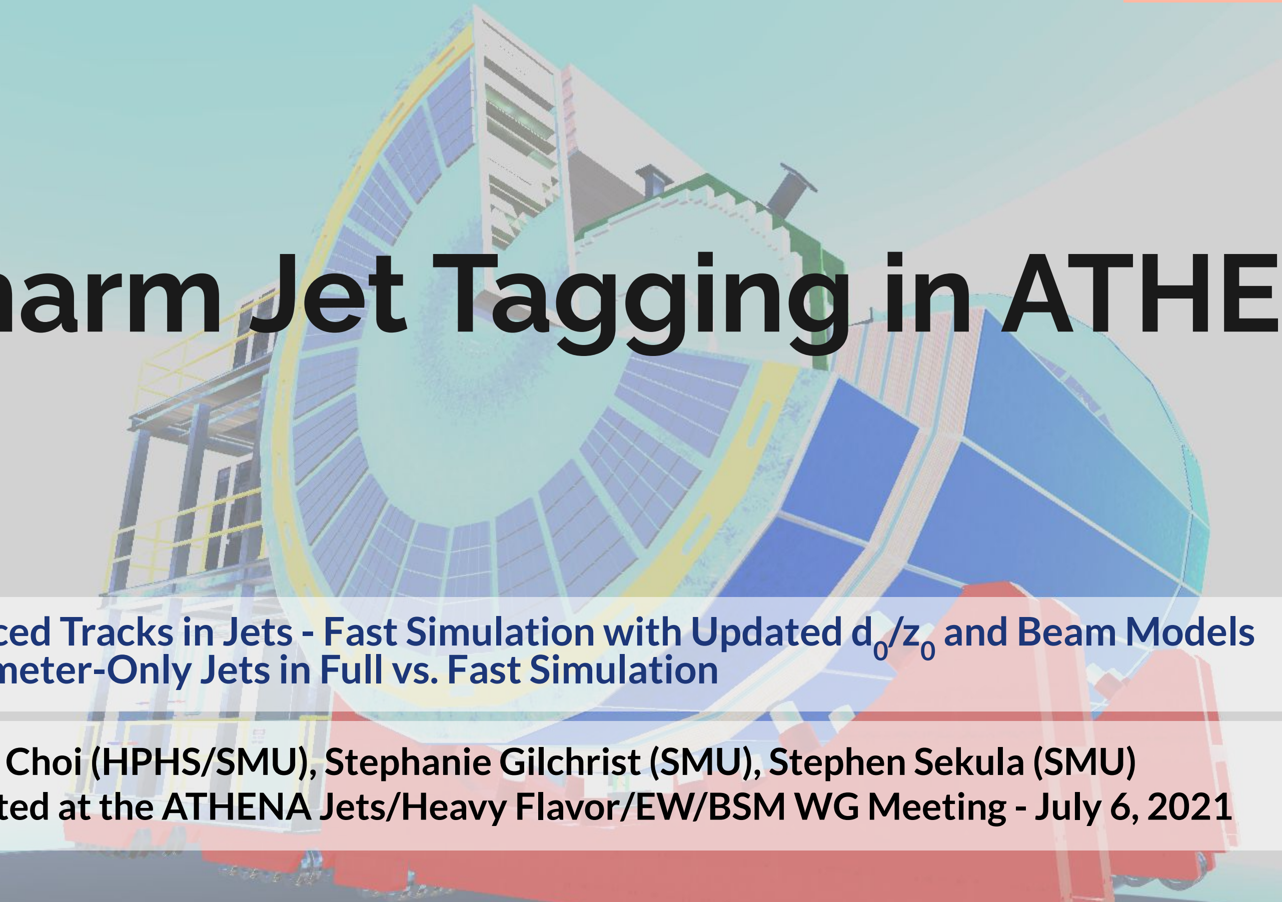

Virtual Reality visuals by Sean Preins

Charm Jet Tagging in ATHENA

Displaced Tracks in Jets - Fast Simulation with Updated d_0/z_0 and Beam Models
Calorimeter-Only Jets in Full vs. Fast Simulation

Justine Choi (HPHS/SMU), Stephanie Gilchrist (SMU), Stephen Sekula (SMU)
Presented at the ATHENA Jets/Heavy Flavor/EW/BSM WG Meeting - July 6, 2021

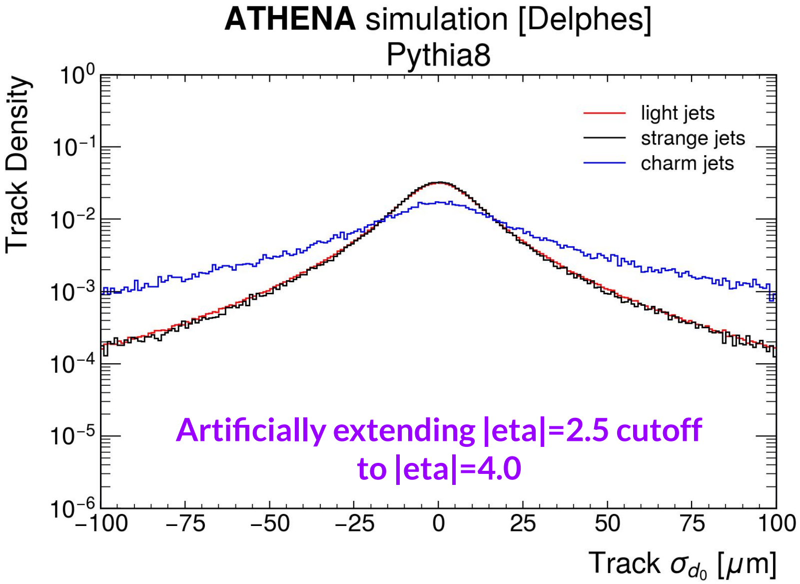
Displaced Tracks in Delphes

- Implementation of full all-silicon d_0/z_0 resolution model
- Implementation of EIC beam model
  - Implications for simplistic displaced track tagging
  - Multivariate displaced track tagging
  - Next steps

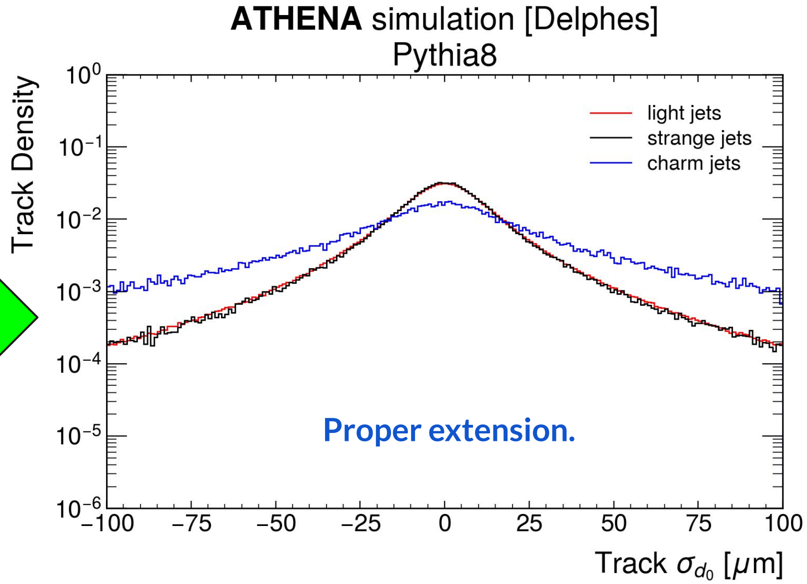
Note: all jets used in this section of the talk are R=1 anti-k_T jets with energy flow, a minimum p_T of 5 GeV/c, and $|\eta|<3.0$ (fast simulation calorimeter extends only to $|\eta|=3.5$)

Thanks to Rey Cruz-Torres: Extended d_0/z_0 Model!

On June 29, Rey provided all-silicon d_0/z_0 resolution numbers to |eta|=4.0. Implemented in delphes_EIC.



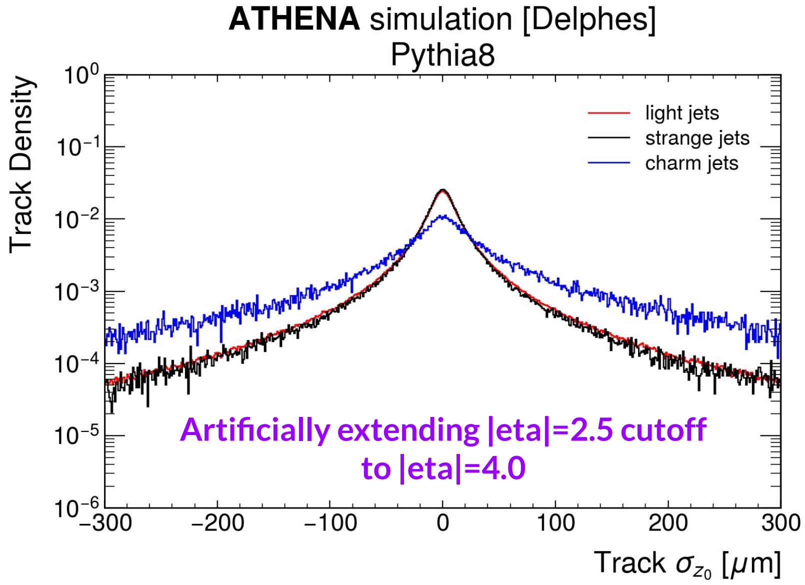
Artificially extending |eta|=2.5 cutoff to |eta|=4.0



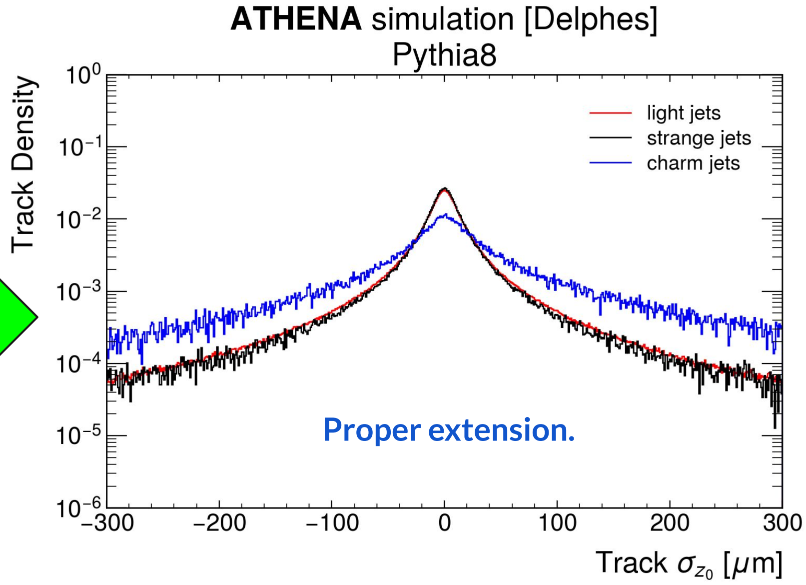
Proper extension.

Thanks to Rey Cruz-Torres: Extended d_0/z_0 Model!

On June 29, Rey provided all-silicon d_0/z0 resolution numbers to |eta|=4.0. Implemented in delphes_EIC.



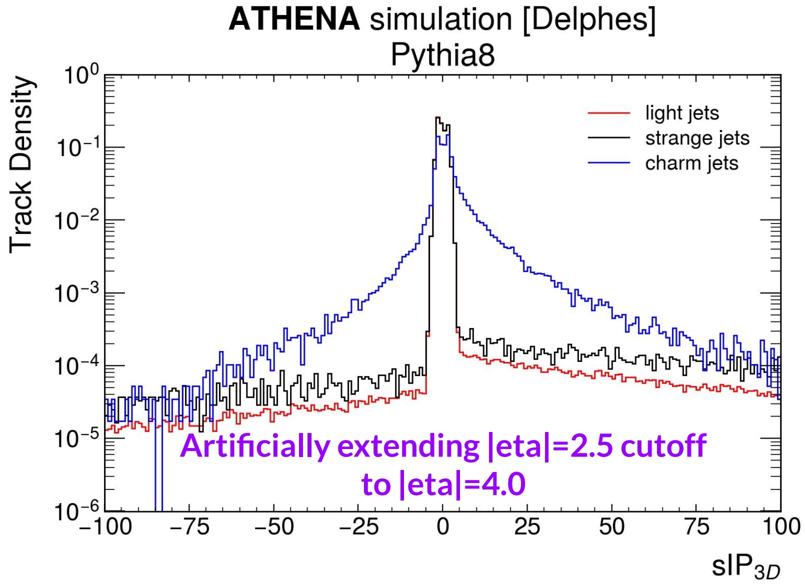
Artificially extending |eta|=2.5 cutoff to |eta|=4.0



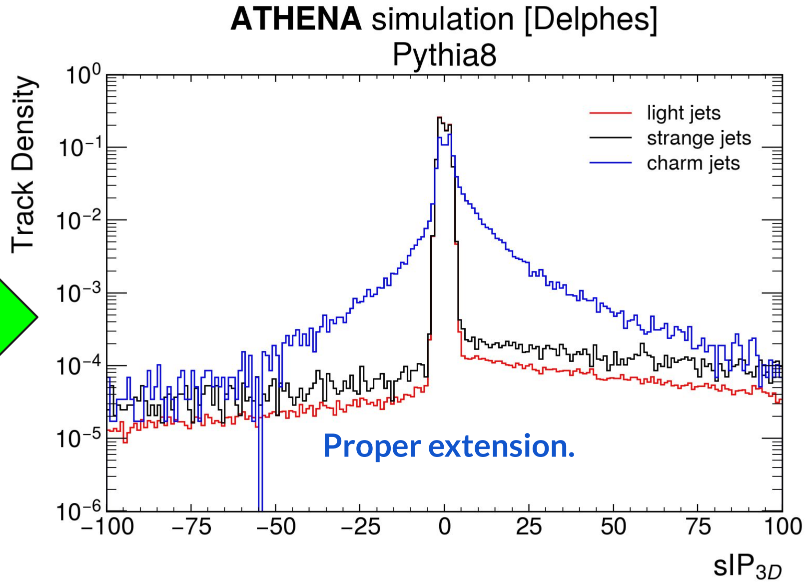
Proper extension.

Thanks to Rey Cruz-Torres: Extended d_0/z_0 Model!

On June 29, Rey provided all-silicon d_0/z_0 resolution numbers to |eta|=4.0. Implemented in delphes_EIC.



Artificially extending |eta|=2.5 cutoff to |eta|=4.0

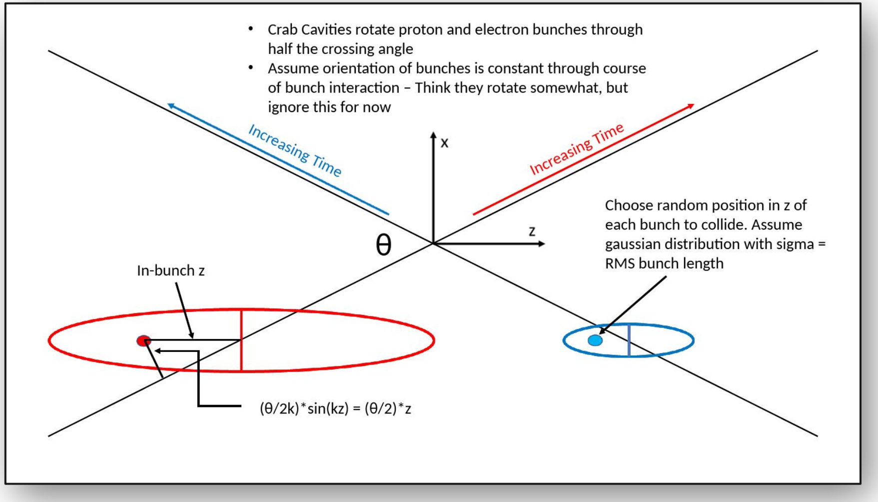


Proper extension.

EIC Beam Implementation

Two-stage process with Delphes:

- **Run eicSimuBeamEffects after modifying `delphes_EIC/pythia8cards/CC_DIS.cmnd` to incorporate the Pythia8-level beam crossing (25mrad)/size effects.**
    - *Generate HepMC2 (Delphes now accepts HepMC3 - no reason to persist in this deprecated format)*
- **Run DelphesHepMC2 to process files from first step.**
    - *Added the BeamSpotFilter module to the `delphes_card_allsilicon_3T.tcl` card to find the actual beamspot, now that it is no longer artificially set to the origin.*



- Crab Cavities rotate proton and electron bunches through half the crossing angle
- Assume orientation of bunches is constant through course of bunch interaction – Think they rotate somewhat, but ignore this for now

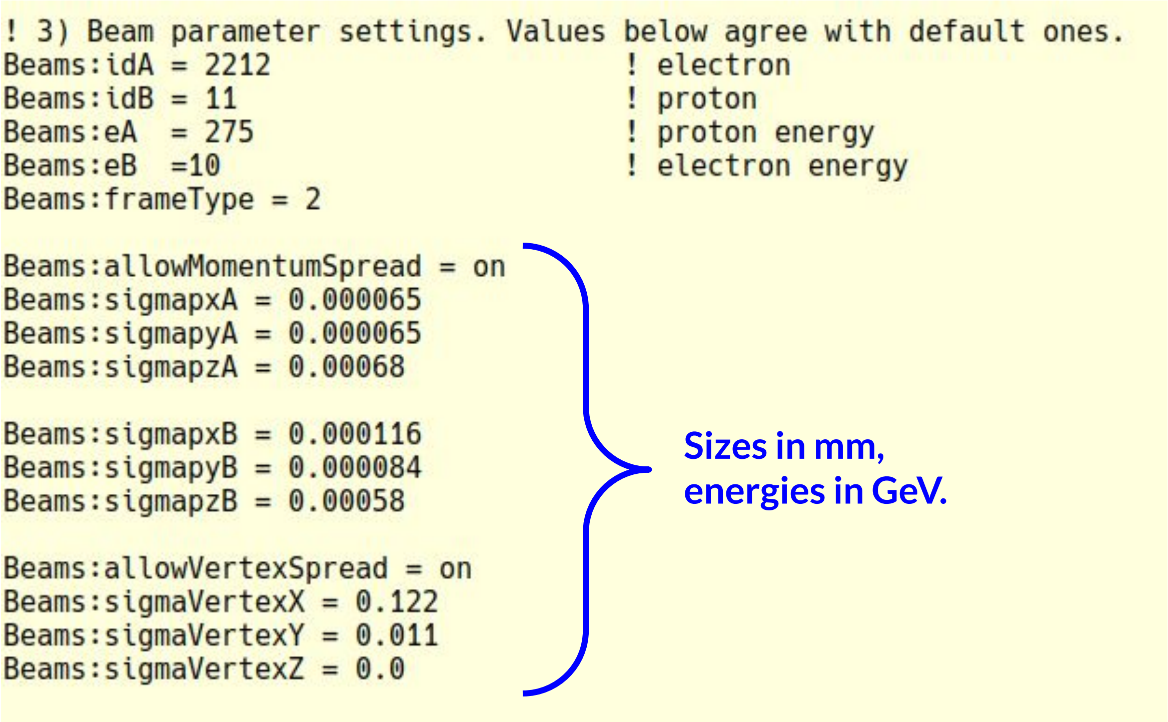
Choose random position in z of each bunch to collide. Assume gaussian distribution with sigma = RMS bunch length

Brian Page's implementation in <u>eicSimuBeamEffects project</u>

Delphes automatically builds the repositioned beamspot into track calculations. <u>No uncertainty on beamspot position assumed.</u>

Beam Parameters

As implemented in Pythia8. An EIC Beam Model class specifically configures and handles the crossing angle aspects of the model, bunch length, etc. This is passed to Pythia8 at compile-time.

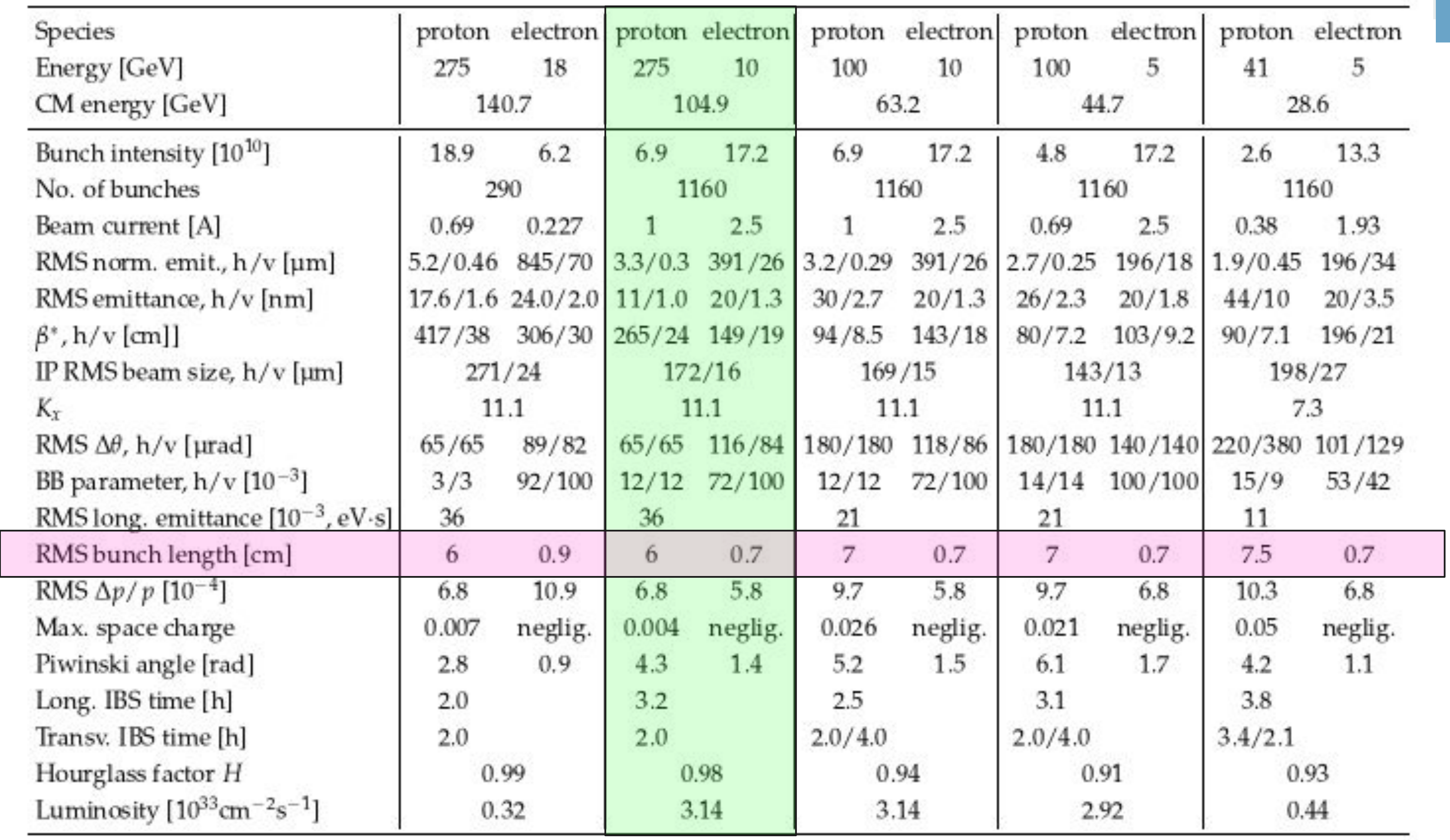


```
! 3) Beam parameter settings. Values below agree with default ones.
Beams:idA = 2212                     ! electron
Beams:idB = 11                       ! proton
Beams:eA  = 275                      ! proton energy
Beams:eB  =10                        ! electron energy
Beams:frameType = 2

Beams:allowMomentumSpread = on
Beams:sigmapxA = 0.000065
Beams:sigmapyA = 0.000065
Beams:sigmapzA = 0.00068

Beams:sigmapxB = 0.000116
Beams:sigmapyB = 0.000084
Beams:sigmapzB = 0.00058

Beams:allowVertexSpread = on
Beams:sigmaVertexX = 0.122
Beams:sigmaVertexY = 0.011
Beams:sigmaVertexZ = 0.0
```

Sizes in mm, energies in GeV.

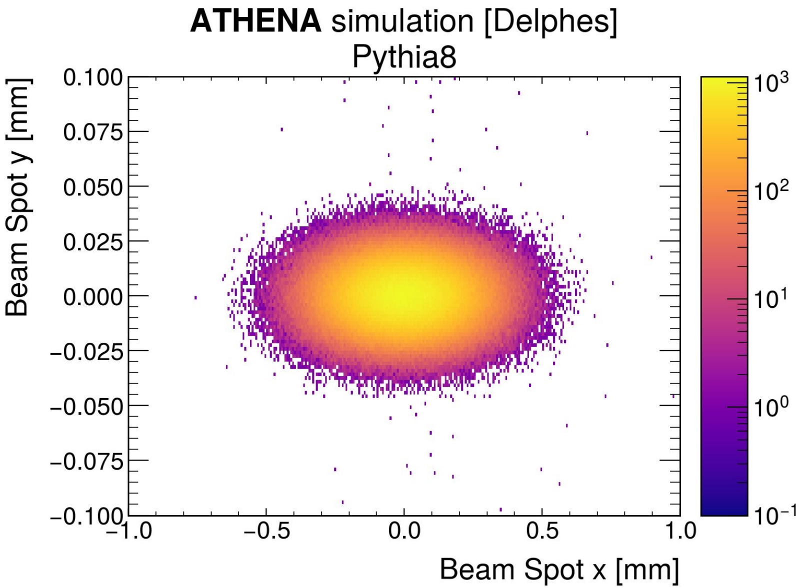
Table from EIC CDR

Table 3.4: EIC beam parameters for different center-of-mass energies $\sqrt{s}$, with strong hadron cooling. High acceptance configuration.

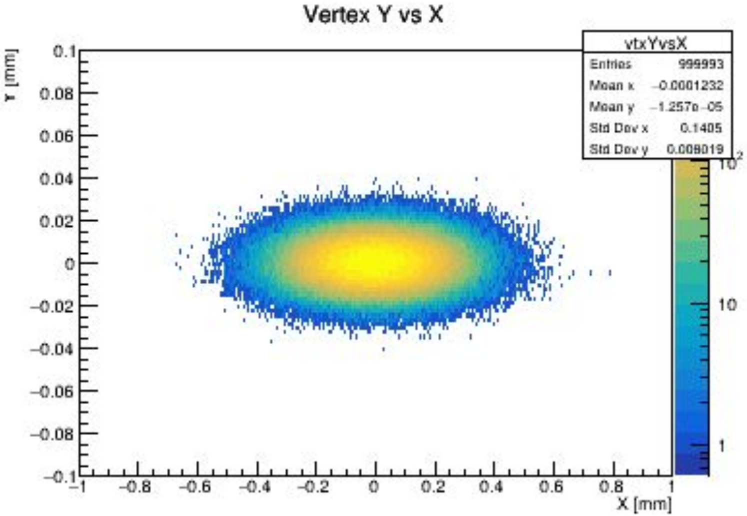


Species	proton	electron	proton	electron	proton	electron	proton	electron	proton	electron
Energy [GeV]	275	18	275	10	100	10	100	5	41	5
CM energy [GeV]	140.7		104.9		63.2		44.7		28.6	
Bunch intensity $[10^{10}]$	18.9	6.2	6.9	17.2	6.9	17.2	4.8	17.2	2.6	13.3
No. of bunches	290		1160		1160		1160		1160	
Beam current [A]	0.69	0.227	1	2.5	1	2.5	0.69	2.5	0.38	1.93
RMS norm. emit., h/v [μm]	5.2/0.46	845/70	3.3/0.3	391/26	3.2/0.29	391/26	2.7/0.25	196/18	1.9/0.45	196/34
RMS emittance, h/v [nm]	17.6/1.6	24.0/2.0	11/1.0	20/1.3	30/2.7	20/1.3	26/2.3	20/1.8	44/10	20/3.5
β^*, h/v [cm]]	417/38	306/30	265/24	149/19	94/8.5	143/18	80/7.2	103/9.2	90/7.1	196/21
IP RMS beam size, h/v [μm]	271/24		172/16		169/15		143/13		198/27	
K_x	11.1		11.1		11.1		11.1		7.3	
RMS $\Delta\theta$, h/v [μrad]	65/65	89/82	65/65	116/84	180/180	118/86	180/180	140/140	220/380	101/129
BB parameter, h/v $[10^{-3}]$	3/3	92/100	12/12	72/100	12/12	72/100	14/14	100/100	15/9	53/42
RMS long. emittance $[10^{-3}$, eV·s]	36		36		21		21		11	
RMS bunch length [cm]	6	0.9	6	0.7	7	0.7	7	0.7	7.5	0.7
RMS $\Delta p/p$ $[10^{-4}]$	6.8	10.9	6.8	5.8	9.7	5.8	9.7	6.8	10.3	6.8
Max. space charge	0.007	neglig.	0.004	neglig.	0.026	neglig.	0.021	neglig.	0.05	neglig.
Piwinski angle [rad]	2.8	0.9	4.3	1.4	5.2	1.5	6.1	1.7	4.2	1.1
Long. IBS time [h]	2.0		3.2		2.5		3.1		3.8	
Transv. IBS time [h]	2.0		2.0		2.0/4.0		2.0/4.0		3.4/2.1	
Hourglass factor H	0.99		0.98		0.94		0.91		0.93	
Luminosity $[10^{33}\text{cm}^{-2}\text{s}^{-1}]$	0.32		3.14		3.14		2.92		0.44	

Beam Spot Position (y vs. x)



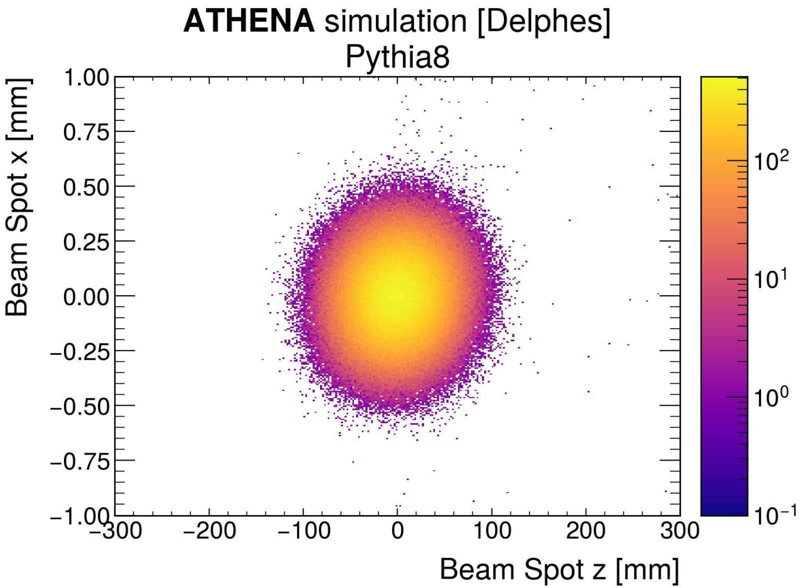
10x275



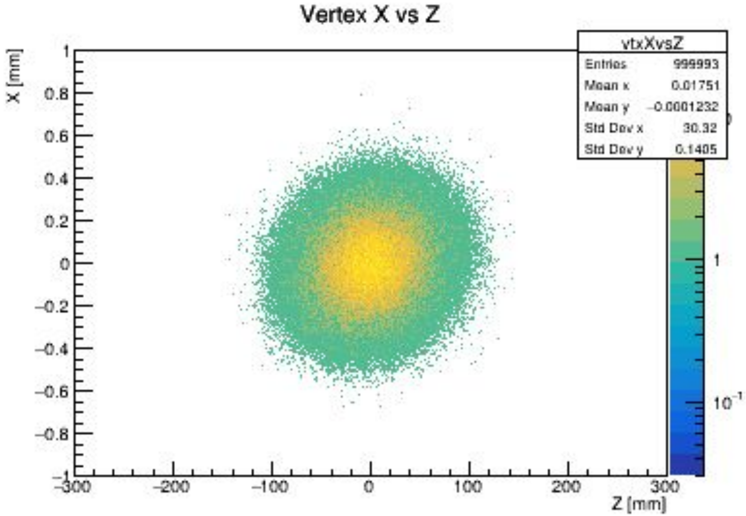
Plot from Brian’s talk on 6/1/21.

18x275

Beam Spot Position (x vs. z)

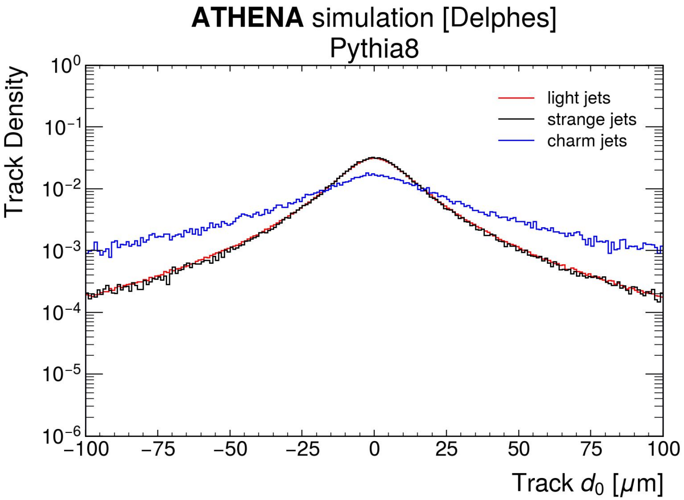


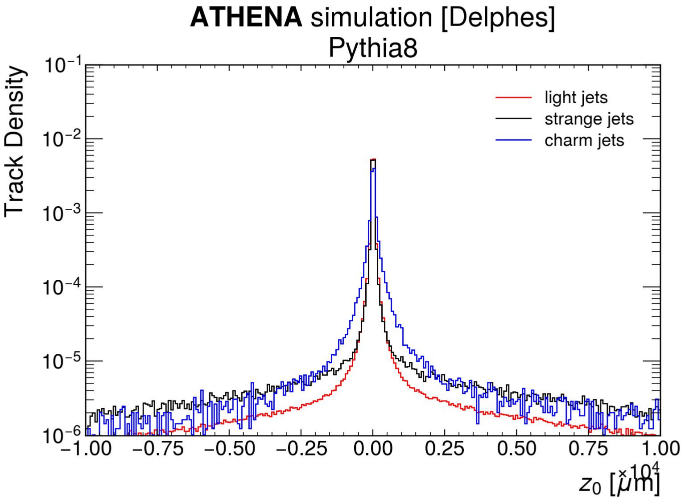
10x275



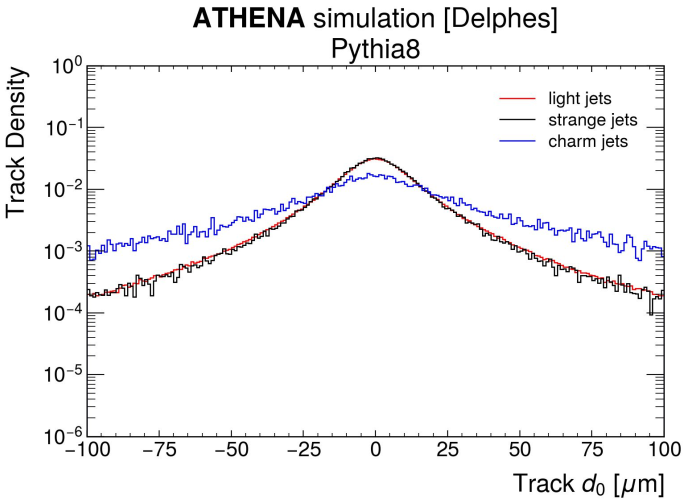
vtxXvsZ	
Entries	999993
Mean x	0.01751
Mean y	−0.0001232
Std Dev x	30.32
Std Dev y	0.1405

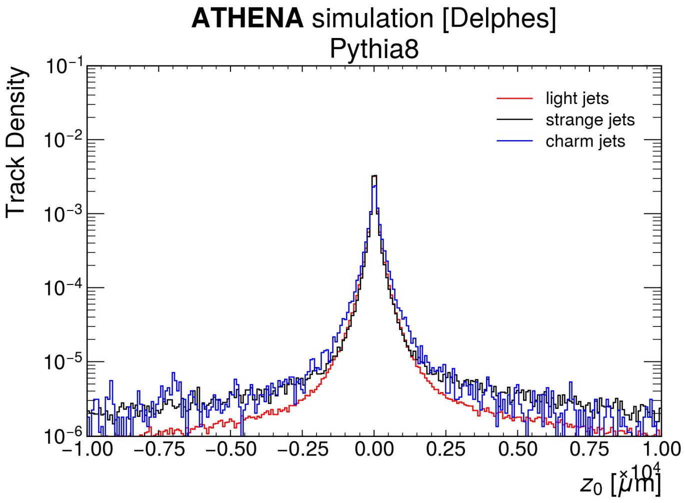
Plot from Brian’s talk on 6/1/21.
18x275

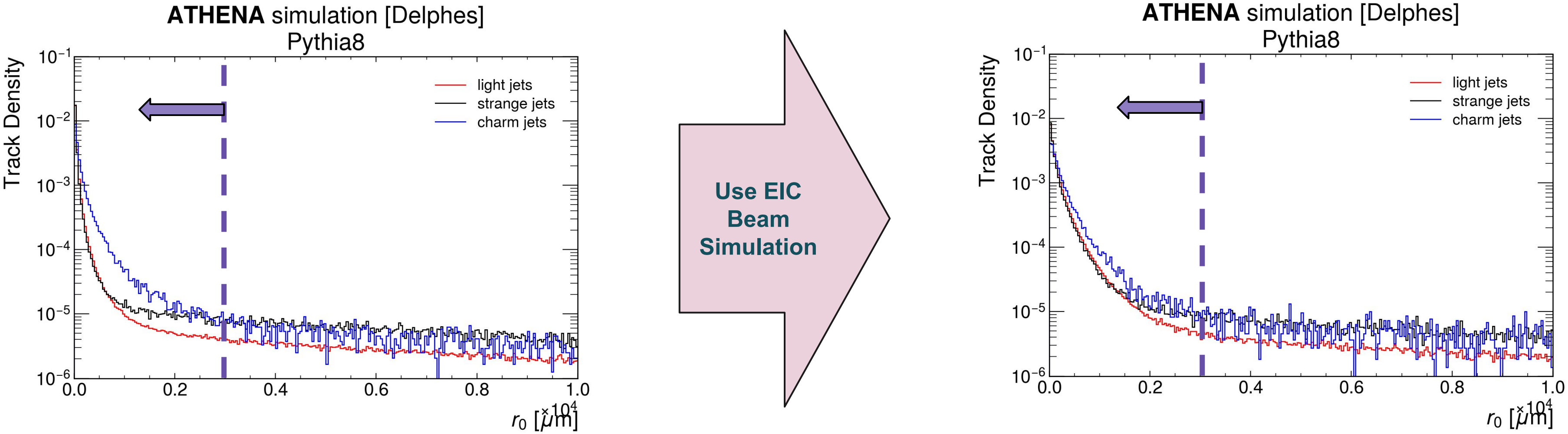

ATHENA simulation [Delphes]
Pythia8


ATHENA simulation [Delphes]
Pythia8

Use EIC Beam Simulation


ATHENA simulation [Delphes]
Pythia8


ATHENA simulation [Delphes]
Pythia8



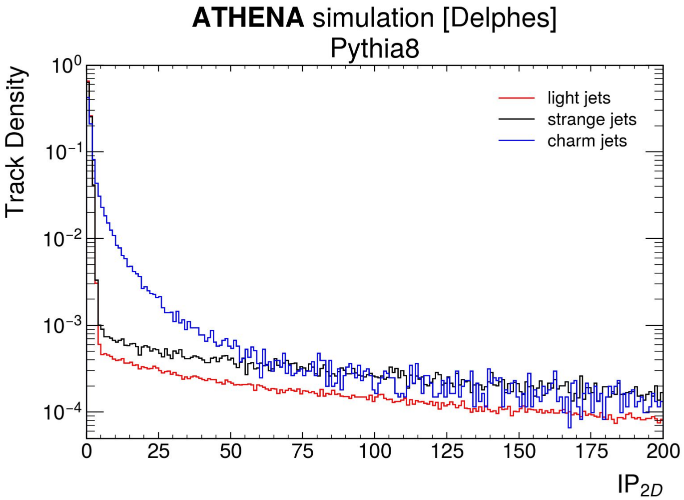
Use EIC Beam Simulation

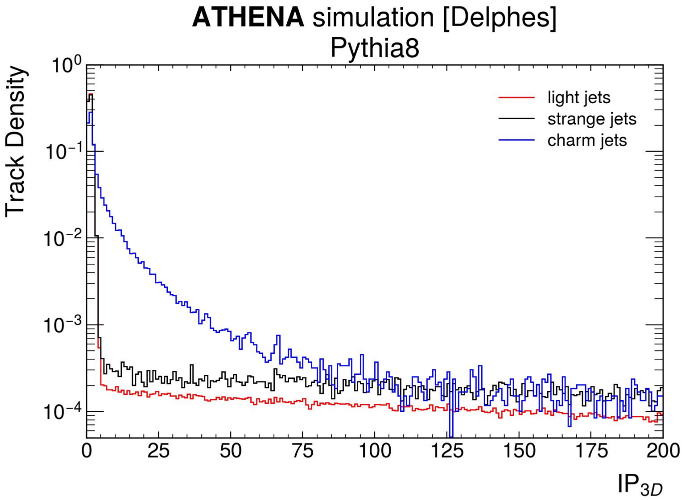
The purple dotted line and arrow indicate the maximum value of r_0 ($=\sqrt{[z_0^2 + d_0^2]}$), **3mm**, that we used in the YR era to eliminate overly long-lived particles like K_s and Λ compared to charm hadrons. This was a “cheap” way to do this without vertexing.

We see so far that with the EIC beam model in place:

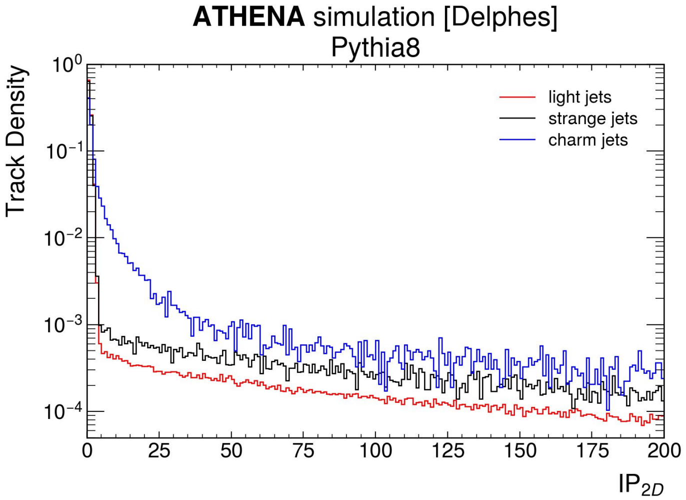
- d_0 is relatively robust, which makes some sense: the beam spot spread in *x* and *y* is not huge (at the ~10s of micron level compared to hadron flight lengths of ~100s of microns or more)
- z_0 is not robust and is greatly diluted by the large spread of the beam spot in z. Again, this is sensible given that this spread (~10,000s of microns) exceeds typical hadron flight lengths.

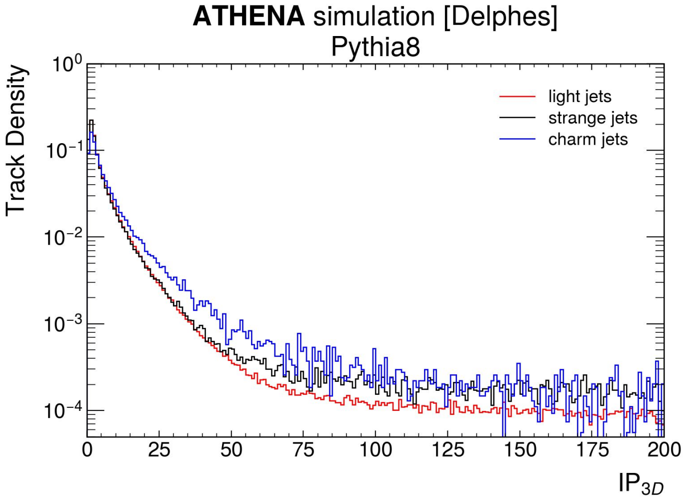
We should expect IP_{2D} to be robust but IP_{3D} to be degraded (and thus also for sIP_{3D} to be degraded).

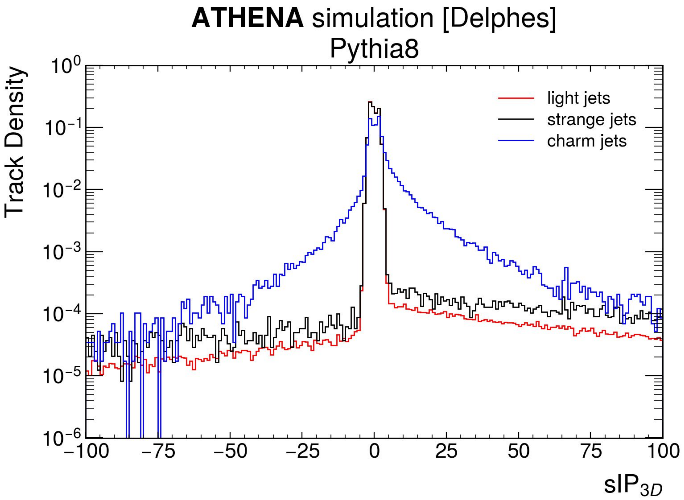




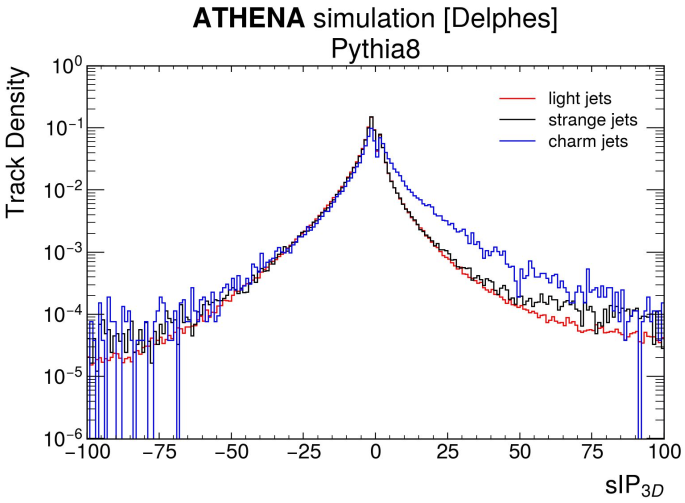
Use EIC Beam Simulation







Use EIC Beam Simulation



- The original **sIP3dTagger** approach is now **too naive** and cannot be expected to be performant.
  - *Indeed, this is what we observe. Leaving the tagger configuration unchanged (see box right), we maintain 20% charm jet efficiency but increase light-jet efficiency from 0.5% -> 10%!!!*
- Use approach more aligned with LHC methodologies:
  - Combine the IPxD variables (IP2D and sIP3D) using a multivariate approach.
  - Specific topology: Use the leading 4 tracks in the jet and classify jets using the pair of variables (IP2D, sIP3D) for each of the 4 tracks (8 total inputs)
  - Alternatively, we could use a likelihood approach, but a Multi-Layer Perceptron NN is just as easily implemented in the code and balances between too simple and too complex.

sIP3DTagger

At least 2 tracks in a jet with …

- $p^{trk}_T > 0.5$ *GeV/c*
- $sIP_{3D} > 3$
- $r_0 < 3$*mm*

CharmIPXDTagger

Training and Validation Approach

I generated a **dedicated Pythia8/Delphes sample for training**. This sample will never be used for performance assessment.

Training/Testing Approach:

- Use 10,000 (100,000) charm (light) jets for training
- Use an equal-sized sample for testing
- Use ReLU for the activation function and transform the inputs so that they are in consistent ranges (e.g. [0,1])
- Architecture: 8:24:1 (input:hidden:output)
- Use 1000 epochs for training

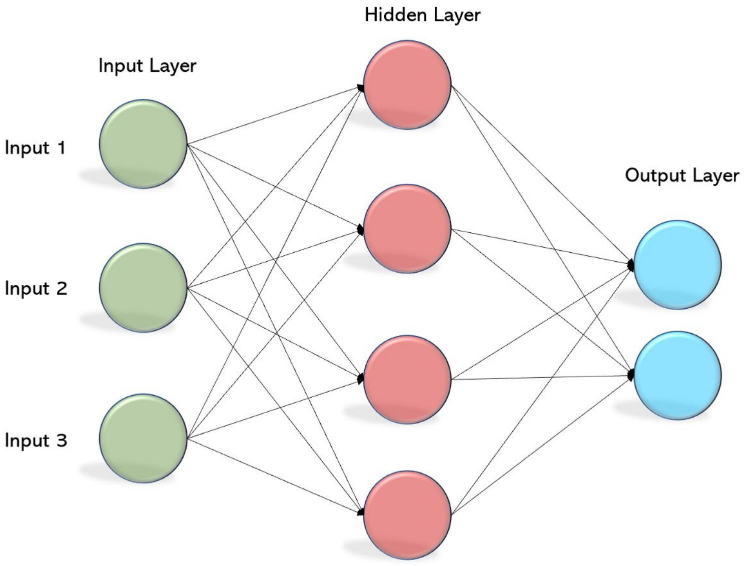
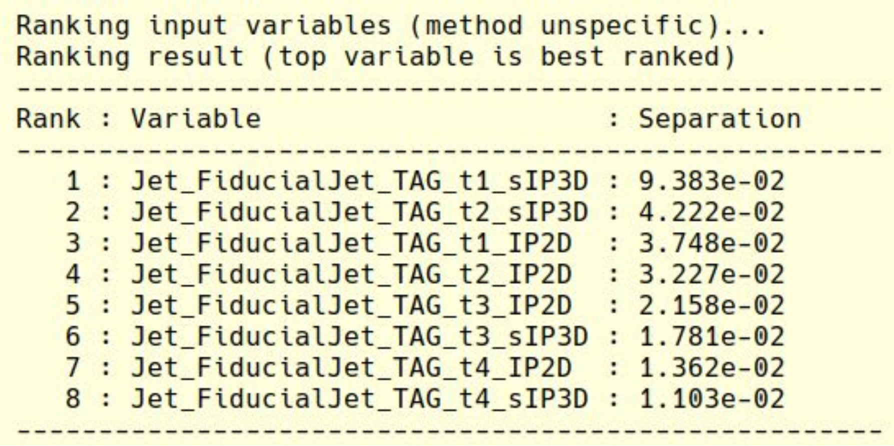
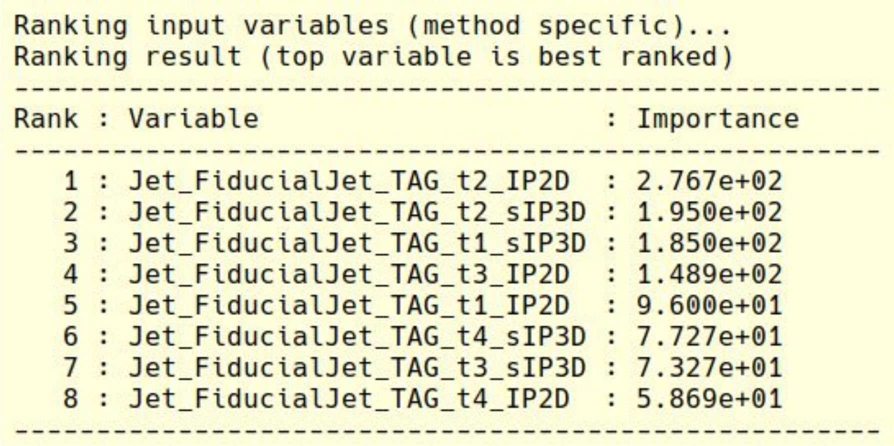


Illustration of basic MLP architecture

Variable Ranking - Before and After Training



```
Ranking input variables (method unspecific)...
Ranking result (top variable is best ranked)
----------------------------------------------------
Rank : Variable                    : Separation
----------------------------------------------------
   1 : Jet_FiducialJet_TAG_t1_sIP3D : 9.383e-02
   2 : Jet_FiducialJet_TAG_t2_sIP3D : 4.222e-02
   3 : Jet_FiducialJet_TAG_t1_IP2D  : 3.748e-02
   4 : Jet_FiducialJet_TAG_t2_IP2D  : 3.227e-02
   5 : Jet_FiducialJet_TAG_t3_IP2D  : 2.158e-02
   6 : Jet_FiducialJet_TAG_t3_sIP3D : 1.781e-02
   7 : Jet_FiducialJet_TAG_t4_IP2D  : 1.362e-02
   8 : Jet_FiducialJet_TAG_t4_sIP3D : 1.103e-02
----------------------------------------------------
```



```
Ranking input variables (method specific)...
Ranking result (top variable is best ranked)
----------------------------------------------------
Rank : Variable                    : Importance
----------------------------------------------------
   1 : Jet_FiducialJet_TAG_t2_IP2D  : 2.767e+02
   2 : Jet_FiducialJet_TAG_t2_sIP3D : 1.950e+02
   3 : Jet_FiducialJet_TAG_t1_sIP3D : 1.850e+02
   4 : Jet_FiducialJet_TAG_t3_IP2D  : 1.489e+02
   5 : Jet_FiducialJet_TAG_t1_IP2D  : 9.600e+01
   6 : Jet_FiducialJet_TAG_t4_sIP3D : 7.727e+01
   7 : Jet_FiducialJet_TAG_t3_sIP3D : 7.327e+01
   8 : Jet_FiducialJet_TAG_t4_IP2D  : 5.869e+01
----------------------------------------------------
```

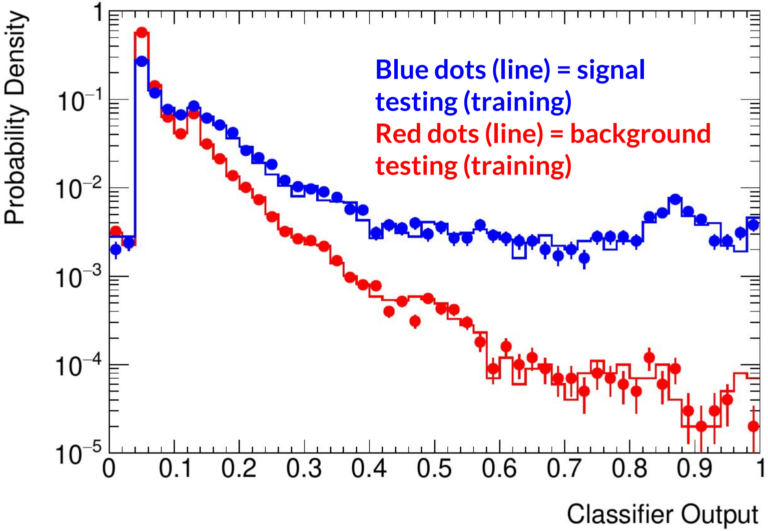
Estimated using separation: $\langle S^2\rangle = \frac{1}{2}\int \frac{(\hat{y}_S(y)-\hat{y}_B(y))^2}{\hat{y}_S(y)+\hat{y}_B(y)}dy$

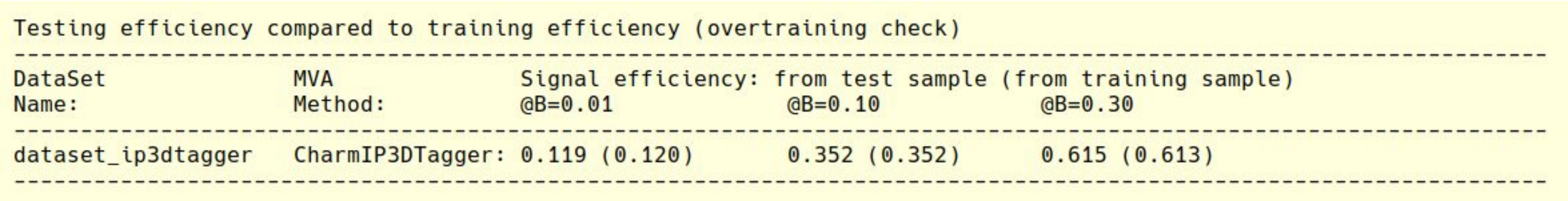
After training, rank using importance (sum of the weights-squared of the connections to nodes in the first hidden layer) → the MLP ranks IP2D as more than or as highly valuable as IP3D, as we would expect!

Overtraining Assessment

No strong evidence of overtraining, though this can certainly be improved (likely too many epochs and/or need more jets in training and testing samples).

- *K-S Test of compatibility of training/testing shapes in signal (background) yield 0.044 (0.83).*



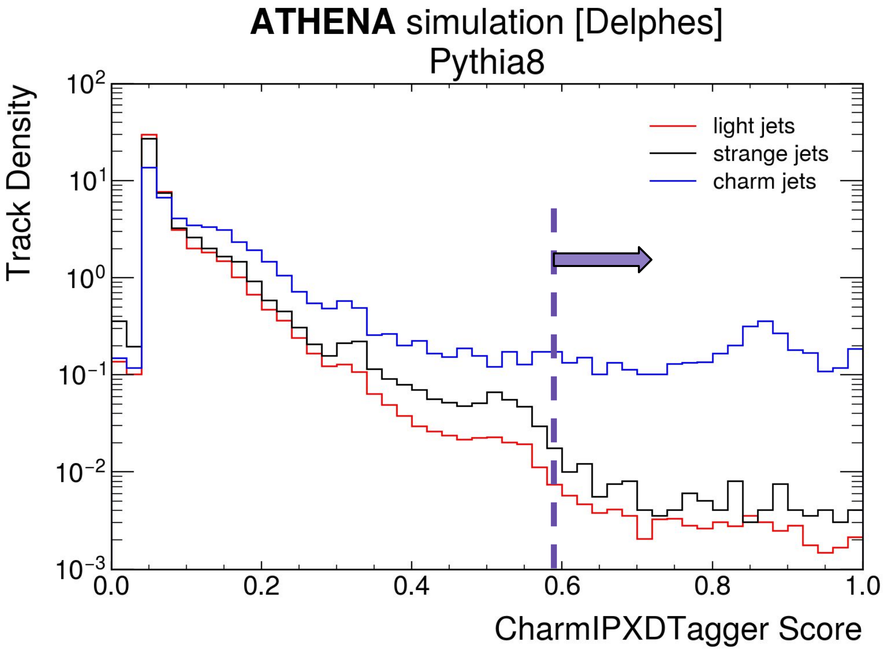


```
Testing efficiency compared to training efficiency (overtraining check)
-------------------------------------------------------------------------------------------------------------
DataSet              MVA              Signal efficiency: from test sample (from training sample)
Name:                Method:          @B=0.01              @B=0.10              @B=0.30
-------------------------------------------------------------------------------------------------------------
dataset_ip3dtagger   CharmIP3DTagger: 0.119 (0.120)        0.352 (0.352)        0.615 (0.613)
-------------------------------------------------------------------------------------------------------------
```

Performance Assessment

Optimize cut on CharmIPXDTagger by minimizing the expected light-jet background subtraction error in a target luminosity of 100fb^{-1}.

- Best selection: **CharmIPXDTagger > 0.58**
- **Charm (*Light+Strange*) Efficiency: 6.8% (*0.14%*)**
  - Expect about 2000 tagged charm jets assuming CT18NNLO proton PDF and using this approach alone (*expect also a comparable number of light+strange jets passing tagging*)



The above figure is an evaluation of the tagger on a sample <u>independent</u> of the training and testing samples.

Next Steps in Jet Tagging

- Beam crossing and shape effects have a significant impact on jet tagging, as expected.
  - Naive approach using single sIP_{3D} variable no longer viable - switch to multivariate approach using 4 leading track IP_{2D} and sIP_{3D} values.
- Next steps:
  - Revisit K-Tagger and e-Tagger in light of beam shape implementation.
  - Vertexing available in Delphes -> move on to look at secondary vertex information for jet flavor tagging (e.g. mass, number of tracks, displacement significance relative to interaction point, etc.)
- ***Translate tagging yield into impact on charm jet population from 1 year of EIC collisions -> impact on intrinsic strangeness assessment.*** *(work in collaboration with Fred Olness at SMU)*

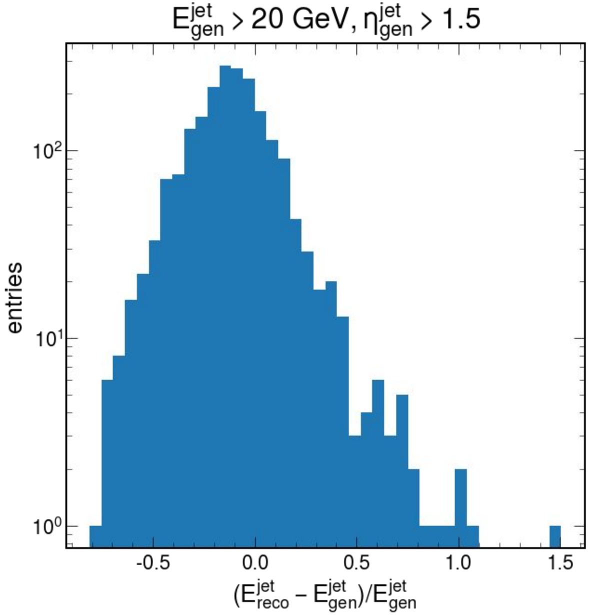
Fast vs. Full Simulation - Calorimeter-Only Jets

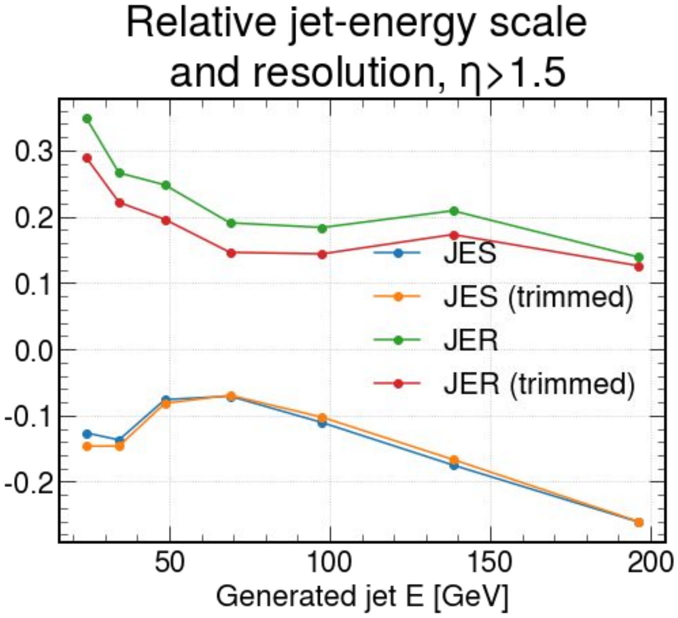
NOTE: fast simulation jets in this section are matched to generator-level (particle-level) jets using a ΔR matching and the requirement that $\Delta R_{max} < 0.5$ and no generator-level jet is used more than once for a match.

Full Simulation

Miguel posted on [Slack chat]() some studies of calorimeter-only jets (R=1) in full simulation using a preliminary calibration of clusters.

He asked me to do the same in Delphes, looking at Calorimeter-Only Jets (jets built from calorimeter towers with no energy flow applied).



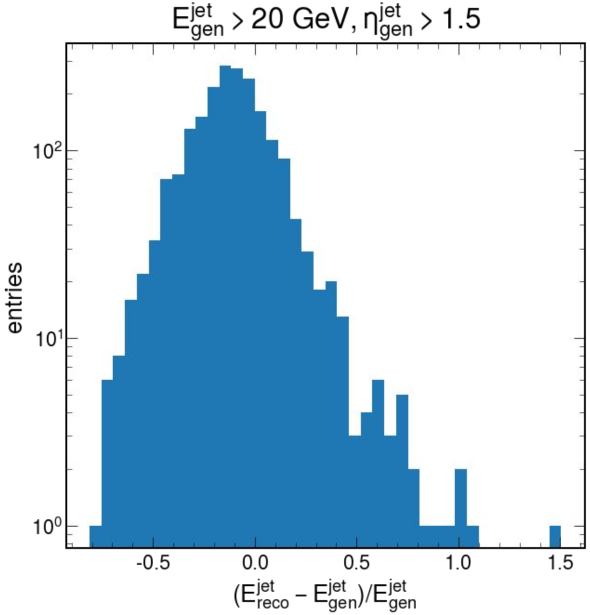


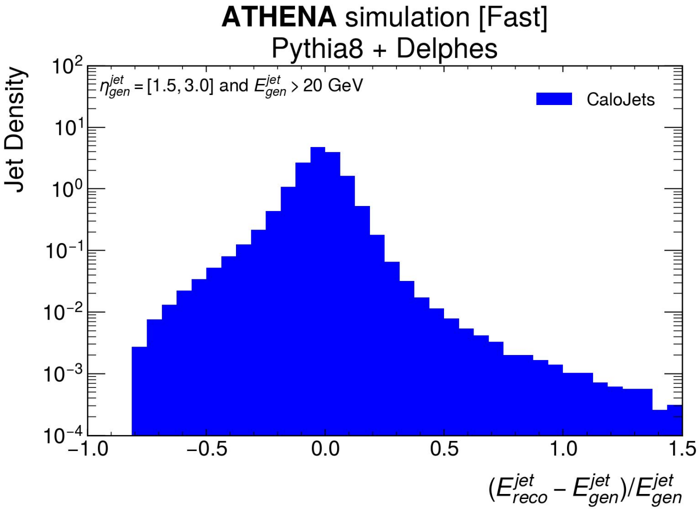
Fast (1M events) Compared to Full Simulation

The JES and JER are determined from the mean and width, respectively, of the quantity

(REC - GEN)/GEN

This quantity is plotted for jet energy on the right in fast simulation., compared to full simulation on the left.

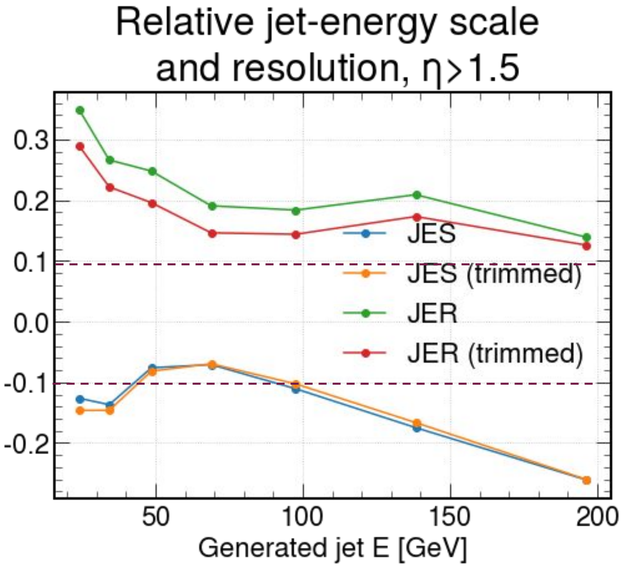


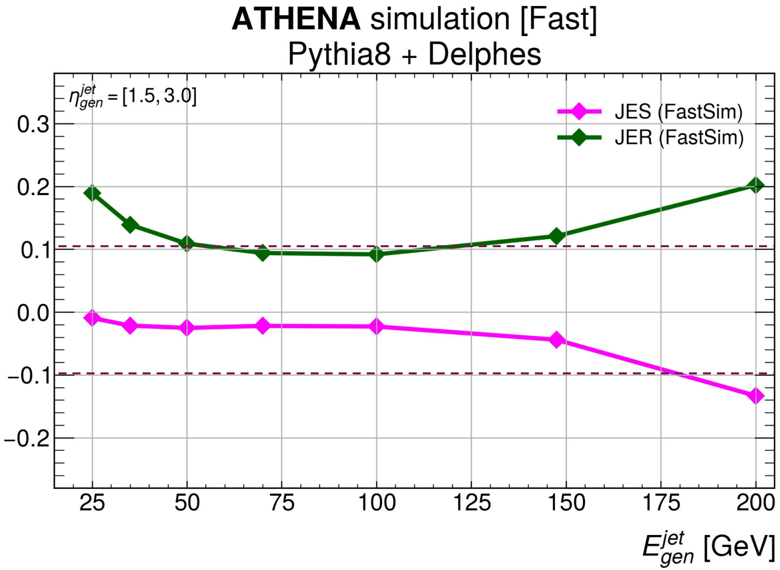


Fast (1M events) Compared to Full Simulation

In Delphes, the JES is closer to zero and the JER closer to 10% - in both cases better than in full simulation.

NOTE: In Delphes, the calorimeters extend only to |η|=3.5, whereas in full simulation they go to 4.0. All jets are R=1 jets.


Relative jet-energy scale and resolution, η>1.5


ATHENA simulation [Fast]
Pythia8 + Delphes

±10% lines indicated on vertical axes for comparison. Calorimeter model in fast simulation comes from YR, with ECAL (HCAL) constant resolution terms of 2% (10%) in the forward direction.

APPENDIX

Basic Ideas

Charm quark jets (e.g. produced from CC DIS s→c reactions) contain long-lived heavy charm hadrons that produce displaced substructures (vertices) within the jet. This can “tag” the jet as heavy flavor.

Current efforts focused on:

- *Displaced track counting*
- *Particle ID of Kaons (fast simulation) and Electrons (Full-Simulation Calorimeter-based approaches)*
  - *c→e + X ≅ 13% (inclusive)*
  - *c→K(OS*) + X ≅ 35% (inclusive)*
  - *c→K(SS*) + X ≅ 5% (inclusive)*

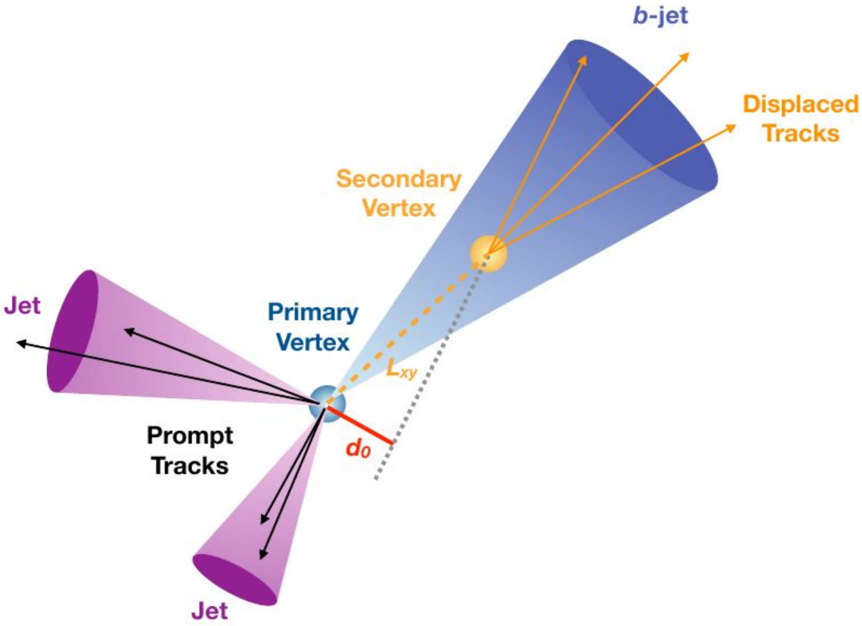


Image from ATLAS Experiment (arXiv:2106.03584). Depicted for pp collisions and b-jets, but applies equally well to charm jets at ep collider.

[*] OS = Opposite-Sign, SS = Same-Sign. Here, “sign” refers to the sign of the (anti-)charm quark charge.

Software Framework

- PYTHIA8(.305) for CC DIS collisions (ep at 10 on 275 GeV) -> 20 million collisions
- DD4hep (athena, ip6, etc.) for full simulation; single-particle events for now, only calorimetry.
- DELPHES for fast simulation of final-state particles/smearing for detector effects (tracks, neutrals, jets, particle flow)
- delphes_EIC for ATHENA-like detector configuration, including tracking, PID, ECAL, and HCAL.
  - All-silicon tracker model (momentum smearing and resolution), 3T magnetic field.
- OLeAA for analysis of DELPHES files
  - A simple analysis framework I sketched up last year (OLeAA = Own Little e-A Analysis)
  - Analyze small output files in Jupyter/UPROOT