PYTHIA-4-BSM — Hands-On Session

P. Skands

April 2010

Preparations

The tutorial files are available from:

`http://home.fnal.gov/~skands/stuff/pythia-mc4bsm5.tgz`

Compile the PYTHIA *library:*

```
g77 -c -O0 pythia-6.4.22.f
```

Alternatively, you can use gfortran:

```
gfortran -c -O0 pythia-6.4.22.f
```

For the exercise, we will be using the LHEF file you generated in the previous tutorial with MadGraph or CalcHEP. If you do not have this file or just want to try something different, you may also look at one of the example LHEF files from the previous year's MC4BSM tutorials, which are also included in the tarball or, alternatively, can be downloaded here:

`http://hep.pa.msu.edu/people/christensen/mc4bsm/mc4bsm-CH.lhe.tar.gz`
`http://www.physics.ucdavis.edu/~conway/research/MC4BSM/mc4bsm-MG-decayed.tgz`
`http://www.physics.ucdavis.edu/~conway/research/MC4BSM/mc4bsm-CH-decayed.tgz`

For documentation, see the PYTHIA manual and update notes available at:

`http://projects.hepforge.org/pythia6/`

General Tip: when you need to look something up in the PYTHIA manual, use the index in the back or the search function of your pdf reader.

1 Processing an LHEF file through PYTHIA

Open the file `pythia-mc4bsm.f` *in emacs or some other editor of your choice.*

1. *Locate the part where it says:*

   ```
   C...1) Open LHEF file on unit LUN, and tell Pythia where to find it.
   ```

   You will see that a file is opened and some switches in PYTHIA *are set to point to the unit number where that file is opened. This is how you tell* PYTHIA *which file to read. Edit the* `OPEN(...)` *command so that it will open the file you have chosen to work with. You do not need to edit the rest of the program.*

2. *Compile* `pythia-mc4bsm.f` *together with* `pythia-6.4.22.o`*:*

```
g77 -o pythia-mc4bsm.x pythia-mc4bsm.f pythia-6.4.22.o
```

3. *Run the program* `pythia-mc4bsm.x`*. Wait a little while it processes the file and generates events Upon finishing, it produces two histogram files* `pythia-mc4bsm-n.dat` *and* `pythia-mc4bsm-pt.dat`*, which contain the hadron-level charged particle multiplicities (number of generator-level tracks, limited to the range 200 to 300 tracks) and* p_T *spectra, respectively, with no cuts applied. Start by just looking at these distributions. Start gnuplot and plot the distribution of the number of tracks by:*

```
plot "pythia-mc4bsm-n.dat" using 1:2 with histeps
```

Q: Why are only the bins with an even number of tracks populated? Now plot the p_T *spectrum of charged particles by:*

```
plot "pythia-mc4bsm-pt.dat" using 1:2 with histeps
```

Q: Where does the p_T *spectrum peak? (x axis is in GeV)*

4. *Run the program* `pythia-mc4bsm.x` *again and study the output while* PYTHIA *initializes. What does it say? Which warnings are given? Try one of the other LHE example files from the web page. Do you get the same initialization output? What are the differences?*

5. *Next study the event records which are printed out while* PYTHIA *is running (tip: a parenthesis* `(rho+)` *around a particle means it has decayed), in particular the correspondence between particle names and PDG codes (*`K(I,2)`*). Identify the PDG codes for: pions, kaons, protons, photons, gluons, and the quarks. (tip: the PDG codes and the event record are described in the* PYTHIA *manual, Chapter 5.)*

6. *Q: how do the multiplicities change if you switch off the underlying event? Open* `pythia-mc4bsm.f` *in an editor and add a bit of code towards the top of the program:*

```
CALL PYGIVE('MSTP(81)=0')
```

Recompile and run `pythia-mc4bsm.x`*. Check how the number printed out at the end of the run changes. How much of the total particle multiplicity in the event is due to the underlying event?*

7. *Q: how do the multiplicities change if you use another tune of the underlying event? Open* `pythia-mc4bsm.f` *in an editor and replace the line you just added by:*

```
CALL PYTUNE(320)
```

Recompile and run `pythia-mc4bsm.x`*. Check first how the stuff printed out during initialization changes (note the output on the tune parameters). When the run finishes, check if/how the average number of tracks printed out at the end of the run changes. Note that the default settings correspond to "Tune A". The settings you are now using, tune 320, are for a brand new tune called "Perugia 0", which uses a somewhat different underlying-event and parton-shower model. How large are the differences in track multiplicity?*

8. *Q: in order to find the BSM resonance itself, you would need to include an event-by-event analysis that plots, e.g., a lepton-lepton invariant mass or similar. If there is time to spare, see if you can insert such an analysis. Note that this is not really crucial to running PYTHIA as such, but will give you some additional experience with the event record and defining an analysis. You may want to look up* `PYJETS` *and* `PYONOF` *(this very useful routine to switch on and off specific decay channels of resonances is more recent than the manual and hence is not documented in the manual - you will have to consult the update notes) in the documentation. If you need a very crude first-idea detector simulation and jet finder, look up* `PYCELL` *in the manual.*