

**Heavy-flavour
hadronisation and
balancing studies in
pp collisions using
PYTHIA**



ALICE

Paul Veen



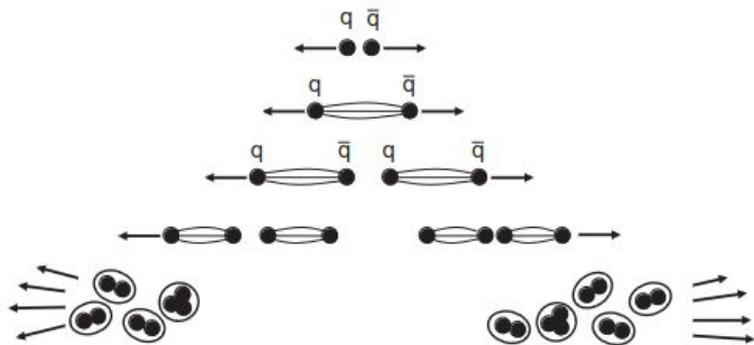
**universit
PARIS-SACL**

Hadronisation in PYTHIA

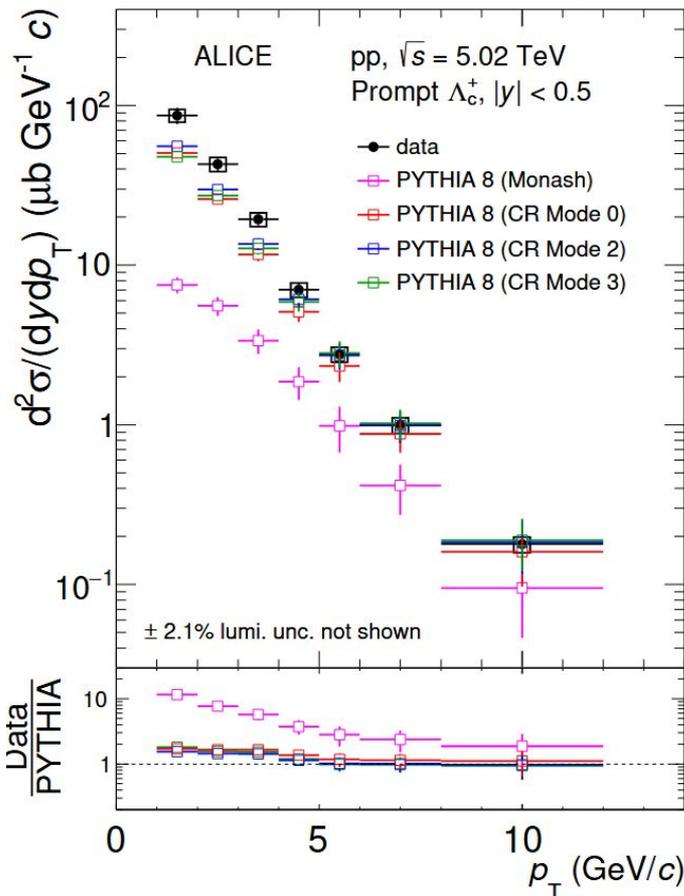
Based on Lund string model

Standard: PYTHIA Monash: underestimate baryon production

In order to understand hadronisation in PYTHIA:
how is colour flow treated from partonic to final state?



pau.veen@cern.ch



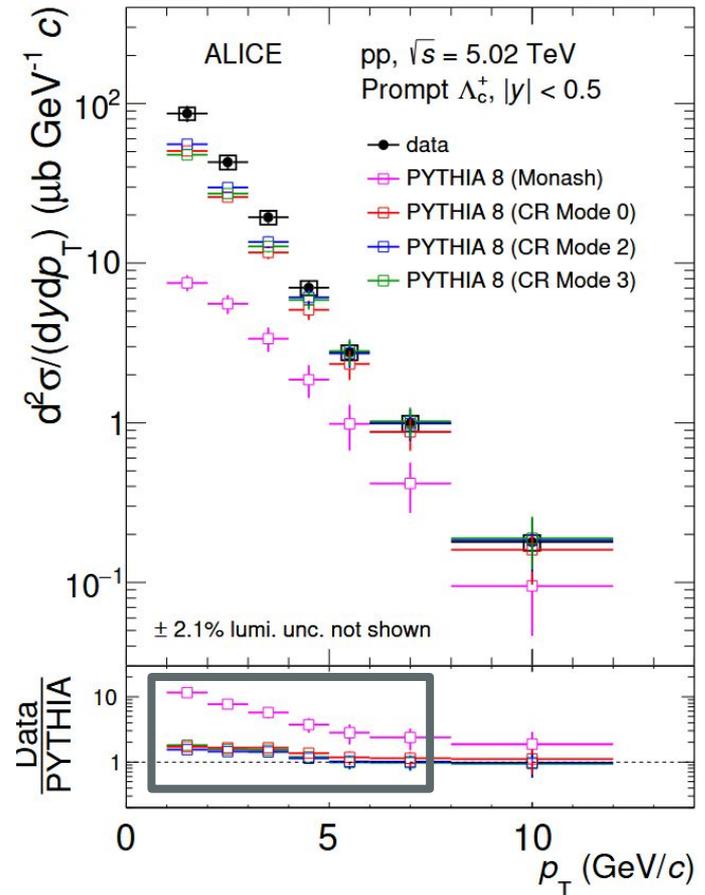
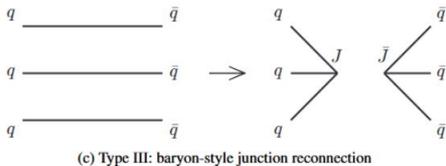
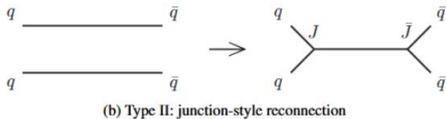
Left: Thomson

Right: Phys. Rev. C 104, 054905

Hadronisation in PYTHIA

PYTHIA Junctions (CR mode 2) also allows junction topologies -> increases baryon production

- Depends on string-length minimisation parameter: generally disfavouring junctions unless large phase space overlap (with MPIs)



Phys. Rev. C 104, 054905

Physics goal

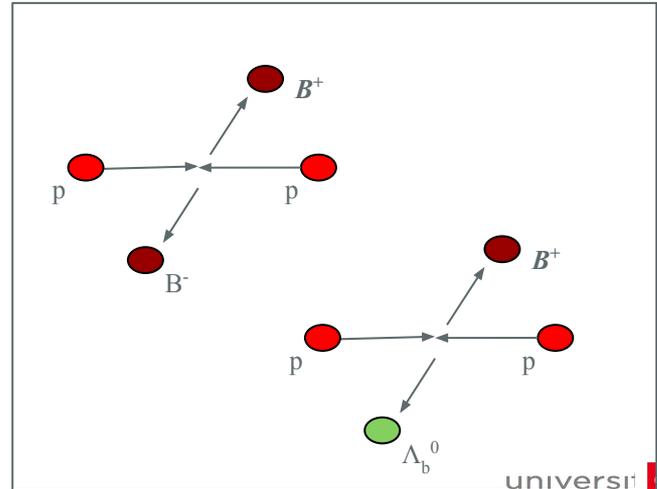
hard-scattering origin

hadronisation mechanisms

Study beauty and charm hadronisation with different PYTHIA tunes (Monash, Junctions) as a function of multiplicity for **future colliders**

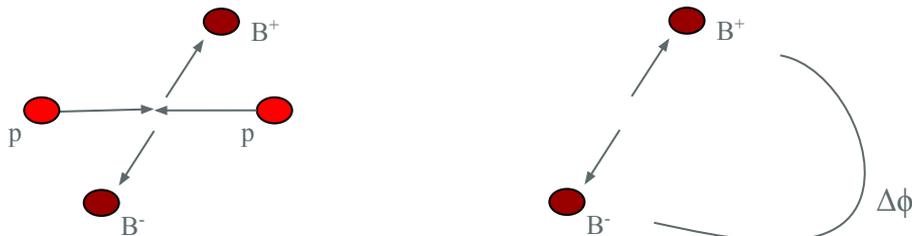
multiparton interactions

Observable: **balancing**

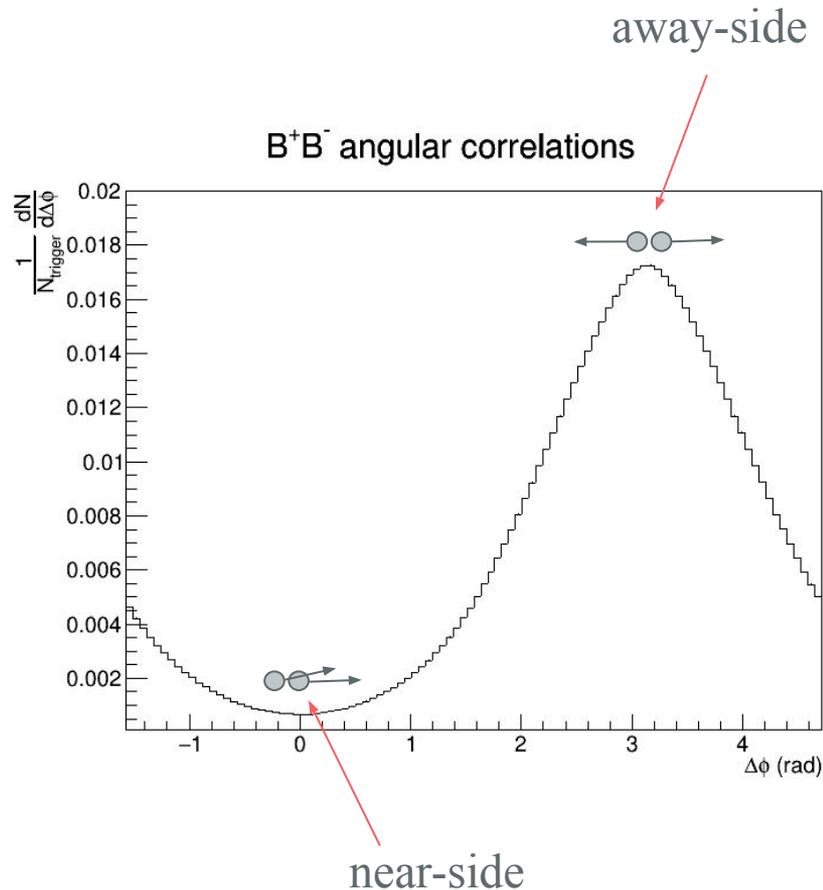


Analysis: angular correlations

Quarks are confined: calculate instead azimuthal angle correlations of hadronisation products (B^+ , B^0 , B^0 -bar, Λ_b^0 , ...)



Normalisation: divide over total number of triggers (B^+ or D^+ mesons) in specified regime



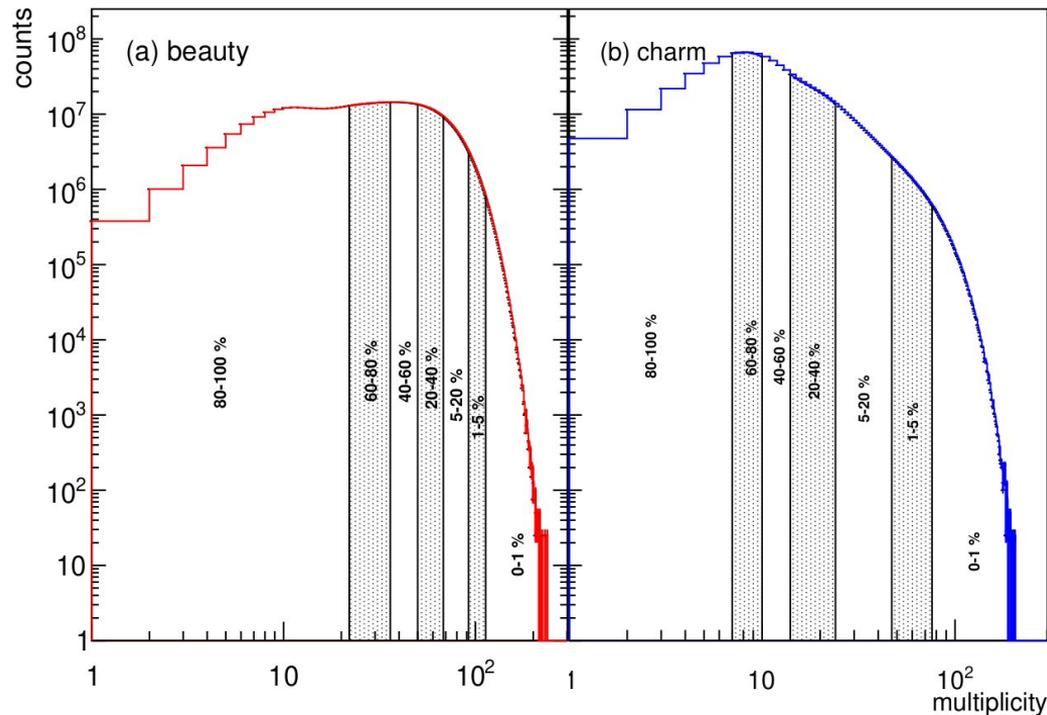
Analysis: multiplicity

Multiplicity defined as number of charged primaries in the event

This notation: % most of events in most multiplicative bin

- top 1% = 1% most multiplicative events
- 80-100% = 20% least multiplicative events

Charm and beauty binning is not 1-to-1 in this definition



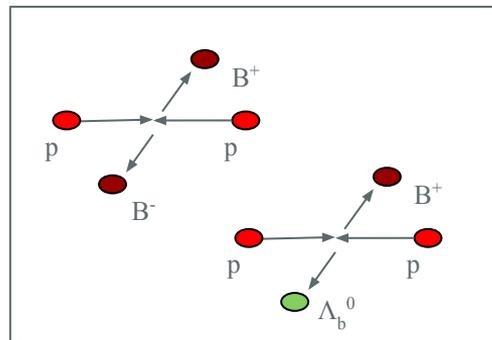
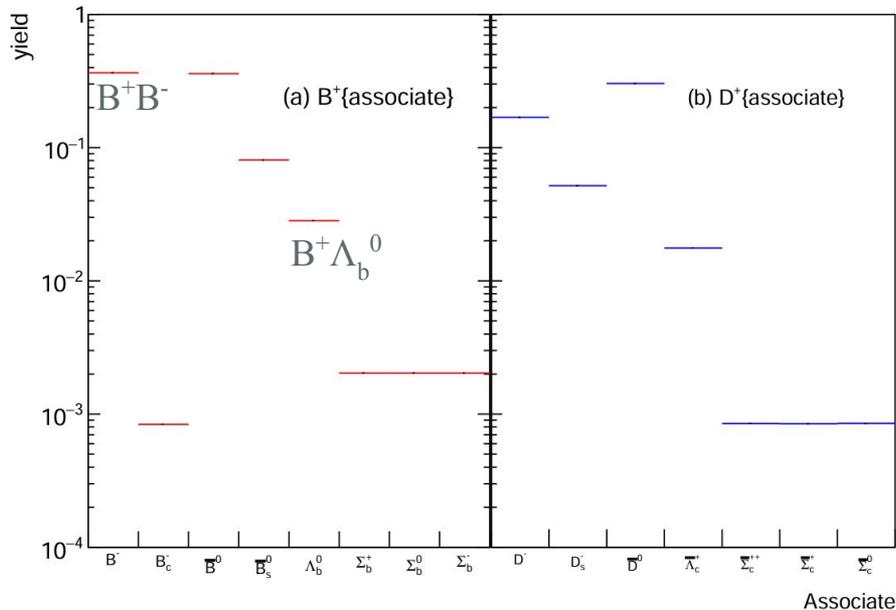
Results

1. Junctions?
2. Multiplicity?
3. Beauty-charm?

By looking at

- **balancing yields and their**
- **baryon/meson ratios**

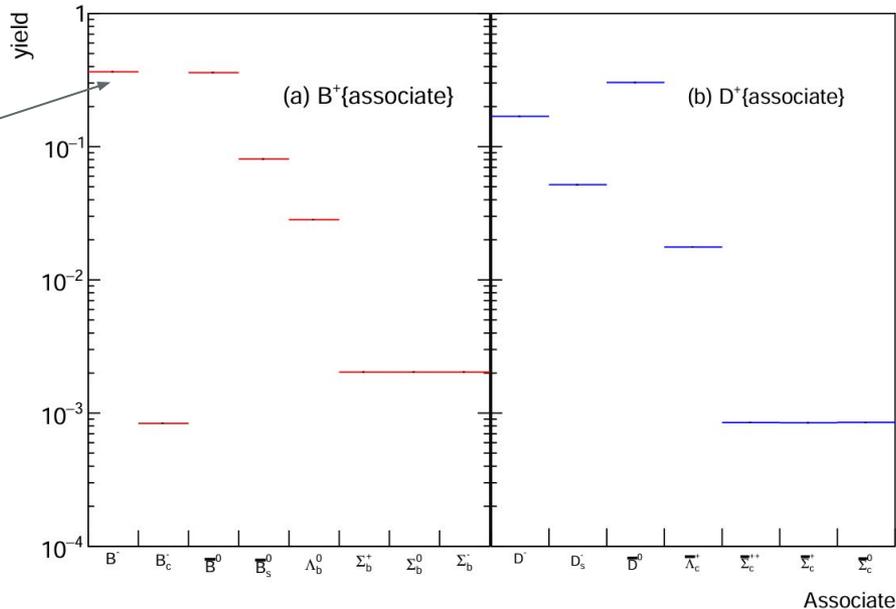
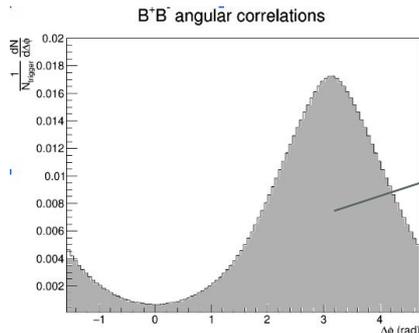
Errors obtained with sub-sampling



(upper)
balancing
example in
Monash

Results

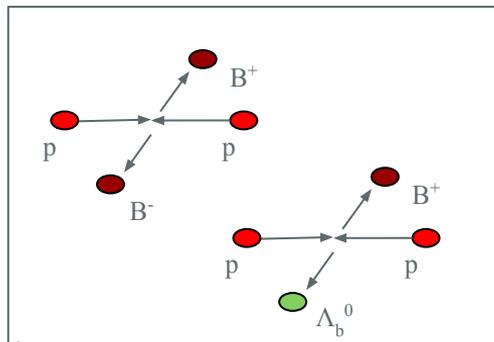
1. Junctions?
2. Multiplicity?
3. Beauty-charm?



By looking at

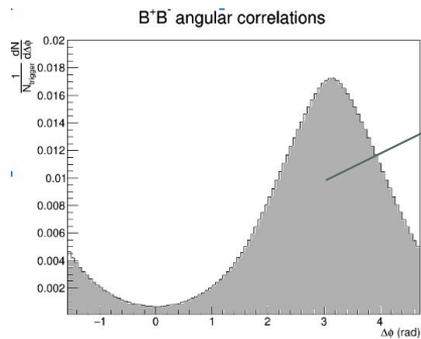
- **balancing yields and their**
- **baryon/meson ratios**

Errors obtained with sub-sampling



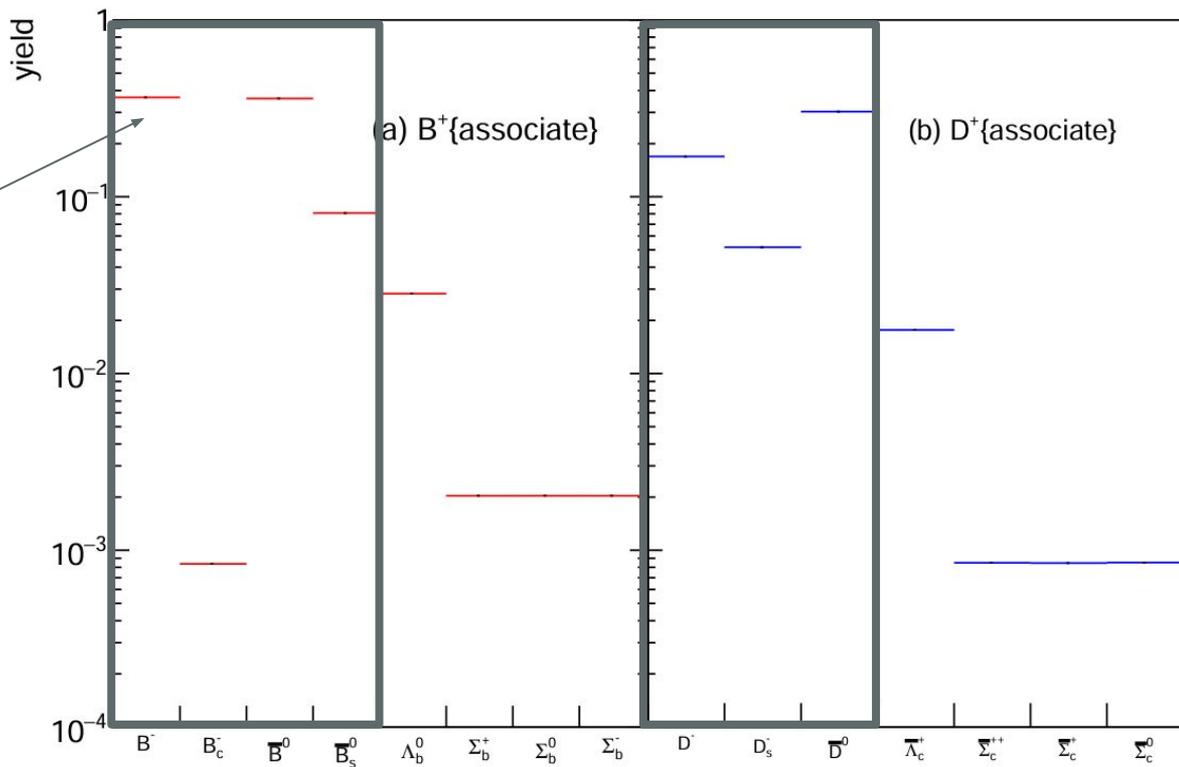
(upper)
balancing
example in
Monash

Results



By looking at

- balancing yields and their
- baryon/meson ratios



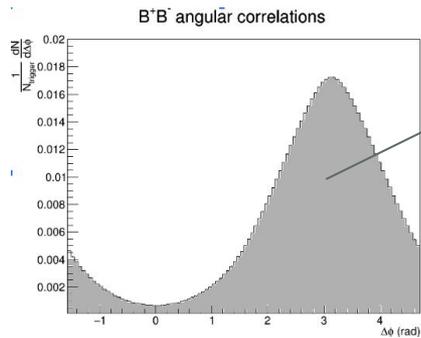
Errors obtained with sub-sampling

MESONS

MESONS

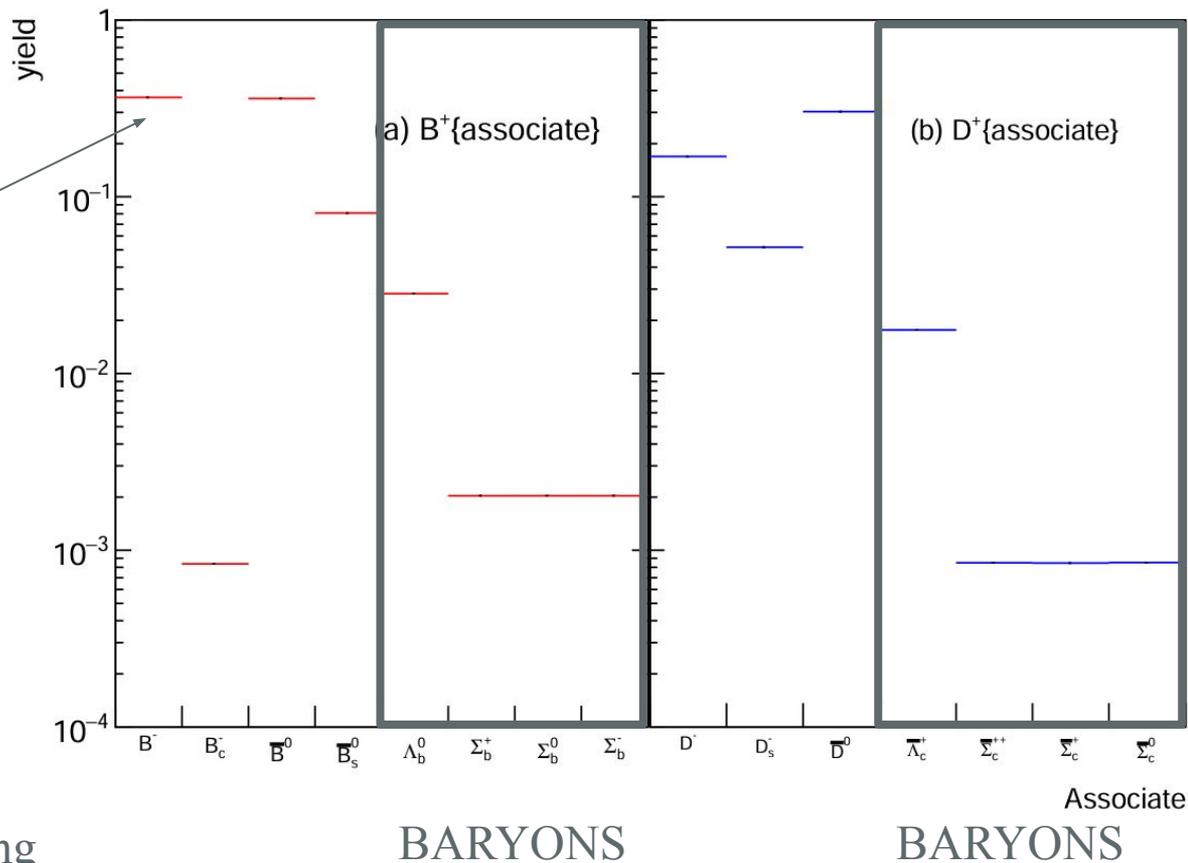
Associate

Results



By looking at

- balancing yields and their
- baryon/meson ratios



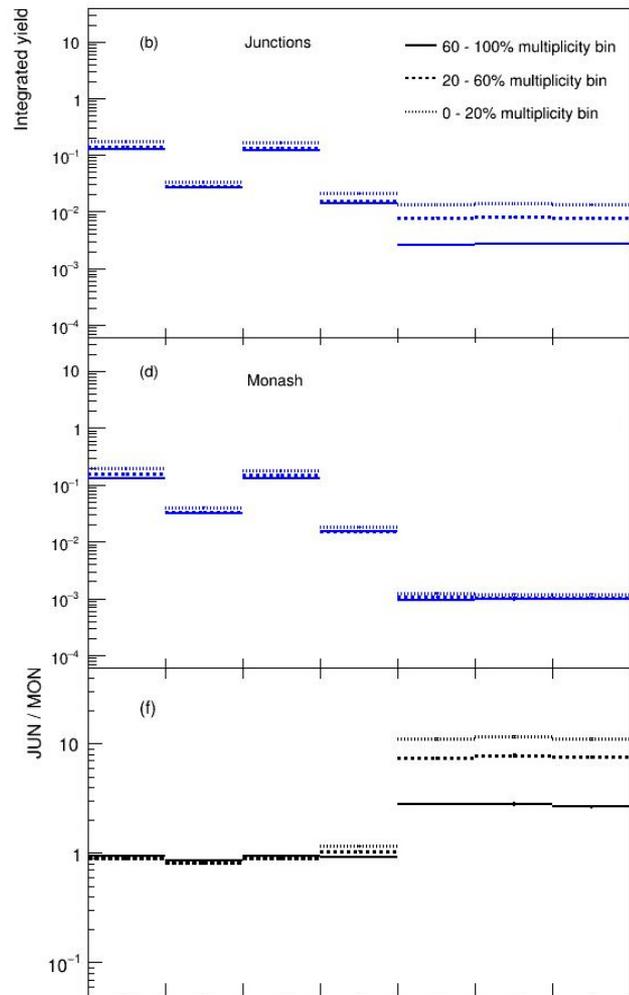
Errors obtained with sub-sampling



Balancing: prompt charm

The expected increase in baryon balancing yield is observed

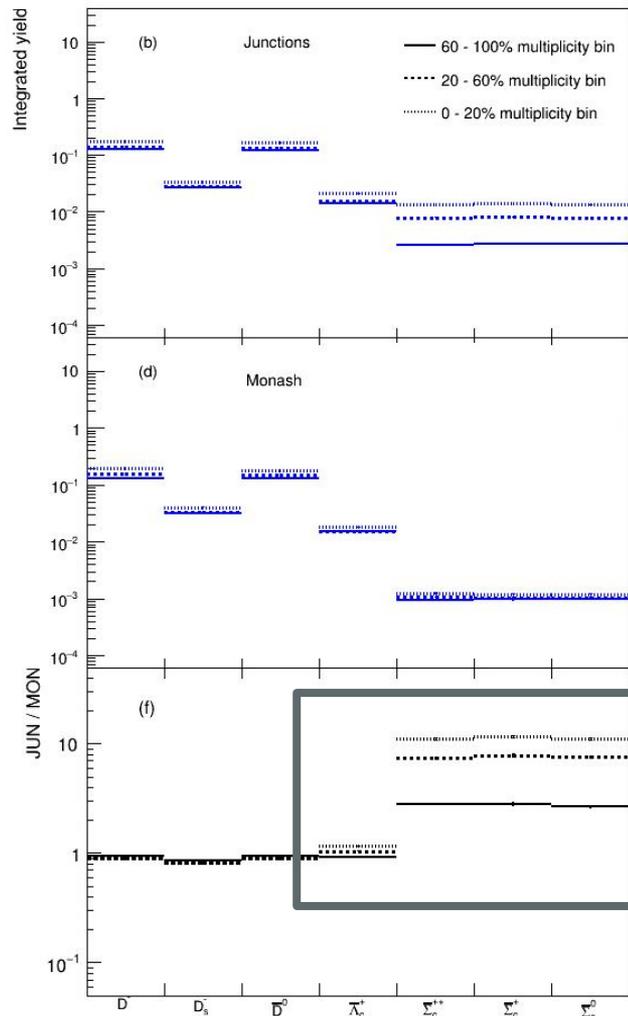
- Even stronger for Σ_c -bar baryons due to isospin suppression in PYTHIA which is 'lifted' by CR (both tunes, but more extreme in Junctions)
- Clear dependence on multiplicity for Σ_c -bar baryons



Balancing: prompt charm

The expected increase in baryon balancing yield is observed

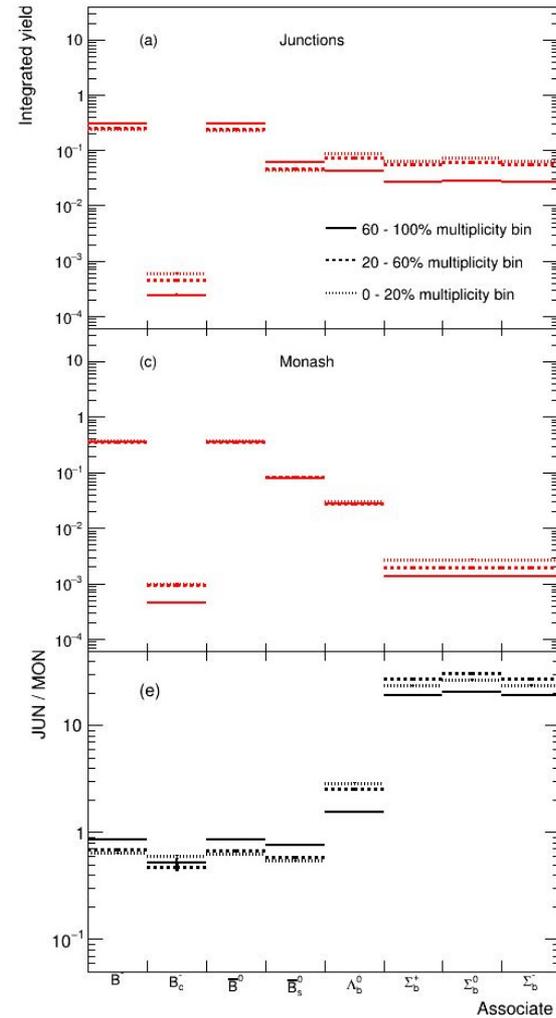
- Even stronger for Σ_c -bar baryons due to isospin suppression in PYTHIA which is 'lifted' by CR (both tunes, but more extreme in Junctions)
- Clear dependence on multiplicity for Σ_c -bar baryons



Balancing: beauty

Behaviour is similar as in the charm case, though with minor differences:

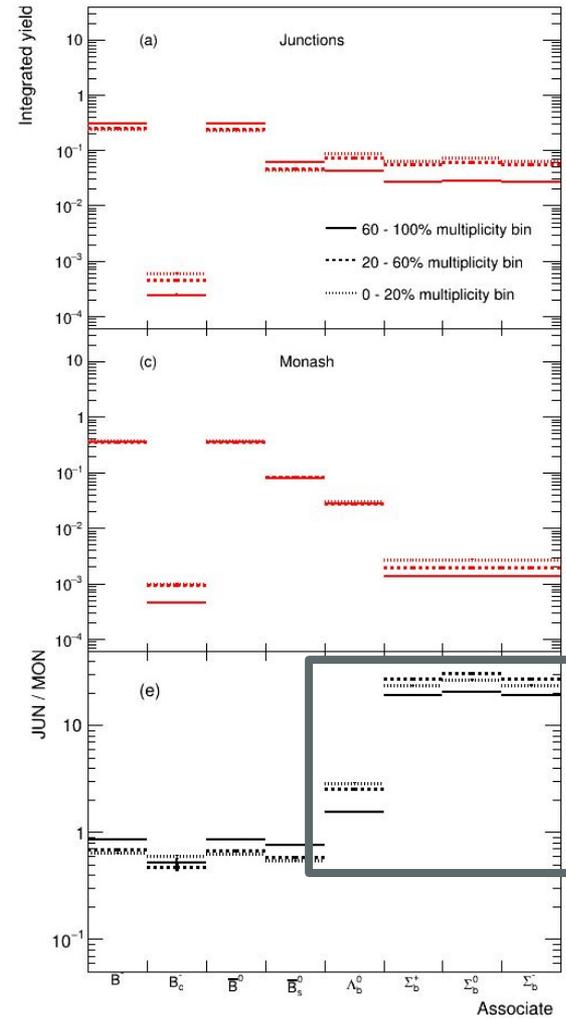
- The increase in Λ_b^0 balancing yield is larger than charm equivalent
- The increase in Σ_b baryons is not as strongly dependent on multiplicity, but still larger than Λ_b^0



Balancing: beauty

Behaviour is similar as in the charm case, though with minor differences:

- The increase in Λ_b^0 balancing yield is larger than charm equivalent
- The increase in Σ_b baryons is not as strongly dependent on multiplicity, but still larger than Λ_b^0

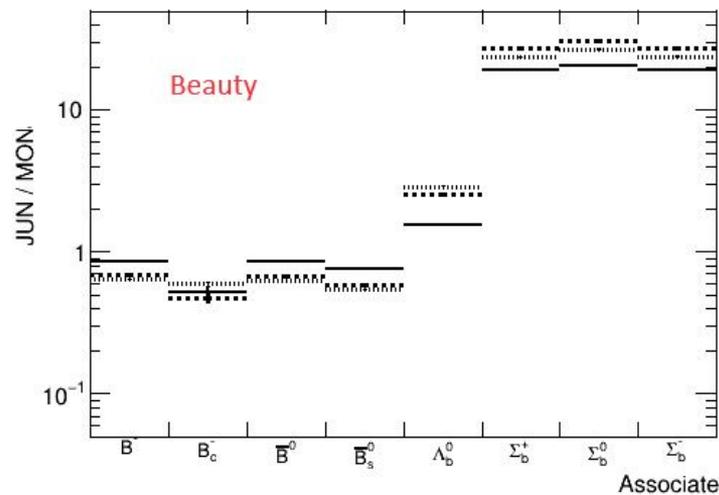
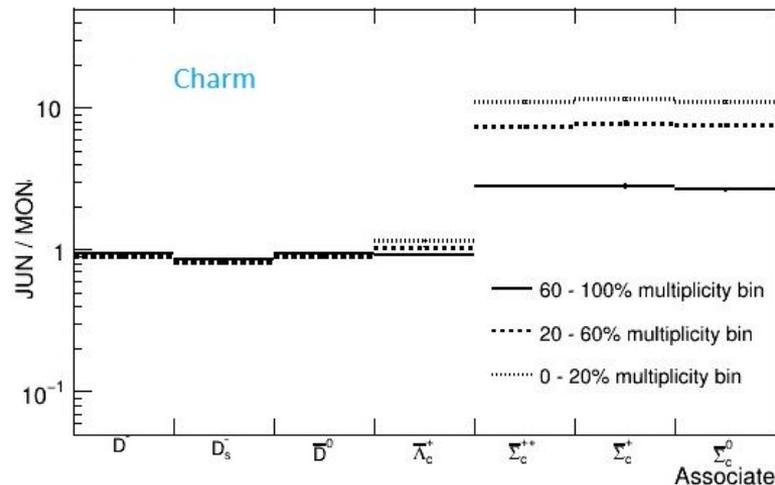


Balancing: differences

In summary:

- Stronger increase in baryon partners in beauty
- Multiplicity dependence for Σ_c -bar more pronounced compared to beauty

Related to different treatment of beauty and charm in PYTHIA + tuning of charm (experimental availability)

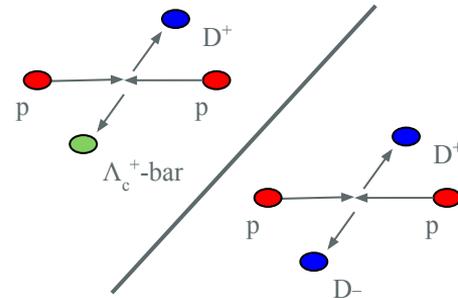
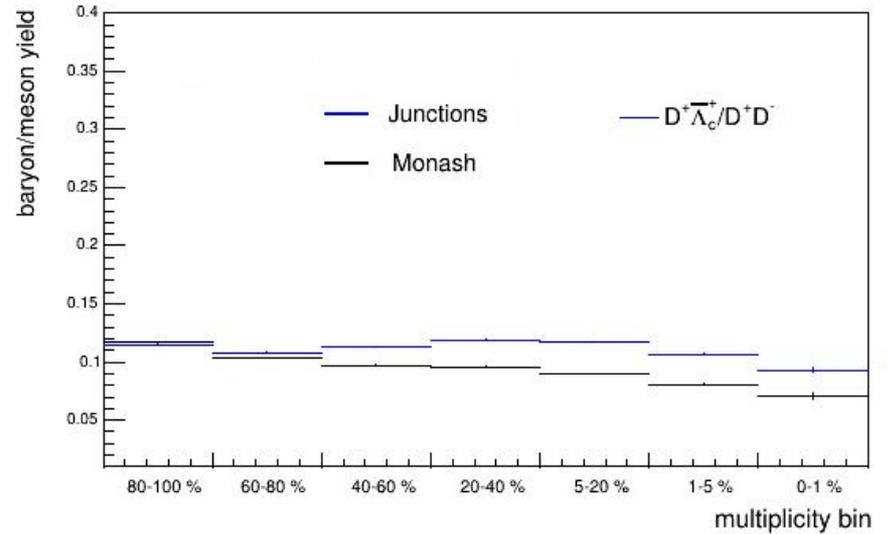


Ratios: prompt charm

Interpreted as *how well Junctions performs at balancing with baryon partners*

Expectation: this ratio increases as a function of multiplicity

- Wiggling behaviour seen in Junctions

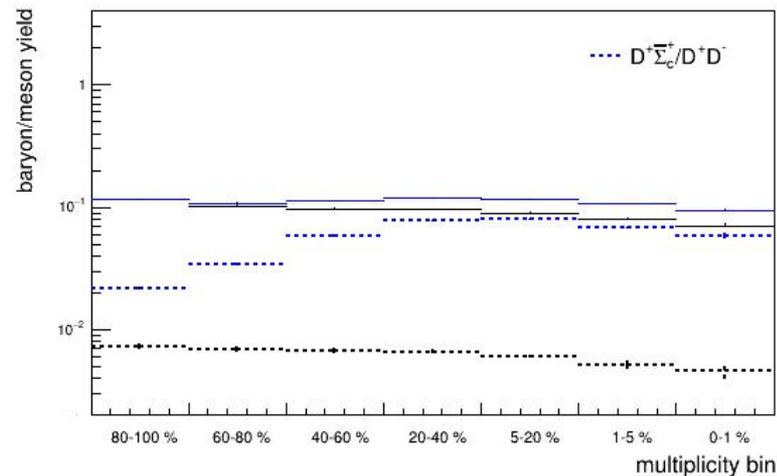
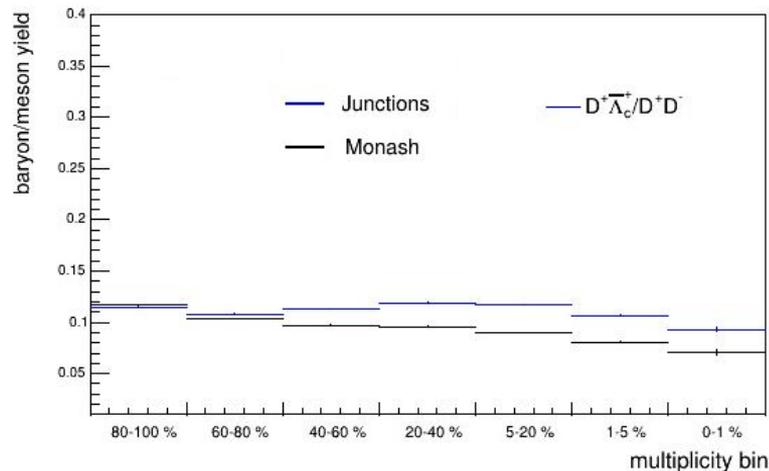


Ratios: prompt charm

Interpreted as *how well Junctions performs at balancing with baryon partners*

Expectation: this ratio increases as a function of multiplicity

- Wiggling behaviour seen in Junctions
- Λ_c^+ -bar and Σ_c^+ -bar show slightly similar trends

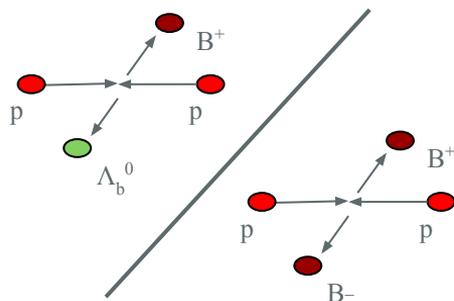
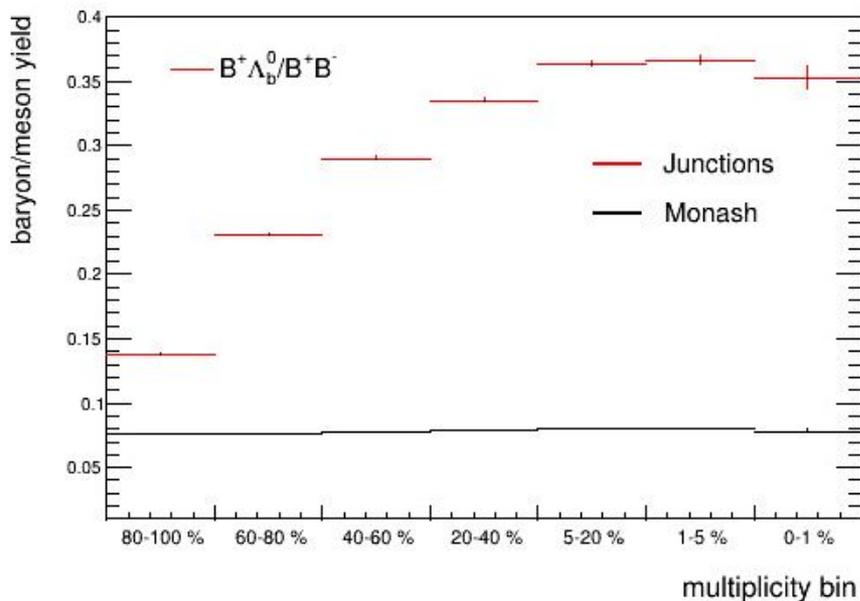


Ratios: beauty

Beauty sector shows an increase as function of multiplicity

At large multiplicity this saturates

- Onset of decreasing trend seems to be shifted towards higher multiplicities due to the larger (beauty) mass

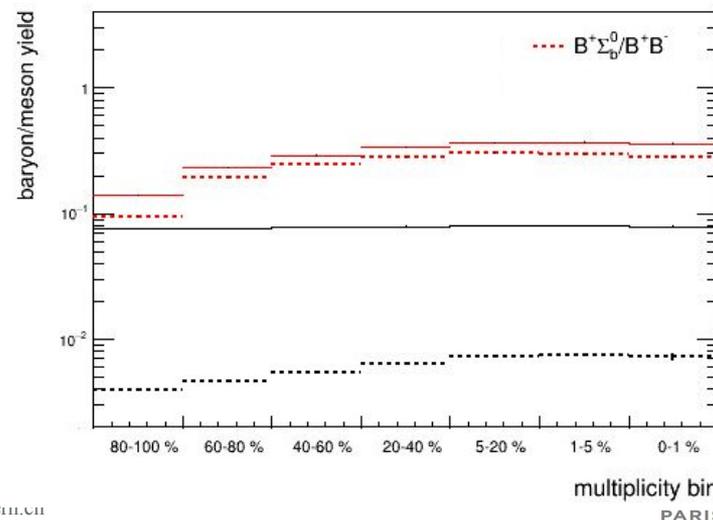
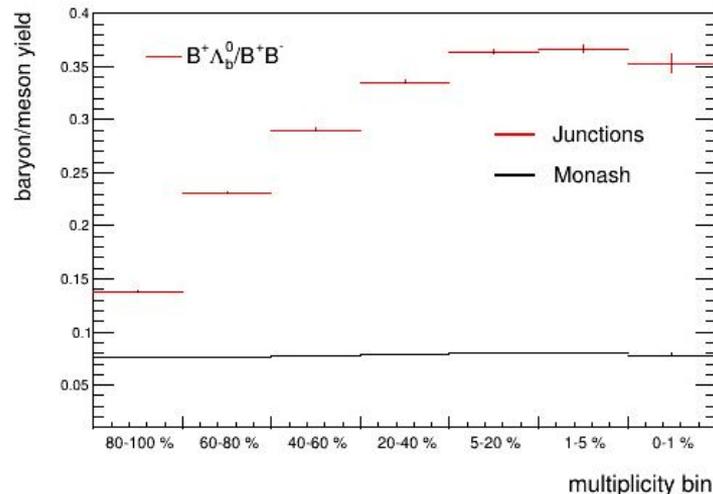


Ratios: beauty

Beauty sector shows an increase as function of multiplicity

At large multiplicity this saturates

- Onset of decreasing trend seems to be shifted towards higher multiplicities due to the larger (beauty) mass
- The Λ_b^0 and Σ_b^0 baryons show identical behaviour for Junctions, differ slightly for Monash

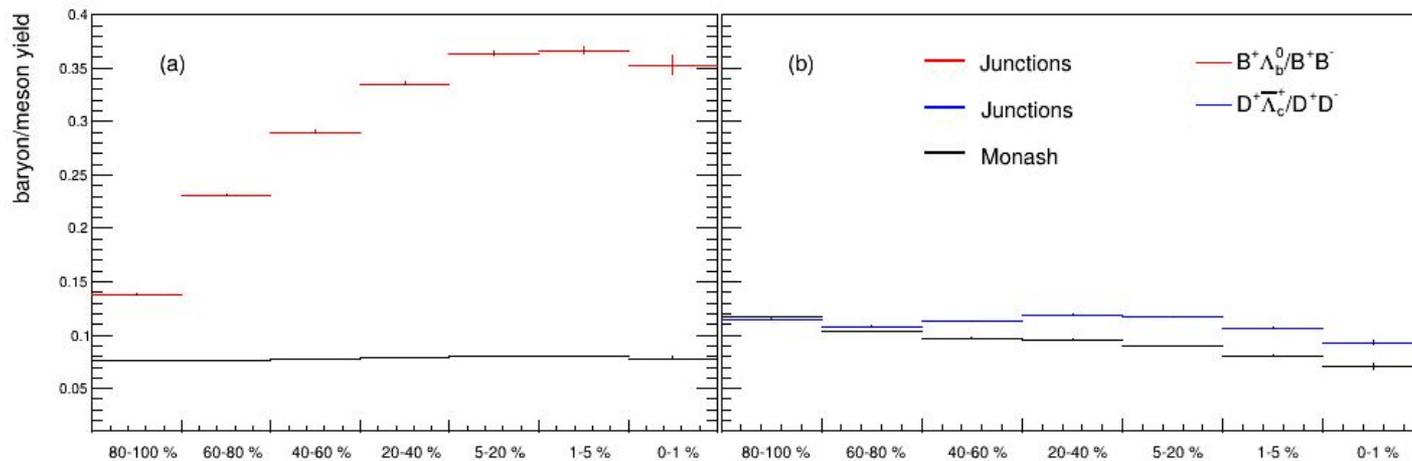


Ratios: differences

Focussing on Junctions: different behaviour for beauty (left) and charm (right)

- Increase and saturation effect seen in beauty
- ‘Wiggling’ in charm

Explicit example: StringFlav:ProbQQ1toQQ0join



Event classification study

Goals:

- investigate multiplicity differences in beauty and charm
- characterise event 'business' without using multiplicity

Based on papers:

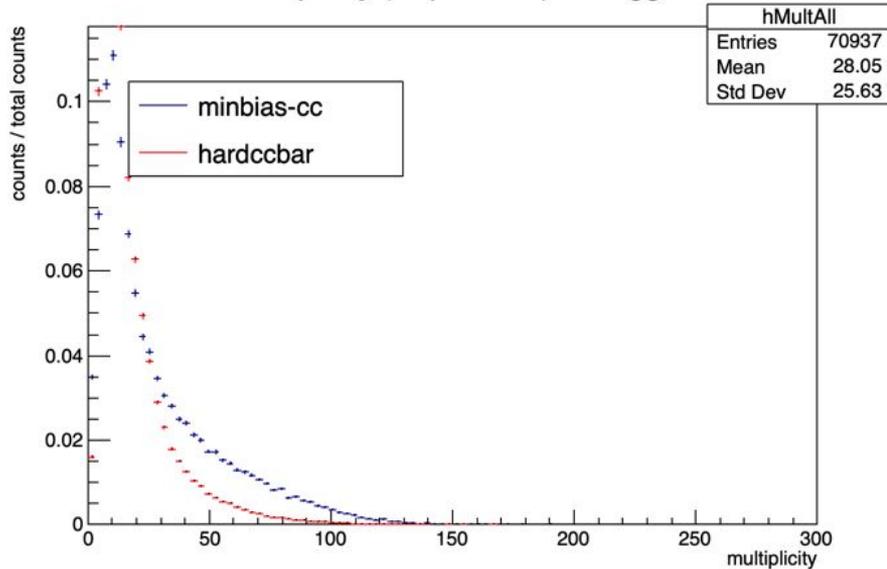
- nCH as nMPI proxy: <https://journals.aps.org/prd/pdf/10.1103/PhysRevD.102.076014>
- Plateau at high nCH: <https://inspirehep.net/files/0fc2ffef7a5e4a6073e599630b3fd2d8>

Conclusion plots

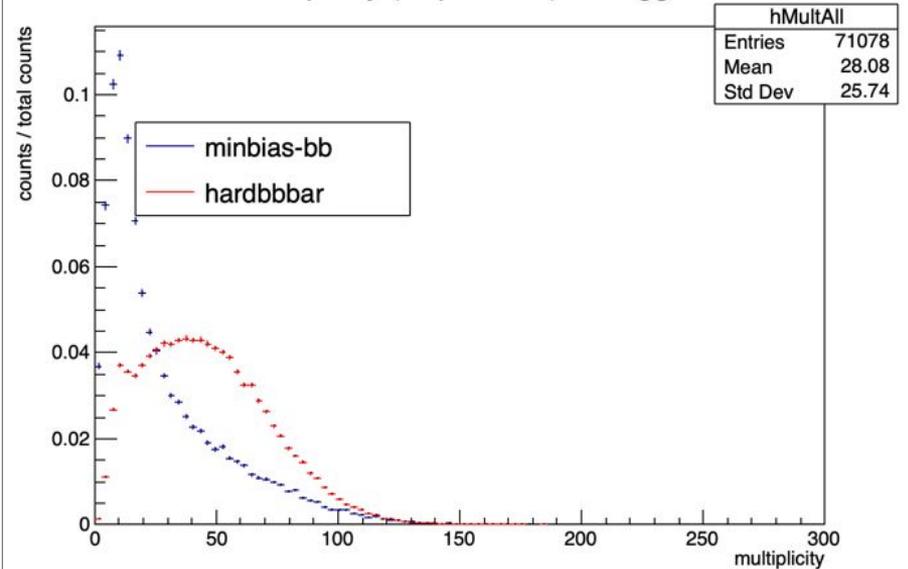
tldr

Monash multiplicity

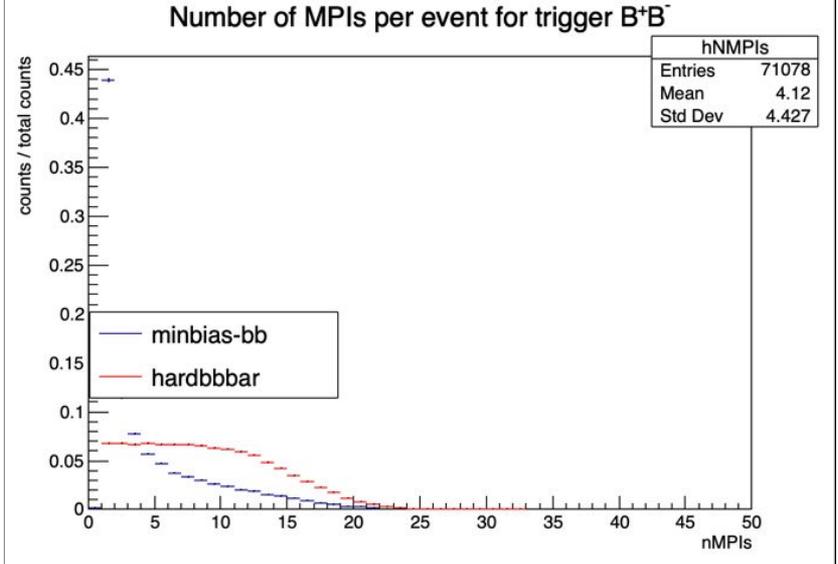
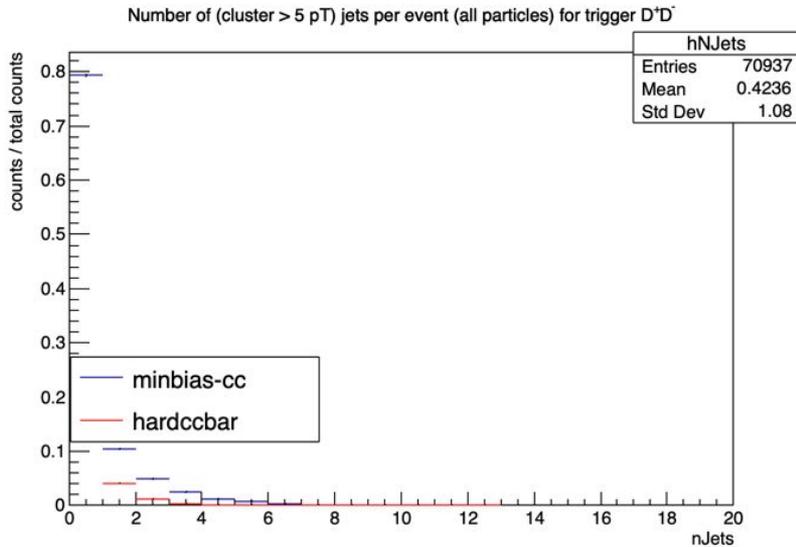
Event multiplicity (all particles) for trigger D^+D^-



Event multiplicity (all particles) for trigger B^+B^-

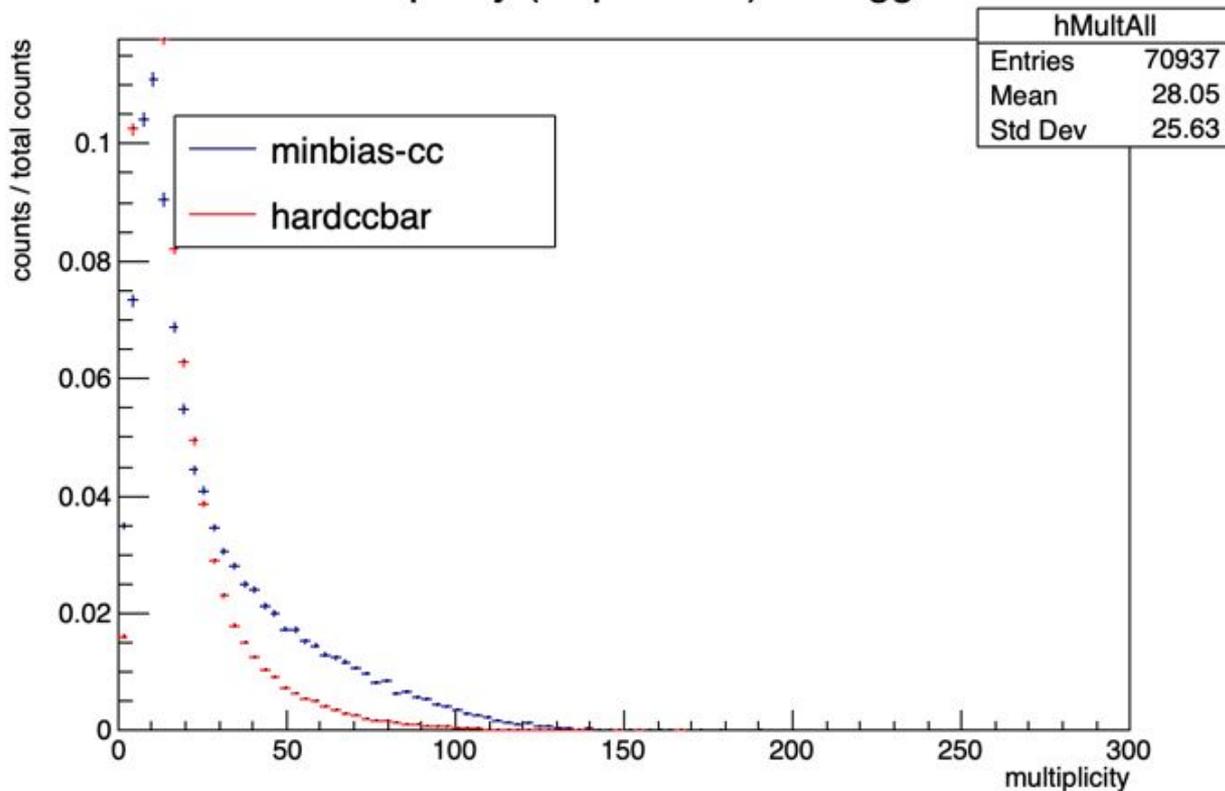


Monash number of jets



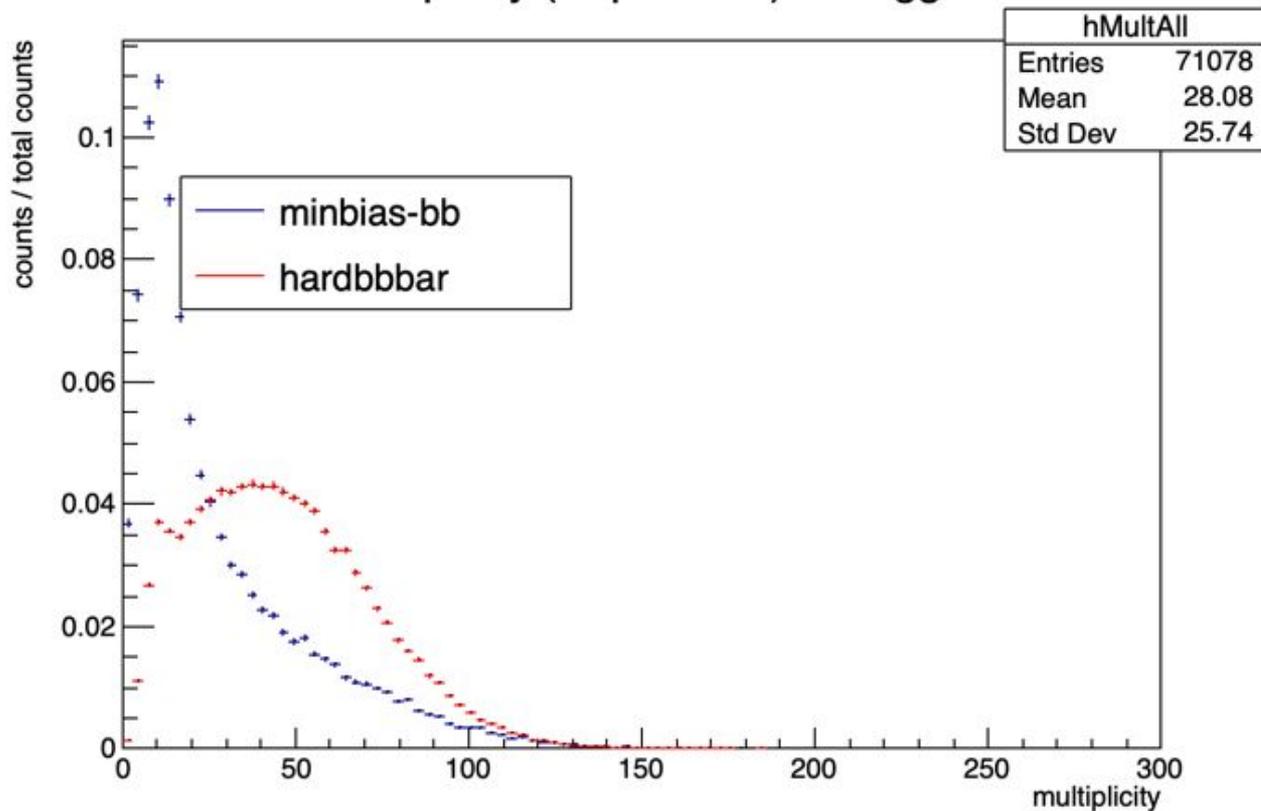
Charm Monash: multiplicity

Event multiplicity (all particles) for trigger D^+D^-



Beauty Monash: multiplicity

Event multiplicity (all particles) for trigger B^+B^-



Comparison of simulation types

4 min hardbbbar = 12 000 \sim 1e4 pairs B+B-

4 min hardqcd:all = 96 \sim 1e2 pairs B+B- ...

pT 1 \rightarrow 5 then 2500 \sim 2.5e3 pairs B+B-

with hardbbbar + hardccbar = 10 000 \sim 1e4 pairs B+B-

Big issue ?

\rightarrow needs $(1e2)^2 = 1e4$ factor extra statistics...

Probably worse for all the other beauty species

Conclusion: multiplicity

Big differences between hardccbar and hardbbbar.. In particular, the hardbbbar has a very particular shape.

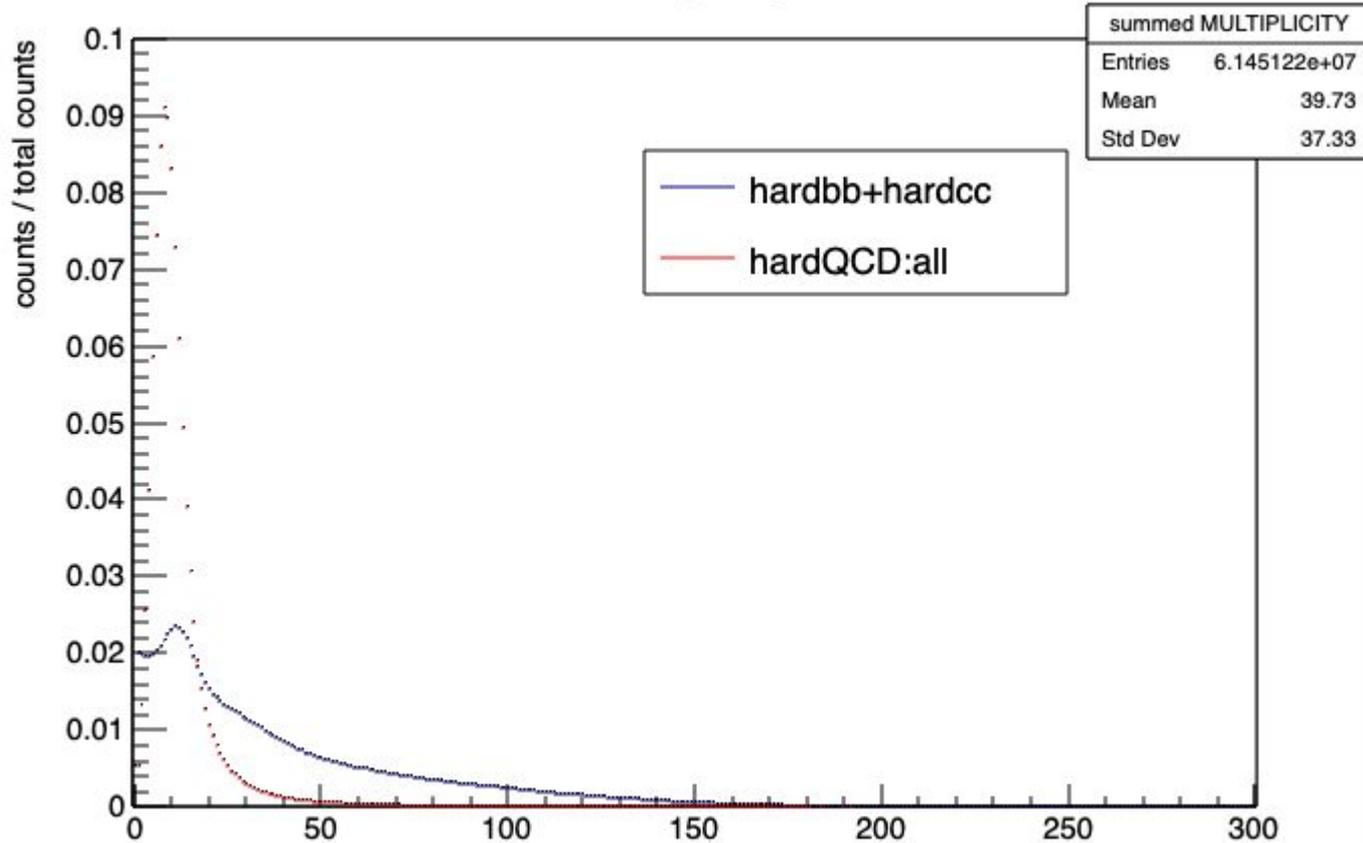
Also unexpected that hardccbar seems to have lower multiplicities than the minbias-cc

Monash and junctions are consistent with each other

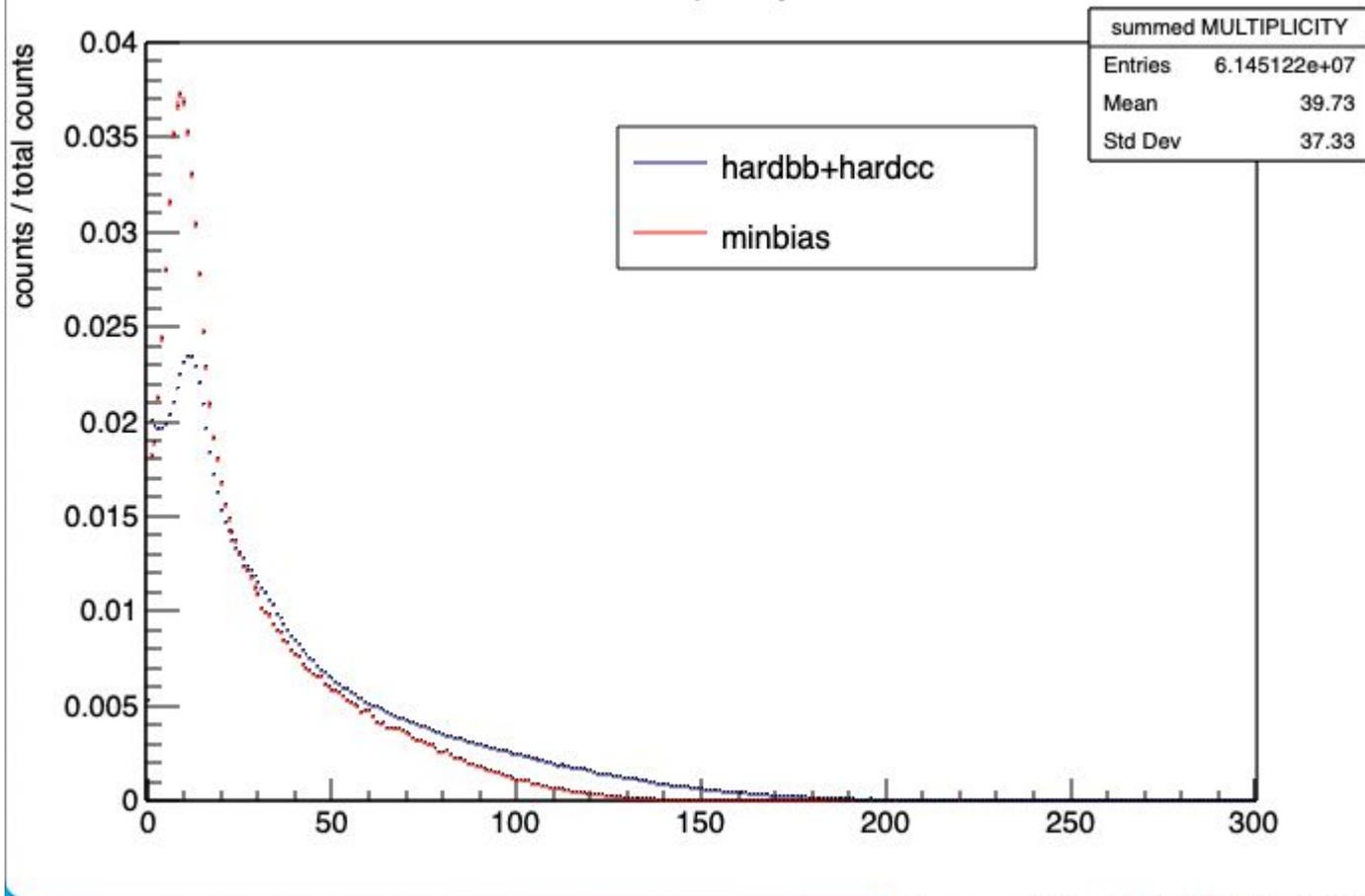
Idea

**05/10/2025. Comparisons using
hardbbbar + hardccbar at the
same time**

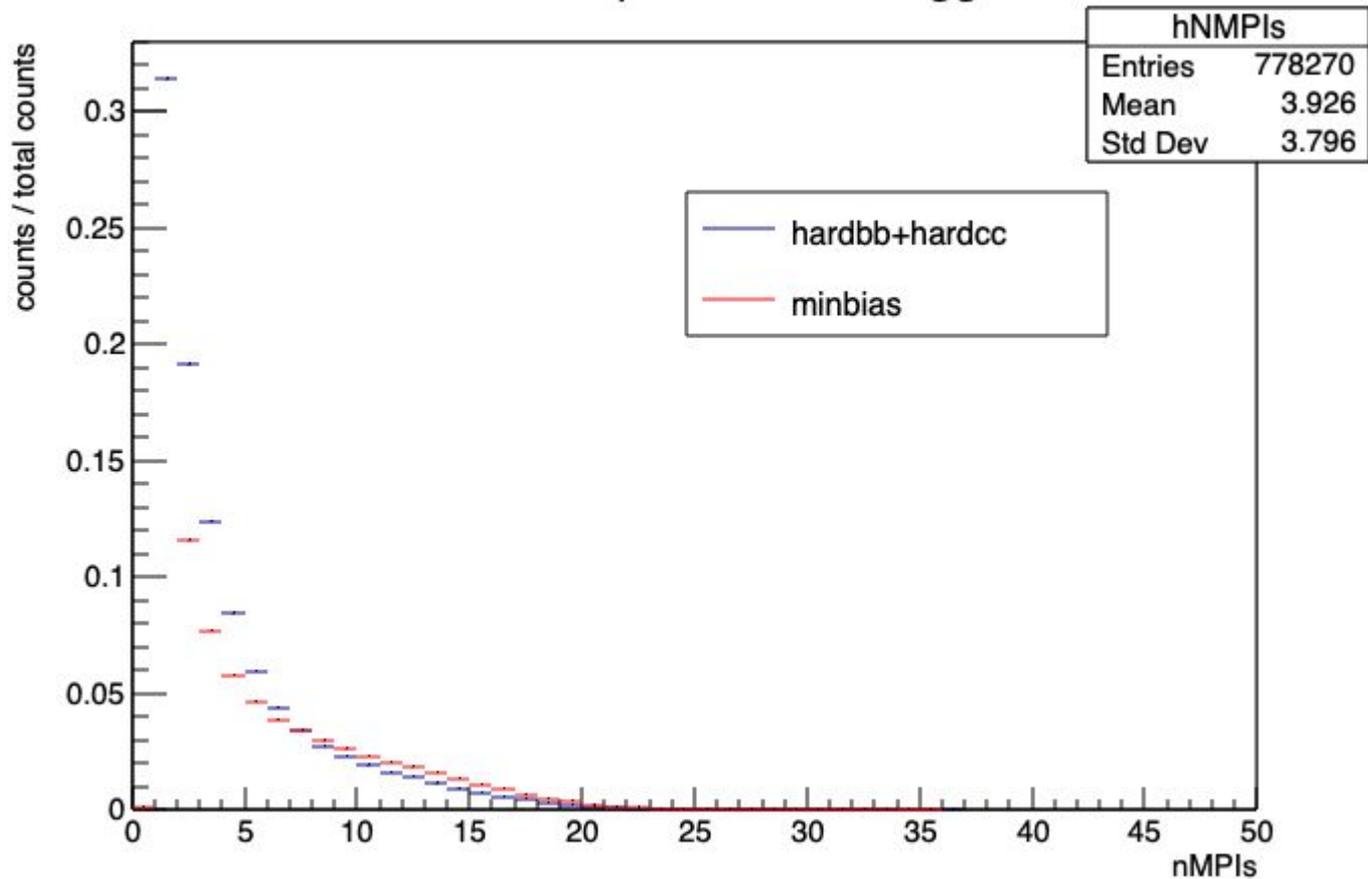
Multiplicity



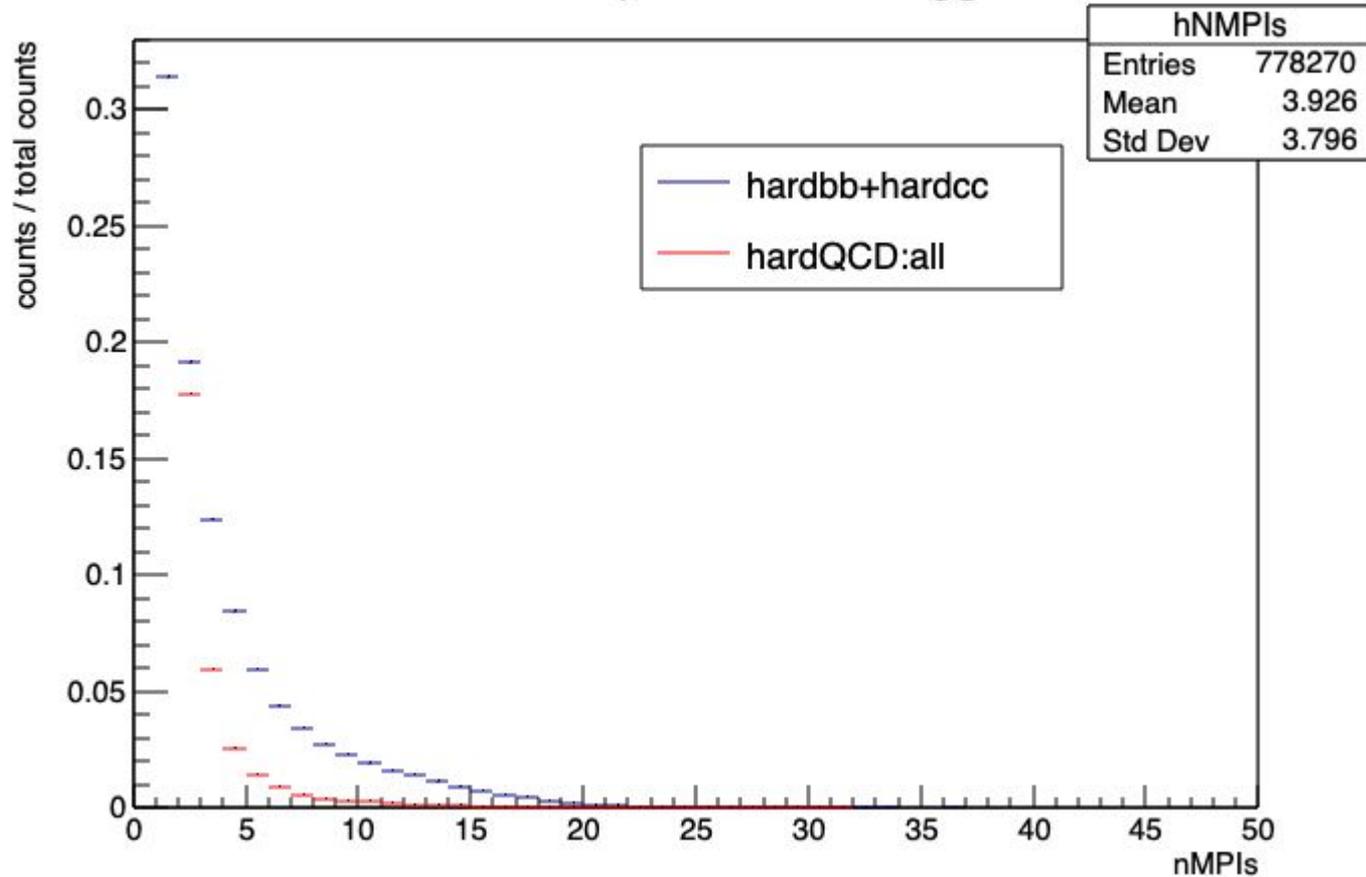
Multiplicity



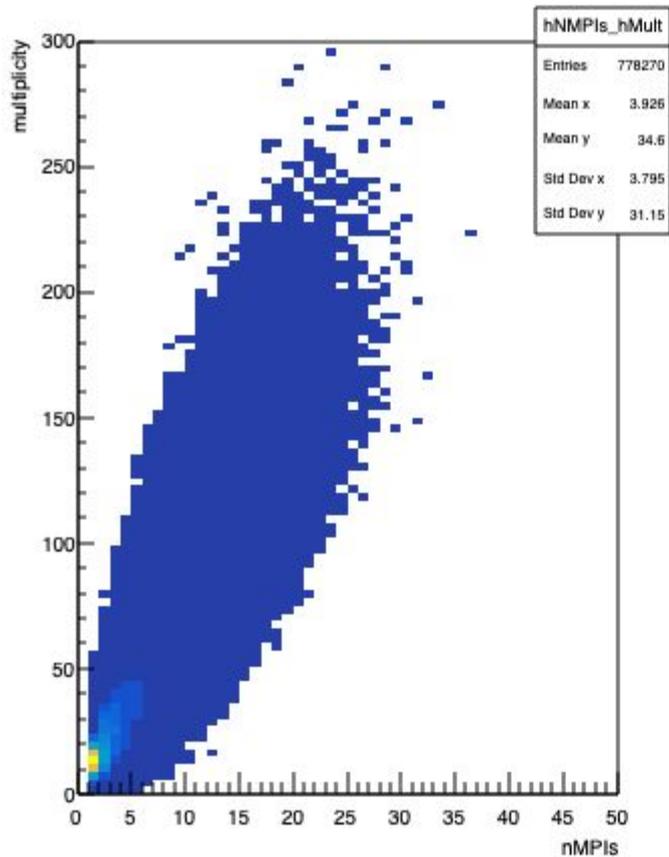
Number of MPIs per event for trigger B^+B^-



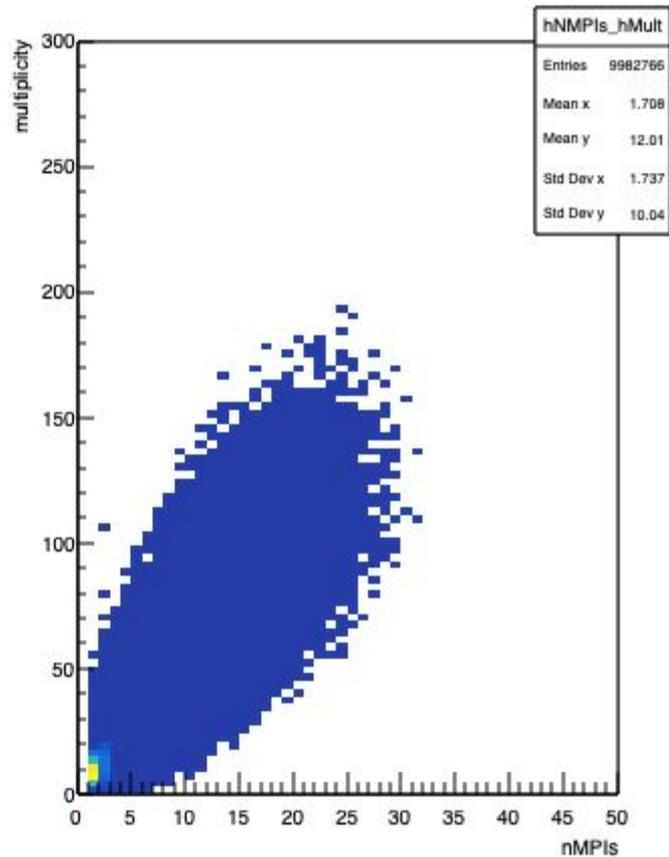
Number of MPIs per event for trigger B^+B^-



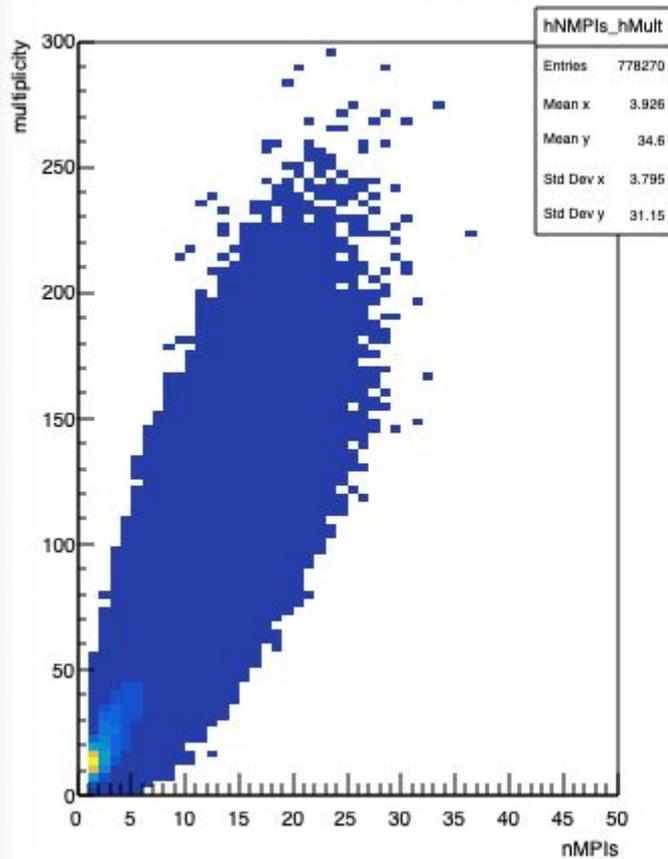
hardbb+hardc_Number of MPIs and multiplicity for trigger B \bar{B}



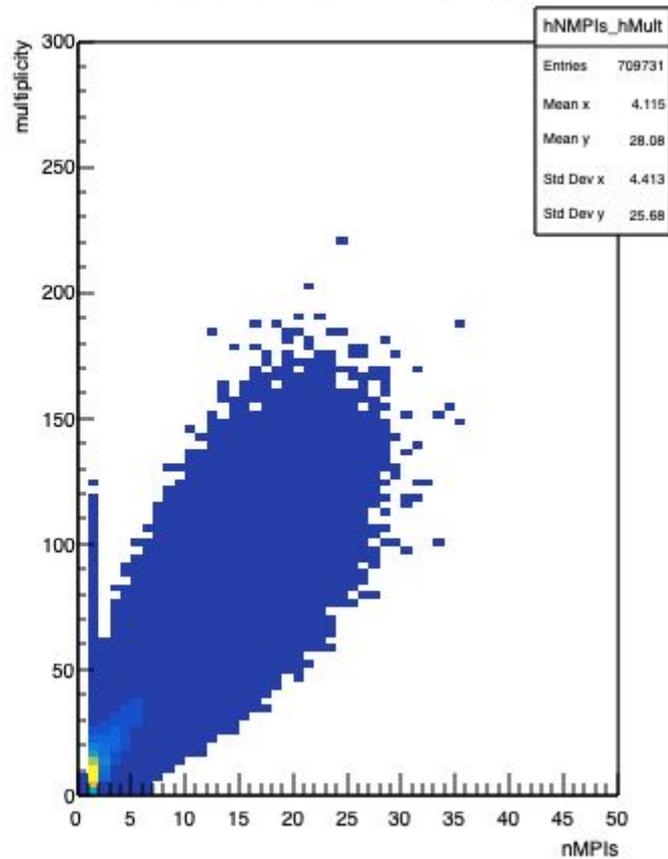
hardQCD:all_Number of MPIs and multiplicity for trigger B \bar{B}



hardbb+hardc_Number of MPIs and multiplicity for trigger B*B



minbias_Number of MPIs and multiplicity for trigger B*B



Conclusions

Interestingly enough, it seems like the $\text{hardbbbar} + \text{hardccbar}$ is more close to minimum bias than it is to hardQCD:all

- Except for a trend of an increased amount of lower multiplicities \rightarrow makes sense due to absence of low p_T jets in $\text{hardccbar} + \text{hardbbbar}$

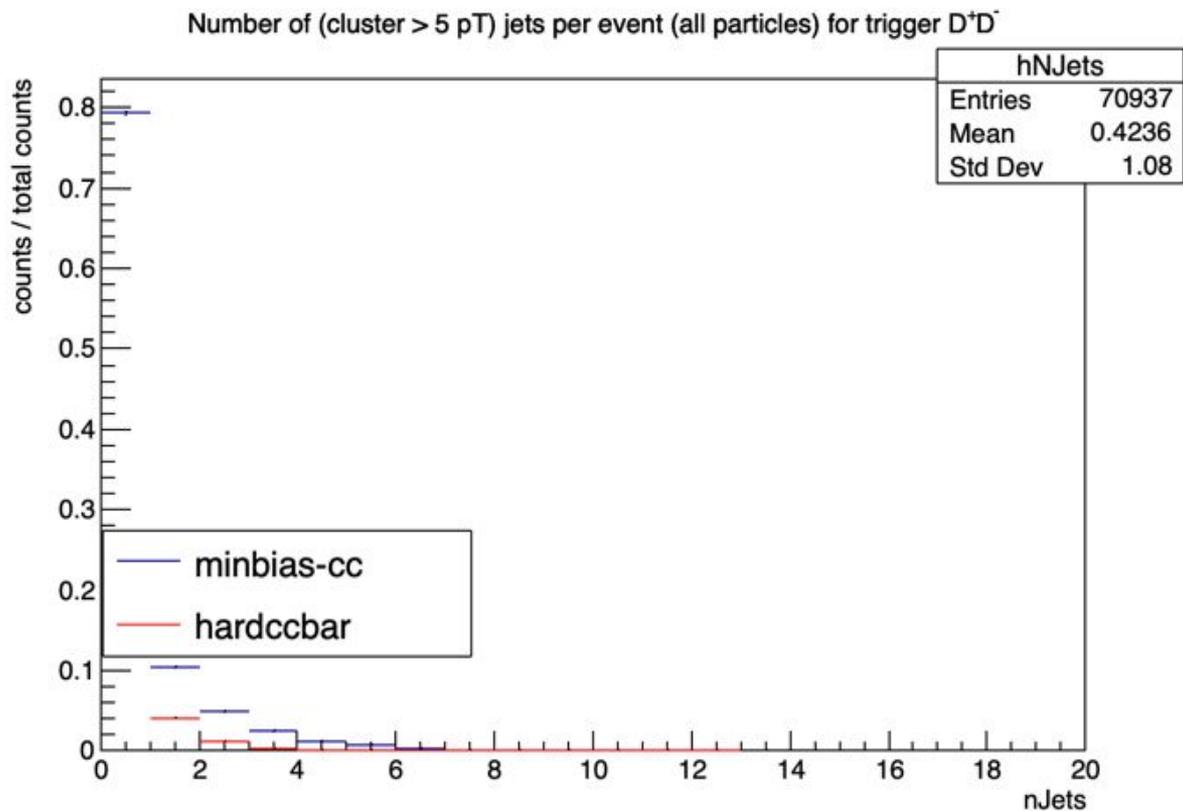
Another huge advantage is that we would now use the same event structure and same multiplicity distribution for both beauty and charm \rightarrow fair comparison, less bias

Comparison might even be better than using hardQCD:all and gives us way more statistics (factor 10^4 difference for beauty)

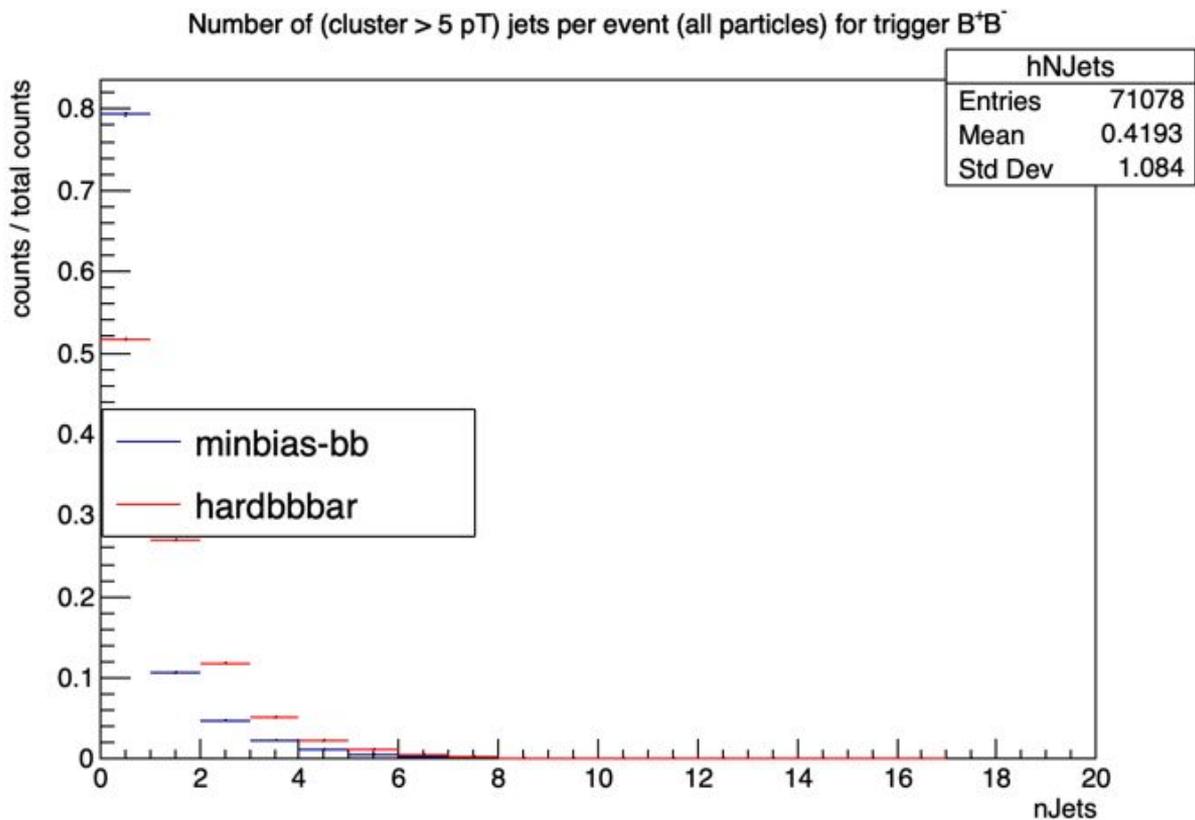
Translation from $n\text{MPIs}$ to multiplicity looks about the same (and better than the MB, as before)

More plots (not that important to look at)

Charm Monash: number of jets



Beauty Monash: number of jets



Conclusion: number of jets

MB has a lot of 0 jet events, in general more events with fewer jets than biased simulations (makes sense).

- Somehow for beauty this trend is mostly inverted. It seems like the hardccbar simulations are more ‘natural’ and closer to MB, while the bb-bar are more artificial. Maybe a hard bb jet is usually formed in the biased bb simulations, while for charm it’s less ‘forced’?
- At high number of jet events this seems to ‘converge’ to the same trend for beauty and for charm

Monash and junctions show similar trends

Variable: sphericity

Sphericity basically describes how ‘spherical’ the event is

- A value close to 0 is less spherical, i.e. more like an e+e- event with a characteristic di-jet.
- A value close to 1 is more spherical, like the typical hadron collision events.

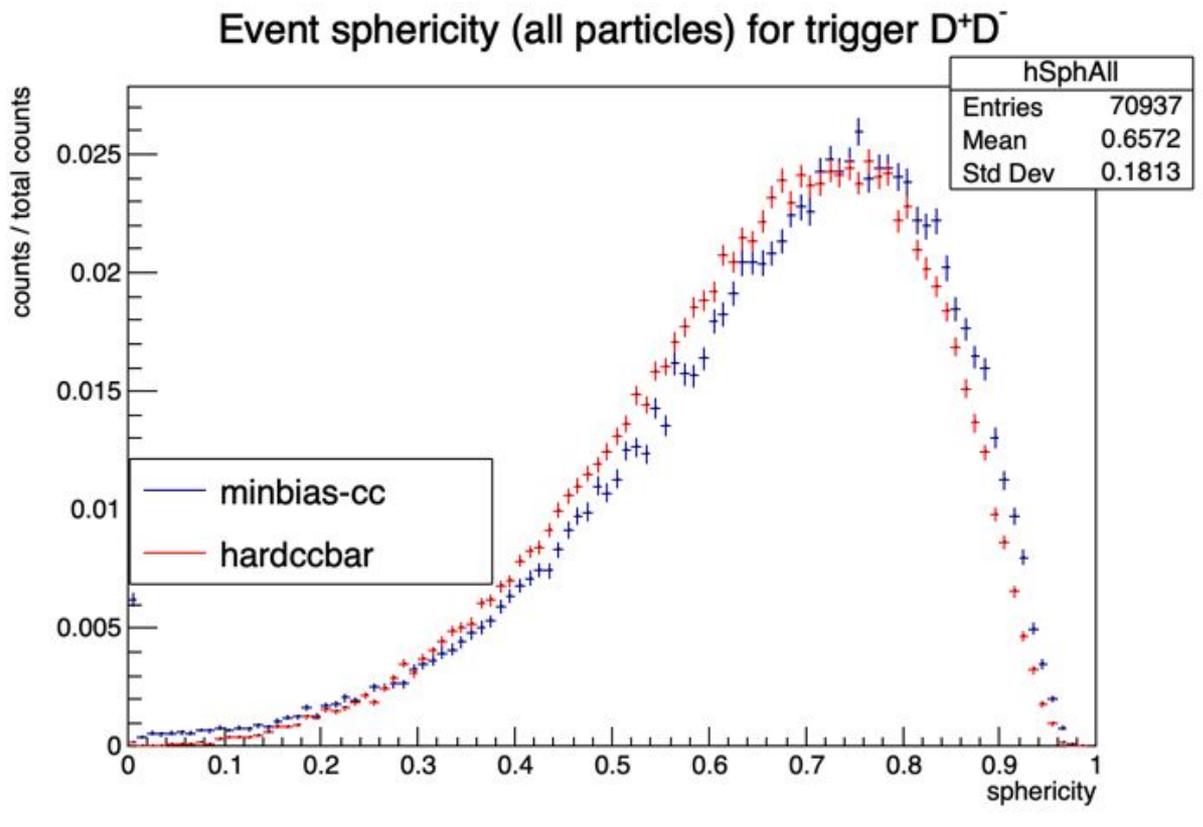
I wanted to see if, like it is used in the [paper](#), the sphericity can be used to check if events have jets, or are more jet-like.

- (i) Transverse sphericity: this quantity allows one to know whether a dijetlike structure is present in the event [33]. It is defined for a unit vector $\hat{\mathbf{n}}_s$ which minimizes the ratio:

$$S_0 \equiv \frac{\pi^2}{4} \min_{\hat{\mathbf{n}}_s} \left(\frac{\sum_i |\vec{p}_{T,i} \times \hat{\mathbf{n}}_s|}{\sum_i p_{T,i}} \right)^2, \quad (1)$$

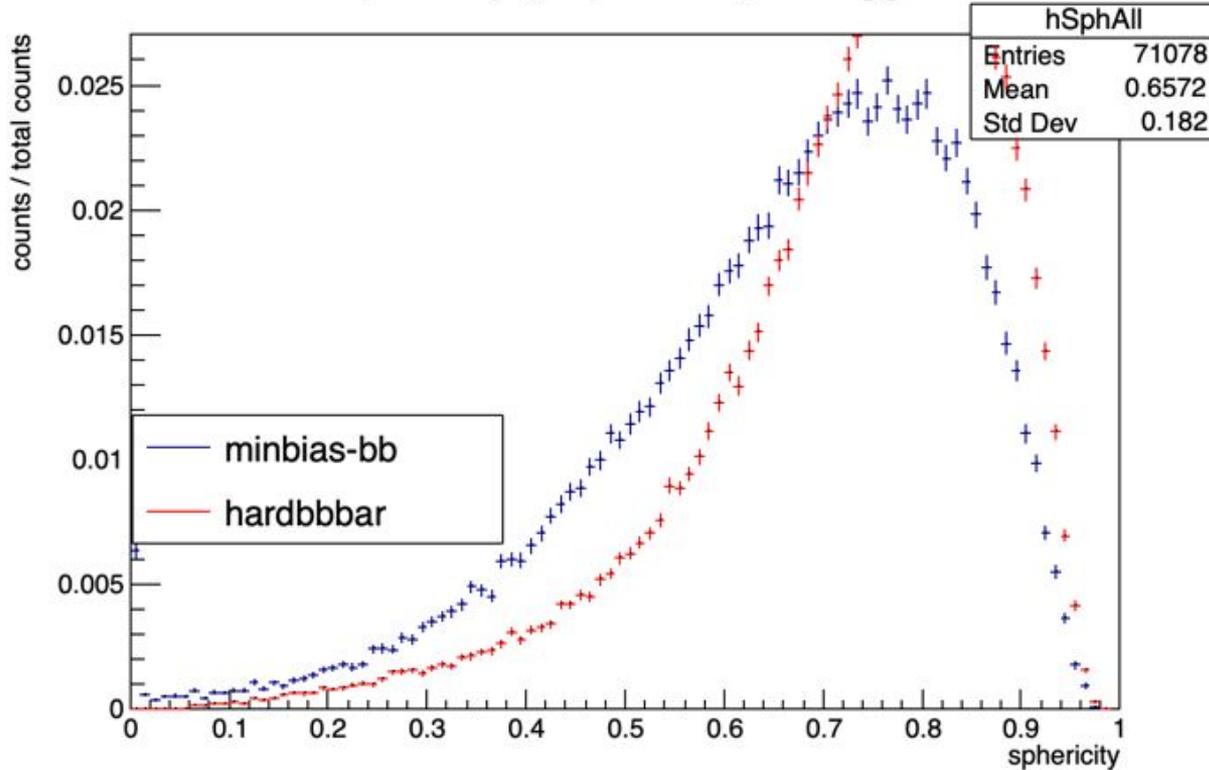
paul.veen@cern.ch

Charm Monash: sphericity (all)



Beauty Monash: sphericity (all)

Event sphericity (all particles) for trigger B^+B^-



Conclusion: sphericity

Soft and hard very similar.

In charm only marginal differences between all categories. Needs more statistics.

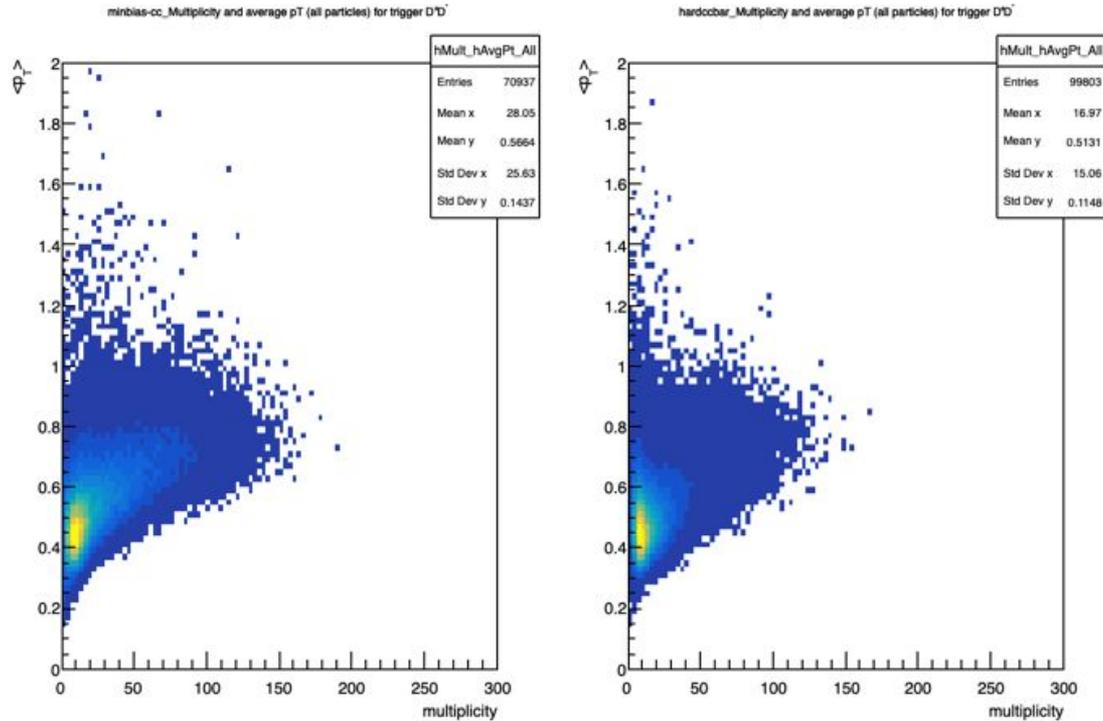
For beauty, something interesting: it seems like the hardbbbar events are a bit more spherical, i.e. less jet-like, which was my hypothesis from the number of jets earlier.

Two-dimensional histograms

Always

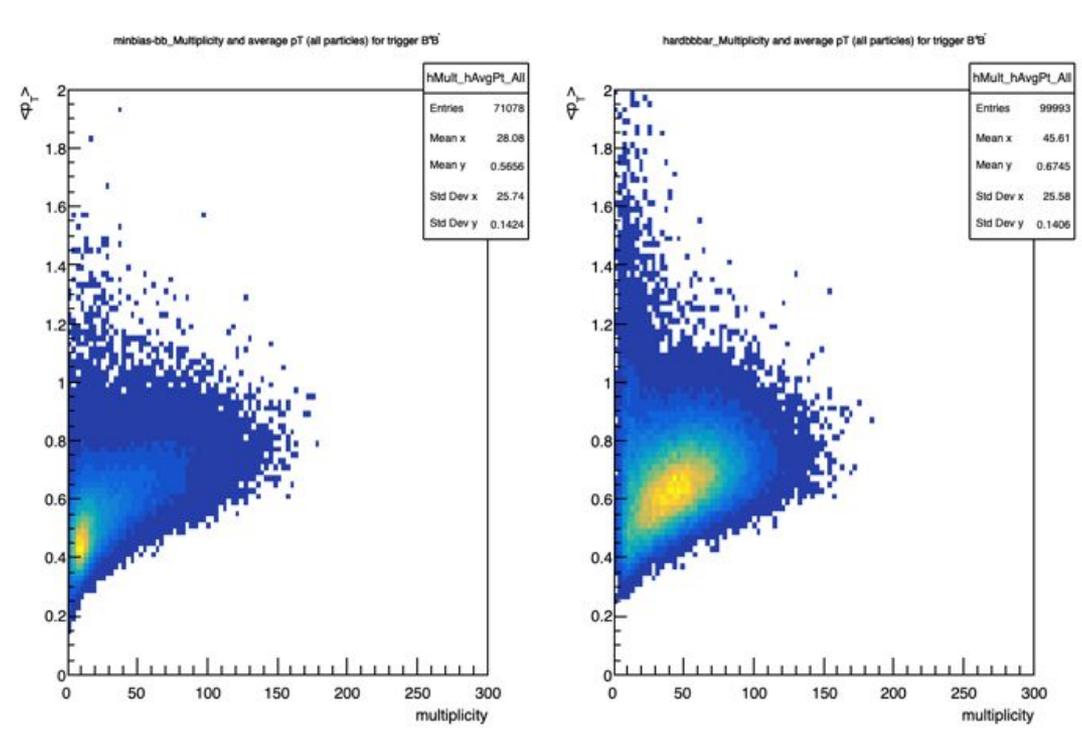
- left: minbias
- right: hardccbar/bbbar

Charm Monash: multiplicity vs $\langle pT \rangle$ (all)



paull@cern.ch

Beauty Monash: multiplicity vs $\langle p_T \rangle$ (all)



Conclusion: multiplicity vs $\langle pT \rangle$

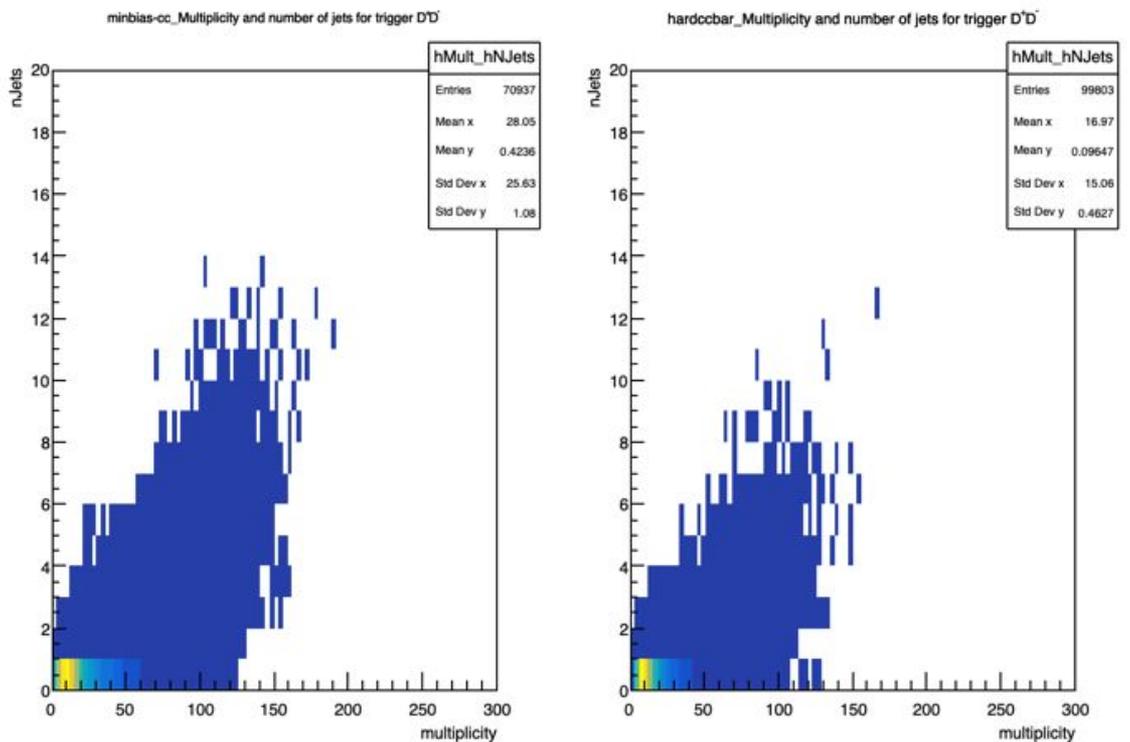
In junctions the trends seem to be sharper than in monash (can also be due to difference in statistics?)

In general, $\langle pT \rangle$ shows increasing trend with increasing multiplicity but saturates rather quickly too. Exactly what I expected and similar to what is seen in the balancing studies.

It is not so clear to me what conclusion to draw from the MB and boosted simulations. Needs more statistics. Though it seems like the trends saturate a bit faster in MB, as I would expect.

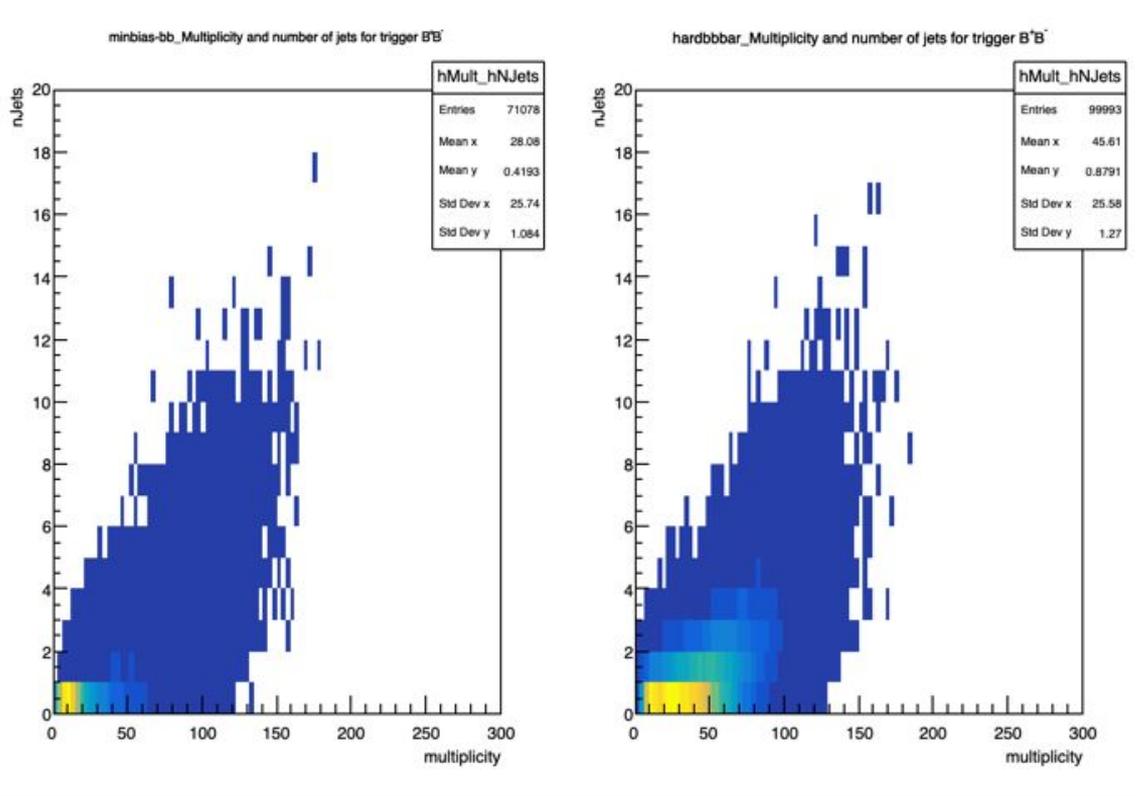
Soft and all have only minor effects due to ‘ignoring’ particles

Charm Monash: multiplicity vs number of jets



paul.veen@cern.ch

Beauty Monash: multiplicity vs number of jets



pau.veen@cern.ch

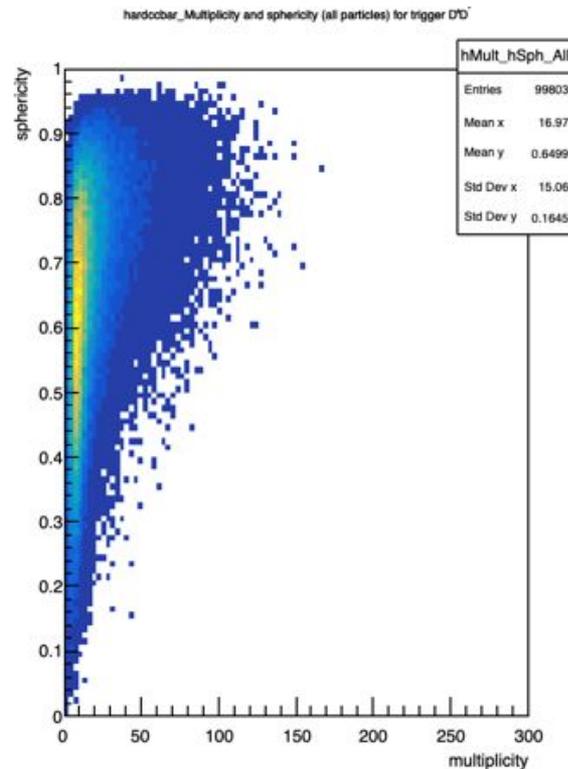
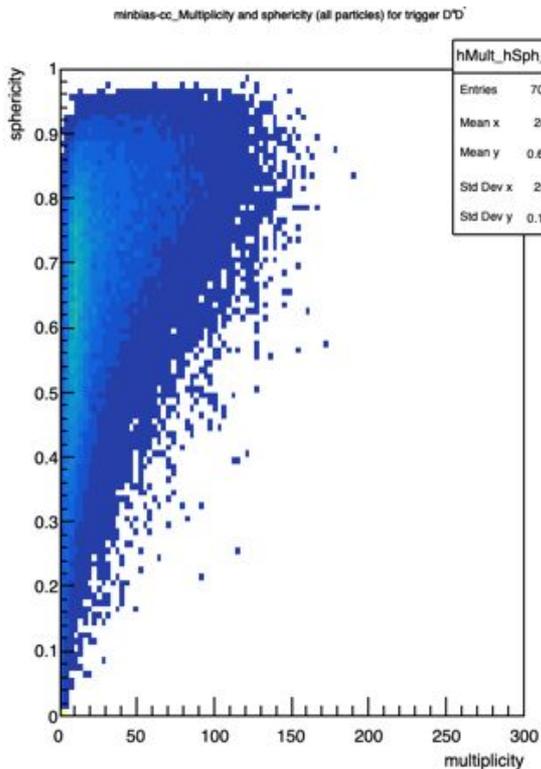
Conclusion: multiplicity vs number of jets

To me it seems like in all categories, there are no big differences.

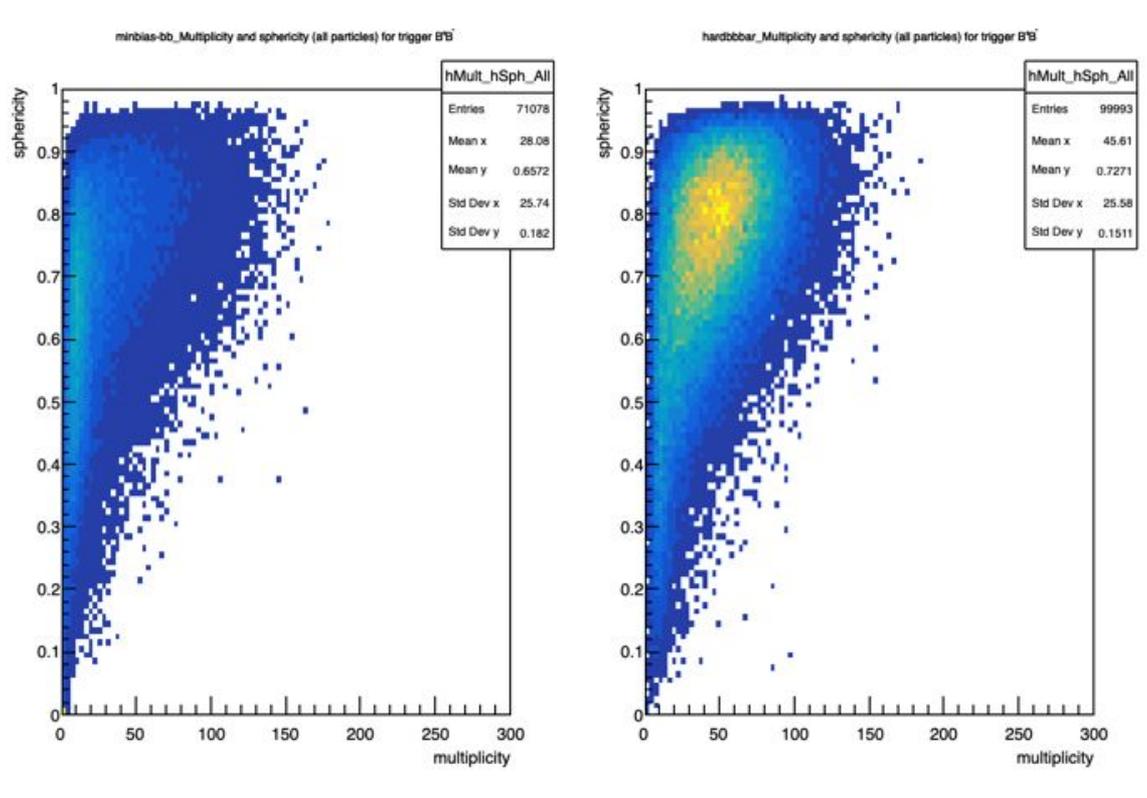
The main trend is that the number of jets increases (sharply) with increasing multiplicity.

Maybe it's statistics, but it could be that monash has on average more jets than junctions.

Charm Monash: multiplicity vs sphericity (all)



Beauty Monash: multiplicity vs sphericity (all)



paul.vech@cern.ch

Conclusion: multiplicity vs sphericity

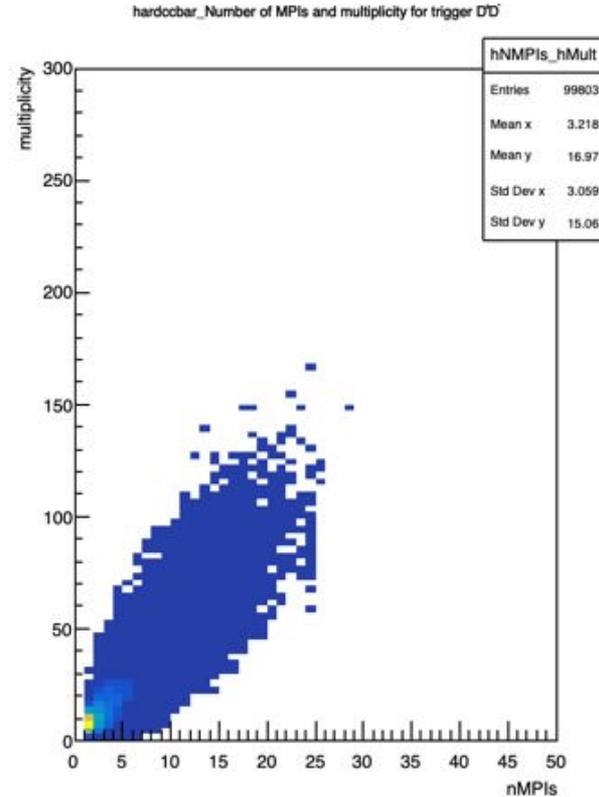
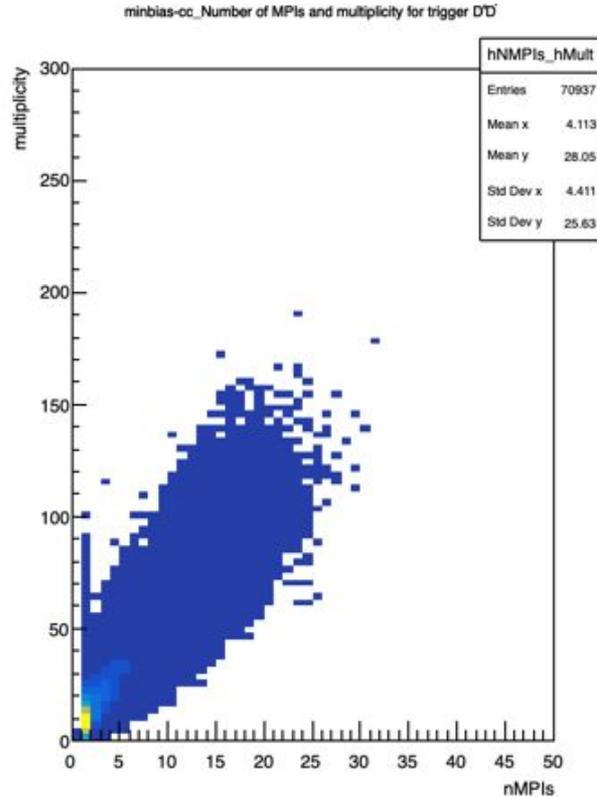
For me sphericity is the most difficult variable to interpret, which is why I included these correlation plots.

The clearest trend is that with higher multiplicity the sphericity gets closer to 1, which makes sense, as events will be more ‘busy’ and hence more ‘spherical’ (hadronic-like).

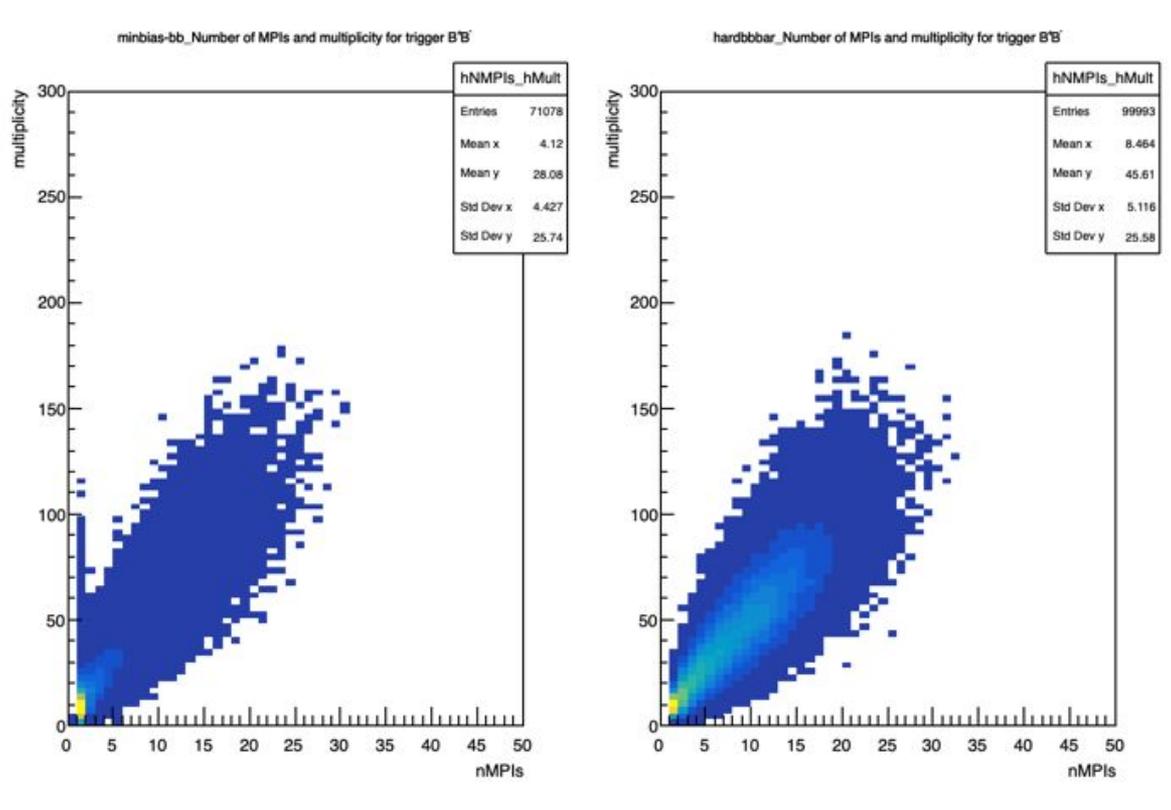
There don’t seem to be large differences in Monash and Junctions, but it seems like the beauty boosted simulations are much more centered at higher multiplicity (but the same sphericity) compared to charm. This is also what we saw in the multiplicity histograms.

- The conclusion is the same: beauty boosted is quite different, while charm MB, charm boosted and beauty MB are similar.

Charm Monash: number of MPIs vs multiplicity



Beauty Monash: number of MPIs vs multiplicity



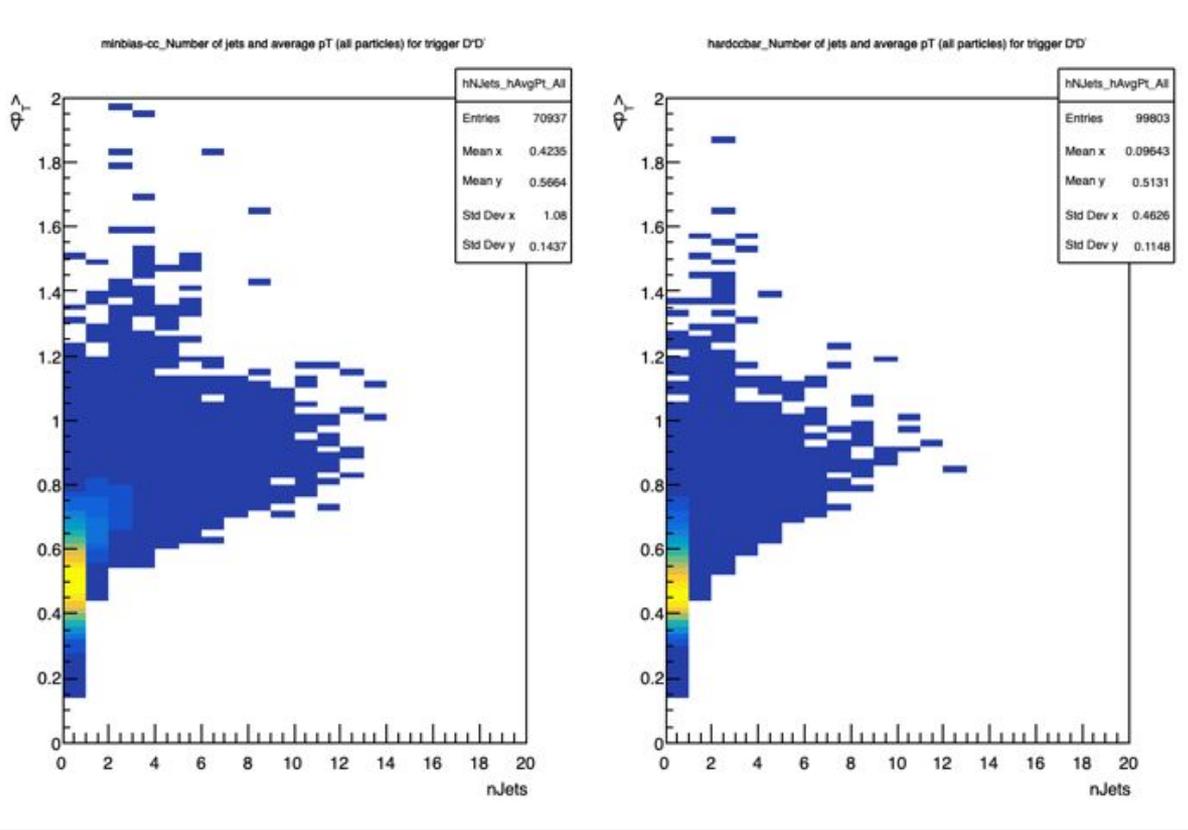
Conclusion: number of MPIs vs multiplicity

These plots are more of a consistency check: is multiplicity a good probe for MPIs and how similar are all the simulations?

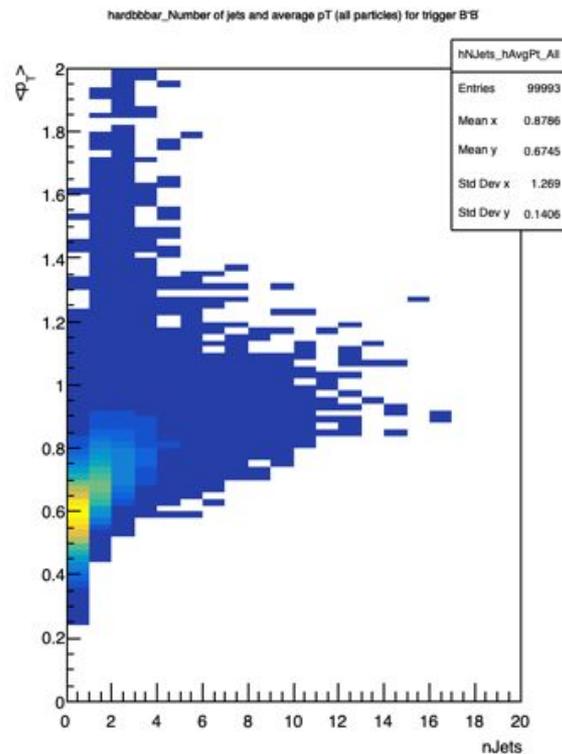
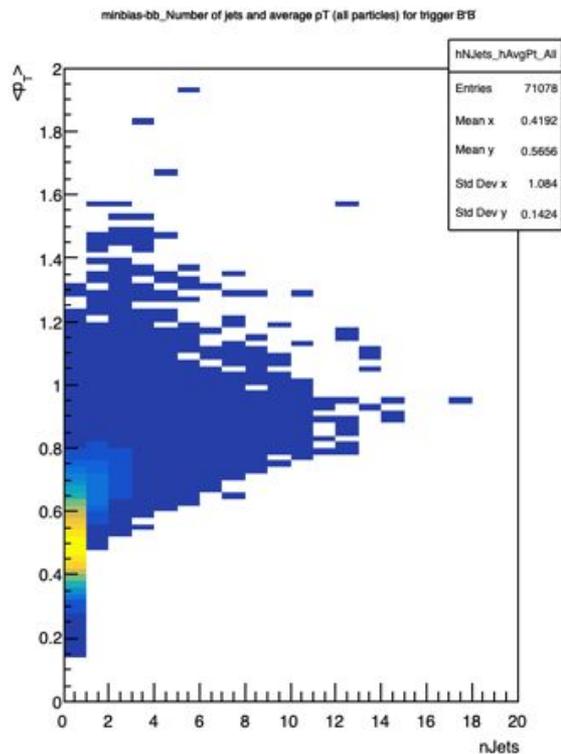
- They all have the (expected) trend that the multiplicity increases as the number of MPIs increases, and all trends in all simulation types are similar.

It might be worth noting the peak at $n\text{MPIs} = 1$ for the MB simulations. This ‘peak’ is absent in the boosted simulations.

Charm Monash: number of jets vs $\langle pT \rangle$ (all)



Beauty Monash: number of jets vs $\langle p_T \rangle$ (all)



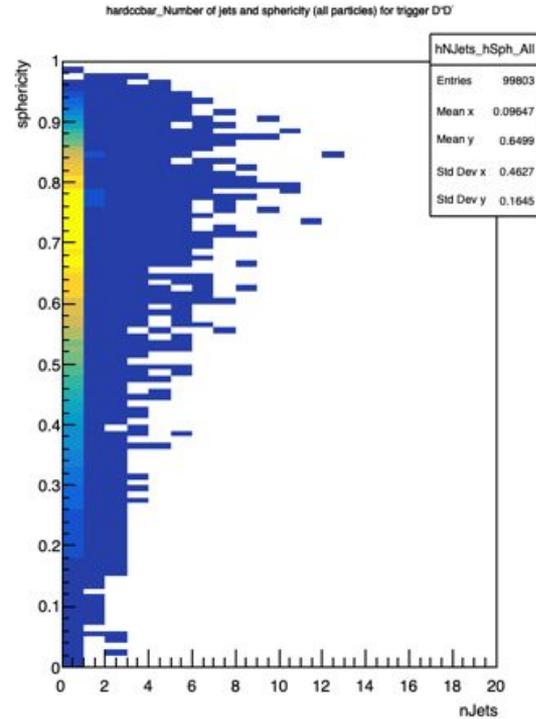
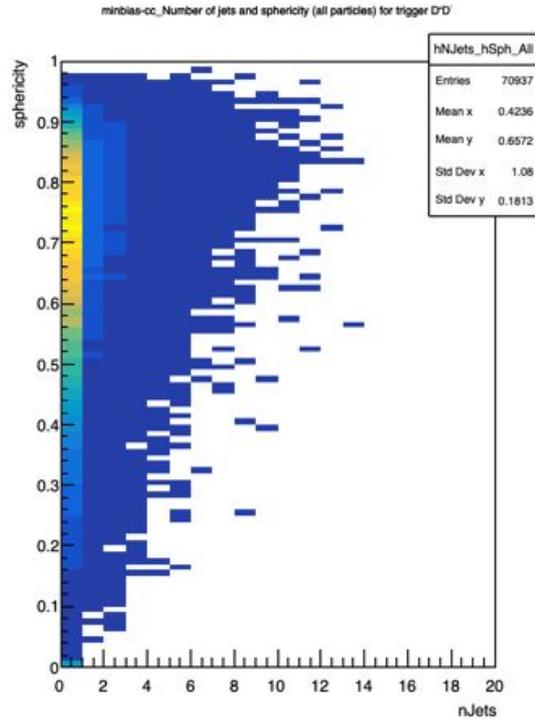
Conclusion: number of jets vs $\langle p_T \rangle$

I don't see many differences in trends between all the simulation types.

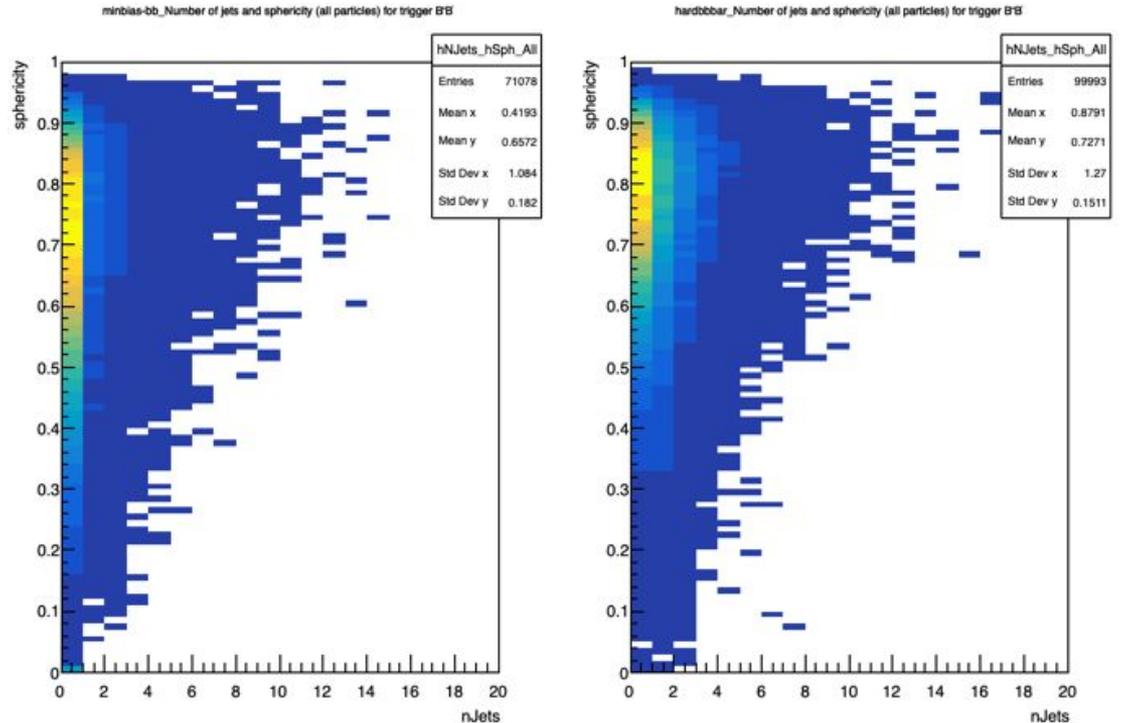
The general trend is what I expected, the $\langle p_T \rangle$ increases as the number of jets increases, but it saturates quickly.

- It is interesting that we don't see the (quite) different behaviour for beauty-charm in the $\langle p_T \rangle$ here.

Charm Monash: number of jets vs sphericity (all)



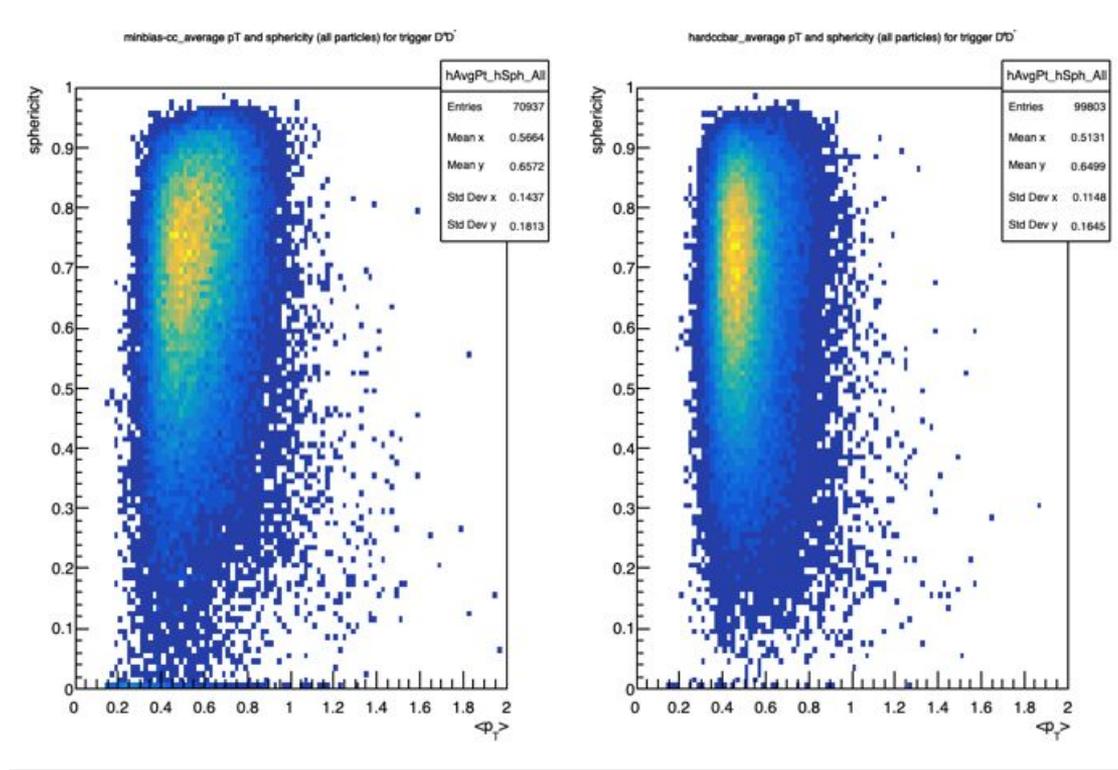
Beauty Monash: number of jets vs sphericity (all)



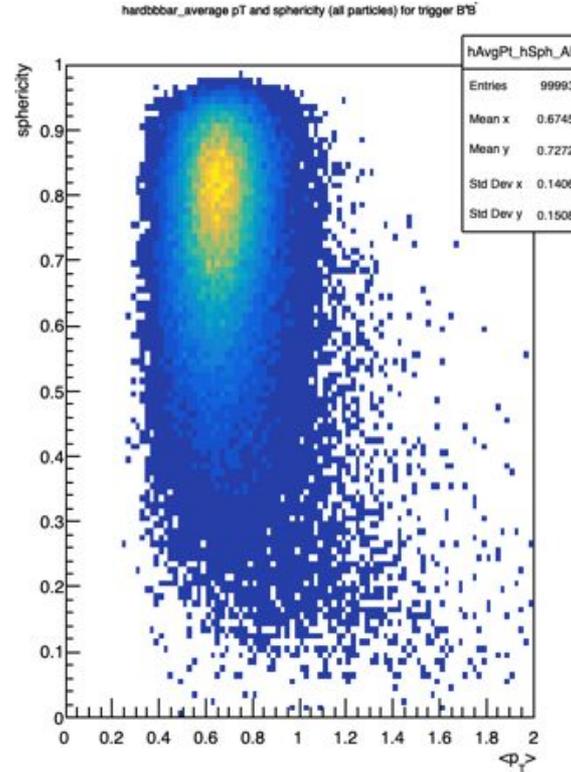
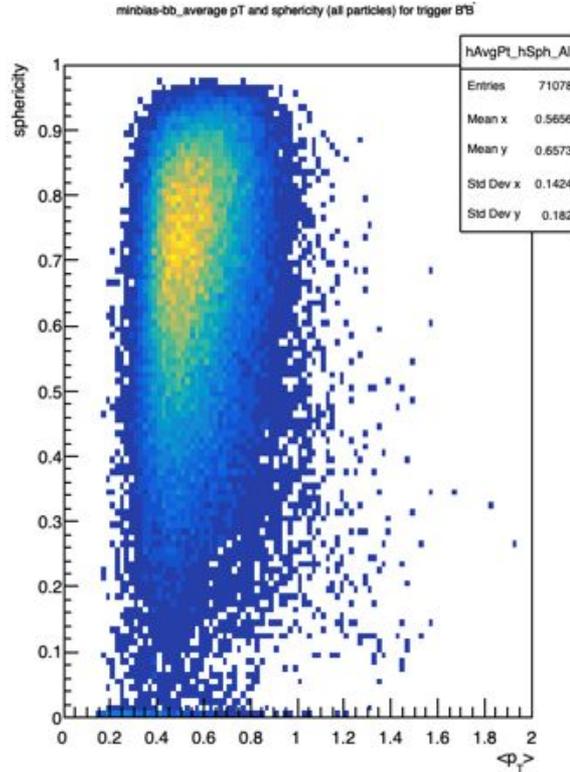
Conclusion: number of jets vs sphericity

Basically this doesn't give new information, because we have the correlation of number of jets and multiplicity. The same sphericity trends as before are seen here.

Charm Monash: $\langle pT \rangle$ vs sphericity (all)



Beauty Monash: $\langle p_T \rangle$ vs sphericity (all)



Conclusion: $\langle p_T \rangle$ vs sphericity

I think the most noticeable effect here is the ‘escaping tail’ for the boosted beauty all simulations. It seems like in those simulations the events with higher $\langle p_T \rangle$ seem to be more ‘jet-like’ and that it occurs more often than in the other ones.

- This does seem to agree with the hypothesis from earlier.

General conclusion

I think the most important conclusion is that charm MB, charm boosted and beauty MB are similar, while beauty boosted is quite different.

- The question is to understand why this is the case and how it can be ‘equalised’?

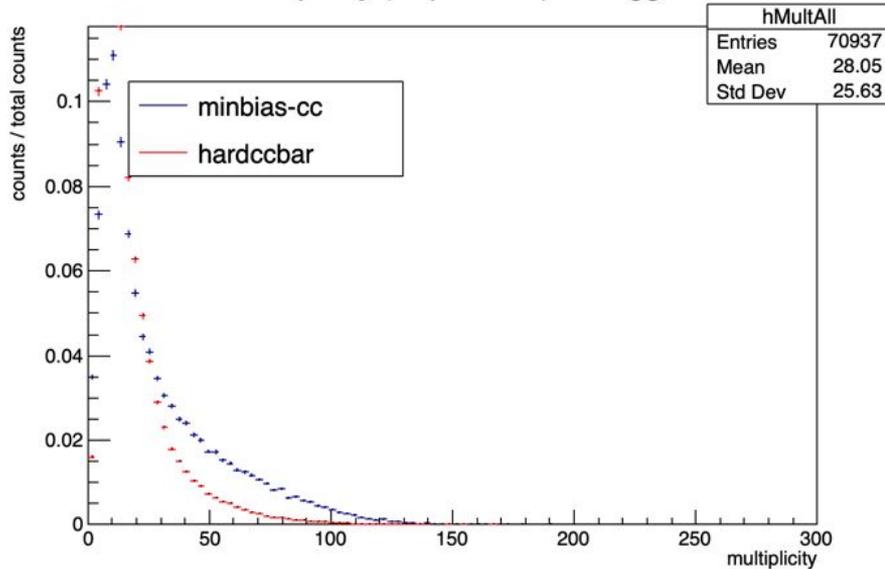
Some hints are found in the multiplicity spectra, the number of jets per event and the event sphericities (see next slides for easy reference to plots). It seems like the hardccbar simulations are more ‘natural’ and closer to MB, while the bb-bar are more artificial. Maybe a hard bb jet is usually formed in the biased bb simulations, while for charm it’s less ‘forced’?

At high number of jet events this seems to ‘converge’ to the same trend for beauty and for charm

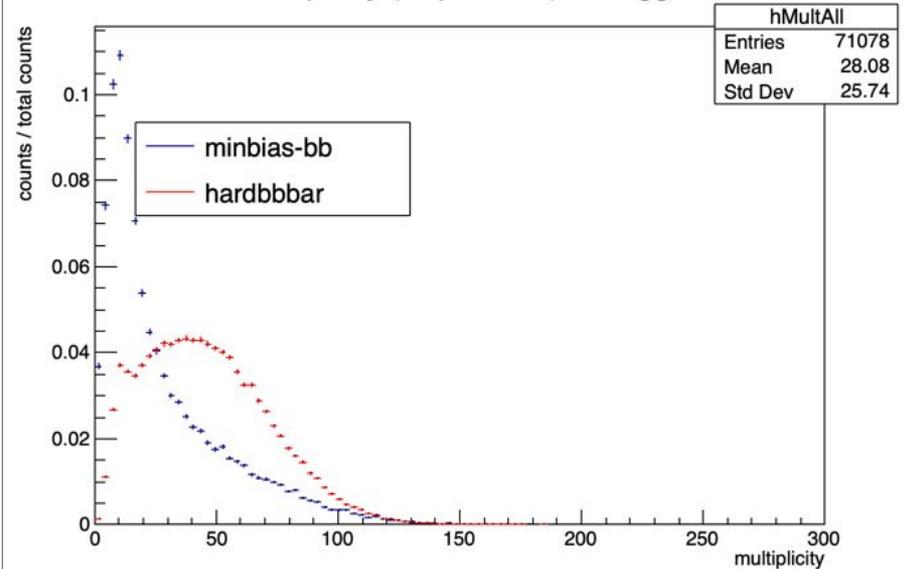
- But some results show the opposite, where hardbbbar events seem to be more ‘spherical’ (i.e. less ‘jet-like’) (e.g. in the 1D sphericity histograms).

Monash multiplicity

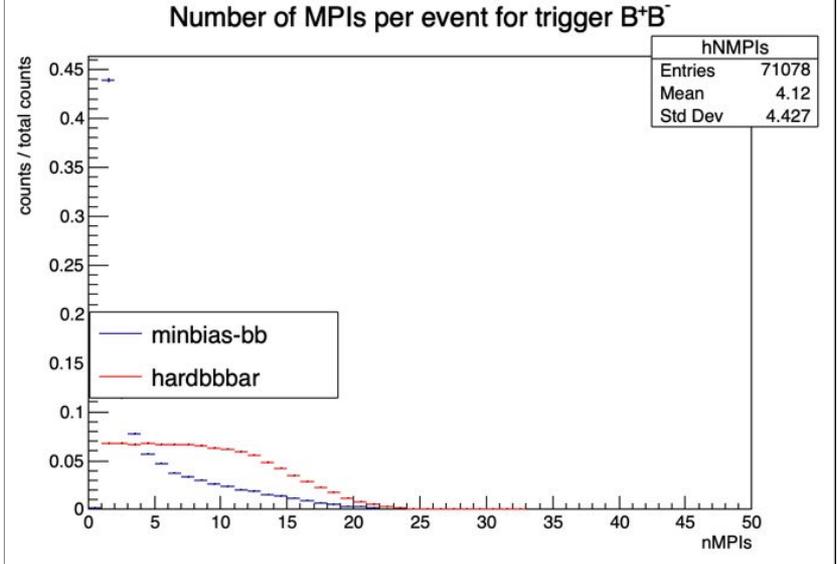
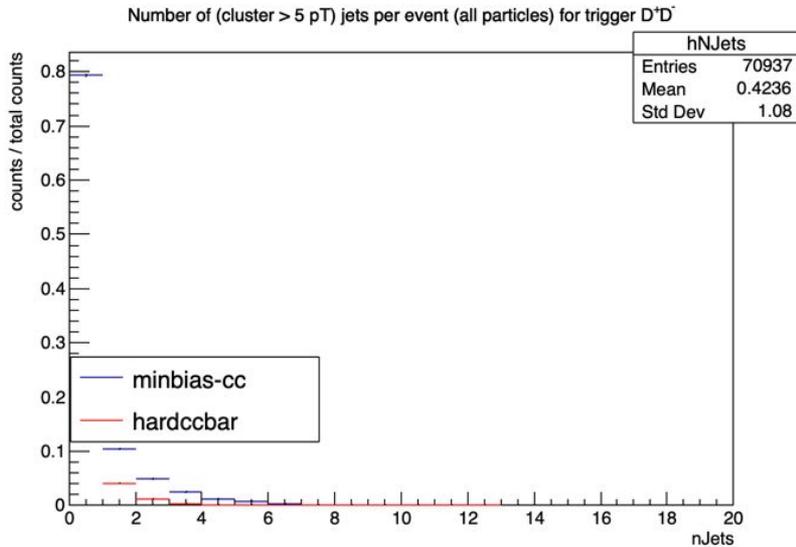
Event multiplicity (all particles) for trigger D^+D^-



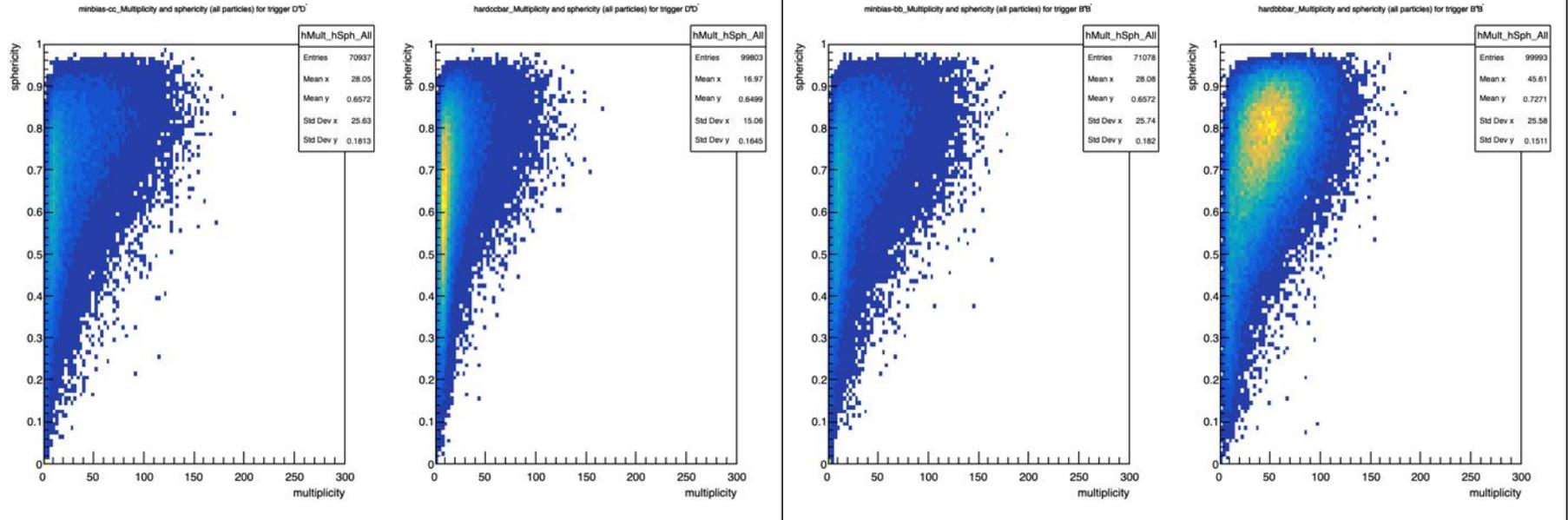
Event multiplicity (all particles) for trigger B^+B^-

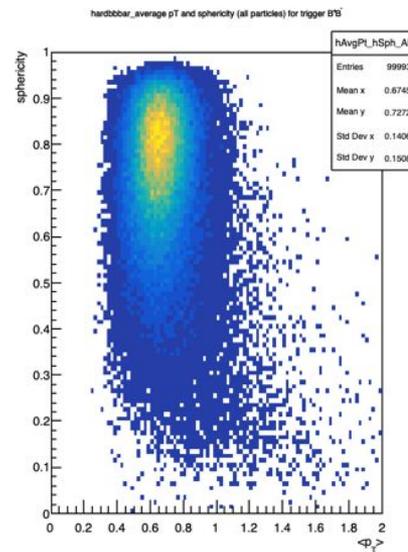
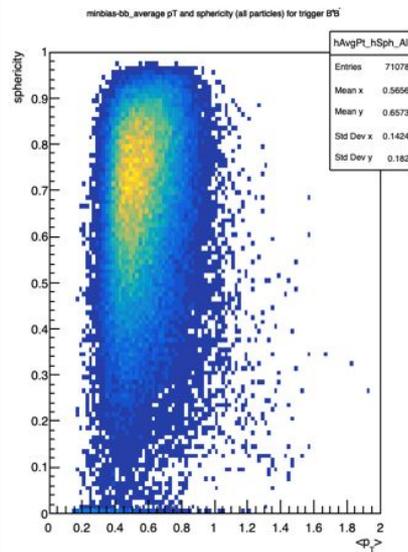
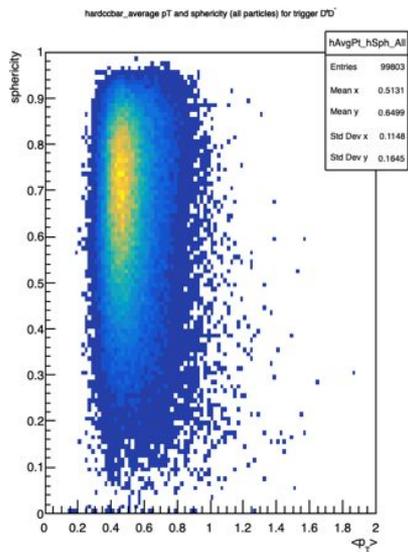
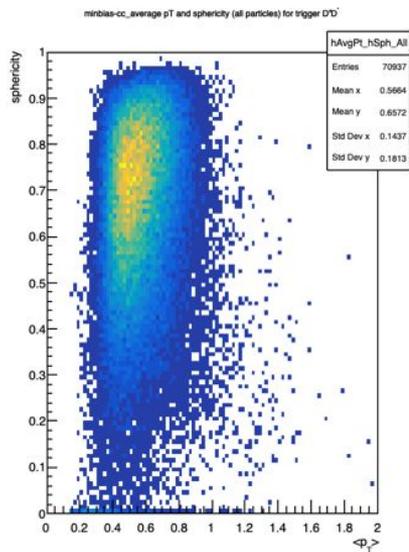


Monash number of jets



Monash multiplicity vs sphericity





General conclusion

Other comparisons:

- Only marginal differences between “soft” and “all”. Easier to focus on one of the two (easiest one is “all”).
- Essentially no differences between monash and junctions -> comparison between the two is “fair” and the same categorisation can be used

Considerations

Jet finding algorithm: many parameters

More statistics?

Pick variables to cut with? Repeat studies with those cuts to see the effect on the balancing?

Noticeable differences in $b\bar{b}$ and $c\bar{c}$ boosted simulations. Perhaps consider using both of them in each simulation, instead of just $b\bar{b}$ for beauty and $c\bar{c}$ for charm?

Repeat study with baryons?

Note on statistics

Code seems to run much faster with the following changes

- hardQCD instead of hardbbbar or hardccbar
- veto events with 0 multiplicity (we're not interested in them anyways..)
- no QC checks (no 'test' plots created for correlations)

Charm/event = 1.2 (hardQCD) vs 1.3 (hardccbar)

Beauty/event = 0.068 (hardQCD) vs 1.6 (hardbbbar)

-> 'loss' of about factor 100 for beauty.. Can we live with that? (consider also that hardQCD might be up to 10x faster..)



HANDS ON
- submitting jobs
on bash script



ALICE

Paul Veen



universit
PARIS-SACL

Updated scripts on github

<https://github.com/ppoava/MasterThesisNikhef/commit/8ea5213b90edc37454598861ace3b4bc55938e5e>

Updated workflow scripts

- merging of beauty and charm in simulation and analysis scripts
- use of hardbbbar and hardccbar for all simulations

```
// Multiplicity is defined as the number of charged primary particles in the final state of the event
// Particles to be considered primaries here are: electron, muon, pion, kaon, proton
if (abs(id) == 11 || abs(id) == 13 || abs(id) == 211 || abs(id) == 321 || abs(id) == 2212)
{ // Needs to be a primary particle
  if (status >= 81 && status <= 89)
  { // Needs to be prompt
    MULTIPLICITY++;
  }
}
```

Multiplicity is calculated for all events.. not just the ones actually containing the beauty and charm we look at?

Multiplicity definition now updated

```
if (particle.isFinal() &&
    (abs(particle.id()) == 11 || // e±
     abs(particle.id()) == 13 || // μ±
     abs(particle.id()) == 211 || // π±
     abs(particle.id()) == 321 || // K±
     abs(particle.id()) == 2212)) // p / p̄
{
    // Exclude weak-decay secondaries
    double ctau = pythia.particleData.tau0(motherID) * 3e10; // [cm]
    if (ctau < 0.01)
    { // lifetime < 0.01 cm → strong decay
        MULTIPLICITY++;
    }
}
```

General idea and motivation

- We need more statistics...
- Let's assume we need around 100 million events (of biased simulations, i.e. HardQCD)
 - Impossible to do with your computer (you can try..)
- On Stoomboot we can submit batch jobs to the cluster

Structure

First simulate pythia job (hardQCD) with X number of events

Submit this Y amount of times

Result is Y groups of X number of events ('batched')

This batching is nice because we can use it to determine the uncertainties (with **subsampling** = repeating the analysis on the separate datasets and comparing the outputs)

Submit again the batched jobs, but now with the analysis script

Proceed to merging (and keep the batched outputs as well)

- I implemented the subsampling in the plotting macro ([link](#)) so that it can be done at run-time. It needs some more testing, however..

Condor cluster

https://kb.nikhef.nl/ct/Stoomboot_cluster.html

https://kb.nikhef.nl/ct/Batch_jobs.html

Uses condor cluster (see also

<https://htcondor.readthedocs.io/en/latest/users-manual/submitting-a-job.html>)

Also important: storage: https://kb.nikhef.nl/ct/Storage_overview.html

sub.sh

```
executable          = test.sh
log                 = test.log
output              = outfile.txt
error               = errors.txt
## Can use "el7", "el8", or "el9" for UseOS or you can specify your own
## SingularityImage but an OS must be specified and in string quotations.
+UseOS              = "el9"
## This job can run up to 4 hours. Can choose "express", "short", "medium", or "lo
+JobCategory        = "short"
queue
```

Exercise

1. Try submitting a simple test job that prints “I love quarks !” by using the cluster.
 - Make sure to use the short queue and be sure to submit only 1 job.. we don't want to overload the system with our small test.
2. If everything works ok, try to submit the `qqbarcorrelations_status.cpp` script, or the one you have been using so far. Do it with ~1000 events and using the short queue. Think carefully about the directory to put this information. The size of the output files will increase by a lot when you simulate more events (`/data` is a good choice for now, but be careful when you really start simulating a lot of files)
3. Try using the `status_analysis.C` macro on the output files. This can be done interactively if the files are not too large, but should be done in another batch job if the outputs start getting bigger.

Note: be careful

Some tips from my own experience:

- Always submit 1 job first, then scale-up. It won't be the first time that a syntax error makes your 1000 jobs fail.
- In general, 1-100 jobs are ok, more can also be done for more statistics, but it's on the larger side. I would do that only after some checks and other analysis has already been done.
- The number of events to simulate can be picked in such a way that you stay under the short queue limit (4 hours). This is much faster than staying in a longer queue.
- Be careful with bash scripts. Never use * unless it's in a full path. Do checks if your directories and paths actually exist, otherwise exit. Bash scripts have the nice feature that they can go to your home directory and execute commands from there if the path you ask cannot be found/doesn't exist. It can have consequences..

