$):

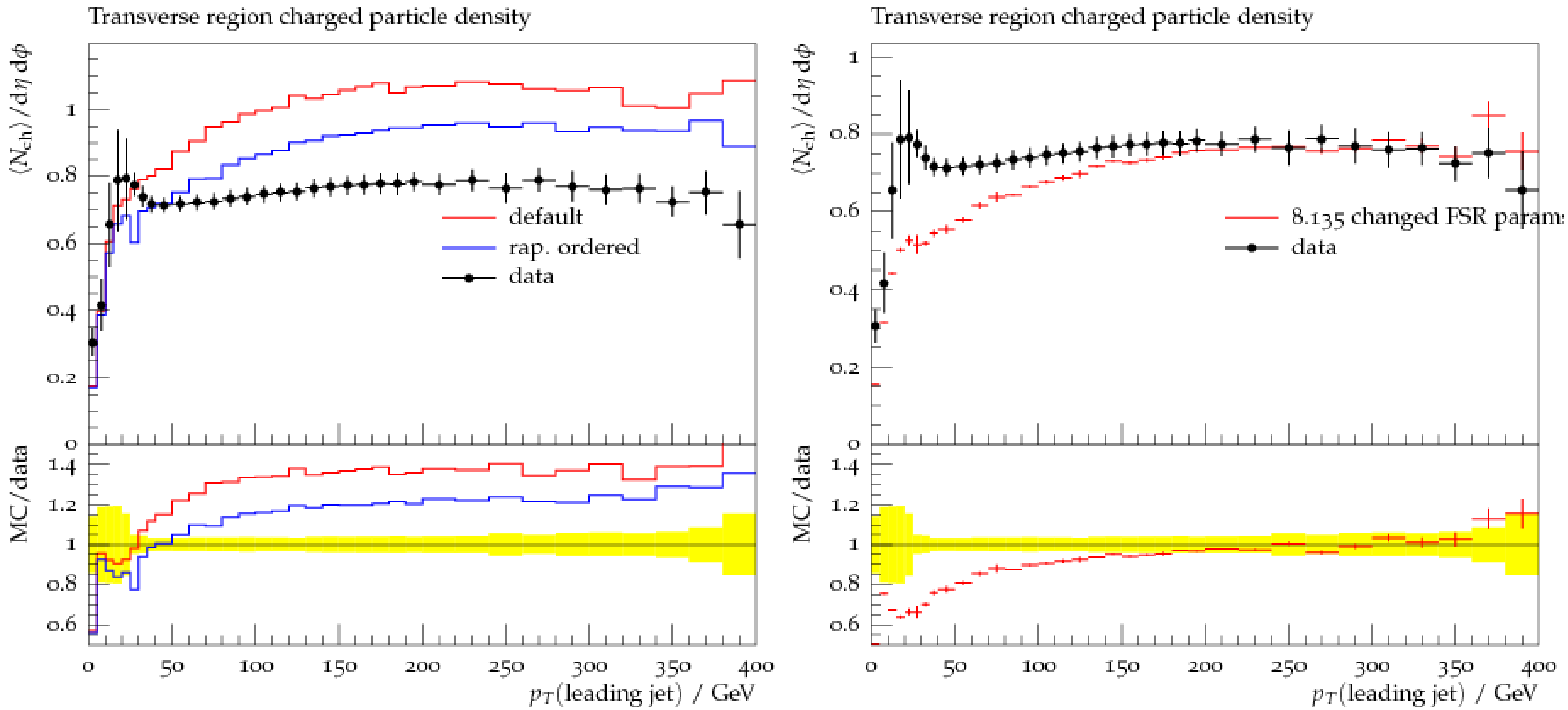


Where did we go wrong?

Studies component by component suggests FSR prime suspect, but ISR could also contribute:



Existing switches used to reduce ISR & FSR ($\sim$ PYTHIA 6.4) fail:



Likely culprit: dipoles where recoil from FSR is taken by beams.

Action: do explicit (semianalytical) study comparing $2 \to 3$ ME's with sum of all possible $2 \to 2$ ME's $\otimes$ PS branchings.

Now gearing up; results before summer (?).

Summary and Outlook

- PYTHIA6 is winding down:
  - $\star$ is supported but not developed;
  - $\star$ realistically main option for current run (sigh!),
  - $\star$ *but not after long shutdown 2012*!
- PYTHIA8 is the natural successor,
  - $\star$ is (sadly!) not yet quite up to speed in all respects,
  - $\star$ but in others already better than PYTHIA6,
  - $\star$ and will continue to move ahead.
- Advise to experimentalists:
  - $\star$ start to use PYTHIA8 to build up experience
  - $\star$ make people aware of existing documentation/examples
  - $\star$ allow the use of existing extensions (pointers!)
  - $\star$ if you want new features (e.g. ψ', Υ') then be prepared to use PYTHIA8
  - $\star$ provide feedback, both what works and what does not

There is no way back!