

# Table of Contents

<b>Notes on PYTHIA8 tuning using PROFESSOR in ATLAS.....</b>	<b>1</b>
PYTHIA 8.....	1
PYTHIA8 in ATLAS.....	1
PROFESSOR.....	1
RIVET.....	2
LHAPDF.....	2
"Official" ATLAS Stuff.....	2
Tuning Workflow.....	2
On NAF.....	4
Tuning Settings.....	4
Scripts/Commands.....	5
AGILe.....	6
How to?.....	6

# Notes on PYTHIA8 tuning using PROFESSOR in ATLAS

## PYTHIA 8

- Documentation: (<http://home.thep.lu.se/~torbjorn/php8145/Welcome.php>)
- To run in any other directory than /example: export  
PYTHIA8DATA=/home/dkar/atlas/tuning/pythia\_with\_old\_hepmc/pythia8145/xmldoc
- export LD\_LIBRARY\_PATH=\$LD\_LIBRARY\_PATH:\$PWD/../local/lib
- To compile: make main32 HEPMCLOCATION=\$PWD/../local
- To run: ./main32.exe main32.cmnd test.out (or my.hepmc to pipe output through Rivet)

## PYTHIA8 in ATLAS

- Current Samples:  
([http://panda.cern.ch/server/pandamon/query/?mode=taskquery&qTaskName=mc10\\_7%25831%25Pythia8%25](http://panda.cern.ch/server/pandamon/query/?mode=taskquery&qTaskName=mc10_7%25831%25Pythia8%25))
- For development: check out and rebuild the Generators/Pythia\_i head.

Pythia8 most writes output in the HepMC GenEvent output format, which can be read into Atlas using ReadEventFromFile in GenAnalysisTools:

```
from AthenaCommon.AlgSequence import AlgSequence
topSequence = AlgSequence()

from ReadEventFromFile.ReadEventFromFileConf import ReadHepMc
read = ReadHepMc()
read.File = "filename.hepmc2g"
topSequence += read

theApp.EvtMax = 500
```

asetup 16.6.5.2

```
Evgen_trf.py 7000 108316 1 -1 1 MC11.108316.Pythia8_minbias_ND.py evgen.root | tee test.out
```

cmE, DSID, START\_RUN\_NO, END\_RUN\_NO, JO, OUTPUT, LOG

<https://twiki.cern.ch/twiki/bin/view/AtlasProtected/RivetForAtlas>

```
source /afs/cern.ch/sw/lcg/external/MCGenerators/rivet/1.5.0/i686-slc5-gcc43-opt/rivetenv.sh
```

```
and rivet-buildplugin RivetATLAS_MINBIAS_MB20.so ATLAS_MINBIAS_MB20.cc -m32
```

## PROFESSOR

- Documentation: (<http://projects.hepforge.org/professor/trac/>)
- Tutorial: (<http://projects.hepforge.org/professor/prof-tutorial.pdf>)

## RIVET

- Documentation (<http://projects.hepforge.org/rivet/trac/wiki/>)

- Tutorial: (<http://projects.hepforge.org/rivet/rivet-tutorial.pdf>)
- Useful RIVet commands:
  - ◆ rivet --list-analyses (-v ATLAS\_)
  - ◆ rivet --analysis=ANALYSIS\_NAME hepmc.fifo
  - ◆ rivet -a analysis1 -a analysis2 hepmc.fifo (or -H results.aida, default Rivet.aida)
  - ◆ Need to rm my.hepmc, and mkfifo my.hepmc
- Analysis (ATLAS): Track UE: ATLAS\_2010\_S8894728, Cluster UE: ATLAS\_2010\_S8994773, MInbias 2.0:ATLAS\_2010\_S8918562
- export RIVET\_PLOT\_PATH, RIVET\_REF\_PATH, RIVET\_ANALYSIS\_PATH, RIVET\_INFO\_PATH
- Documentation (<https://savannah.cern.ch/projects/hepmc/>)
- Latest version 2.06.03 did **not** work with PYTHIA 8.145, switched to 2.05.01
- Download and install: (./configure --prefix=\$PWD/./local/ --with-momentum=GEV --with-length=MM; make; make install; use HEPMCLOCATION=\$PWD/./local OR mkdir build install, work in build, ../HepMC-2.05.01/configure -prefix=\$PWD/./hepmc/install -with-momentum=GEV -with-length=MM and then make; make install; reconfigure PYTHIA with ./configure --with-hepmc=\$PWD/hepmc/hepmc/install --with-hepmcversion=2.05.01; (re)make. )

## LHAPDF

- Documentation (<http://projects.hepforge.org/lhapdf/manual>)
- Download and Install (./configure --prefix=\$PWD/./local/; make; make install; Use bin/lhapdf-getdata to put PDFsets in share/lhapdf/PDFsets/)
- Link: export LHAPATH=\$PWD/./local/share/lhapdf/PDFsets

PDF Sets: MRSTMCal: LO\*\*, CT10.LHgrid" LO\*, CTEQ6L1: LO

```
./configure --with-hepmc=$PWD/./local/ --with-hepmcversion=2.05.01 --with-lhapdf=$PWD/./local/lib/
```

```
g++ -O2 -ansi -pedantic -W -Wall -Wshadow -fbounds-check -Wno-shadow -I../include -I/home/dkar/atlas/bin/main32.cc -o ../bin/main32.exe \
-L../lib/archive -lpythia8 \
-lhepmcinterface \
-L/home/dkar/atlas/tuning/pythia_with_old_hepmc/pythia8145/runpythia/./local/lib -lHepMC \
-L/home/dkar/atlas/tuning/pythia_with_old_hepmc/pythia8145/runpythia/./local/lib -lLHAPDF
```

```
g++ -O2 -ansi -pedantic -W -Wall -Wshadow -fbounds-check -Wno-shadow -I../include -I $HOME/tuning
```

## "Official" ATLAS Stuff

- Get: `svn co svn+ssh://svn.cern.ch/repos/atlasoff/Generators/Tuning/trunk/firstdata/analyses`
- Rivet: need Rivet\*\*\*.so and \*.aida files to include an internal analysis and link: `export RIVET_ANALYSIS_PATH=$PWD`

## Tuning Workflow

1. Setup the directory structure. Create mc/XXX directories. [NOT NEEDED]. Script: `make_mkdir.sh`
1. Decide which parameters to tune.

- ◆ Start off by MPI tune.
- ◆ By prof-scanparams -> prepares used\_params files. They should reside in mc/XXX/.
- ◆ Use: prof-scanparams -i \$PWD/./mc -o \$PWD/./mc -N 100 mpi\_params. (created MC/XXX dirs too)/ prof-sampleparams -o \$PWD/./ -N 5 mpi\_params

1. Decide which analysis/datasets to tune to.

- ◆ For MPI tune (and to avoid jet slices, for now), use ATLAS 900 and 7000 MB and UE (track and cluster), CDF 1.8 and 1.96 MB. (8 in total).
- ◆ Only needs to generate soft QCD.
- ◆ For each mc/XXX, there will be different directories corresponding to cmE.

1. Generate MC samples.

- ◆ Input run\_params\_cmE (from run directory) and used\_params (from /mc/XXX/)
- ◆ The scripts to run will be run\_cmE.sh, where cmE will determine which run\_params\_cmE file and Rivet analysis is included.
- ◆ Arguments to the run script: begin-run-no (000) end-run-no (099)
- ◆ Should result in mc/XXX/cmE/Rivet.aida for each cmE.
- ◆ Merge by: merge\_runs.sh (change end-run-no to argument)
- ◆ Extend 1 param: previous number of runs x Delta(p)\_ext/Delta(p)\_prev

2. Weights file with weights assigned to different analysis.

- ◆ Use Pythia6 MPI tuning file?
- ◆ :X:Y denotes X-axis range where to tune to. For different weights on different ranges, multiple lines corresponding to the same histogram.
- ◆ Copy reference data from /afs/cern.ch/atlas/groups/Generators/Tuning/firstdata/allref, store them in "ref" directory.
- ◆ To extract all analysis: sed -n 's/AidaPath/&/p' filename >
- ◆ To add weight "1.0" to all: sed "s/\$/ 1.0/g" filename >

1. Look at how runs envelop the data.

- ◆ prof-envelopes --mcdir mc --datadir ref -o envelopes1 --weights w
- ◆ make-plots envelopes2/envelopes/\*.dat --pdf
- ◆ ./makegallery.py -s envelopes/envelopes/ pdf envelopes.html
- ◆ For just one run comparison: compare-histos Rivet.aida; make-plots --pdf \*.dat; then makegallery. [maximal all-in-one way: rivet-mkhtml mc1.aida:"MC1 label" mc2.aida"MC2 label"]

1. Create a runcombs-file which contains combination of MC runs to be used.

- ◆ By prof--runcombs. (prof-runcombs --mcdir mc -c 0:1)

1. Parameterise the generator response. By prof-interpolate, results in folder ipols that contains a generator parameterisation file.

- ◆ prof-interpolate --mcdir mc --datadir ref --obsfile w2 --runsfile runcombs.dat (-o ipol)
- ◆ Interpolation set written to ref/ipol

1. Finally tune. Using prof-tune, stored in results/.

- ◆ prof-sensitivities --datadir ref --ipoldir ipol --runsfile runcombs.dat --obsfile w2 --plotmode extremal -o sensitivity\_plots
- ◆ prof-tune --datadir . --weights w4 --ipoldir ref/ipol --runs runcombs.dat (stored in tunes)
- ◆ prof-sensitivities --mcdir mc --datadir . --runs runcombs.dat --weights wt\_04 --plotmode extremal -o sensitivity\_plots --ipol cubic --ipoldir ref/ipol
- ◆ prof-sensitivities --mcdir mc --datadir . --runs runcombs.dat --weights wt\_04 --plotmode colormap -o sensitivity\_plots --ipol cubic --ipoldir ref/ipol
- ◆ Show the sensitivity of observables to the parameters varied: savesensitivities.py --datadir . --outdir splots --observables weights1

- ◆ Show how well the generator runs "enclose" data: prof-envelopes --mcdir mc --refdir ref -o envelopes --weights weights
- ◆ Show scatter plot for each tuning parameter.

1. Get a way to compare to jetX data, run jetslices and merge. [HS: n Pythia6 we used the following CKIN(3) cuts for the QCD runs: 0, 10, 20, 50, 100, 150]

Note: if you source the local/env.sh script, you will get the Genser versions of AGILe (1.2.2), Rivet (1.5.0), and Professor (1.2.1) in your environment. If you *additionally* source the local/rivetenv.sh file, you will change the Rivet version to the ATLAS tuning build of Rivet 1.4.0.

So analyses which want to keep using Rivet 1.4.0 **for now** can keep sourcing rivetenv.sh. For everyone else, and particularly if you are starting something new, please only source the env.sh script and use Rivet 1.5.0 from Genser. At some point -- when the Py8 runs are done! -- we'll completely remove the local builds.

## On NAF

- source /afs/desy.de/project/glite/UI/etc/profile.d/grid-env.sh OR source /afs/cern.ch/project/gd/LCG-share/current/external/etc/profile.d/grid-env.sh
- voms-proxy-init -rfc
- gsissh atlas.naf.desy.de
- qsub [resource requirements] script [script parameter]
- If you have many jobs using the same script and you want to parallelize them, you can make an array job out of it with the switch: -t from-to:step

## Tuning Settings

- MPI Tune parameters: (/afs/cern.ch/atlas/groups/Generators/Tuning/firstdata/more tune\_11\_mpi/parameter.ranges)

MultipleInteractions:pT0Ref (0.5 -10, 2.15)

MultipleInteractions:ecmPow (0-5, 0.24)

MultipleInteractions:pTmin (0.1 -10, 0.2)

For :bProfile = 2

MultipleInteractions::coreRadius (0.1-1-, 0.4)

MultipleInteractions:coreFraction (0 -1, 0.5)

For bProfile=3

MultipleInteractions:expPow (0.1 -10, 1)

- PYTHIA8 Run settings:

The main32 example is very useful for running Rivet. It reads a runcard and writes HepMC events.

```
#fixed_params
```

```
PDF:pSet = 8
```

```
MultipleInteractions:bProfile = 3 _(double gaussian=2)_
```

```
MultipleInteractions:alphaSvalue = SigmaProcess:alphaSvalue
```

```
SpaceShower:rapidityOrder = on
```

```
#UE200.params
```

```
! 1) Settings that will be used in a main program.
```

Main:numberOfEvents = 3000000 ! number of events to generate  
 Main:timesToShow = 1000 ! show how far along run is this many times  
 Main:timesAllowErrors = 30 ! abort run after this many flawed events  
 Main:showChangedSettings = on ! print changed flags/modes/parameters  
 #Main:showAllSettings = on ! print all flags/modes/parameters  
 Main:showChangedParticleData = on ! print changed particle and decay data  
 #Main:showAllParticleData = on ! print all particle and decay data

! 2) Beam parameter settings. Values below agree with default ones.

Beams:idA = 2212 ! first beam, p = 2212, pbar = -2212  
 Beams:idB = 2212 ! second beam, p = 2212, pbar = -2212  
 Beams:eCM = 200 ! CM energy of collision

! 3) Pick processes and kinematics cuts

SoftQCD:minBias on

### Tune Parameters:

Parameter	Tune 2C	Tune 2M	Tune 4C
SigmaProcess:alphaSvalue	0.135	0.1265	0.135
SpaceShower:rapidityOrder	on	on	on
SpaceShower:alphaSvalue	0.137	0.130	0.137
SpaceShower:pT0Ref	2.0	2.0	2.0
MultipleInteractions:alphaSvalue	0.135	0.127	0.135
MultipleInteractions:pT0Ref	2.320	2.455	2.085
MultipleInteractions:ecmPow	0.21	0.26	0.19
MultipleInteractions:bProfile	3	3	3
MultipleInteractions:expPow	1.60	1.15	2.00
BeamRemnants:reconnectRange	3.0	3.0	1.5
SigmaDiffractive:dampen	off	off	on
SigmaDiffractive:maxXB	N/A	N/A	65
SigmaDiffractive:maxAX	N/A	N/A	65
SigmaDiffractive:maxXX	N/A	N/A	65

R. Corke & TS, arXiv:1011.1759 [hep-ph]

## Scripts/Commands

```

> source profenv.sh
> source /afs/cern.ch/sw/lcg/external/MCGenerators/professor/1.2.1/x86_64-slc5-gcc43-opt/setup.sh
> source /afs/cern.ch/sw/lcg/external/MCGenerators/rivet/1.5.0/x86_64-slc5-gcc43-opt/rivetenv.sh

```

### HS:

```

> prof-runcombs -c 0:1 --mcdir
/afs/cern.ch/atlas/groups/Generators/Tuning/firstdata/tune_10_shower/mc3-min -o runcombs_0_1.dat

> prof-interpolate --mcdir /afs/cern.ch/atlas/groups/Generators/Tuning/firstdata/tune_10_shower/mc3-min
--runs runcombs_0_1.dat -o test --ipol cubic --weights
/afs/cern.ch/atlas/groups/Generators/Tuning/firstdata/tune_10_shower/weights04

> prof-tune --runs runcombs_0_1.dat --ipoldir test/ipol --ipol cubic --weights
/afs/cern.ch/atlas/groups/Generators/Tuning/firstdata/tune_10_shower/weights04 -o test --refdir

```

*/afs/cern.ch/atlas/groups/Generators/Tuning/firstdata/allref/*

## AGILe

Use v1.2.1. *how to pass parameters?*

```
> source /afs/.cern.ch/sw/lcg/external/MCGenerators/agile/1.2.0/x86_64-slc5-gcc43-opt/agileenv.sh
```

```
> agile-runmc Pythia6:425 --beams=LHC:7000 -p PYTUNE=341 -p "CKIN(3)=10" -n 100 -o test.out
```

## How to?

```
uname -m gcc --version
```

```
-- DeepakKar - 25-Feb-2011
```

---

This topic: Main > PythiaProf

Topic revision: r30 - 2014-04-23 - DeepakKar



Copyright &© 2008-2019 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback