Summary of CMS Minimum Bias Pythia 8 Tunes

Hanyi (Livia) Yi

2013 NSF/REU Program

Physics Department, University of Notre Dame

Advisors:

Dr. Steve Merenna, Fermilab

Professor Mike Hildreth, University of Notre Dame

Professor Kevin Lannon, University of Notre Dame

Abstract

Pythia 8 is a computer program designed to simulate the physics processes that occur during a high energy collision, such as collisions between protons or electrons. Upgraded from Pythia 6, however, 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. Improving Pythia 8's performance in describing collider data involves two parts: first, adding new CMS analyses into Rivet, a toolkit for MC event generators and experimental data comparison, in order to expand the base of analyses to be compared to; second, running CMS analyses using Pythia 8 with different settings and compare the output data with real CMS data in order to get the most optimized global tune to describe collisions between protons. We use Professor-1.3.1 as a tuning instrument, and we come to the conclusion that the new tune provides a more accurate and reliable prediction for CMS collisions.

I Introduction

Event generators are software packages that generate simulated high-energy collision events. It is important to understand why they play such a critical role in high-energy physics. They share the same event reconstruction framework and subsequent physics analysis with the real machines such as LHC and Tevatron. Therefore, they help us gain knowledge of what may be going on in the real colliders by demonstrating to us how an original input is developed and distorted under relatively controlled physical conditions. Conversely, because of the fact that the event generators are programmed based on current knowledge of particle physics, the differences between the output data we get from them and the real LHC data shows to us the deficiencies of the current physics models and how we might improve them.

Pythia 8 is one of the major Monte Carlo event generators used for simulating LHC events. Monte Carlo methods rely on repeatedly generating random numbers running simulations numerous times in order to get statistically optimized outcomes to solve a problem. In Pythia, random numbers are used to make choices about what is going to happen next in order to reproduce the quantum mechanical probabilities for different results. The user interacts with Pythia in three stages: he first gives Pythia an initial setting where the tasks are to be performed; then Pythia runs by generating simulated events based on the specified conditions; after it finishes running, the final statistics are made available. (Torbj¨orn Sj¨ostrand, Stephen Mrenna and Peter Skands, October 2007)

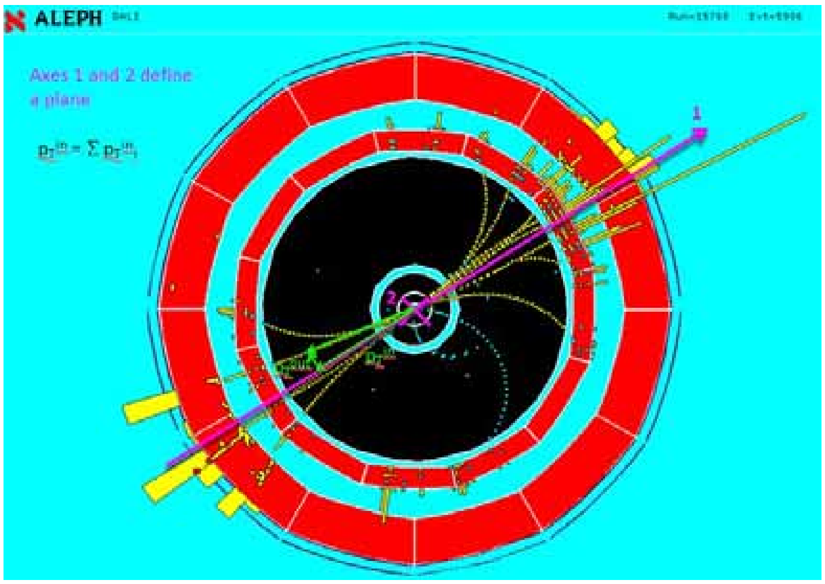
To tune Pythia 8 means to refine Pythia 8 predictions of the particle physics “truth”. Here, the "truth" is a list of particles produced from the proton-proton collision that will then propagate through the CMS detector. We use two major comparison tools to tune Pythia 8 – Rivet and Professor. Rivet is a validation system for MC generators. It produces plots from a piece of MC generator analysis code. In other words, one only needs to write the code once and the code can be used to validate and compare every generator that is able to simulate it. It makes sure that consistent comparisons are always done, which is important when trying to understand one generator compared to another one. It is useful for tuning because its analysis code and reference data can be used as an input.

We also use Professor as a major tuning instrument. Its advantage exists in its ability to interpolate the results of an analysis based on a generated grid of data. There are so many different settings that need to be tuned that without an automated program like Professor, one would need to generate an infinite number of MC samples to find the optimal settings. But once we run Pythia 8 with different settings and feed the output data to Professor, it is able to look for the optimization without automatically.

II Pythia 8 Tuning with regard to ALEPH Experiment

ALEPH was an experiment that ran at CERN before the LHC using the LEP collider. Although our main goal is to use data collected by the CMS experiment at the LHC, ALEPH experiment is a useful starting point because it has been used in tuning Pythia 6 for so long. We first reproduced analysis ALEPH_2004_S5765862 within the CMS framework, and tested Rivet comparison machinery and Professor tuning tools by finding Pythia 8's optimized parameter set with respect to the variable pT(in).

The pT(in) spectrum is the projection of the transverse momentum of charged particles onto the so-called "thrust axis" of the event, which is the axis of maximum energy flow, and it can be demonstrated more clearly by the following graph:



(Michael Hildreth, June 2013)

It is one of the variables that has always been hard to describe, so our goal is to produce a gradient map of pT(in) versus the Pythia 8 parameters. We did multiple runs, varying the parameters of the default tune one-at-a-time. In particular, we varied the following parameters each by 10% up and down, except for TimeShower:alphaSorder, which is either 1 or 2. We used Rivet to do statistical comparison between the real ALEPH data and MC data. The first piece of information we want is whether any of these settings improve the pT(in) description. The chi^2 map is produced for this purpose (the bolded ones are the default settings):

pT(in), E(CMS) = 189GeV

Chi^2/n	**TimeShower:**	TimeShower:

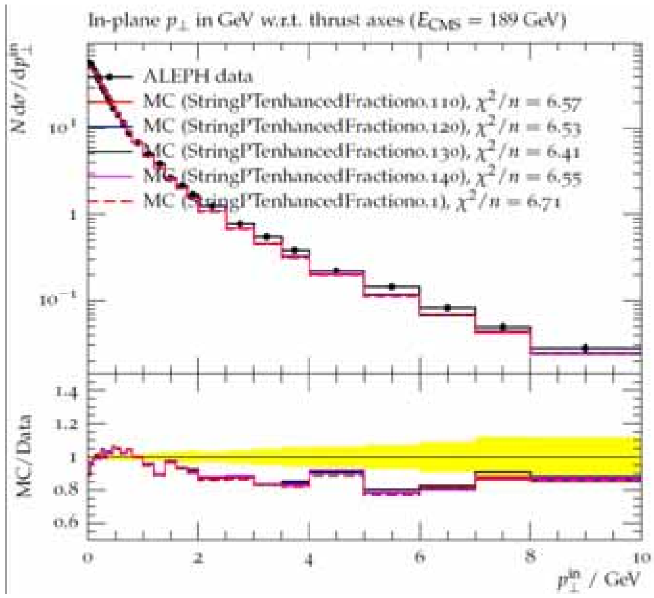
	alphaSorder = 1	alphaSorder = 2
StringPT:sigma = 0.30096	7.74	
StringPT:sigma = 0.304	**7.96**	
StringPT:sigma = 0.3344	6.88	
StringZ:aLund = 0.27	7.55	
StringZ:aLund = 0.33	8.65	
StringZ:bLund = 0.72	9.05	
StringZ:bLund = 0.88	6.73	
TimeShower:alphaSvalue = 0.12447	20.10	
TimeShower:alphaSvalue = 0.1383	**7.96**	
TimeShower:alphaSvalue = 0.15213	28.54	
TimeShower:alphaSvalue = 0.10620		63.77
TimeShower:alphaSvalue = 0.1180		22.93
TimeShower:alphaSvalue = 0.12980		10.95
TimeShower:pTmin = 0.36	7.19	20.73
TimeShower:pTmin = 0.40	**7.96**	
TimeShower:pTmin = 0.44	7.83	20.73
TimeShower:pTminChgQ = 0.36	7.97	
TimeShower:pTminChgQ = 0.40	**7.96**	
TimeShower:pTminChgQ = 0.44	8.20	

pT(in), E(CMS) = 189GeV, TimeShower:alphaSorder = 1

Chi^2/n		StringPT:enhancedFraction										
		0.0	**0.01**	0.1	0.13	0.15	0.2	0.25	0.3	0.4	0.5	1.0
StringPT :enhance dWidth	1.0			8.20	8.20	8.20	8.20	8.20	8.20	8.20	8.20	
	2.0	8.20	**7.96**	6.71	6.41	6.54	7.82	8.77	10.55	15.46	22.11	64.42
	3.0			7.23	8.42	9.68	14.68	20.56	28.84	45.87	66.84	
	4.0			8.99	12.41	15.61	25.25	37.89	51.99	83.26	117.90	
	5.0			11.46	17.56	22.84	37.24	53.56	74.78	119.37	165.05	
	6.0			14.31	23.07	29.42	49.78	73.47	98.56			
	7.0			17.23	26.93	35.73	60.38	88.37	119.11			
	8.0			19.81	32.41	42.04	70.25	101.31	136.23			
	9.0			21.92	36.58	47.33	79.59	113.94	153.90			
	10.0					52.72	86.72	125.96	167.57			

The two sensitive parameters we see are StringPT:enhancedFraction and StringPT:enhancedWidth. We increased the parameters in small steps of 0.1 to see the change in chi^2. It turns out that the chi^2 appears to be the smallest when StringPTenhancedFraction=0.13 and StringPT:enhancedWidth=2.0.

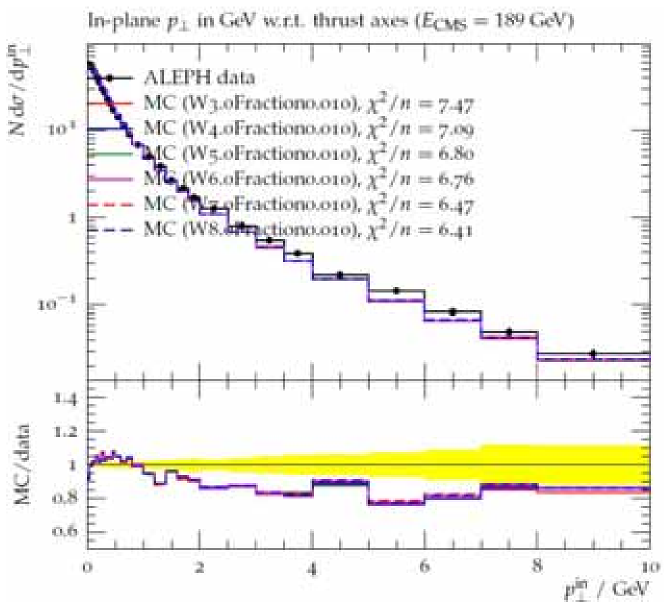
Furthermore, we also want to understand how the shape changes. For this we check the ratio plots.



However, because Rivet requires us to manually change parameter settings and run events, there could certainly be promising parameter sets that we missed. That is where our second tool comes in – the Professor project. Professor is relatively automated and can be used to do both full-scale tuning and local tuning.

We ran a 2-D grid with the parameter StringPT:enhancedFraction from 0.00 to 0.16 and StringPT:enhancedWidth from 1 to 8. Professor's interpolation told us that if we fix Fraction at a very small value, the chi^2 decreases when I increase Width to a certain extent. The minimum professor found was F=-0.00136, W=6.86.

We then used Rivet to test if Professor gave the right interpolation. Since Fraction can't be negative, I fixed it at F=0.01 (which is its default value) and increased Width. The comparison plots can be found below. We did not go look for the exact minimum because tuning to the ALEPH data was not a priority, but we could see the trend.



III Pythia 8 Tuning with regard to CMS Experiment

As I described earlier, we found a seemingly most-optimized parameter setting, and then Professor found us a new minimum, which we manually tested its validity using Rivet comparison tools. We consider Professor has proved its . We also took advice from *Summary of ATLAS Pythia 8 tunes*, in which the ATLAS Collaboration presented the latest ATLAS[1] Pythia 8 minimum bias and underlying event tunes.

We picked three analyses from CMS experiments database: CMS_QCD_10_006, CMS_QCD_10_010, and CMS_QCD_10_024. They are all CMS measurements of the underlying activities in the scattering processes in proton-proton collision. “Underlying” means the “soft” (e.g. low momentum”) part of the collision that happens because the two protons are breaking up. Although the “hard” part of the collision depends on exactly what particles were produced, the soft, underlying piece is the same for all collisions (i.e. universal).

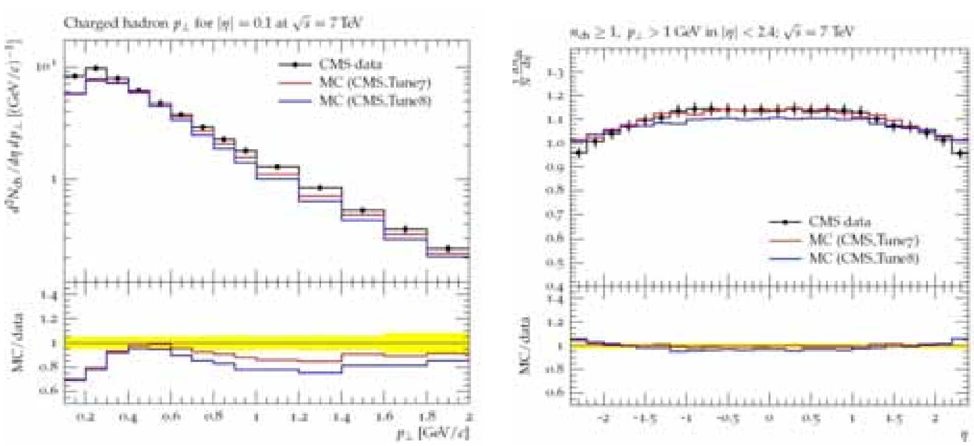
[1] Another detector at the LHC

Tune: pp 7 and Tune: pp 8 are both ATLAS Pythia 8 tunes. In fact, the ATLAS Collaboration did five tunes in total:

PDF	pT0Ref	ecomPow	a1	reconnectRange	Tune:pp
Minimum-bias tunes: A2					
CTEQ 6L1	2.18	0.22	0.06	1.55	7
MSTW2008 LO	1.90	0.30	0.03	2.28	8
Underlying event tunes: AU2					
CTEQ 6L1	2.13	0.21	0.00	2.21	9
NNPDF 2.1 LO	1.98	0.18	0.04	3.63	–
MSTW2008 LO	1.87	0.28	0.01	5.32	10
NNPDF 2.1 NLO	1.74	0.17	0.08	8.63	–
CTEQ 6.6	1.73	0.16	0.03	5.12	–
CT10	1.70	0.16	0.10	4.67	11
MSTW2008 NLO	1.51	0.19	0.28	5.79	–
MRST2007 LO*	2.39	0.24	0.01	1.76	–
MRST2007 LO**	2.57	0.23	0.01	1.47	–

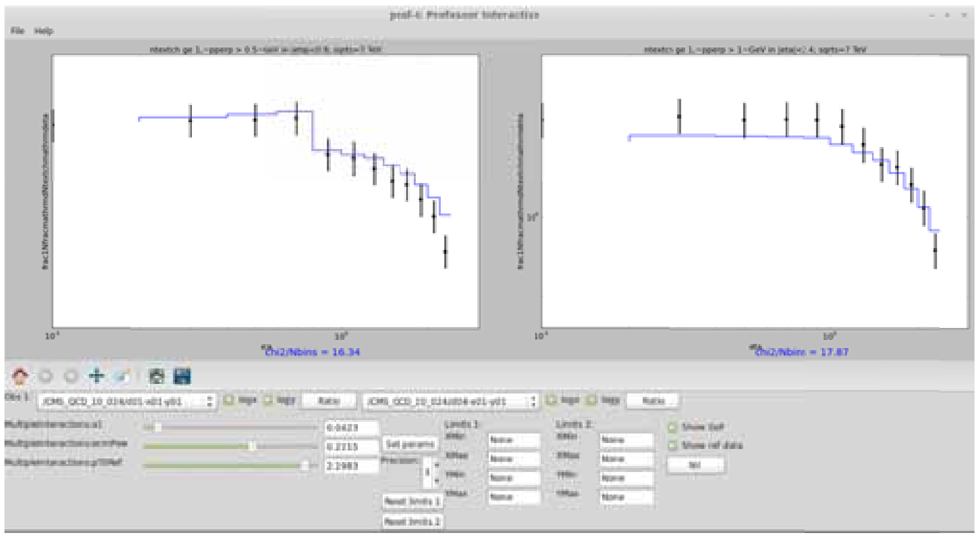
Table 5: Tuned MPI parameters for the A2/AU2 Pythia 8 tunings.

We did a simple rivet comparison of the CMS measurements with the two ATLAS tunes, and Tune: pp 7 appears to behave better in general for CMS analysis. Therefore, we focused on tuning Tune: pp 7 for now.



The parameters we tuned are MultipleInteractions:pT0Ref (subsequently referred to as pT0Ref), MultipleInteractions:ecmPow (subsequently referred to as ecmPow) and MultipleInteractions:a1 (subsequently referred to as a1). We ran a 5*6*5 grid and fed the output data to Professor, with pT0Ref varying through 2.16, 2.17, 2.18, 2.19

and 2.20, ecmPow varying through 0.19, 0.20, 0.21, 0.22, 0.23 and 0.24, and a1 varying through 0.04, 0.05, 0.06, 0.07 and 0.08.



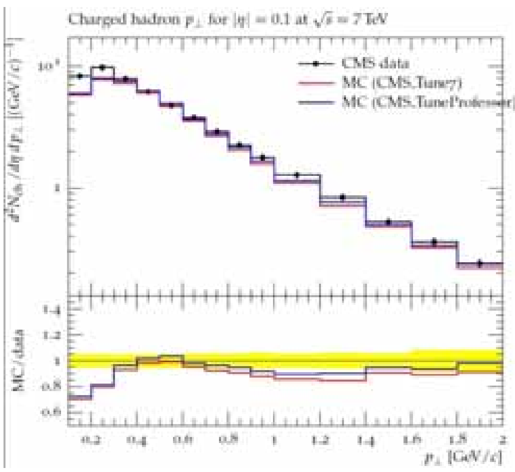
The optimization Professor found with the goodness of fit calculated turned out to be:

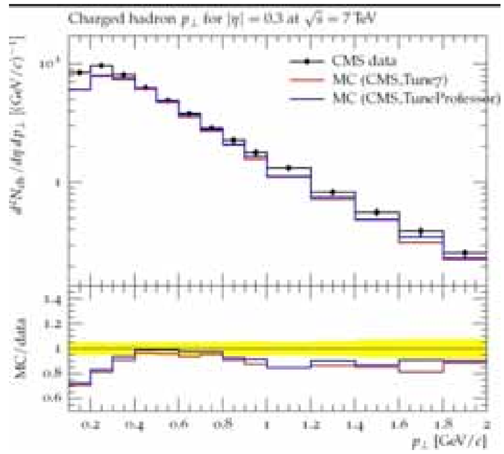
Parameter	Value
MultipleInteractions:pT0Ref	2.171734e+00
MultipleInteractions:ecmPow	2.045149e-01
MultipleInteractions:a1	4.024240e-02

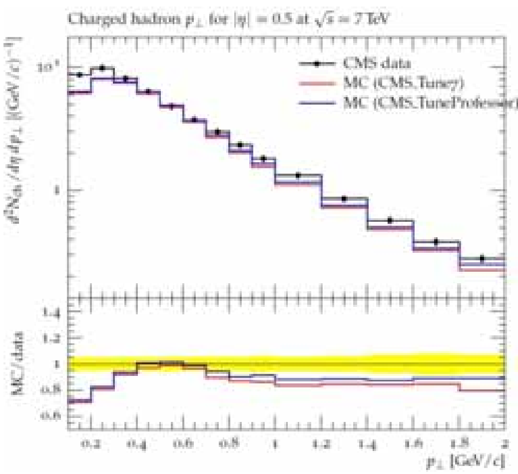
Goodness of fit: 2.93e+04; Ndf: 602; Goodness of fit/Ndf: 4.87e+01

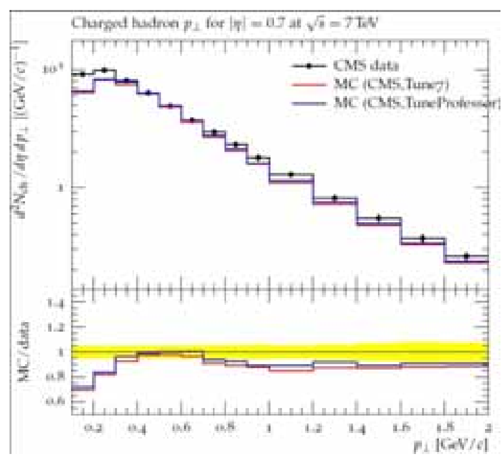
IV Summary

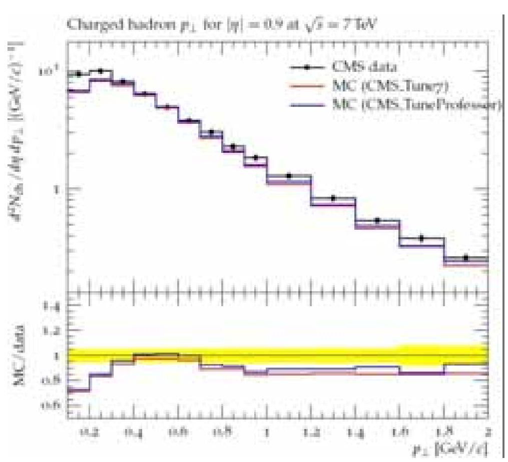
In this paper I have presented a new CMS tune of Pythia 8 event generators. The comparison between this new Pythia 8 tune and Tune: pp 7 can be found below:

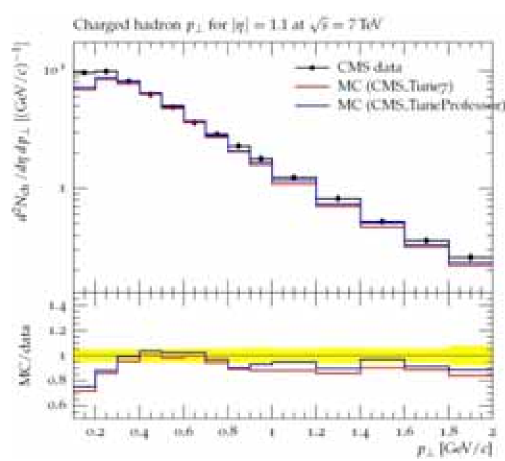

Charged hadron $p_\perp$ for $|\eta| = 0.1$ at $\sqrt{s} = 7$ TeV
CMS data
MC (CMS,Tune7)
MC (CMS,TuneProfessor)
MC/data
$p_\perp$ [GeV/c]

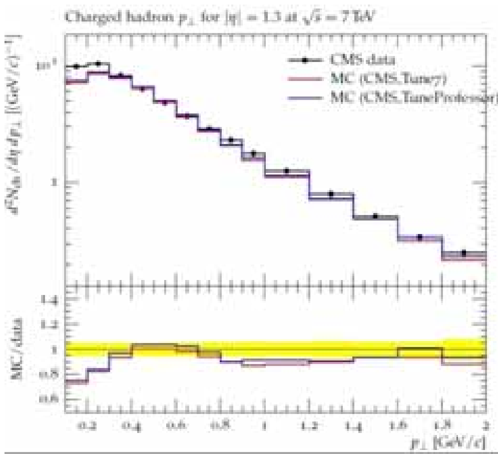

Charged hadron $p_\perp$ for $|\eta| = 0.3$ at $\sqrt{s} = 7$ TeV
CMS data
MC (CMS,Tune7)
MC (CMS,TuneProfessor)
MC/data
$p_\perp$ [GeV/c]

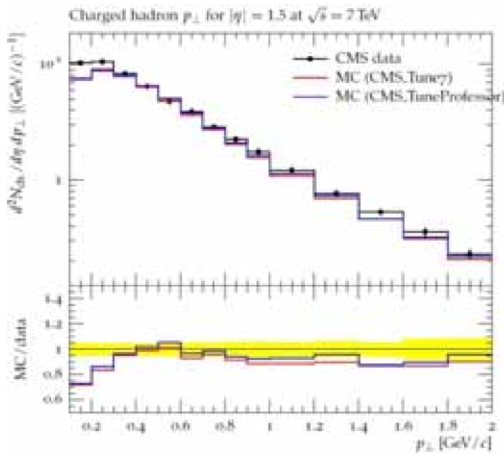

Charged hadron $p_\perp$ for $|\eta| = 0.5$ at $\sqrt{s} = 7$ TeV
CMS data
MC (CMS,Tune7)
MC (CMS,TuneProfessor)
MC/data
$p_\perp$ [GeV/c]

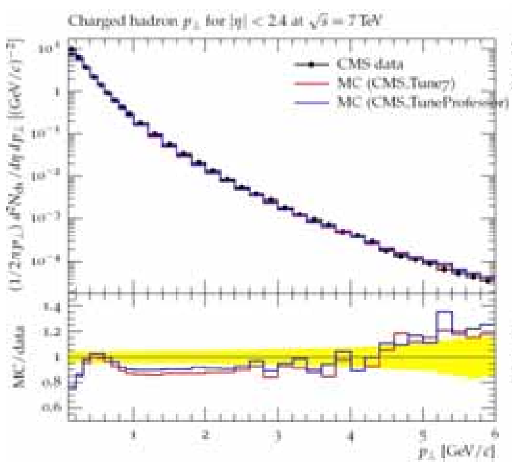

Charged hadron $p_\perp$ for $|\eta| = 0.7$ at $\sqrt{s} = 7$ TeV
CMS data
MC (CMS,Tune7)
MC (CMS,TuneProfessor)
MC/data
$p_\perp$ [GeV/c]

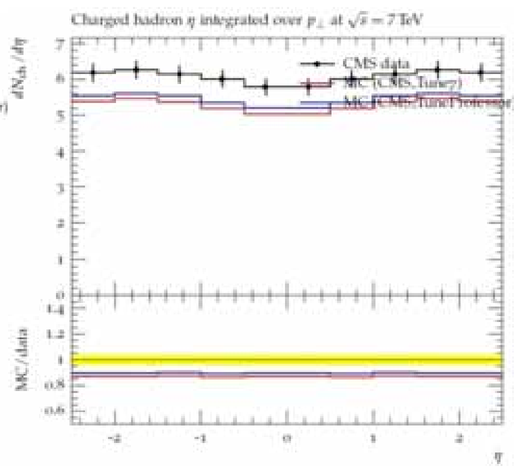

Charged hadron $p_\perp$ for $|\eta| = 0.9$ at $\sqrt{s} = 7$ TeV
CMS data
MC (CMS,Tune7)
MC (CMS,TuneProfessor)
MC/data
$p_\perp$ [GeV/c]

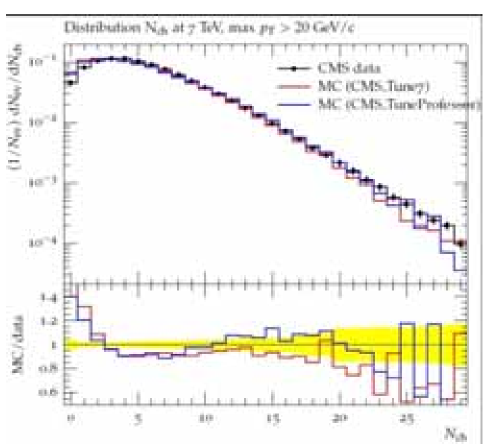

Charged hadron $p_\perp$ for $|\eta| = 1.1$ at $\sqrt{s} = 7$ TeV
CMS data
MC (CMS,Tune7)
MC (CMS,TuneProfessor)
MC/data
$p_\perp$ [GeV/c]

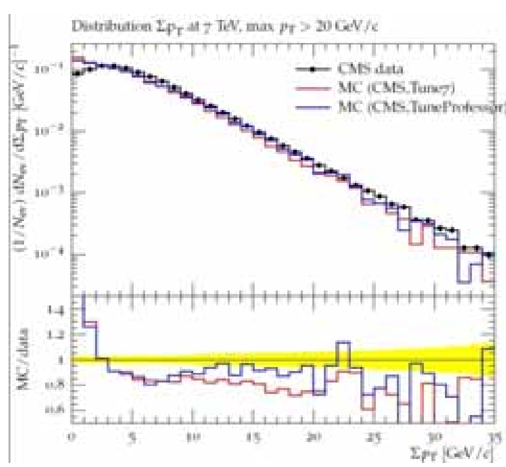












In the meantime, we have also coded up a new CMS analysis –

CMS_QCD_10_029 and put it into Rivet, for the purpose of benefiting future generator comparison and generator tuning. The work we have done this summer is just a beginning. There are many more data sets to compare and parameters in Pythia to be tuned. We have already started looking at additional measurements, and we will try to do more extensive tuning in the future.

References

Torbj¨orn Sj¨ostrand, Stephen Mrenna and Peter Skands, *A Brief Introduction to Pythia 8.1*, CERN-LCGAPP-2007-04, FERMILAB-PUB-07-512-CD-T, LU TP 07-28, October 2007

Torbj¨orn Sj¨ostrand, *Monte Carlo Generators*, hep-ph/0611247, CERN-LCGAPP-2006-06, November 2006

The ATLAS Collaboration, *Summary of ATLAS Pythia 8 tunes*, ATL-PHYS-PUB-2012-003, August 22, 2012