Roteiro de Atividades do Doutorado
2º Semestre de 2017

André Vieira

29 de Agosto de 2017

Resumo

Na busca por uma compreensão dos eventos de raios cósmicos em altas energias, estamos interessados em usar o ferramental computacional fornecido pelos geradores FPMC, PYTHIA e o EPOS-LHC, com o objetivo de simular processos elementares que podem ocorrer em colisões em altas energias na atmosfera.

Conteúdo

1 Introdução

Em um gerador de eventos, o objetivo principal é usar computadores para gerar eventos tão detalhados, tal como podem ser observados por um detector perfeito, ou seja, com 100 % de eficiencia, para um caso mais próximo da realidade são utilizados simuladores de detectores, por exemplo, o GEANT4 [1].

A função do gerador de eventos é simular um comportamento médio e as flutuações quânticas dos dados reais. Nos geradores, as técnicas de Monte Carlo[1] são utilizadas para selecionar as variáveis relevantes de acordo com as distribuições de probabilidade desejadas, assim, garantir a aleatoriedade de uma forma aproximada das partículas no estados finais.

Not Possible without Monte Carlo!

Monte Carlos generators allow to include theoretical models, phase space integration in multiple dimensions, inclusion of detector effects, and efficiency and acceptance determination for new physics processes [Lecture 6: Monte Carlo Generators Simulating high energy pp-collisions ...][a].

[a] https://www.scribd.com/document/212643325/Lecture-6

2 Geradores

2.1 FPMC - Forward Physics Monte Carlo [1, 2]

```
//DataCard: Single Diffraction with Dijets.
OUTPUT      1
OUTPUTLHE   0
NTNAME      'SDDijets.ntp' //h2root ---> 'SDDijets.root'
LHEFILE     'FPMC_SDDijets_13TeV_PTMIN_20.lhe'
MAXEV       1000     // Number of collisions
PART1       'E+'     // Type of beam 1 particle
PART2       'P'      // Type of beam 2 particle
TYPEPR      'INC'    // Inclusive reaction
TYPINT      'QCD'    // Switch between QED and QCD process
ECMS        13000.   // GeV
IPROC       11500    // Type of process to generate
NFLUX       9        // QCD factorized model, Pomeron flux
NRN1        33781
NRN2        11776
YJMAX       6.
YJMIN      -6.
YWWMIN      0.
```

[1] http://home.thep.lu.se/~torbjorn/talks/cern12a.pdf

```
YWWMAX      0.1
PTMIN       20.
IFIT        100
```

2.2 PYTHIA 8226 [3]

```
// Set up generation.

  // Interface for conversion from Pythia8::Event to HepMC event.
  HepMC::Pythia8ToHepMC ToHepMC;

  // Specify file where HepMC events will be stored.
  HepMC::IO_GenEvent ascii_io("hepmcout41.hepmc", std::ios::out);

  // Generator. Process selection. LHC initialization. Histogram.
  Pythia pythia;
  pythia.readString("Beams:eCM=8000.");

  //                 ProcessGroup:ProcessName
  pythia.readString("HardQCD:all=on");
  pythia.readString("PhaseSpace:pTHatMin=20.");
  pythia.init();

//-------------------------------------------------------------
// Create Pythia instance and set it up to generate hard
// QCD processes above pTHat = 20 GeV for pp collisions at 14 TeV.
  Pythia pythia;

  // Initialisation, p pbar @ 14 TeV
  //                 ProcessGroup:ProcessName
  pythia.readString("HardQCD:all=on");
  pythia.readString("Beams:idB = -2212");
  pythia.readString("Beams:eCM=14000.");
  pythia.readString("PhaseSpace:pTHatMin=20.");
  pythia.init();   //  pp default

 // init(...) call is executed that all the settings values are prop-
 // agated to the various program elements, and used to precalculate
 // quantities that will be used at later stages of the generation.

  // Set up the ROOT TFile and TTree.
  TFile *file = TFile::Open("pytree.root","RECREATE");
  Event *event = &pythia.event;
  TTree *T = new TTree("T","Tree");
  T->Branch("event",&event);
  T->Branch("px",&px);
```

```
  T->Branch("py",&py);
  T->Branch("pz",&pz);
  T->Branch("En",&En);


  //-----------------------------------------------------------

"ProcessGroup":ProcessName

"SoftQCD":minBias,elastic, singleDiffractive,
          doubleDiffractive (or 'all')
"HardQCD":gg2gg, gg2qqbar, qg2qg, qq2qq, qqbar2gg, qqbar2qqbarNew,
          gg2ccbar, qqbar2ccbar, gg2bbbar, qqbar2bbbar
"PromptPhoton":qg2qgamma, qqbar2ggamma, gg2ggamma,
               ffbar2gammagamma, gg2gammagamma
"WeakBosonExchange": ff2ff(t:gmZ), ff2ff(t:W)
"WeakSingleBoson": ffbar2gmZ, ffbar2W, ffbar2ffbar(s:gm)
"WeakDoubleBoson": ffbar2gmZgmZ, ffbar2ZW, ffbar2WW
"WeakBosonAndParton": qqbar2gmZg, qg2gmZq, ffbar2gmZgm, fgm2gmZf
                      qqbar2Wg, qg2Wq, ffbar2Wgm, fgm2Wf
"Charmonium": gg2QQbar[3S1(1)]g, qg2QQbar[3PJ(8)]q, ...
"Bottomonium": gg2QQbar[3S1(1)]g, gg2QQbar[3P2(1)]g, ...
"Top": gg2ttbar, qqbar2ttbar, qq2tq(t:W),ffbar2ttbar(s:gmZ),
       ffbar2tqbar(s:W) FourthBottom, FourthTop,
       FourthPair (fourth generation)
"HiggsSM": ffbar2H, gg2H, ffbar2HZ, ff2Hff(t:WW), ...
"HiggsBSM" :h, H and A as above, charged Higgs, pairs
"SUSY": qqbar2chi0chi0 (not yet completed)
"NewGaugeBoson": ffbar2gmZZprime, ffbar2Wprime, ffbar2R0
"LeftRightSymmmetry": ffbar2ZR, ffbar2WR, ffbar2HLHL, ...
"LeptoQuark": ql2LQ, qg2LQl, gg2LQLQbar, qqbar2LQLQbar
```

2.3 CRMC v1.6.0: EPOS LHC [4]

```
Run the program by executing

//   ./bin/crmc -h

to get the following help:

Options of CRMC:
  -h [ --help ]                  description of options
  -v [ --version ]               show version and exits
  -o [ --output ] arg            output mode: "hepmc""(default)",
                                              hepmc.gz, "root", lhe,
                                              lhe.gz
  -s [ --seed ] arg              random seed (default: random)
```

```
  -n [ --number ] arg           number of collisions
  -m [ --model ] arg            model [0=EPOS_LHC (default),
                                  1=EPOS_1.99, 2=QGSJET01,
                                  6=Sibyll_2.1, 7=QGSJETII-04,
                                  11=QGSJETII-03]
  -p [ --projectile-momentum ] arg  momentum/(GeV/c) (usually  >0)
  -P [ --target-momentum ] arg      momentum/(GeV/c) (usually =<0)
  -i [ --projectile-id ] arg    PDG or Z*10000+A*10 (default=proton)
  -I [ --target-id ] arg        PDG or Z*10000+A*10 (default=proton)
  -c [ --config ] arg       config file
  -f [ --out ] arg          output file name (auto if none provided)
  -t [ --produceTables ] [=arg(=1)] create tables if none are found
  -T [ --Test ] [=arg(=1)]          Test mode
  -x [ --cross-section ] [=arg(=1)]  calculate and
                                     print cross section only

for projectile and target Id the following shortcuts are allowed :
 1    = PDG(2212)        = proton
-1    = PDG(-2212)       = anti-proton
 12   = PDG(1000060120) = Carbon
 120  = PDG(211)         = pion+ (not for -I)
-120  = PDG(-211)        = pion- (not for -I)
 208  = PDG(1000822080) = Lead
using these shortcuts with automatic output file name generation will
create more human readable names.

#Example to generate 1000 13 TeV pC collisions with EPOS LHC:
// CM
bin/crmc -o root -p6500 -P-6500 -n1000 -m0 -i2212 -I12

          >> crmc <<

  seed:                 421568191 (automatic)
  projectile id:        2212 (p)
  projectile momentum:6500
  target id:            12 (C)
  target momentum:      -6500

  number of collisions: 1000
  parameter file name:  crmc.param
  output file name:   /home/andre/crmc_eposlhc_421568191_p_C_6500.root
  HE model:            0 (EPOS-LHC)

Opening: libEpos.so
initializations ...
###################################################################
#        EPOS LHC      K. WERNER, T. PIEROG                       #
```

```
#                       Contact: tanguy.pierog@kit.edu                    #
###################################################################
#         WARNING: This is a special retuned version !!!                  #
#      Do not publish results without contacting the authors.             #
###################################################################

The environment variable $CRMC_OUT can be set to define the path
the path of the output files, otherwise $PWD is used as default path.

//Example to generate 100 7 TeV pp collisions with EPOS LHC:
bin/crmc -o hepmc -p3500 -P-3500 -n1000 -m0


//Example to generate 1000 pPb collisions with EPOS LHC:
bin/crmc -o hepmc -p3500 -P-1380 -n1000 -m0 -i2212 -I822080

//Example to generate 1000 pC collisions with EPOS LHC:
bin/crmc -o hepmc -p3500 -P-1380 -n1000 -m0 -i2212 -I12

//Example to generate 100 1.38 ATeV PbPb collisions with EPOS 1.99:
bin/crmc -o hepmc -p1380 -P-1380 -n100 -i208 -I208 -m1

//Example to generate 100 4.4 ATeV pPb collisions with QGSJetII-04:
bin/crmc -o hepmc -p3500 -P-1380 -n100 -m7 -i2112 -I822080

//Example to test Sibyll2.1
bin/crmc -T -m6
```

3 PYTHIA 8: main71.cc

```
// main71.cc is a part of the PYTHIA event generator.
// Copyright (C) 2017 Richard Corke.
// PYTHIA is licenced under the GNU GPL version 2,
// see COPYING for details.
// Please respect the MCnet Guidelines, see GUIDELINES for details.

/*
 * Simple example of fastjet analysis. Roughly follows analysis of:
 * T. Aaltonen et al. [CDF Collaboration],*/
 # Measurement of the cross section for W-boson production in
 # association with jets in ppbar collisions at sqrt(s)=1.96$ TeV
 * Phys. Rev. D 77 (2008) 011108
 * arXiv:0711.4044 [hep-ex]
 *
 * Cuts:
```

```
 *   ET(elec)       > 20GeV
 *   |eta(elec)|    < 1.1
 *   ET(missing)    > 30GeV
 *   ET(jet)        > 20GeV
 *   |eta(jet)|     < 2.0
 *   deltaR(elec, jet) > 0.52
 * Not used:
 *   mT(W)          > 20GeV
 */


   // Interface for conversion from Pythia8::Event to HepMC event.
  HepMC::Pythia8ToHepMC ToHepMC;

  // Specify file where HepMC events will be stored.
  HepMC::IO_GenEvent ascii_io("pythia.hepmc", std::ios::out);
```

4 RIVET [5]

```
/*
The Rivet toolkit (Robust Independent Validation of Experiment and
Theory) is a system for validation of Monte Carlo event generators.
 It provides a large (and ever growing) set of experimental
analyses useful for MC generator development, validation, and tuning,
 as well as a convenient infrastructure for adding your own analyses.
*/

https://rivet.hepforge.org/

~/Desktop/RIVET_PYTHIA$ ls
pythia.hepmc
~/Desktop/RIVET_PYTHIA$ rivet pythia.hepmc -a CDF_2008_S7541902

RIVET:
Input: "pythia.hepmc"

https://rivet.hepforge.org/analyses:
"CDF_2008_S7541902"
# Measurement of the cross section for W-boson production
# in association with jets in ppbar collisions at sqrt(s)=1.96$ TeV)

// Finished event loop at 2017-08-04 12:41:03
// Cross-section = 1.985145e+03 pb

Rivet.Analysis.Handler: INFO  Finalising analyses
Rivet.Analysis.Handler: INFO  Processed 29323 events
```

```
The MCnet usage guidelines apply to Rivet:
see http://www.montecarlonet.org/GUIDELINES
Please acknowledge plots made with Rivet analyses, and cite
arXiv:1003.0694 (http://arxiv.org/abs/1003.0694)

Rivet run completed at 2017-08-04 12:41:03, time elapsed = 0:03:31

Histograms written to /home/andre/Desktop/RIVET_PYTHIA/Rivet.yoda

~/Desktop/RIVET_PYTHIA$ ls
pythia.hepmc  "Rivet.yoda"

~/Desktop/RIVET_PYTHIA$ rivet-mkhtml --mc-errs Rivet.yoda:"PYTHIA8"
Making 12 plots
Plotting ./rivet-plots/CDF_2008_S7541902/d01-x01-y01.dat
Plotting ./rivet-plots/CDF_2008_S7541902/d02-x01-y01.dat
Plotting ./rivet-plots/CDF_2008_S7541902/d03-x01-y01.dat
Plotting ./rivet-plots/CDF_2008_S7541902/d04-x01-y01.dat
Plotting ./rivet-plots/CDF_2008_S7541902/d05-x01-y01.dat
Plotting ./rivet-plots/CDF_2008_S7541902/d05-x01-y02.dat
Plotting ./rivet-plots/CDF_2008_S7541902/d05-x01-y03.dat
Plotting ./rivet-plots/CDF_2008_S7541902/d05-x01-y04.dat
Plotting ./rivet-plots/CDF_2008_S7541902/d06-x01-y01.dat
Plotting ./rivet-plots/CDF_2008_S7541902/d07-x01-y01.dat
Plotting ./rivet-plots/CDF_2008_S7541902/d08-x01-y01.dat
Plotting ./rivet-plots/CDF_2008_S7541902/d09-x01-y01.dat

./rivet-plots/CDF_2008_S7541902$ make-plots --eps *.dat
Making 12 plots
//Plotting d01-x01-y01.dat (12/12 remaining)
Plotting d02-x01-y01.dat (11/12 remaining)
Plotting d03-x01-y01.dat (10/12 remaining)
Plotting d04-x01-y01.dat (9/12 remaining)
Plotting d05-x01-y01.dat (8/12 remaining)
Plotting d05-x01-y02.dat (7/12 remaining)
Plotting d05-x01-y03.dat (6/12 remaining)
Plotting d05-x01-y04.dat (5/12 remaining)
Plotting d06-x01-y01.dat (4/12 remaining)
Plotting d07-x01-y01.dat (3/12 remaining)
Plotting d08-x01-y01.dat (2/12 remaining)
Plotting d09-x01-y01.dat (1/12 remaining)
```

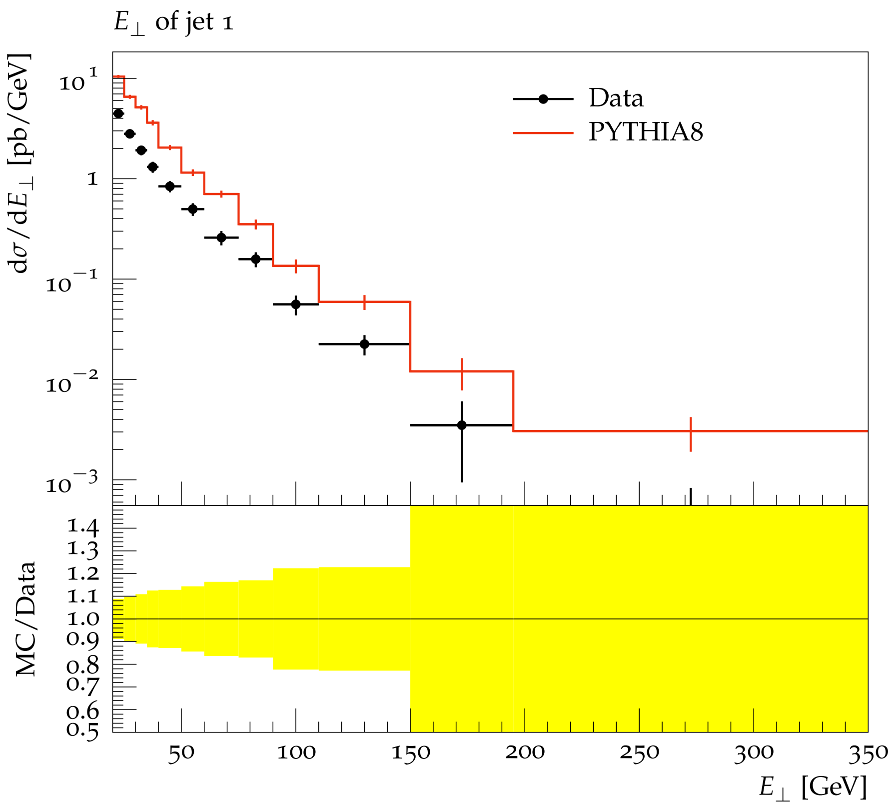

Figura 1: Plotting d01-x01-y01.dat

Referências

[1] S. Agostinelli et al. "GEANT4: A Simulation toolkit". Em: *Nucl. Instrum. Meth.* A506 (2003), pp. 250–303. DOI: 10.1016/S0168-9002(03)01368-8.

[2] M. Boonekamp et al. "FPMC: A Generator for forward physics". Em: (2011). arXiv: 1102.2531 [hep-ph].

[3] Torbjörn Sjöstrand et al. "An Introduction to PYTHIA 8.2". Em: *Comput. Phys. Commun.* 191 (2015), pp. 159–177. DOI: 10.1016/j.cpc.2015.01.024. arXiv: 1410.3012 [hep-ph].

[4] T. Pierog et al. "EPOS LHC: Test of collective hadronization with data measured at the CERN Large Hadron Collider". Em: *Phys. Rev.* C92.3 (2015), p. 034906. DOI: 10.1103/PhysRevC.92.034906. arXiv: 1306.0121 [hep-ph].

[5] Andy Buckley et al. "Rivet user manual". Em: *Comput. Phys. Commun.* 184 (2013), pp. 2803–2819. DOI: 10.1016/j.cpc.2013.05.021. arXiv: 1003.0694 [hep-ph].