Part I: understanding `main31`

The reason why the number of entries in the histograms varied

I still don't know. Based on what I learned from printing output and eyeballing it, I *think* it's a technical detail and nothing to worry about. I'll keep trying to figure it out.

By the way, I checked carefully that my hard scale histograms really did contain what their labels said.

Understanding the hard scale

In Powheg's $t\bar{t}$ events, `SCALUP` is (from what I've checked just by looking at print output) simply the kinematic p_T of the emitted gluon.

This explains why `pThardMode = 1`, `pTdefMode = 0` (*use minimum relative* p_T *of Powheg emission and other partons, where* p_T *is the kinematic* p_T) gives the same hard scale as `pThardMode = 0` (*use SCALUP*).

Why is the hard scale sometimes greater than `SCALUP`?

For `pThardMode = 1` and `pTdefMode = 1` (Powheg "p_T"), I could not reproduce the effect. Maybe it was an artefact of some technical implementation?

For `pThardMode = 1` and `pTdefMode = 2` (Pythia/Lund "p_T"), I could. In fact, it turns out that in many cases, `pThard` is just a little bit larger than `SCALUP`, typically $\mathcal{O}(1\text{ GeV})$. Sometimes more. Conjecture: this is no bug but an effect of the different "p_T" definition. Apparently the Lund "p_T" is sometimes a little greater than the kinematic p_T.

I've since confirmed this conjecture, see slide 7!

The weird behaviour related to the `vetoCount` parameter

I still don't know what's causing it! If I run over a handful of events and check some print output by eyeballing, everything looks sensible. But with larger samples, I can also confirm the strange behaviour I showed you last week. (So it's not a plotting mistake or something.)

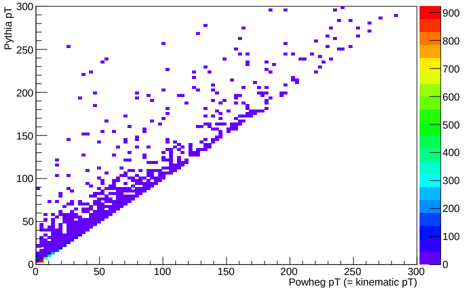
It is most likely due to an implementation detail, if the veto counters are not incremented carefully.

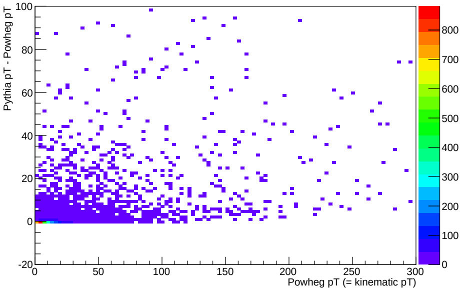
This is proving somewhat tedious to study. If I can't figure it out, I could send Stefan Prestel an email.

On the plus side, I can confirm that the option `vetoCount = 0` does *not* correspond to checking all emissions, even though this is claimed in the Pythia configuration file, `main31.cmnd`. This is already something to notify the authors about.

Scale of initial state radiation emissions: different “p_T” definitions

Here I’ve compared the “p_T” of an emission calculated using the two different definitions.

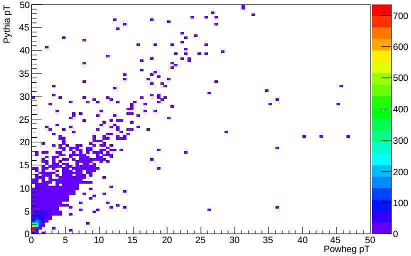


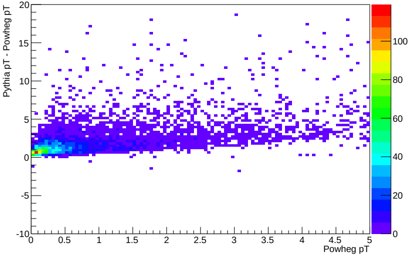


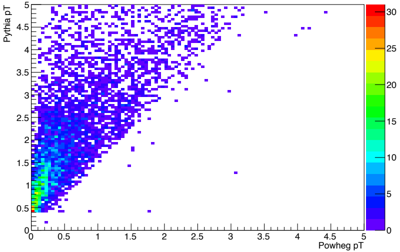
In ISR, the Powheg “p_T” of an emission is equal to the kinematic p_T of the emitted parton.

The Pythia “p_T” is greater than or equal to this, so using the Pythia “p_T” definition should lead to more vetoing (for a given hard scale to veto against).

Scale of final state radiation emissions: different “p_T” definitions







Zoomed in on the soft/collinear corner.

Options I don't want to study unless I have to

…because I consider them pretty irrelevant and think I shouldn't waste to much time on them now.

Here's what they do and why I consider them unimportant to my study:

`pTemt` *Defines between which particles the emission scale is calculated.* Only default option seems physically sensible. Non-default options will decrease emission scale and hence number of vetos.

`emitted` *Specifies how the emitted parton is selected in FSR.* Only affects FSR. Default option is Pythia default behaviour.

…and that covers all the relevant options

So `main31` is now “understood” at some level. Quotes because it’s too complex for it to be meaningful to talk about completely understanding it at this point in my study.

Part II: gap fractions in $t\bar{t}$ events

Done so far

I’ve generated events with various choices of options and analysed them with Rivet.

The resulting plots are quite interesting, but not well organised (labels cluttered, choice of samples unnecessarily confusing, etc.), so I’m omitting them here.

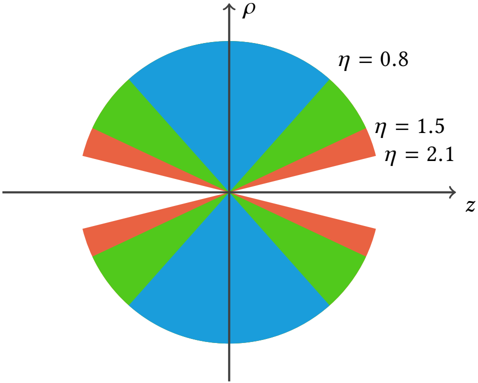
Possible caveat in my Rivet analysis using ATLAS data (i.e. `ATLAS_2012_I1094568`)

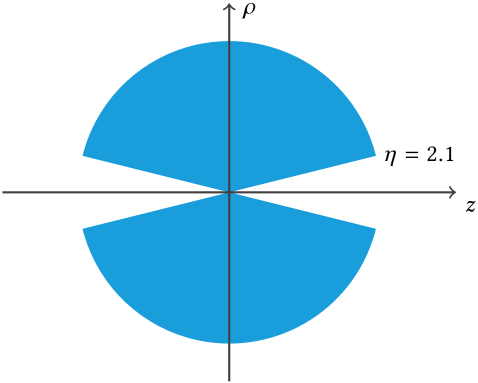
I often only generate $t\bar{t}$ events in the *dileptonic* decay channel to compare to data! This has the advantage of giving my better statistics for a given computing time, but in the end, I should use a sample that is inclusive with respect to decays, to take possible misidentification effects into account.

I'm now generating 100k $t\bar{t}$ events with all decay channels open. Dileptonic decay ratio only $\sim (2/9)^2 \approx 5\%$!

Data/MC agreement as a function of veto region — the observed general behaviour

Note: the analysis actually considers four *rapidity* regions. Visualised here as four *pseudo*rapidity regions.





- good agreement
- discrepancy systematic, but within data uncertainty
- discrepancy

Data/MC agreement as a function of veto region

I think the above pattern might give us important clues for what is going on!

I'll spend some time thinking about it while the computer's generating samples for a more systematic study of the effect of different stages of the generation (hard process, parton shower, hadronisation, adding MPIs, top decay in Powheg or Pythia 8).

Present above & more (Rivet plots!) to ATLAS MC enthusiasts on March 27?

I've proposed this to James Monk and Thorsten Kuhl; haven't heard back yet.

P.S.: Software versions used

Pythia 8.183

Powheg-hvq (Version? Most recent as of 27/02/2014)

LHAPDF 5.9.1

FastJet 3.0.3

Rivet 2.1

As well as GSL 1.16, BOOST 1.55.0, HepMC 2.06.06, and YODA 1.0.5.