December 21, 2021

Muon Collider detector simulation

Potential topics of interest

N. Bartosik [a]

[a] INFN Torino *(Italy)*

Simulation of the Beam Induced Background

Detector simulation workflow

Full simulated event obtained via three distinct stages:

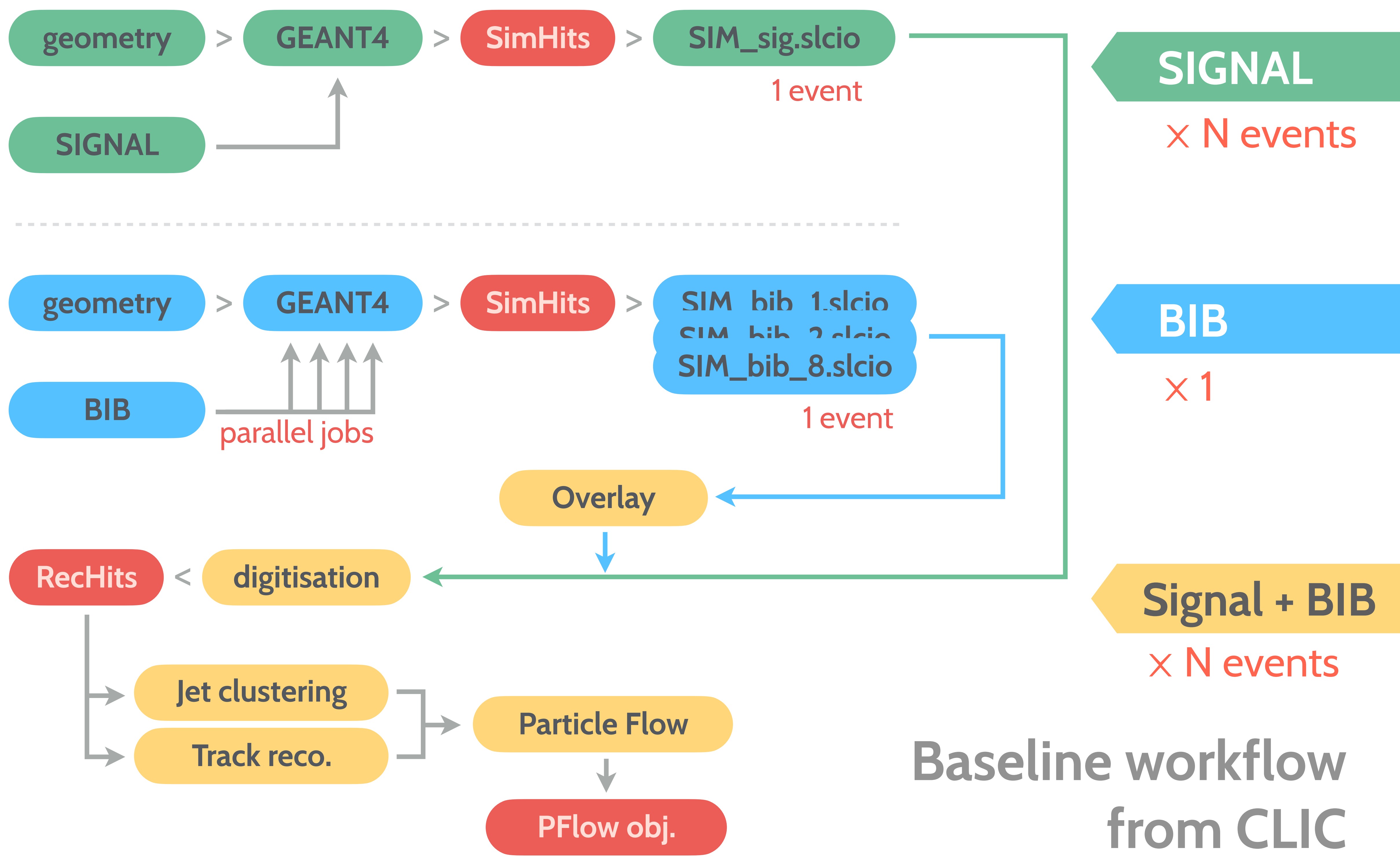
GEANT4 simulation of Signal: straightforward and fast

GEANT4 simulation of BIB: ~10^8 particles/event
↳ **extremely slow** → need a pool of reusable events

Overlay of BIB: performed in each event before digitisation
↳ sensitive to the # of BIB SimHits and merging logics

Reconstruction speed of higher-level objects strongly depends on the amount of input RecHits from BIB

- especially relevant for track reconstruction *(combinatorics)*
- BIB contribution has to be suppressed as early as possible



Baseline workflow from CLIC

BIB contribution creates tremendous amount of data → every step requires careful treatment of computing resources

DISK STORAGE | DISK I/O | CPU TIME | RAM USAGE | DISTRIBUTION

Properties of the BIB contribution

BIB has several characteristic features → **crucial for its effective suppression**

1. Predominantly very soft particles (p << 250 MeV) except for neutrons

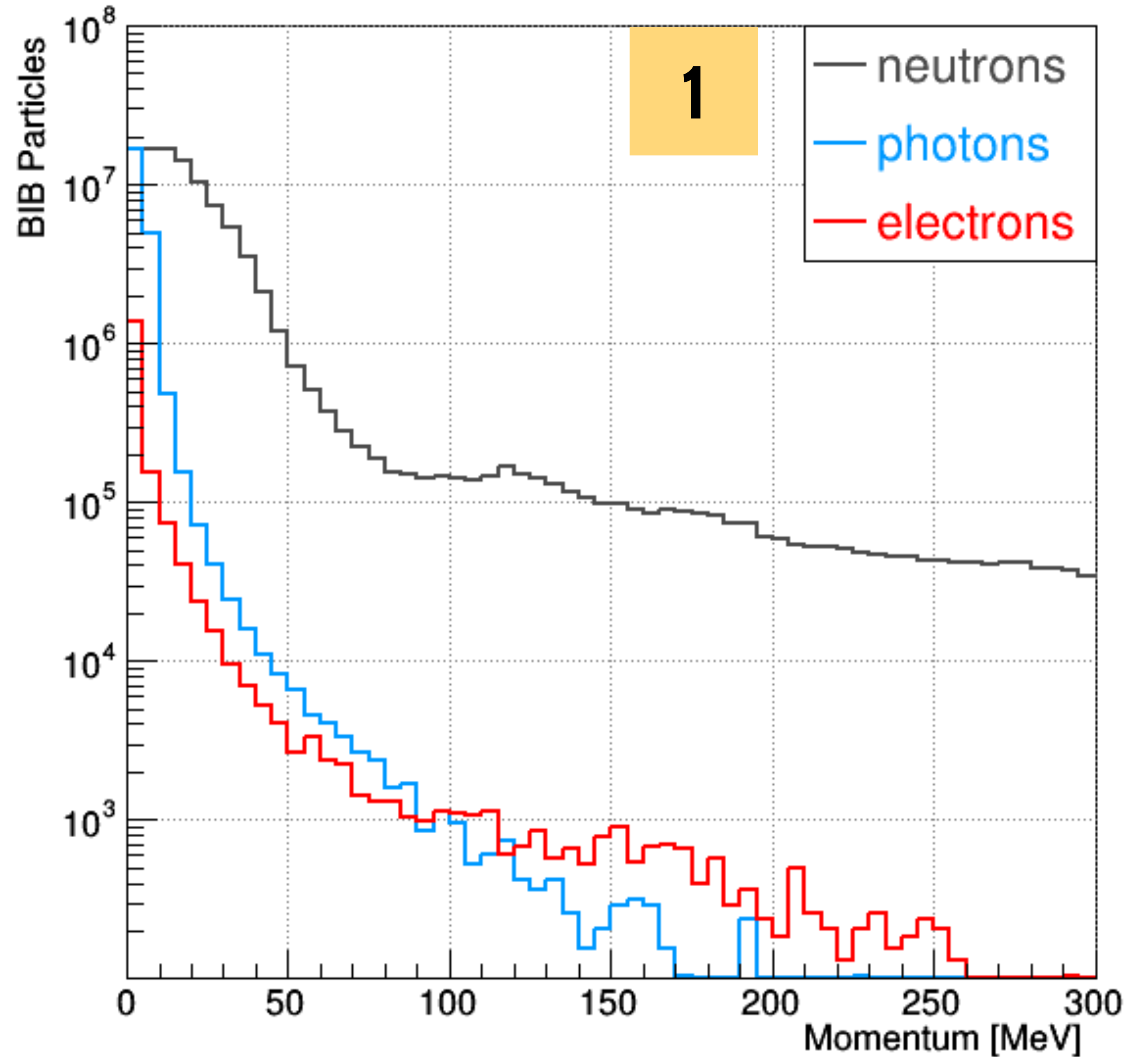
fairly uniform distribution in the detector → no isolated signal-like deposits
↳ conceptually different from pile-up contributions at the LHC

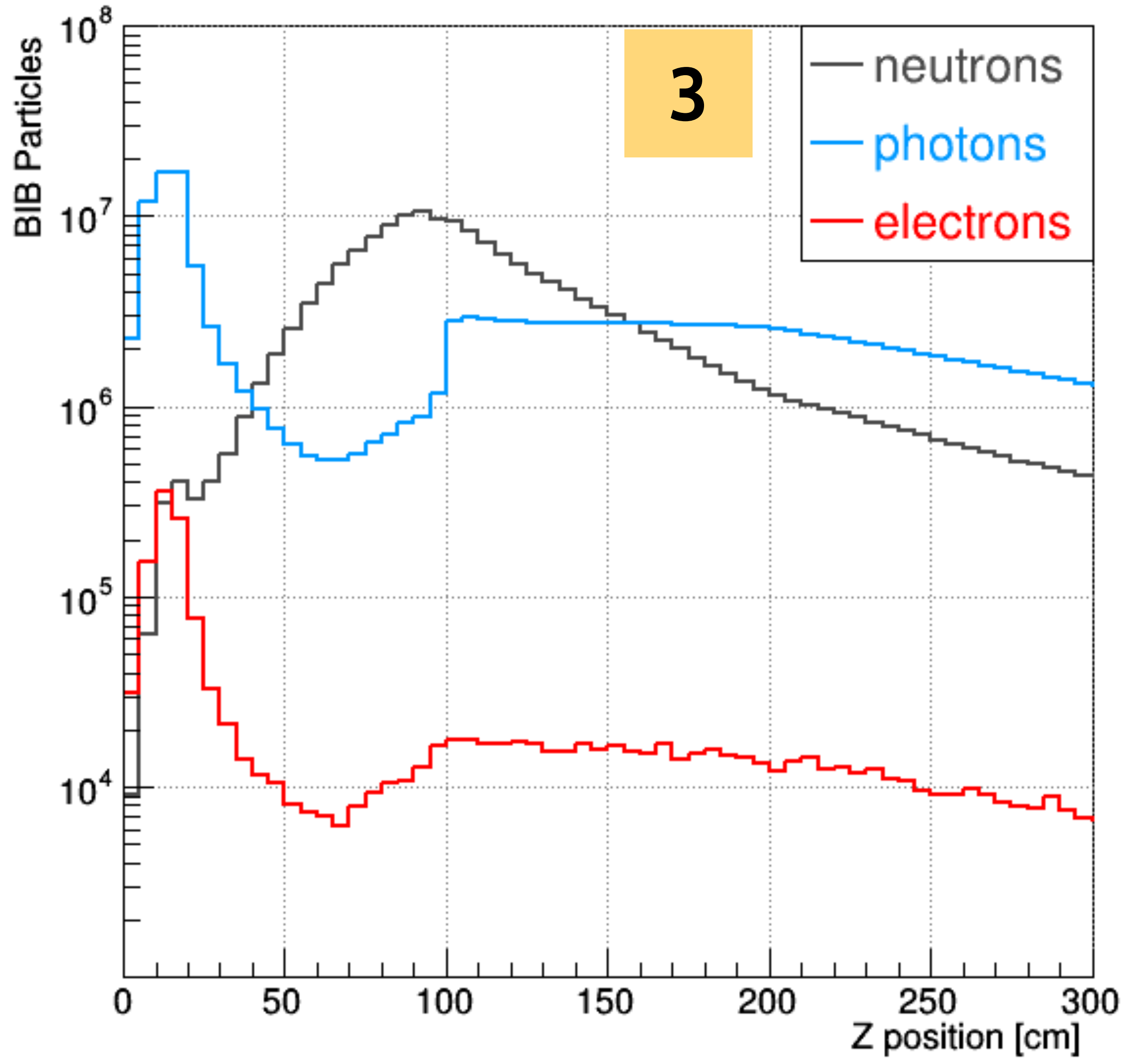
2. Significant spread in time (few ns + long tails up to a few μs)

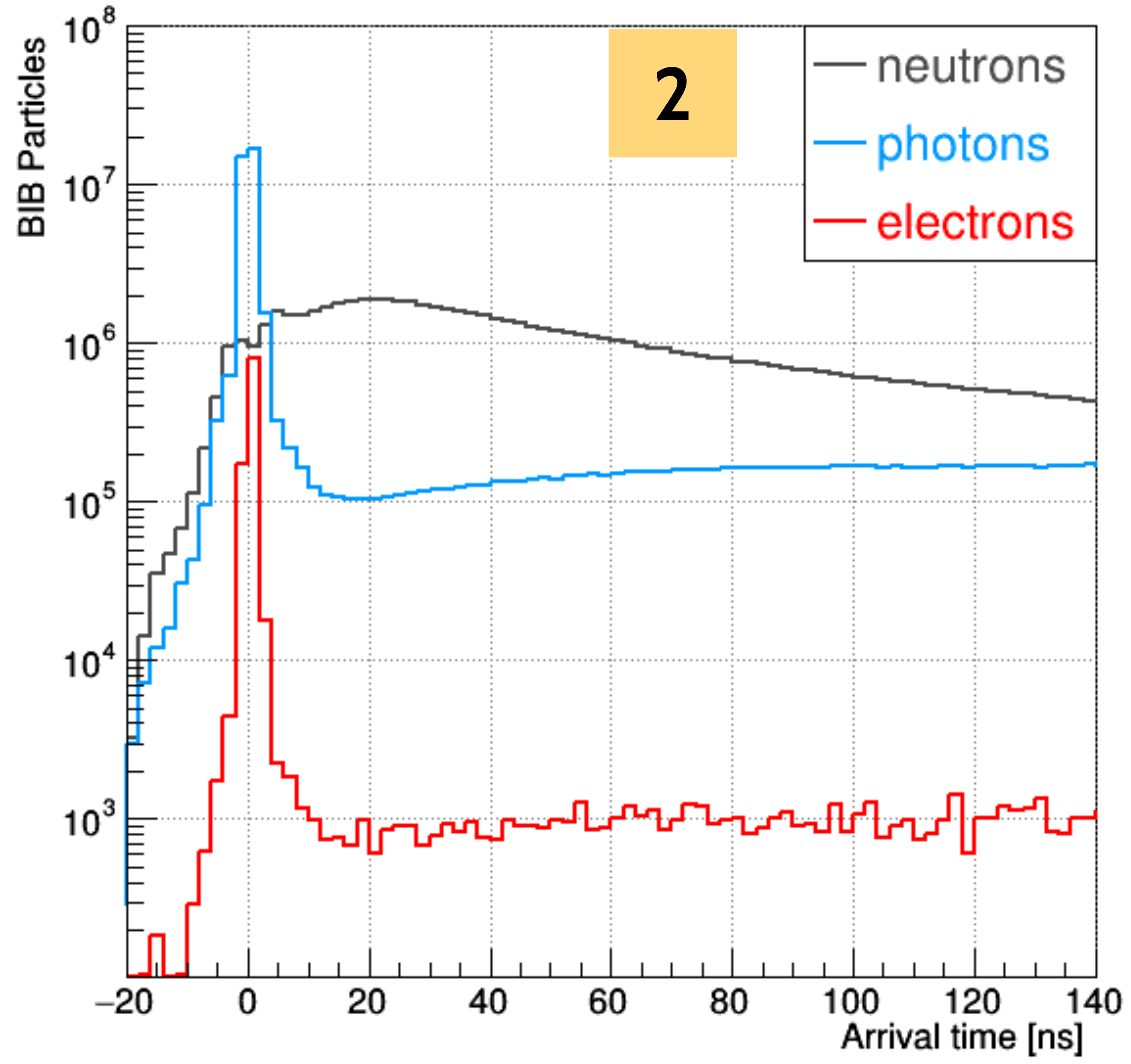
$\mu^+\mu^-$ collision time spread: 30ps (defined by the muon-beam properties)
↳ strong handle on the BIB → requires state-of-the-art timing detectors

3. Large spread of the origin along the beam

different azimuthal angle wrt the detector surface
+ affecting the time of flight to the detector
↳ relevant for position-sensitive detectors







Sophisticated detector technologies and event-reconstruction strategies required to exploit these features of the BIB

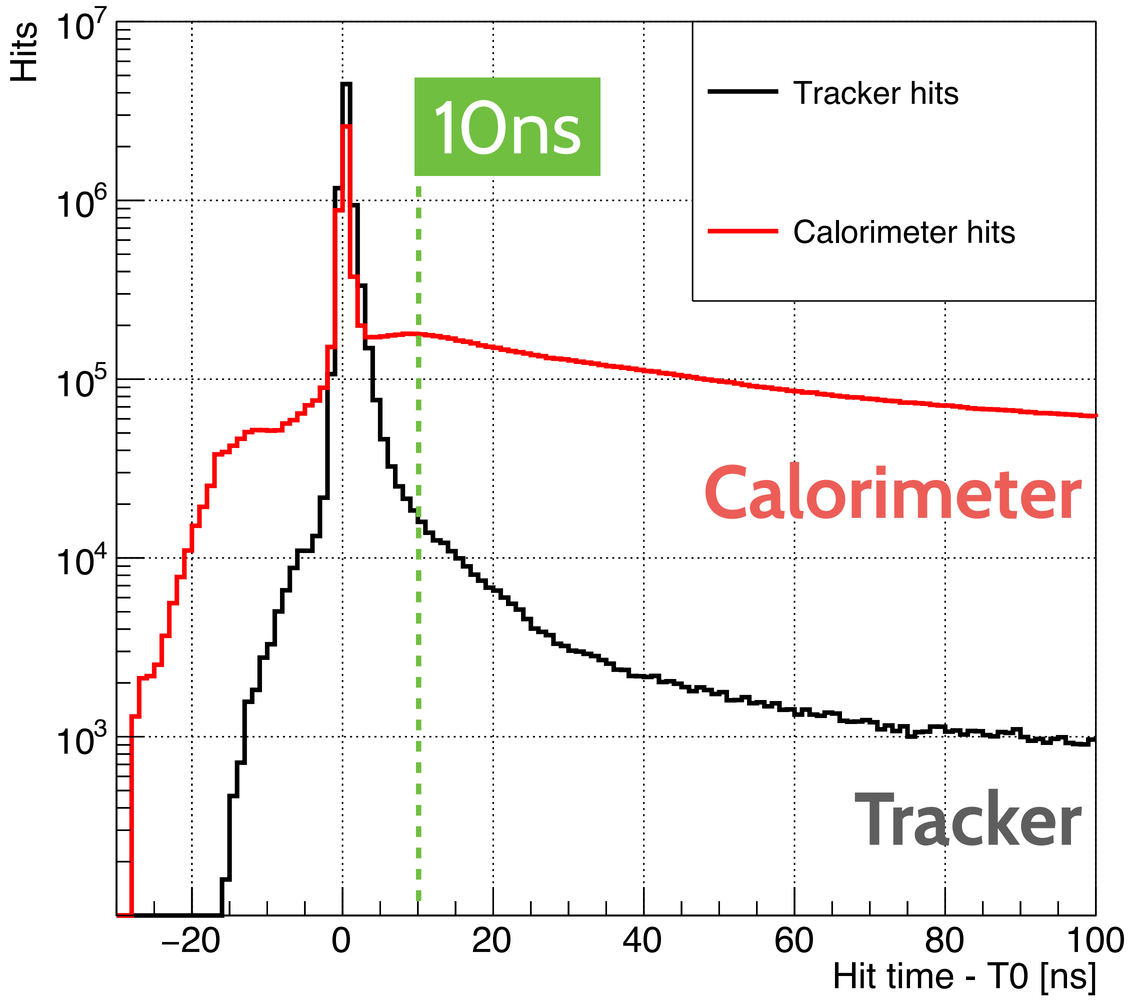
+ detailed simulation needed
to properly evaluate their potential

Current optimisations

Not all of the ~10^8 BIB particles arriving to the detector are relevant for its performance in a real experiment
↳ detectors have finite readout time windows → only a subset of particles relevant for the event reconstruction

1. No GEANT4 simulation of particles arriving too late ×6 less CPU

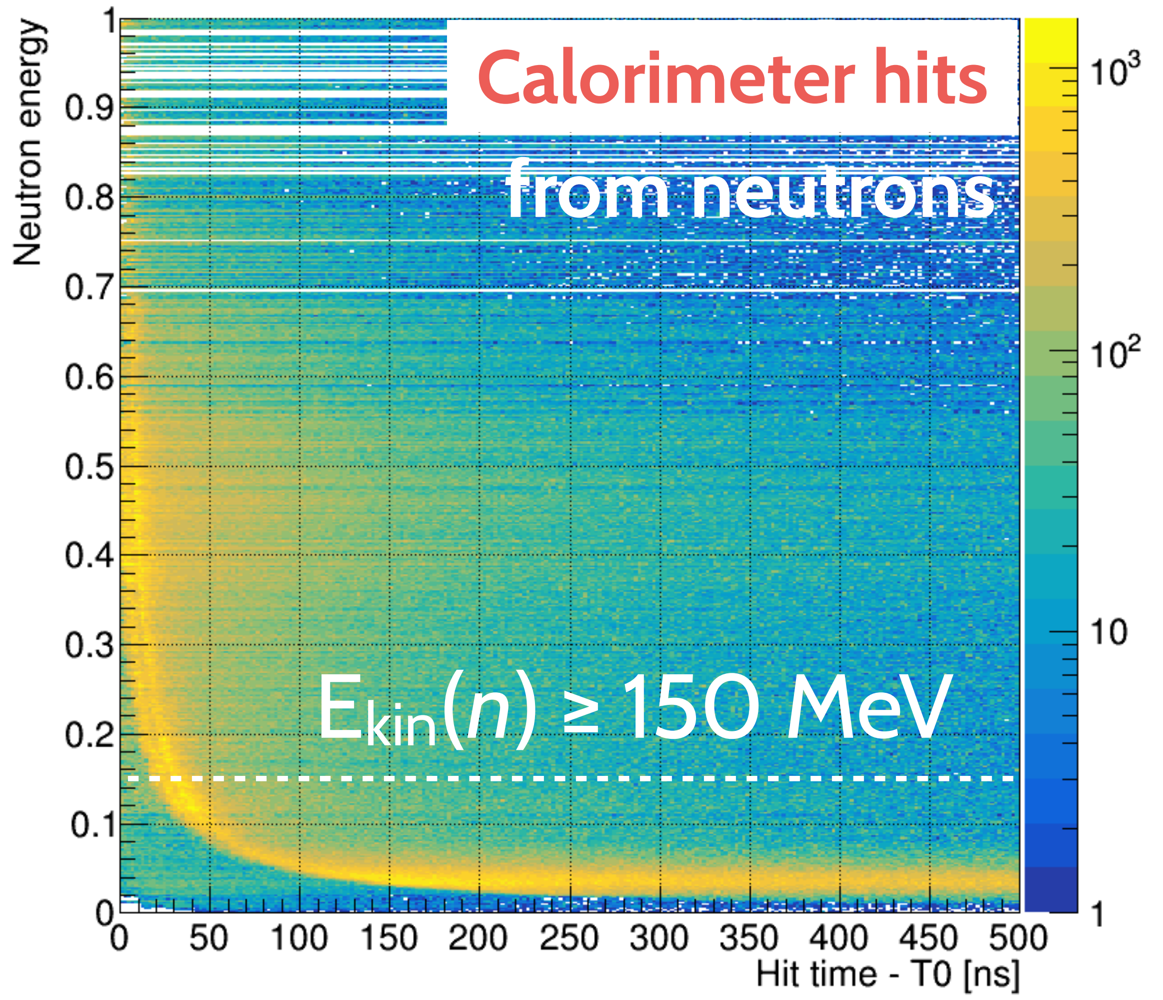
hits at t > 10ns will be outside of the realistic readout time windows
↳ all particles with t > 25ns at the MDI surface are discarded *(accounting for TOF)*



1

2. No GEANT4 simulation of low-energy neutrons ×20 less CPU

high-precision neutron model required for accurate simulation: `QGSP_BERT_HP`
but they are slow → arrive to the detector with a significant delay
↳ neutrons with E_{kin} < 150 MeV can be safely excluded + faster model: `QGSP_BERT`



2

GEANT4 simulation of a single BIB event improved from 127 days → 1 day
↳ ~10-100 reusable events can be generated in several days *(parallelisation)*

3. Russian roulette sampling can be used for the most abundant particles *(used in CMS)*

individual neutrons and photons from BIB are not reconstructable → only combined energy deposits in calorimeters are relevant for digitisation/reconstruction
↳ simulate a fraction (1/f) of particles with a weight (f) → less CPU/RAM/DISK

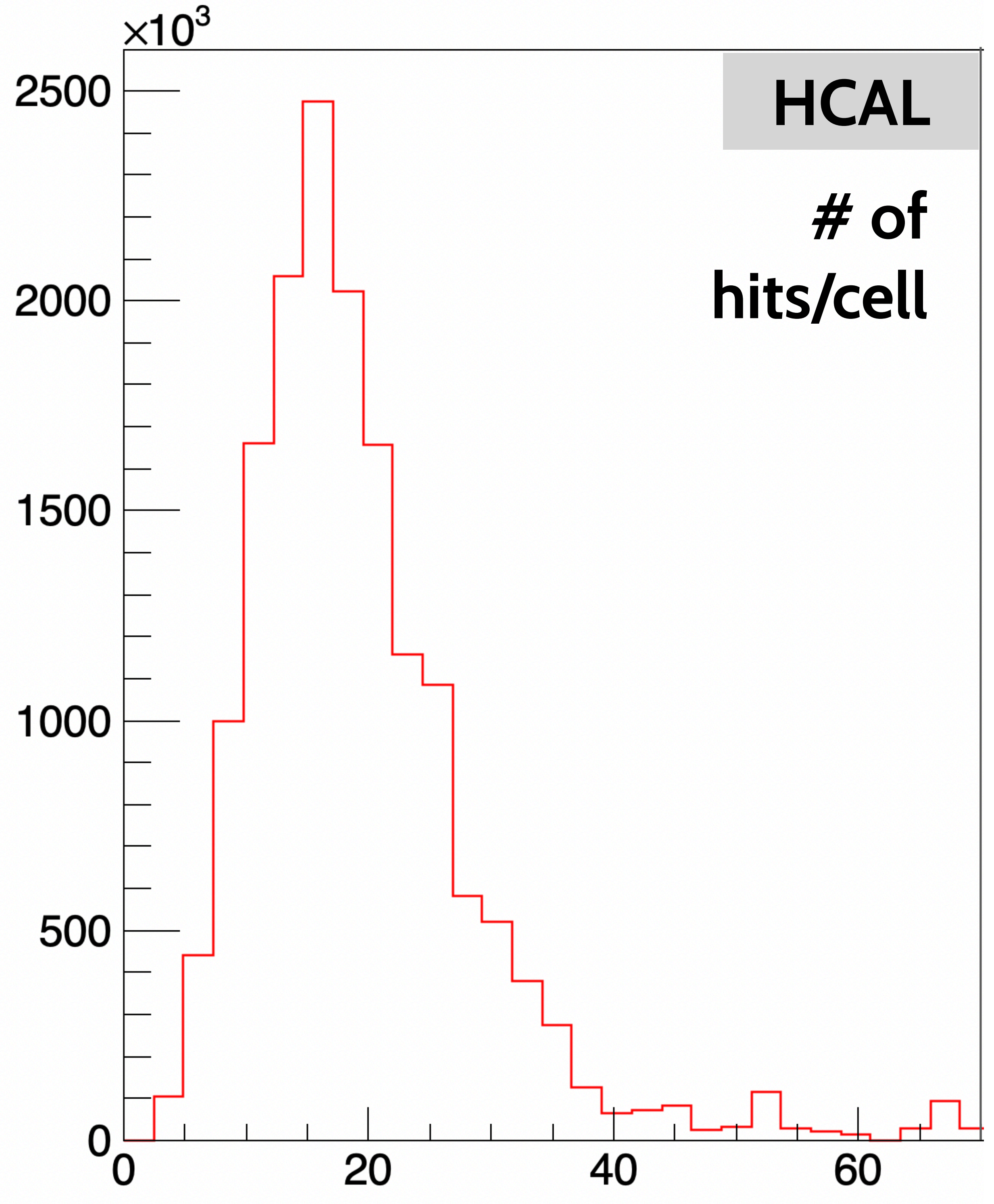
Potential further optimisations

Only combined energy deposit in a single cell is relevant

- **large cell size,** at least 5×5 mm
- **less precise timing,** order of 100ps - 1ns
- **spatial distribution is fairly uniform**

Individual contributions in each cell occupy extra space

on average ×15 more than needed → great potential for CPU savings



1. Russian roulette sampling can be used for the most abundant particles *(used in CMS)*

individual neutrons and photons from BIB are not reconstructable → only combined energy deposits in calorimeters are relevant for digitisation/reconstruction

↳ simulate a fraction (1/f) of particles with weight (f) → less CPU/RAM/DISK

2. Machine Learning methods for skipping GEANT4 simulation completely

train a Generative Adversarial Network (GAN) for producing SimHits directly based on a sample of input MCParticles

↳ ultimate performance required for large statistically independent MC samples

Performance improvement of track reconstruction

Reconstruction of tracks suffers from huge combinatorial background

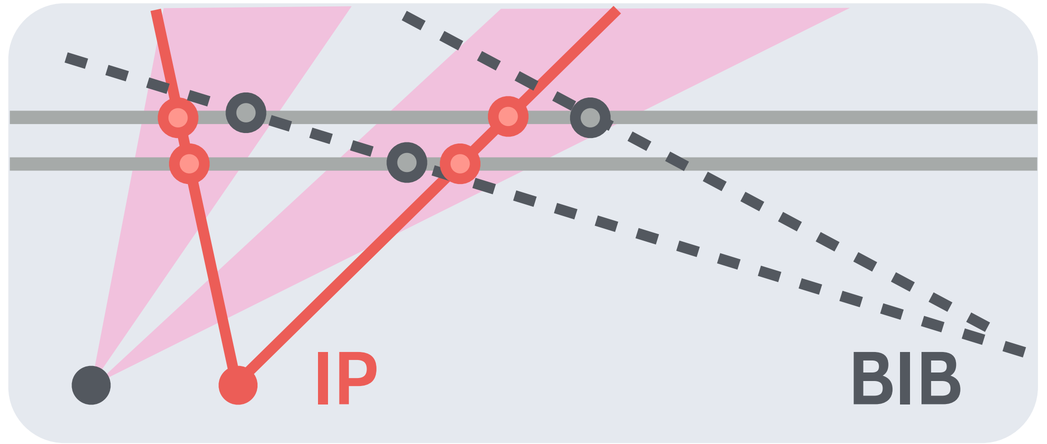
↳ **need suppression of BIB hits + efficient tracking strategies/algorithms**

1. Selection of hits in the narrow time window tailored to the sensor position

↳ limited by the time resolution + beamspot time spread + slow-particle TOF

2. Selection of hit doublets aligned with the IP (double layers in the Vertex Detector)

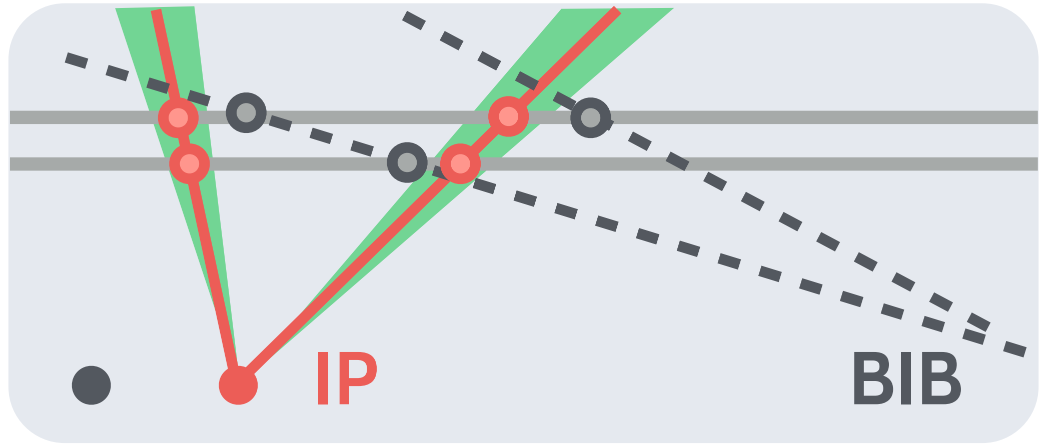
↳ limited by the IP position resolution → requires multi-stage tracking strategy



→

Determine IP position
with faster track reconstruction
- only central region
- inward search from ROI

→

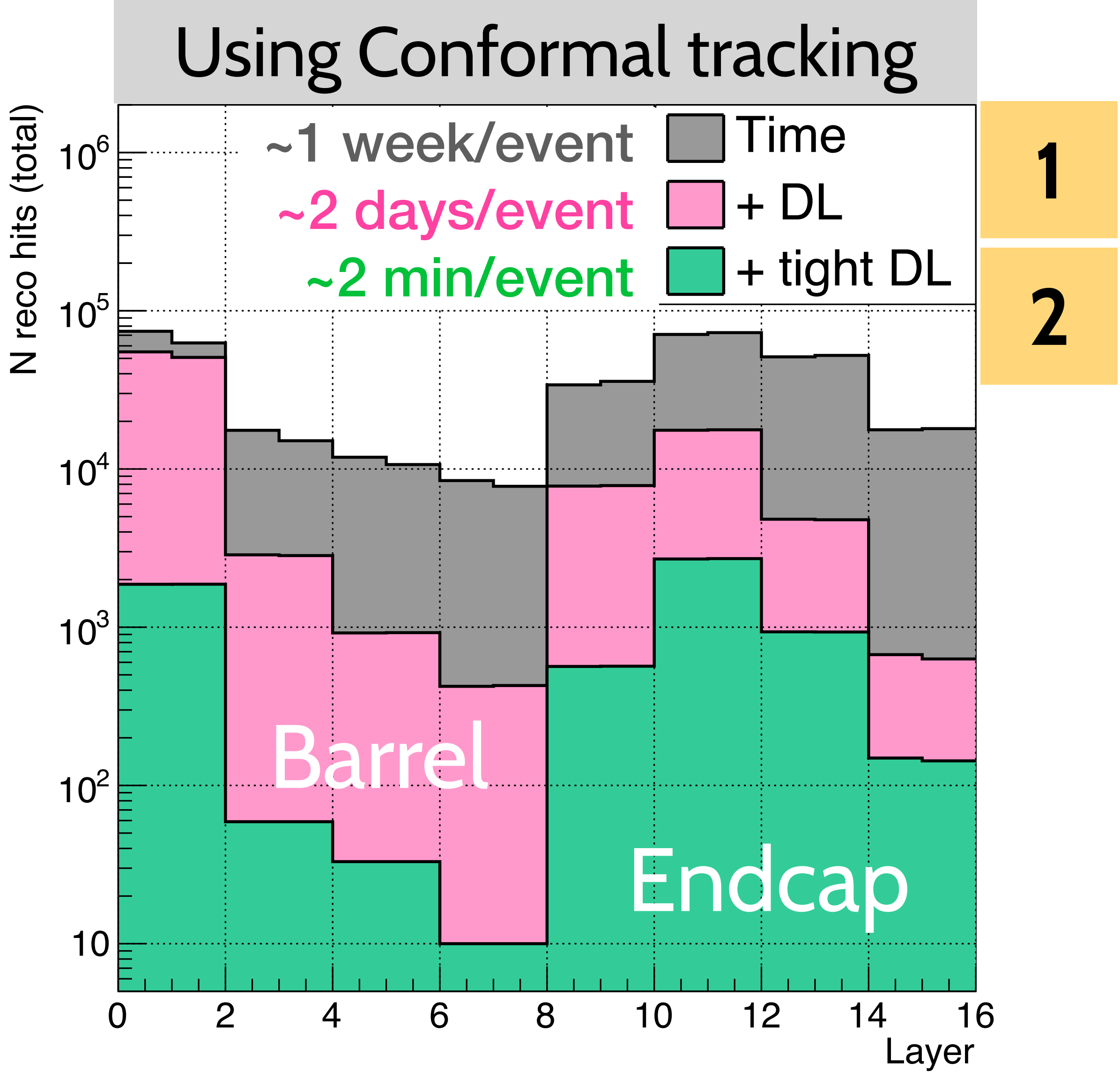


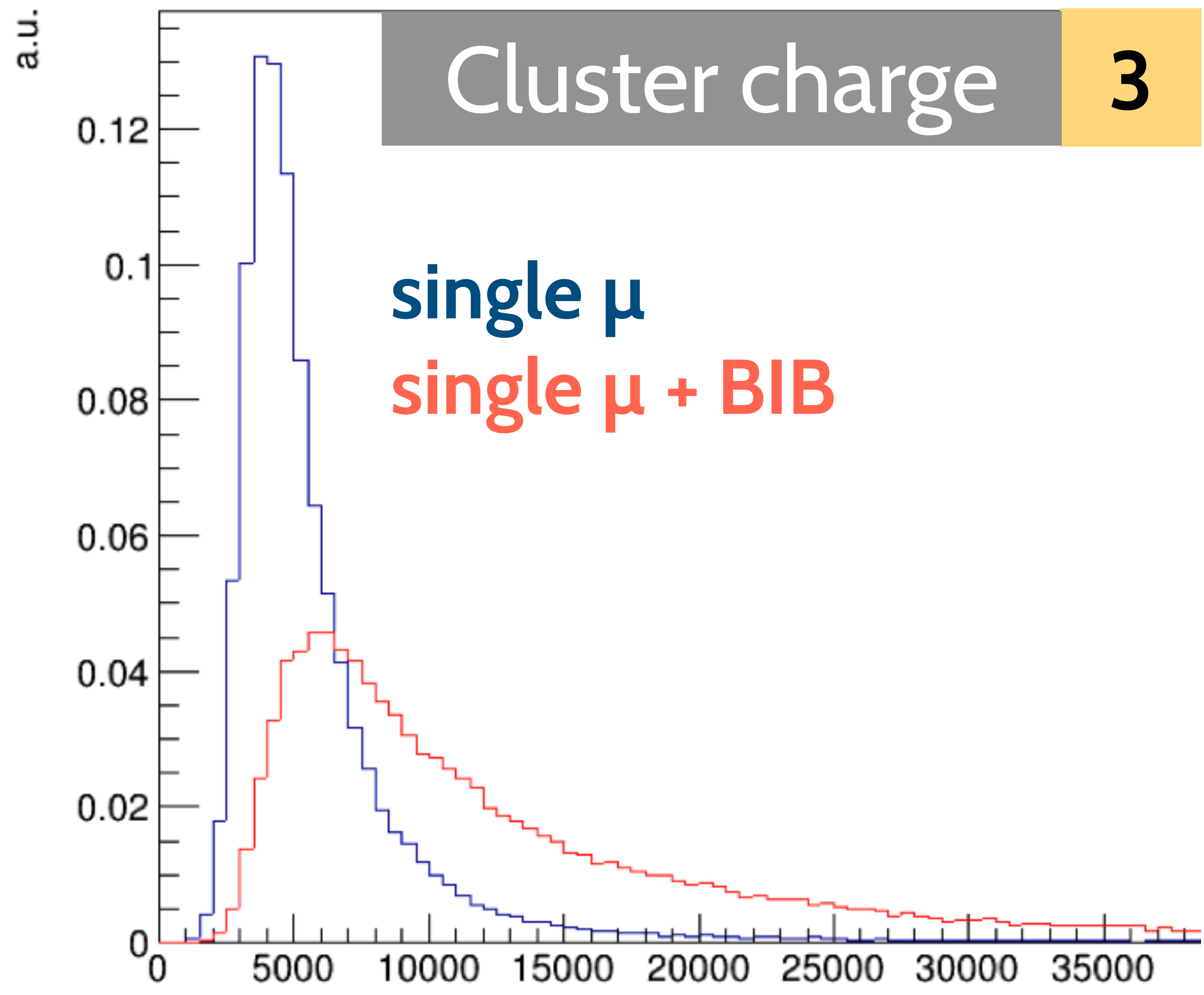
3. Cluster-based BIB suppression (shape and charge of hit clusters)

sensitivity to the particle direction/type in a single layer → realistic digitisation in progress

4. Further promising optimisations to be explored:

- **faster code → integration of ACTS is in progress** *(vectorised calculations, GPUs)*
- **optimised track seeding → regions of interest** *(muons, calorimeters)*





Track reconstruction: loopers

Majority of BIB hits in the Vertex Dedector are from very soft electrons looping along the beam direction (loopers)

↳ many hits from the same particle on the same layer would form a characteristic hit pattern

Computationally efficient identification and removal of such hits can significantly reduce combinatorics

- **neural networks?**
- **Hough transforms?**