FACULTY OF PHYSICS
UNIVERSITY OF WARSAW

Status of the long-lived particles reconstruction study at the ILD

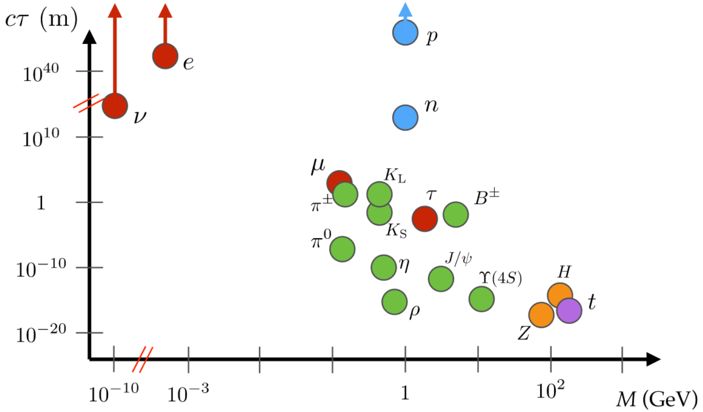
Jan Klamka, A.F. Żarnecki
University of Warsaw

Long-lived particles

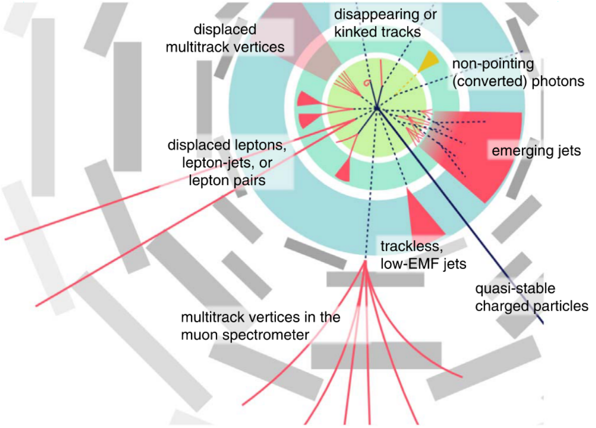
- Many particles with macroscopic lifetimes **already in the SM**
- Various BSM models also predict LLPs, e.g. SUSY, ALPs
- Long lifetimes lead to very **exotic signatures**: displaced vertices, tracks/photons not pointing to IP
- Multiple searches at the LHC (see e.g. 1903.04497)
- LHC sensitive to high masses and couplings

→ e+e- competitive in the complementary region: small masses, couplings and mass splittings

- ILD promising with the TPC, but no dedicated analysis yet



From: https://indico.cern.ch/event/607314/contributions/2542308/



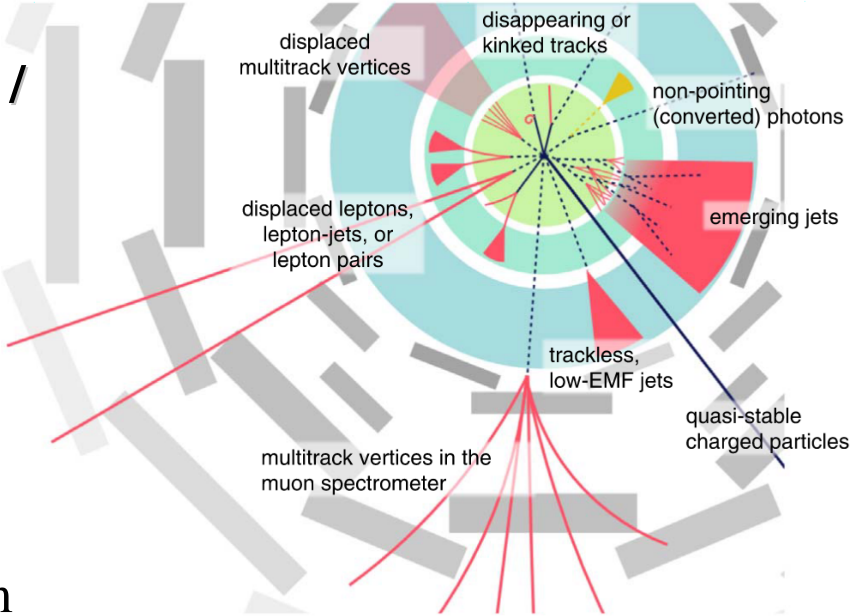
LHC signatures: https://indico.cern.ch/event/607314/contributions/2542309

Long-lived particle signatures

What do we need for tracks that do not **originate from** / **point to** the IP:

- Efficient hit/segment finding
- Good track reconstruction
- Secondary vertex finding
- Particle ID

All can be challenging for tracks with small momentum



Long-lived particles

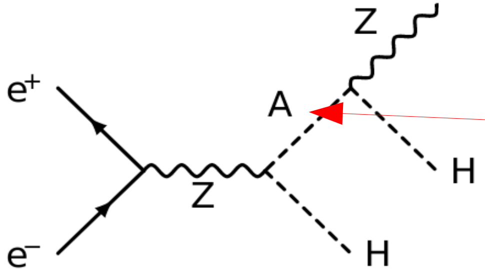
This study:

- **Track reconstruction**
- Events with **displaced vertices**

We first considered reconstruction of SM LLPs (w.r.t. results by U. Einhaus for V0 Finder)

As a more challenging case (much smaller boost) we considered:

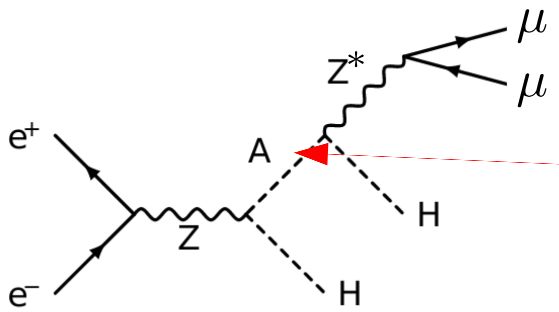
→ (unphysical) Inert Doublet Model sample with small mass splitting, $Z^* \to \mu\mu$



Long-lived, with $c\tau = 1\,\mathrm{m}$

$$m_A - m_H = 1\,\mathrm{GeV}, 5\,\mathrm{GeV}$$

Simulation of displaced vertices



Long-lived, with $c\tau = 1\,\mathrm{m}$

$$m_A - m_H = 1\,\mathrm{GeV}, 5\,\mathrm{GeV}$$

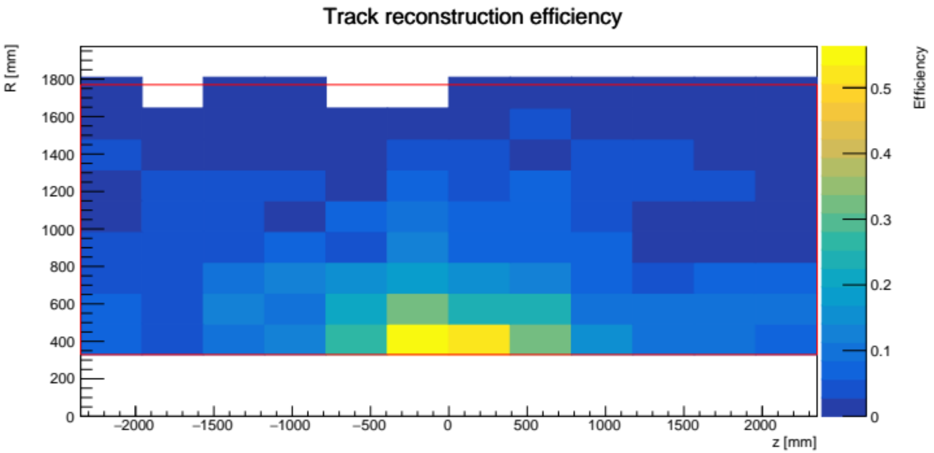
Vertex displacement cannot be simulated in Whizard in a straightforward way

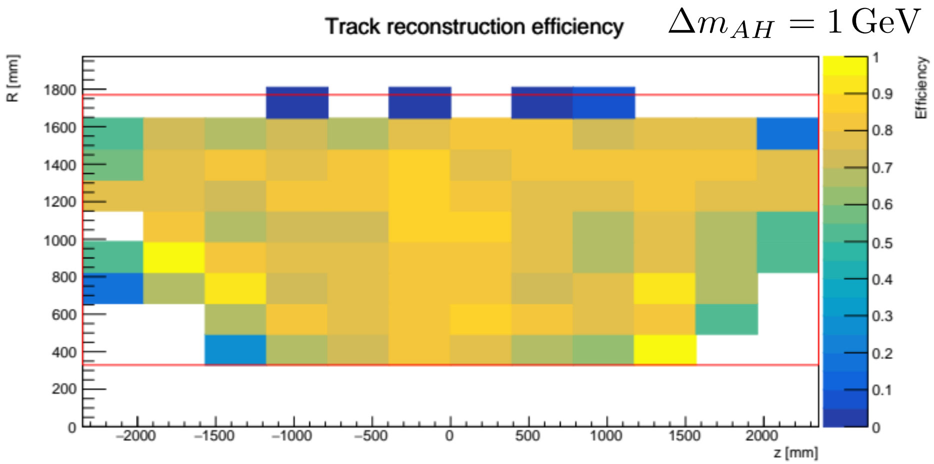
Currently there are two possibilities:

- <u>Define LLP in Whizard as stable and hadronise in Pythia with setting it as non-resonant</u>
- Change the LLP properties in Geant4 Particle Table provided to ddsim

First limitations

Tracking efficiency strongly suppressed by default cuts d0, z0 < 500 mm in the *FullLDCTracking_MarlinTrk* processor → simply remove (or loose) the cut



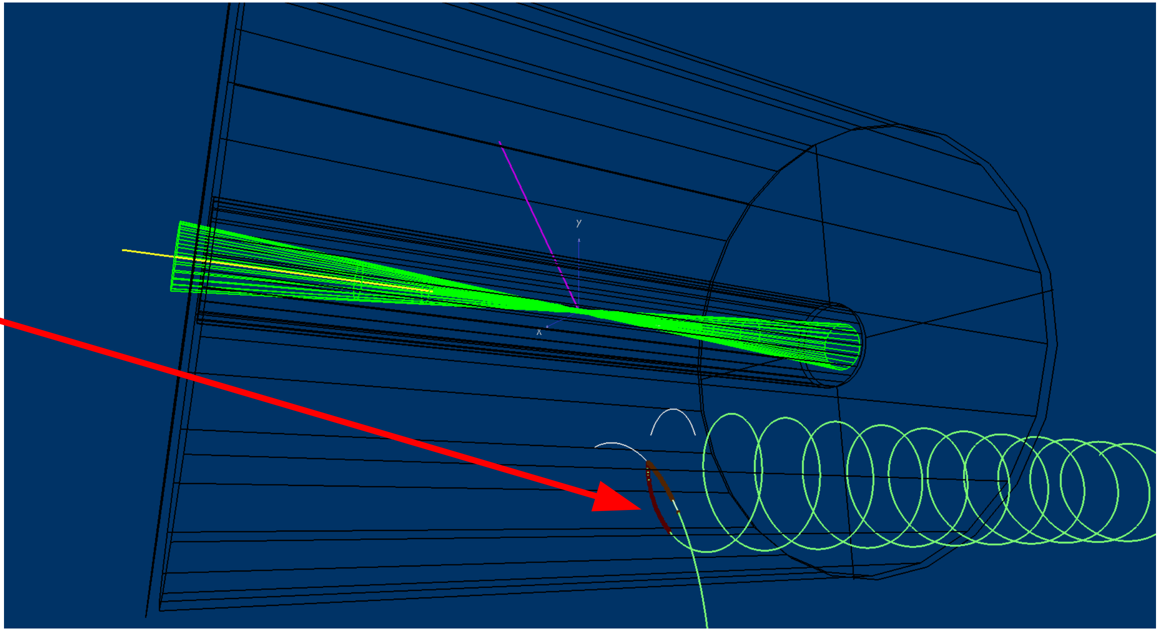


First limitations

Track direction (often even charge) not matching MC truth - as first/last hit often distant from true vertex

Extreme case with inverted direction:

	Px	Py	Pz
MC:	0.113	-0.339	0.061
Reco:	-0.103	0.344	-0.062



- Switch direction in first (last) hit if Pz does not point into Z coordinate of the last (first) hit
- Efficiency improvement by ~10%

Tracking efficiency

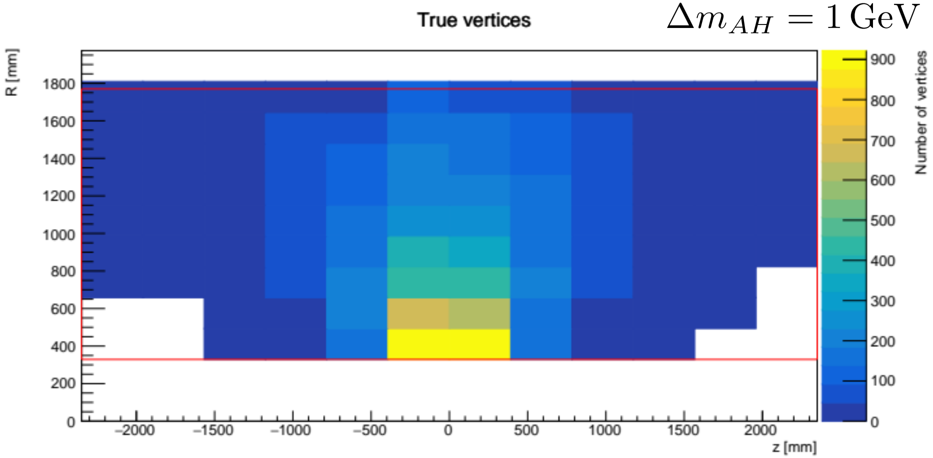
Take only LLP decays inside the TPC

Track state in the first or the last hit (for now take closer to the true vertex)

In matching to MC require:

- Angular separation < 0.3 between true and reco. direction
- Distance < 100 mm between true vertex and first/last hit position
- Good charge sign

$\Delta m_{AH} = 1\,\mathrm{GeV}$



Tracking efficiency

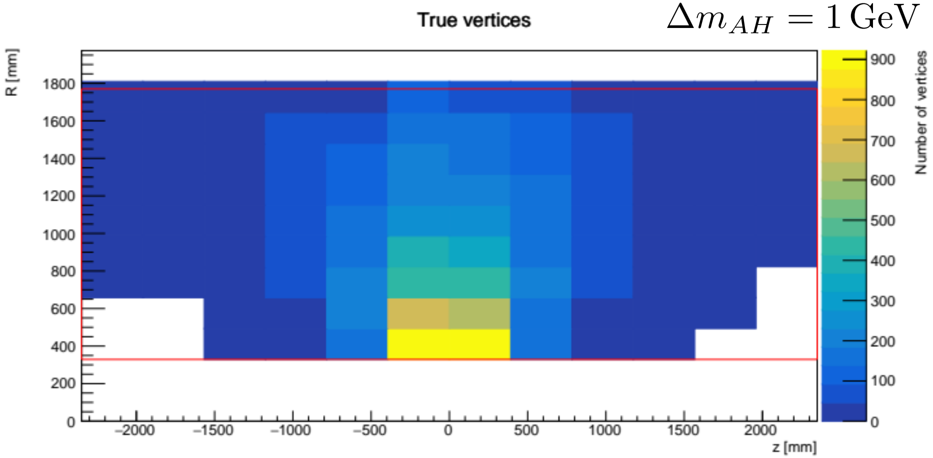
Take only LLP decays inside the TPC

Track state in the first or the last hit (for now take closer to the true vertex)

In matching to MC require:

- Angular separation < 0.3 between true and reco. direction
- Distance < 100 mm between true vertex and first/last hit position
- Good charge sign

$\Delta m_{AH} = 1\,\mathrm{GeV}$



With fixes from slides 6-7, reco. efficiency:

$$\sim 75\%\ (\Delta m_{AH} = 1\,\mathrm{GeV})$$

$$\sim 85\%\ (\Delta m_{AH} = 5\,\mathrm{GeV})$$

Further improvement attempts

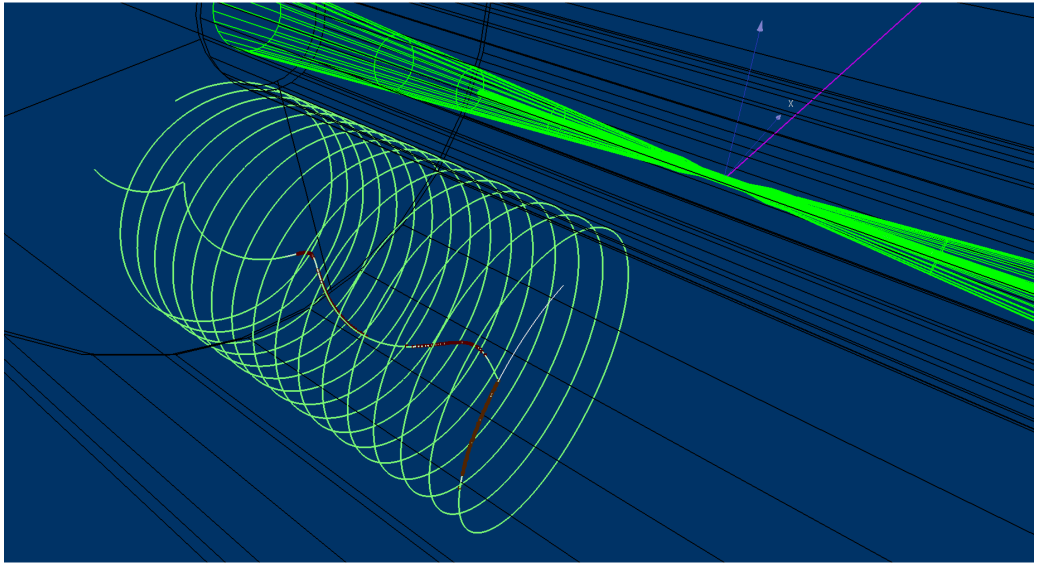
- Standard **fit procedure seems to work** for most events (trajectory is OK)
- Obtained **track parameters do not match MC** (mostly for curlers)

So:

1) Perform fit as usual, with default setup

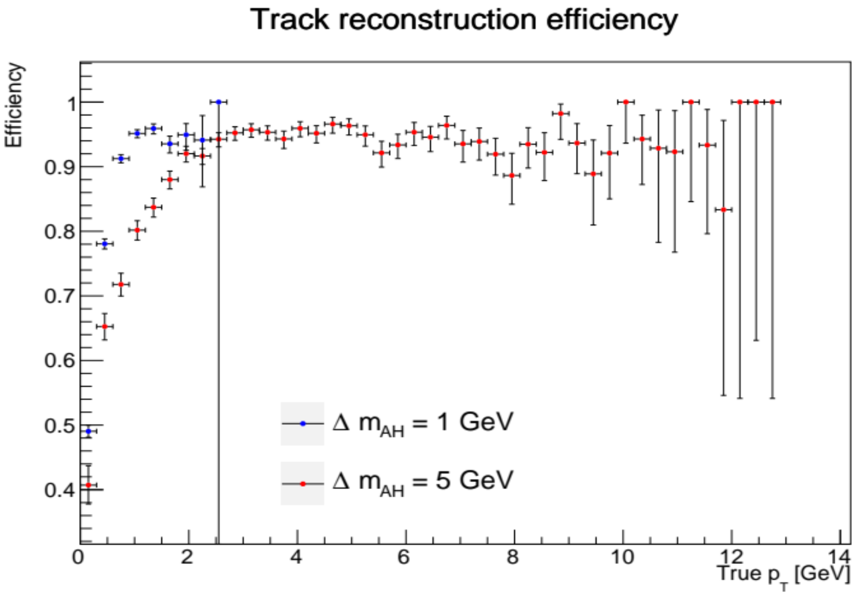
2) Take hits in fit and sort by Z coordinate

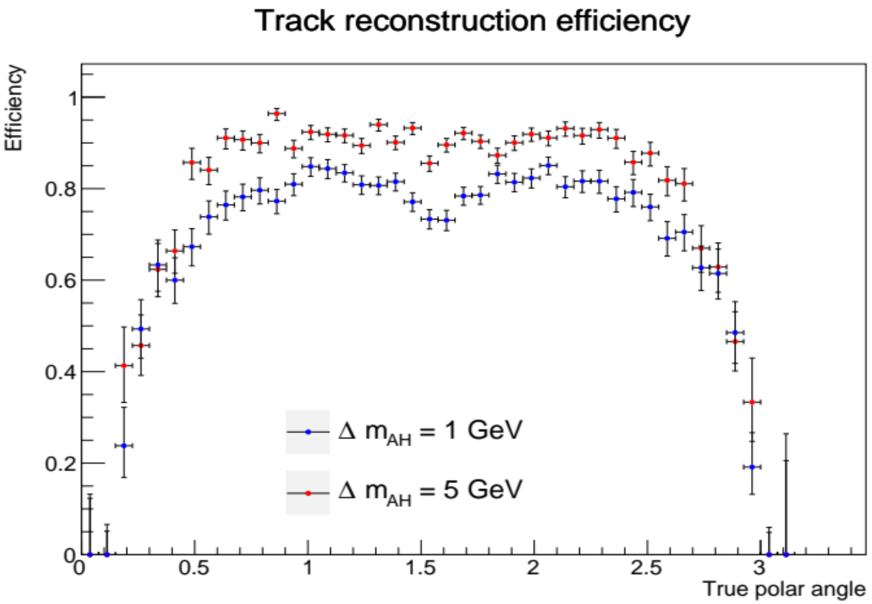
3) Save TrackStates in the first and last hit, sorted by Z



...however, efficiency improved only by 1-2%

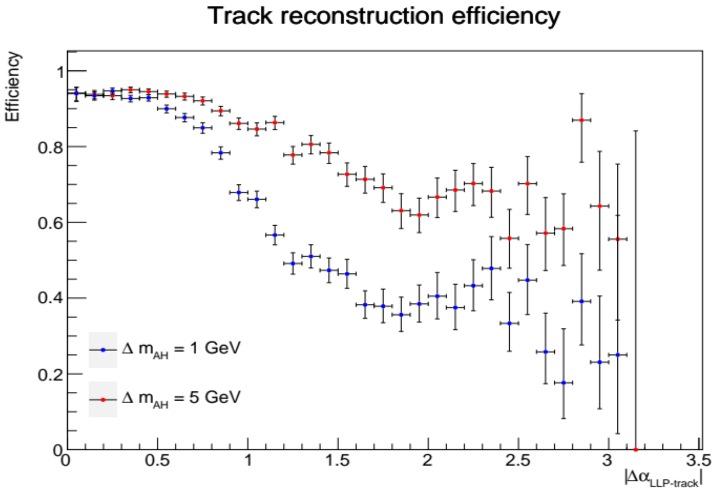
Efficiency distributions



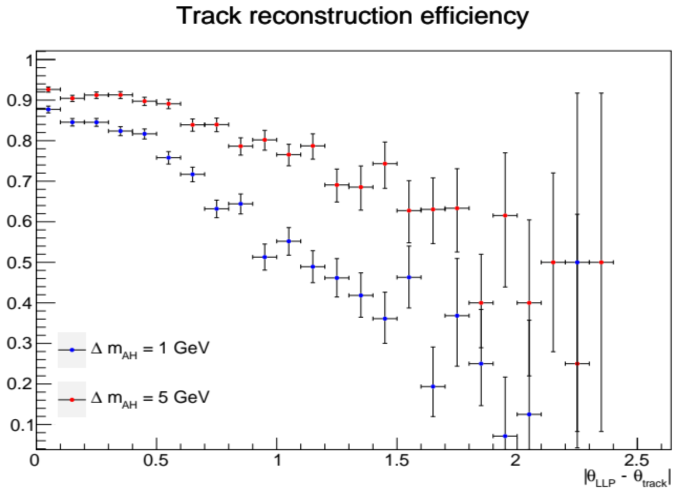


- **Good performance for the high pT**, but no consistent dependence for different mass splittings
- Low efficiency for the forward tracks; small decrease in the central region
  → curlers at very high angles (perpendicular) and LLP decays next to the outer TPC wall

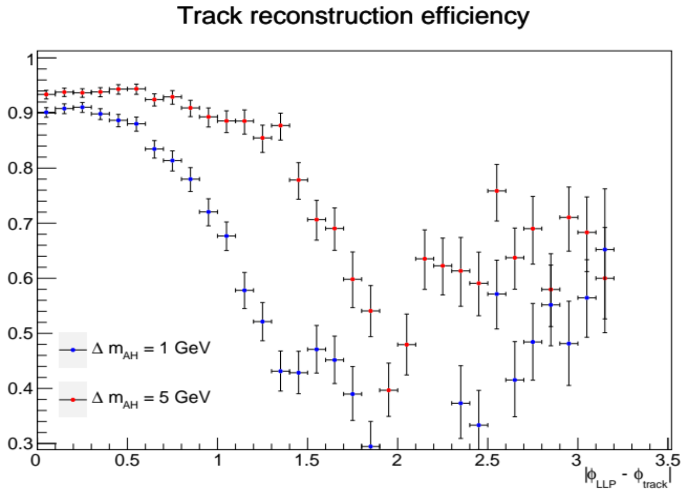
Efficiency distributions



True 3D angle between LLP and track vectors



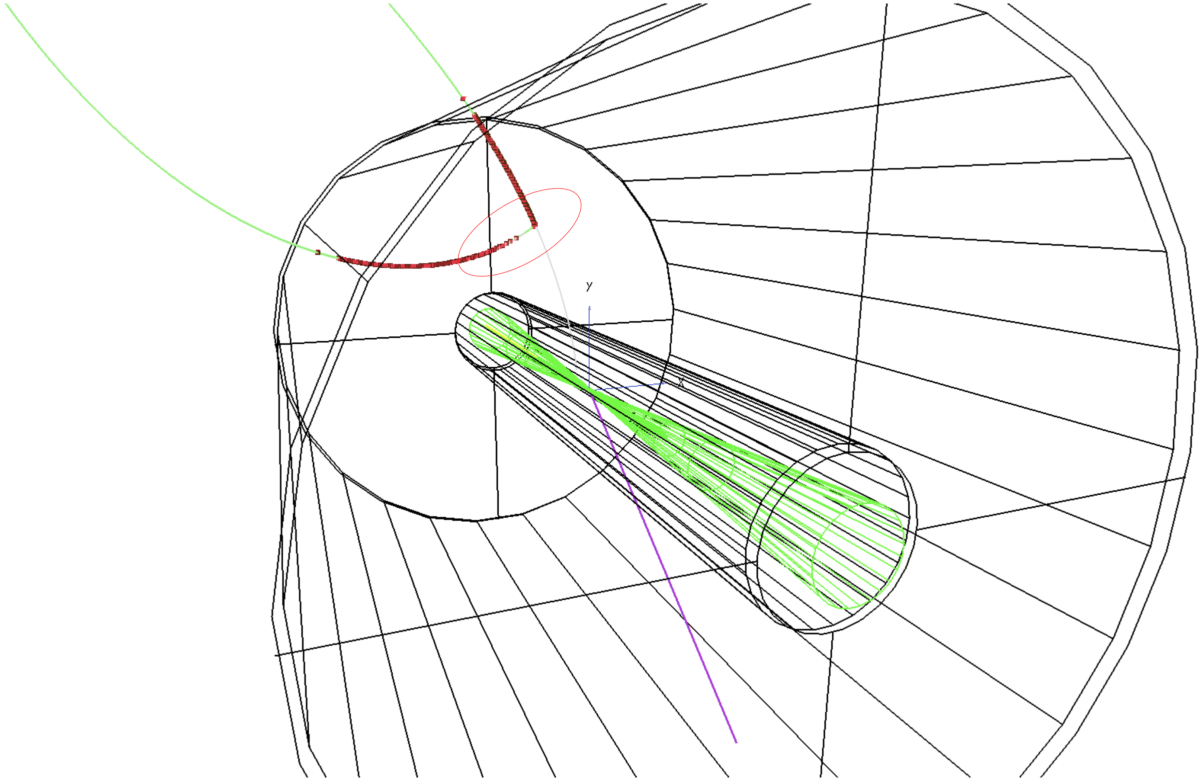
True polar angle difference between LLP and track



True azimuthal angle difference between LLP and track

- High efficiency for the small angles between track and LLP **when tracks point into the IP**
- For higher boost, more tracks at small angles and better overall efficiency

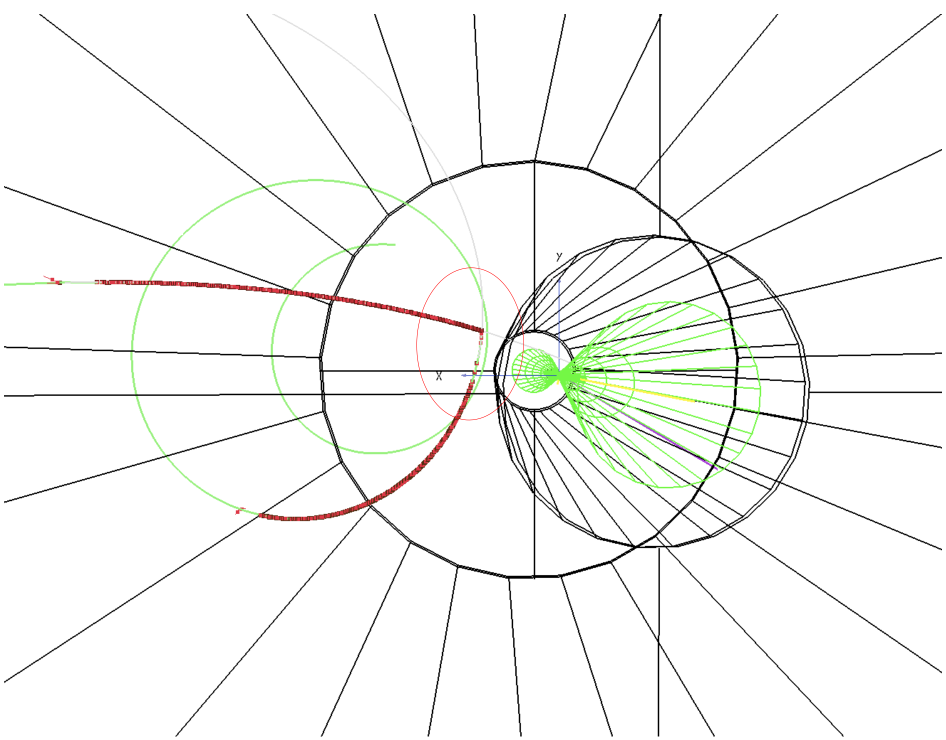
Central φ_{LLP}-φ_{track} region (dM = 5 GeV)



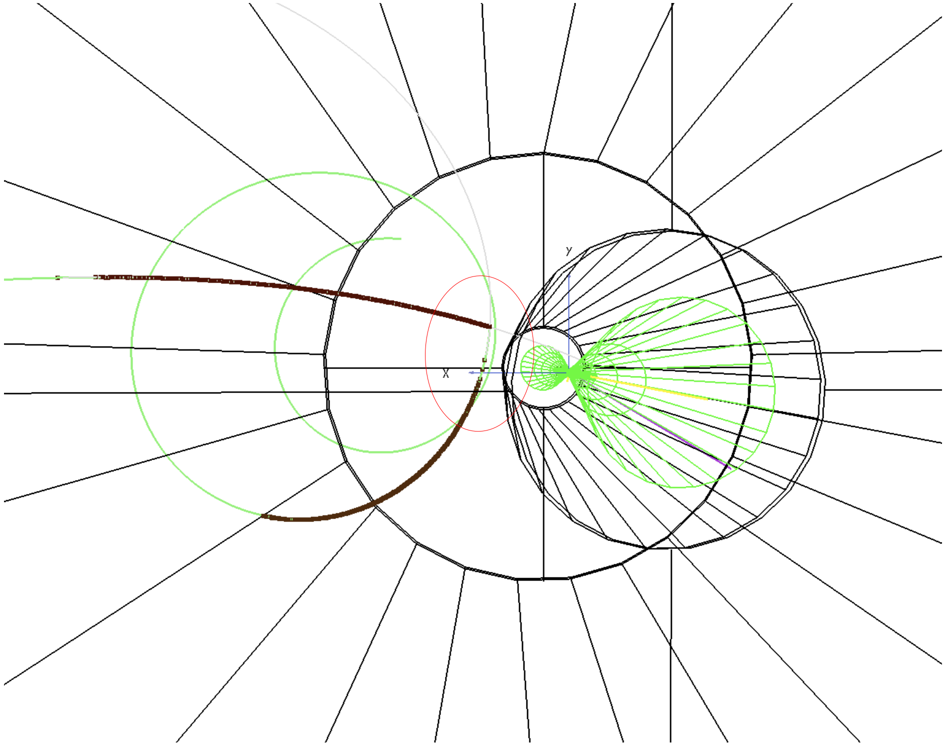
Long distance between first hit and true vertex leads to wrong track parameters

Central φ_{LLP}-φ_{track} region (dM = 5 GeV)

Red: TPCTrackerHits



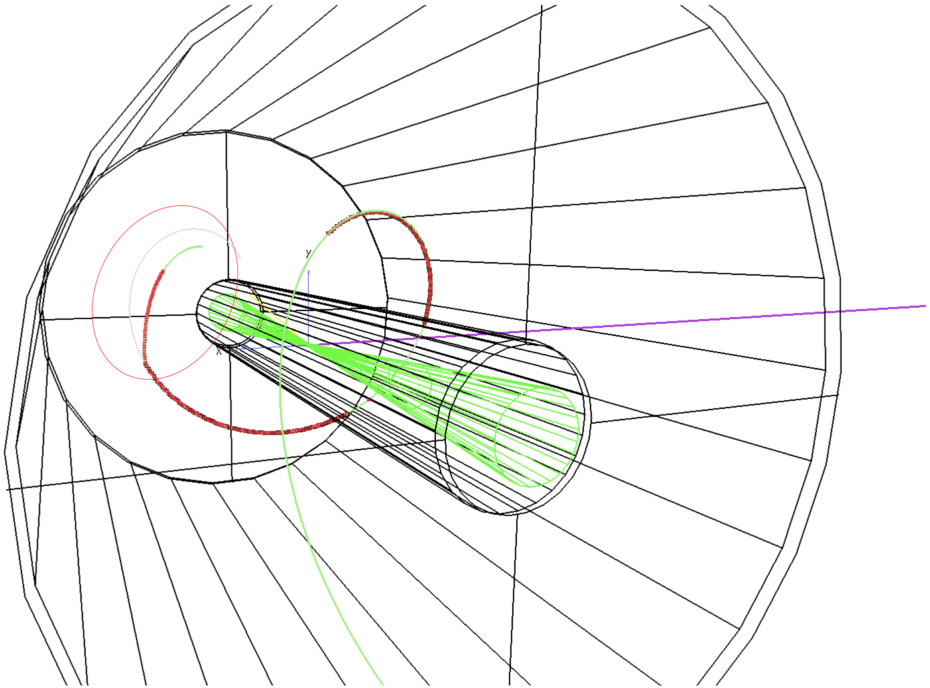
Brown: hits in track



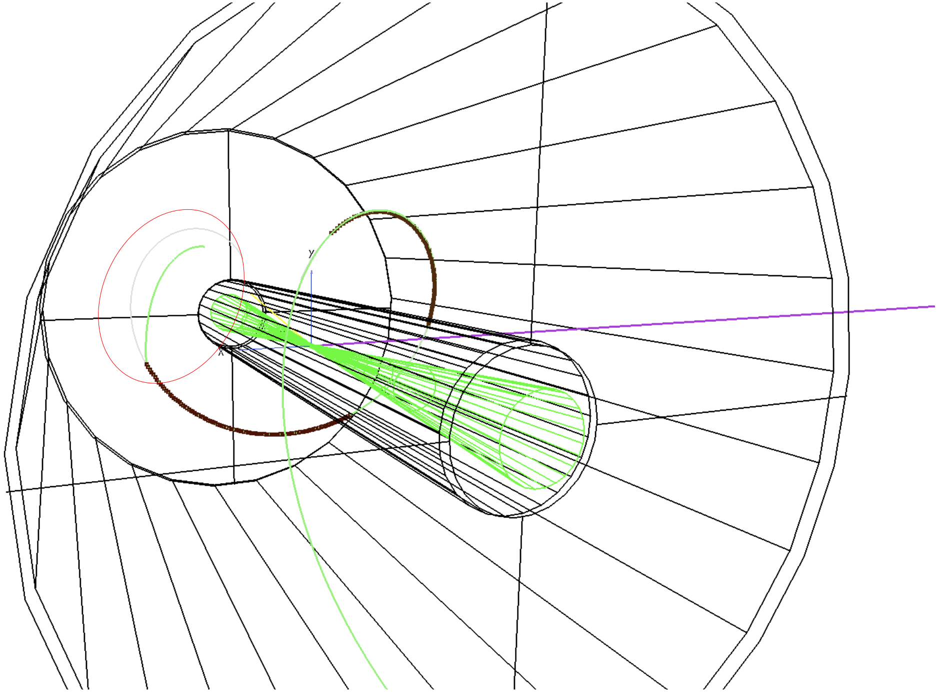
Long distance between first hit and true vertex leads to wrong track parameters

Central φ_{LLP}-φ_{track} region (dM = 5 GeV)

Red: TPCTrackerHits



Brown: hits in track



One of the tracks not found at all or rejected at some point

Conclusions

- First look into track reconstruction in the long-lived particle production events
- We found workaround to simulate vertex displacement on the generator level
- Events with **small mass splitting** and **low-momenta products** studied
- Reconstruction efficiency decreases for large decay angles
- Difficult to identify one general problem to solve and increase the efficiency
- Work in progress

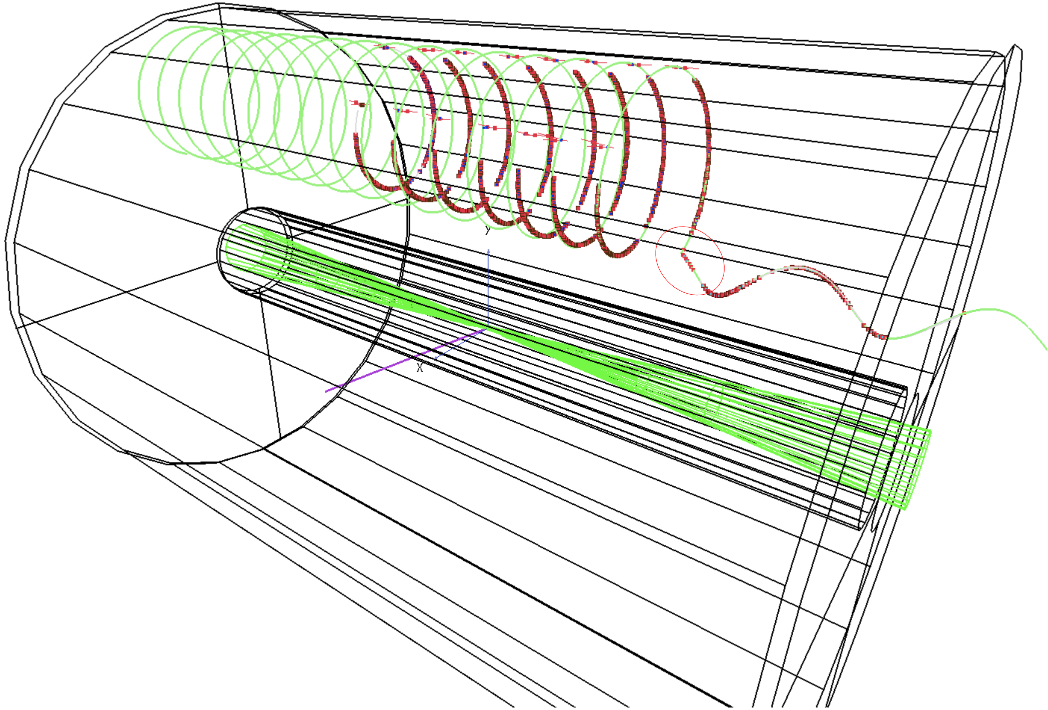
Thank you!

Conclusions

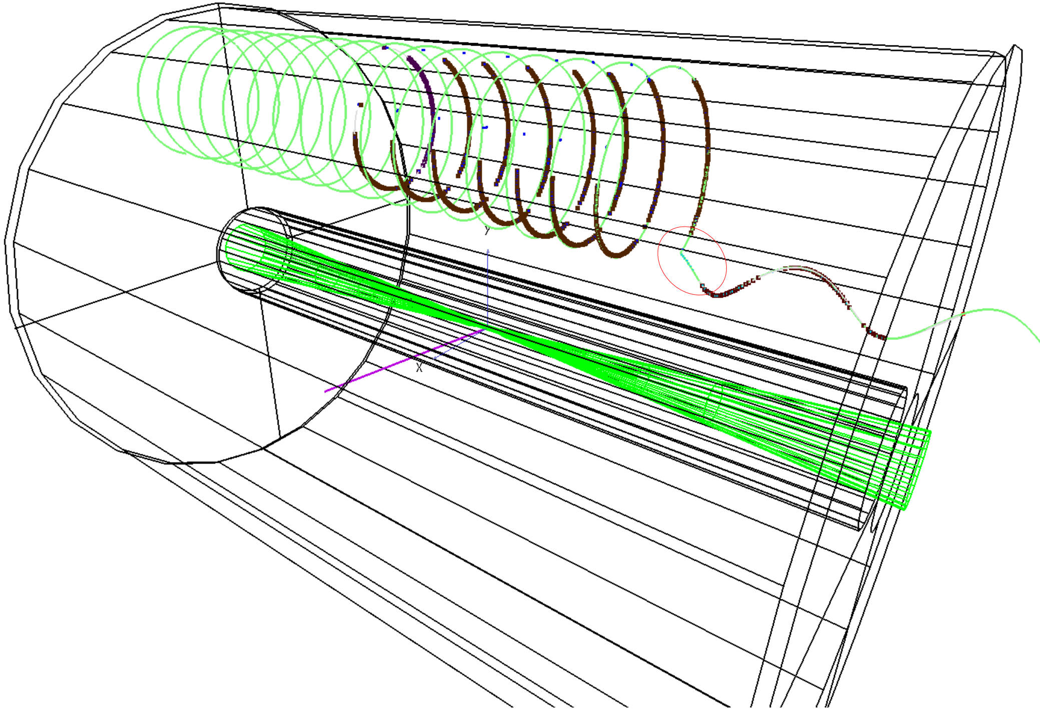
BACKUP

Central φ_{LLP}-φ_{track} region (dM = 1 GeV)

Red: TPCTrackerHits



Brown: hits in track



Long distance between first hit and true vertex leads to wrong track parameters