Chapter 1

Monte Carlo

The Monte Carlo(MC) method was invented by scientists working on the atomic bomb in the 1940s. Its core idea is to use random samples of parameters or inputs to explore the behavior of a complex system or process. Nowadays, MC are essential part of both theoretical and experimental particle physics research. This chapter gives an overview of ATLAS experiment simulation scheme, simulation methods and software used. Also, a techniques for fast simulation will be discussed.

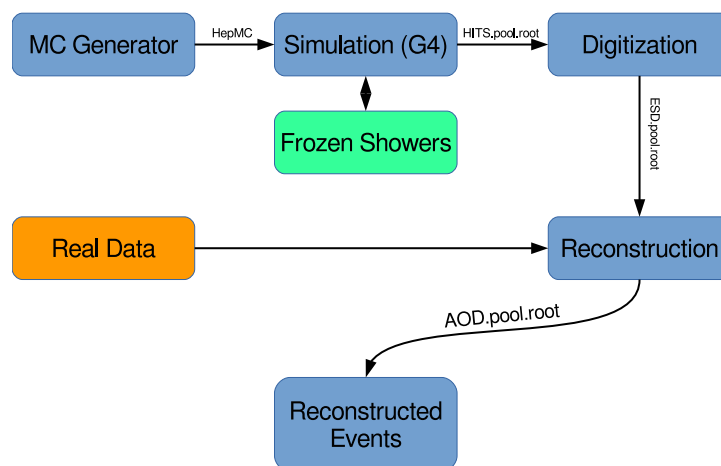## 1.1 Monte Carlo simulation at ATLAS experiment

Fig. 1.1: Diagram of the ATLAS MC production chain

MC is allowing to make different analysis, such as compare data with predictions, study detector or selection algorithms performance. All of this applications are requiring MC precision. Simulation software expects to use precise physics models for sampling and have large enough statistics, to exclude statistical uncertainties (usually 5-10 times more, than expected in a data). ATLAS simulation software is integrated into Athena and usually used during large production of events. Simulation chain is generally divided into 4 main steps (Figure 1.1):

Event generation Simulation of hard interaction and a resulting high-energy particles parameters. This step is independent of ATLAS detector geometry.

Simulation Simulation of energy depositions ("hits") done by final state particles in ATLAS detector.

18 **Digitalization** Simulation of detector responce using "hits" information: first, inputs to the read out
19   drivers (ROD's), called "digits" are constructed, then, ROD functionality is emulated. Detector
20   noise effects are added at this stage.

21 **Reconstruction** Production of the Analysis Object Data (AOD) files, which are containing sufficient
22   information for physics analysis. This stage is identical for both data and MC

23 This is scheme is allowing to use computing resources efficiently, than with a single-step simulation,
24 and simplifies software validation, since it is possible to reuse files from previous stages. In the
25 following sections event generation and simulation will be described more detailed
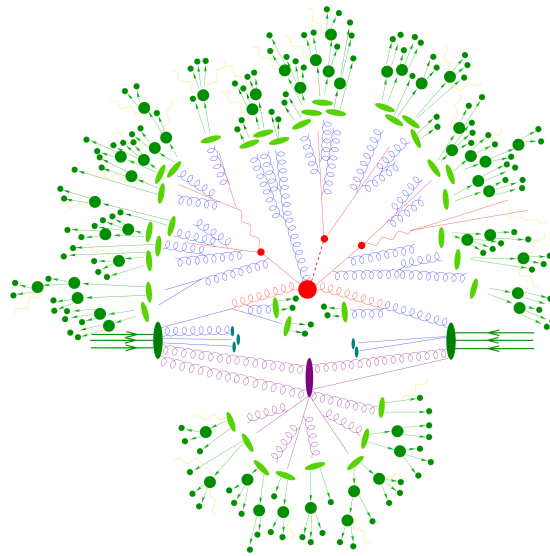
26 ### 1.1.1 Event generators

Fig. 1.2: Schematic view of a $t\bar{t}H$ event produced in a pp-collision: the hard scattering is shown as
    a red blob with the solid and dashed lines as the resulting three particles. Independently
    happening multi-particle interactions are indicated by the violet blob. Parton showers are
    shown with curly lines. Hadronization yields hadrons as shown in light green, while the final
    state particle are dark green.

27   The outcome of the hard interaction could be simple scattering, their annihilation or a combination
28 of two. In any case, the final state consist of a large multiplicity particles. The main goal of event
29 generator is to provide a complete picture of this final states: description of the particle types and
30 momenta on event-by-event basis. The factorisation theorem allows to make event generation in
31 independent stages, which are dominated by different dynamics:

32 **Modelling of hard subprocess** Hard subprocess is happening at the smallest scales of times and
33   distance, where all of the colliding partons are considered free and their initial momentia are
34   given by selected PDF set. Process of interest is simulated by selecting production channels and
35   calculating corresponding matrix elements (ME) in the desired level of accuracy in pertrubation
36   theory . Most of the generators have leading order or next to leading order ME in α_s. Scale of α_s

37 is can be determined through renormalisation of factorisation. Also, inclusion of higher-order
38 correction reduces scale dependency, however it makes calculations technically more difficult.

39 **Parton showering** Quarks and gluons from hard process can QCD radiation, resulting on the
40 dozens of additional partons associated with the event. This process calculated as step-by-
41 step evolution of momentum transfer scales from highest (hard subprocess), to the lowest
42 (around 1 GeV). There is a possiblity of double counting between showers and hard subpro-
43 cess. This can be avoided using matching approach, for which higher order corrections to ME
44 are integrated with parton showers, or merging strategy, there jet resolution scale is used as
45 an threshold between matrix elements and parton showers.

46 **Hadronisation** Final stable color-neutral particles, what can be detected in experiment, are formed
47 during hadronisation. This occures at larger nonpertrubative scales and usually implemented
48 using different phenomenological models.

49 **Modelling underlying event** Parallel to the main process other collisions of partons can occur,
50 called underlying event. These additional interaction can produce partons which contribute to
51 the final state. This is one of the least understood aspect of hadronic collisions.

52 Schematic plan of simulation of ttbar event is shown on a figure 1.2. The hard scattering itself
53 is shown as a red blob with the solid and dashed lines indicating the resulting particles, which
54 themselves decay further. Underlying event is indicated by the violet blob. Parton showers are shown
55 with curly lines. Hadronization yields hadrons as shown in light green, while the final state particles
56 are dark green.
57 The current analysis uses samples generated with the following generators:

58 Powheg Powheg is generator with NLO ME, that can be interfaced with other generator(such as Pythia
59 or Herwig) for higher precision of showering.

60 Pythia Pythia is a general purpose generator for hadronic, hadron-lepton and leptonic collisions. It
61 can model initial and final state showers, hadronisation and decays, underlying event (via multi
62 parton interactions). Pythia contains library with around 240 processes with LO ME. It uses
63 Lund String model for hadronisation.

64 Herwig Herwig is a LO general purpose event generator for simulation lepton-lepton, hadron-lepton
65 and hadron-hadron collisitons. The main difference between Pythia and Herwig is that it is
66 uses angular ordering in the parton showers and also models the hadronisation step based on
67 the cluster fragmentation

68 Sherpa Sherpa is a generator with tree level of matrix elements, featuring its own implementation
69 of parton shower and hadronisation models.

70 Photos Precision tool for QED radiative corrections in W and Z decays.

71 Tauola Generator, used to describe leptonic ans semi-leptonic τ-decays.

1.1.2 Simulation in Geant4

73 After event generation, simulation software obtains hardware response for a final state particles.
74 The main method used by ATLAS, referred to as *Full Simulation*, makes use of the Geant4 [**?**]. It is
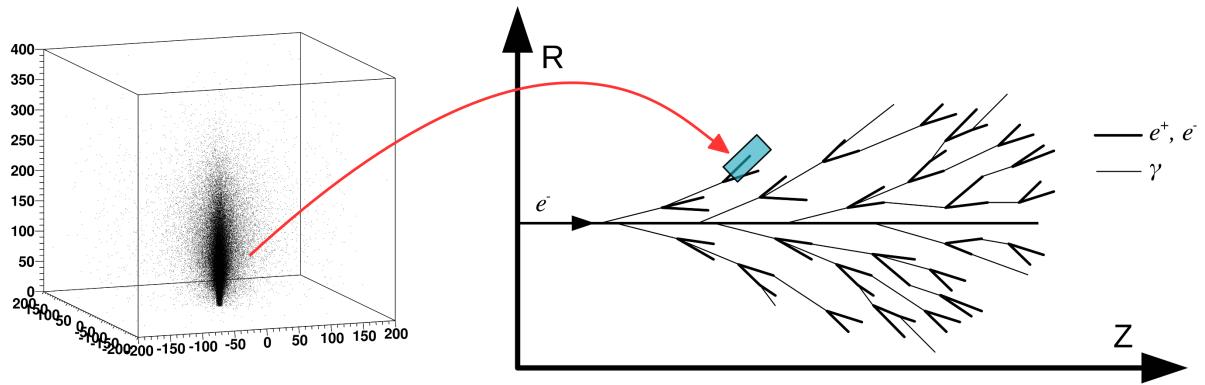
Fig. 1.3: Diagram showing the shower substitution of the low-energy particle, during the high-energy particle simulation.

C++ based toolkit for the simulation of the passage of particles through matter. It is used in a wide range of experiments in a high energy and nuclear physics.

Geant4 can simulate complex detector structure with sensitive detector material and corresponding infrastructure. It can also calculate basic properties of materials, like radiation and interaction length. For detector Geant4 stores "hits" information - snapshots of physical interactions. In Geant4 events and particles are simulated separately and each particle is moved step by step. Size of this step is chosen to preserve both CPU performance and required precision. Physics processes are handled as a discrete either at rest, along step or after it. Geant4 package has different models and approximations of hadronic and electromagnetic processes. Some of them are not so precise, but computationally fast. It allows to choose set of the models, called physics list, depending on a case needs. There are several reference physics lists, that are validated for each new release. ATLAS experiment is using one of this lists.

It is necessary to have a mass MC production for each data taking, what is taking most of the resources. Some of the Run-1 analysis are dominated by available MC statistics. It is possible to improve in CPU usage by tuning physics list or reworking of B-field access. Also there are long term developments for multi-threading and vectorisation of the code. Yet, Run-2 has a higher pileup and luminosity, so even more MC events are needed. This means that fast and accurate simulation approach is essential. During simulation largest time is spend on calorimeters. This is the motivation for development of fast calorimetry techniques.

There are two main methods used in ATLAS:

- Parametrisation of calorimeter cells responce. Spacial energy responce is simulated using longitudial and lateral energy profiles.

- Frozen Showers. This technique will be described more detailed in the following section

1.1.3 Frozen Showers

Main principle of this library is described by it's name. It is using pre-simulated "frozen" showers generated in a full simulation. Particles below minimum energy thresholds are killed and replaced with with this showers. All of the other particles are using full simulation. This process is schematically shown on a Figure 1.3.
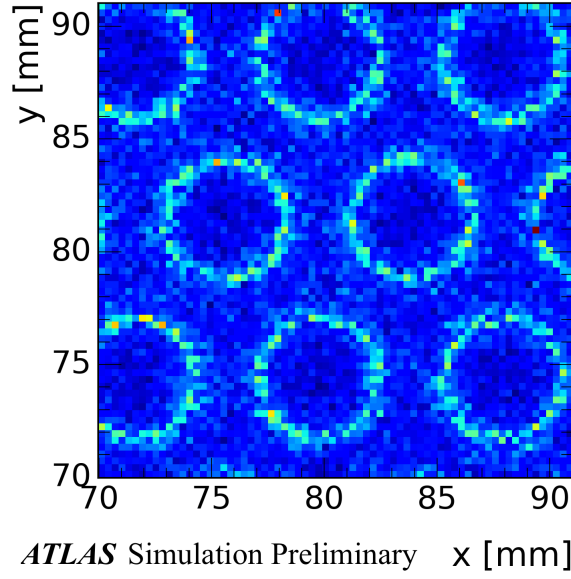
Fig. 1.4: Energy response for electrons in a calorimeter in a x vs y plane and in a distance plane

103   The library itself organized as follows: it contains basic simulation parameters, like Geant4, geom-
104 etry and atlas software release version and physics list used. It stores information about position of
105 initial particle in a bins,while energy remains unbinned. Each shower stores lateral and transverse
106 size and information about all of the hits: four vector (time and position) and energy deposition.

Production use of Frozen Showers

107

108 During a simulation, if an energy of particle falls below cut-off energy, particle algorithm examines
109 resulting shower containment. It checks that particle is far from the edges of calorimeter, so what
110 shower will be by 90% inside calorimeter. This dependes also on a energy of particle, because
111 shower sizes are growing with energy. When particle is removed and substituted by shower taken
112 from corresponding eta and distance bin with the closest energy found. Energies of the hits in shower
113 found are scaled to fully correspond to particle energy. Additionally, shower direction is changed to
114 the direction of the shower.

     Frozen Showers have been used in Atlas Monte-Carlo production since run-1. This method is
applicable for all LAr calorimeters in ATLAS, but currently it is enabled for simulation of forward
calorimeters (FCAL), since it is showing the smallest differences, compared to the other fast sim-
ulation methods (e.g parametrisation). This is because of large of non-uniformly distributed sensi-
tive material, which is giving different responce, than a dead material (Figure 1.5). Resolution of a
calorimeter can be written as:

$$\frac{\sigma}{E} \approx \frac{1}{\sqrt{E}} \oplus \frac{1}{E} \oplus const, \tag{1.1}$$

115 where symbol $\oplus$ indicates a quadratic sum. The first term is 'stocastic term', which is including
116 intrinsic shower fluctuations, second takes into account readout noise effects and pile-up fluctua-
117 tions. Constant term derives from non-uniformities in a detector, what are causing large fluctuation
118 of the energy loss. Resolution of high-energy electrons is mostly dominated by this term. Example
119 of this fluctuations is shown on a Figure 1.4. It can be seen, that particles inside sensitive material
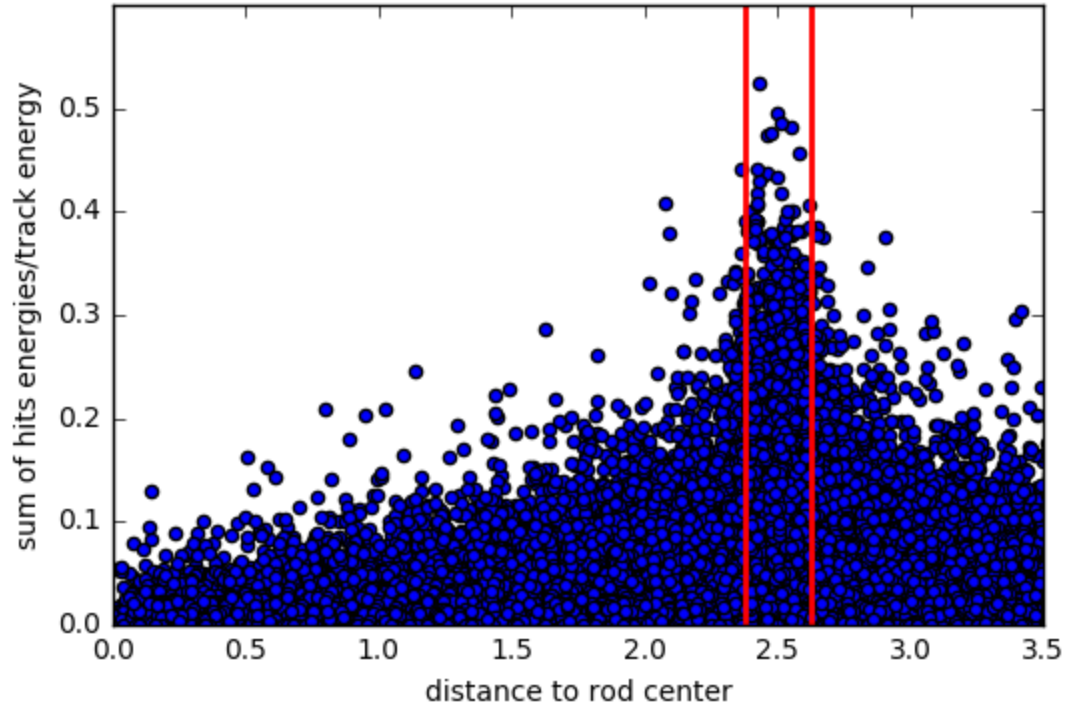
Fig. 1.5: Energy response for electrons in a calorimeter for all electrons in a library
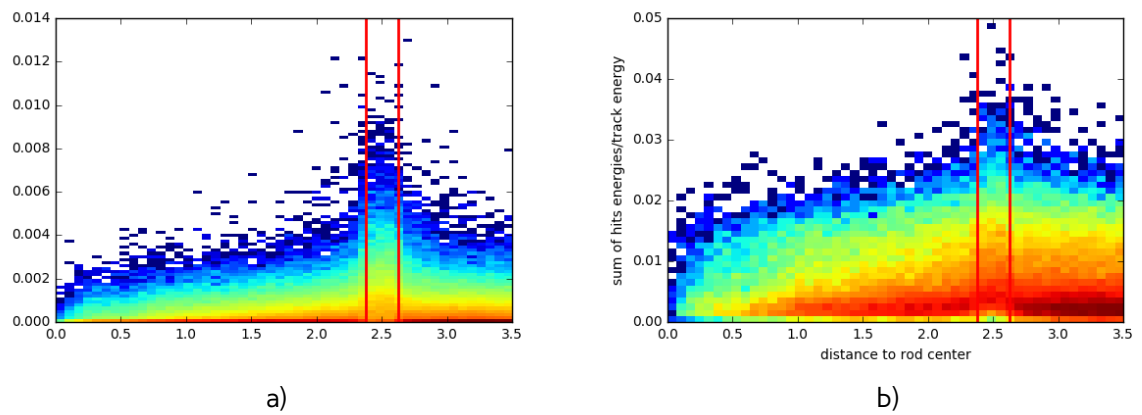

a)

b)

Fig. 1.6: Energy response for electrons in a calorimeter for a) electrons with energy less than 100 MeV
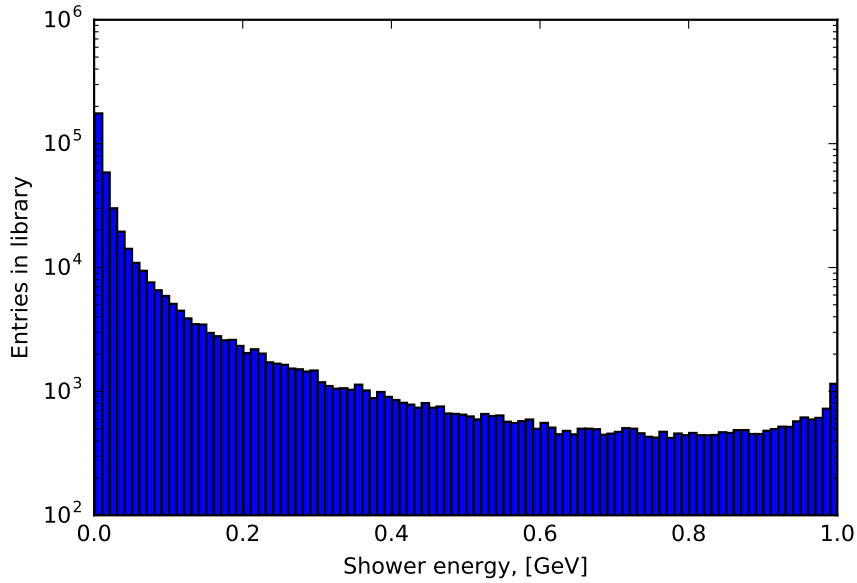b) electrons with energy bigger, than 100 MeV

Fig. 1.7: Distribution of shower energy used in production of 1000 GeV electrons.

are having more energetic showers, than particles in a dead material. Introducing additional spacial binning, that is defined as distance to the closest rod center is allowing to capture this structure. Dependency of summed energy of shower on the distance defined above is shown on a Figure 1.5. Gap is marked by the red lines. Size of this differences between sensitive material and dead material is depending on a initial particle momentia (Figure **??** a and b). For electrons with energy greater than 500 GeV they are almost negligible. This is why upper limit for frozen showers was set to 1000 MeV.

Performance of frozen showers is also depending on a lower limit of a method. Distribution of shower energies, used for production of high-energetic electron (1000 GeV in that case), is shown on a Figure . More than 50% of them are having energy less than 20 MeV. Studies have showed, that Frozen showers are slower, than a standard Geant4 simulation for showers with energy 3 MeV. This is happening due to library non-binned structure for energy. This makes search of closest energy shower in a library slower, than simulation of shower with zero or one hit in a sensitive detector.

Generation of Frozen Showers library

In a Frozen Shower method there are separate libraries for each particle and subdetector used. Showers should cover fully energy and pseudorapidity region and be able to describe data, that is needed during simulation. This is why 2 stages simulation approach have been used. The first stage takes initial particle parameters from a physical processes (ttbar or a single electron).

The first stage is to take initial particle parameters, that later will be used in a library from a physical process. This is done using simulation of some process (e.g. ttbar or single electron). Every time, when particle becomes eligible for Frozen Showers, it parameters are saved in a HepMC format. Particles inside calorimeter tend to cluster tightly around initial track, so random truncation of initial particles is used to obtain better detector coverage. On the second stage, this primary particles are propagated through the calorimeter using standart Atlas simulation infrastructure. Resulted shower

The general frozen showers parameters	
Detectors used	FCAL1, FCAL2, FCAL3
Type of the particle	photons, electrons, neutrons
Energy range	$E_\gamma < 10$ MeV, $E_e < 1000$ MeV, $T_n < 100$ MeV
Containment requirement	$\Delta E_{shower} > 98\%$
The library post-processing parameters	
Generation clustering cutoff	$(\Delta R_{cluster})^2 < 25$ mm
Generation truncation cutoff	$R_{hit}^2 < 50000$ mm, $\Delta E_{shower} < 1\%$

Table 1.1: Main parameters used for the frozen shower libraries in FCAL

parameters are saved in a library. This procedure allows to take into account sampling fluctuations and charge-collection effects on a hit information automatically. Additionally, in order to save disc space as well as a memory consumption, hit information is compressed. This compression is done in a two steps, hit merging and truncation:

- if the distance between any two hits is smaller, than a given parameter R_{min}, then hits are merged into one deposit at the energy weighted center of them. This process is done iterativelly.

- hits whose energies are below the fraction f of the total energy sum of all hits, are truncated. The energy of remaining hits is rescaled back to preserve the total deposited energy.

Unfortunately, for a Frozen Showers, generated for Run-1 monte-carlo, additional tuning of electron libraries was needed. This was done using reconstructed energy of electrons. Frozen Showers tend to underestimate fluctuations of energy loss, that is leading to a smaller electron resolution for a high energies. Correction is done by enlarging bin, corresponding to a gap position. Also, correction of the mean shift is done by scaling energy response of all showers. After this frozen showers are showing good agreement with full simulation. This procedure needs to be done every time, when something is changing in software. Because tuning is done manually, lots of a manpower is needed for each Monte Carlo mass production campaign.

Distance binning problem

As it was mentioned before, process of library generation can be comlicated and take a lot of the time because of the needed tuning. In this subchapter possible ways to improve frozen showers performance have been studied.

As it was mentioned before, that there are two type of material used in a FCAL. Showers within them are giving different response, what is affecting overall reconstructed electron energy resolution. At the first generations distance bin have been corresponding to LAr gap or dead material positions. During tuning bin with LAr was enlarged to gain a better agreement with full simulation. So, one of the basic ideas to improve frozen showers performance is to change a size of LAr gap in a library generation.

It was decided to treat showers, that have been born near LAr gap and crossed it on a radiation length, in a same way with showers in sensitive material gap, and call them sensitive material showers. Oppositely, showers, that haven't crossed LAr gap, are called dead showers. This model leads to a bigger gap width by a definition. One of the possible ways to find this bin position automatically is to use machine learning tools.
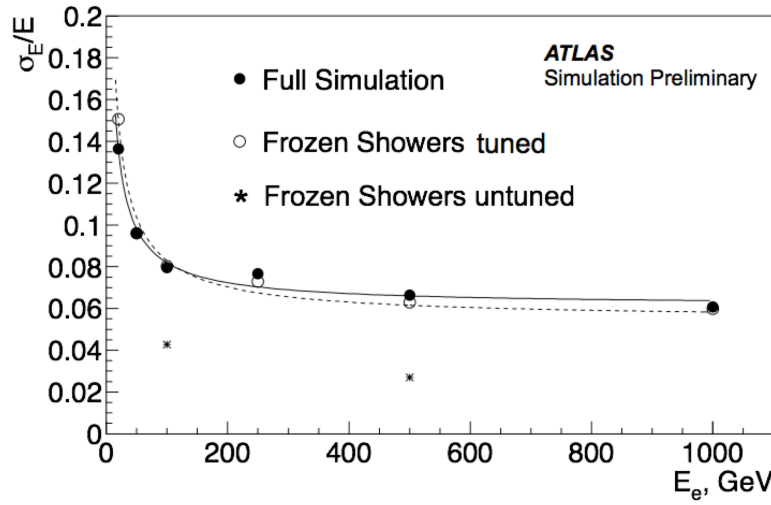
Fig. 1.8: Electron resolution for full simulation, tuned and untuned frozen showers

Machine Learning is a set of algorithms, what allows computers to learn and give a predictions without being specifically programmed. This is a modern field of computer science, that is wildly used in a different fields like computer vision, natural language processing, data science etc. There are two main types of machine learning algorithms: supervised, where example of desired output is given by the "teacher" and the goal is to learn a general rule, that maps inputs to outputs and unsupervised learning, then there are no labels given to algorithm, and algorithms is discovering hidden patterns in data. Initial data parameters of interest, that are used in algorithm to learn are called features. It is important to have right proper set of features and good training sample.

From a geometrical point of view, one of the main parameter is a direction of the shower. Eta momentum distribution is showed on a Figure 1.8 . Most of the showers are collinear to an electron direction. Because of this it was decided to use as a training sample simulation results for electrons with energies less than 1 GeV and momentum uniformly distributed between eta 3.0 and 4.0. This allowing to study equally low and high energy showers equally.

From out definition of 2 classes of showers, it is simple to construct a pre-labelled training sample. This is done by reducing initial sample and taking showers near rod center and inside liquid argon gap. Output of this classifier, that was trained on with sample with shower features, such as energy response and number of hits, than can be used to expand our labels to a full distance range. Than it can be used as an input to a second classifier, which will separate two types of showers using particle parameters, such as energy and distance to a rod center. For a first step decision trees have showed good classification efficiency (around 97%). For a second classifier support vector machines have been used. This method is trying to reconstruct a hyperplane, that is dividing two classes. Outputs of both of this classifiers are shown on a figure . New gap position is determined using borders of hyperplane. This procedure is giving expected from the initial model results. Gap is wider, than and original one. It is also getting bigger with bigger energy, because of the radiation length growth. Validation results for two different eta bins are shown on figure   a) and b). In a bin this new binning is performing better, than original one without any additional tuning. Unfortunately this is not true for all of the bins, as we can see on a figure   b). This eta bin have showed worst performance for a new binning, but it is performing still better, than original binning without tuning.
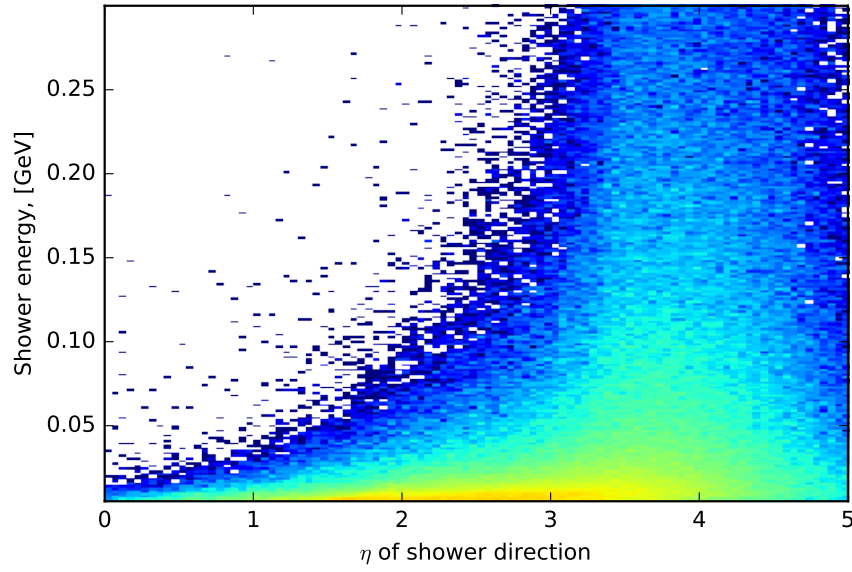
Fig. 1.9: Distribution of showers used in production of 1000 GeV electrons on shower energy vs $\eta_{momentum}$ plane.

This binning was used in a production of new libraries for Monte Carlo in a Run-2. It is planned to use more precise training sample for a future iterations of this procedure for improving performance of outlying eta bins.
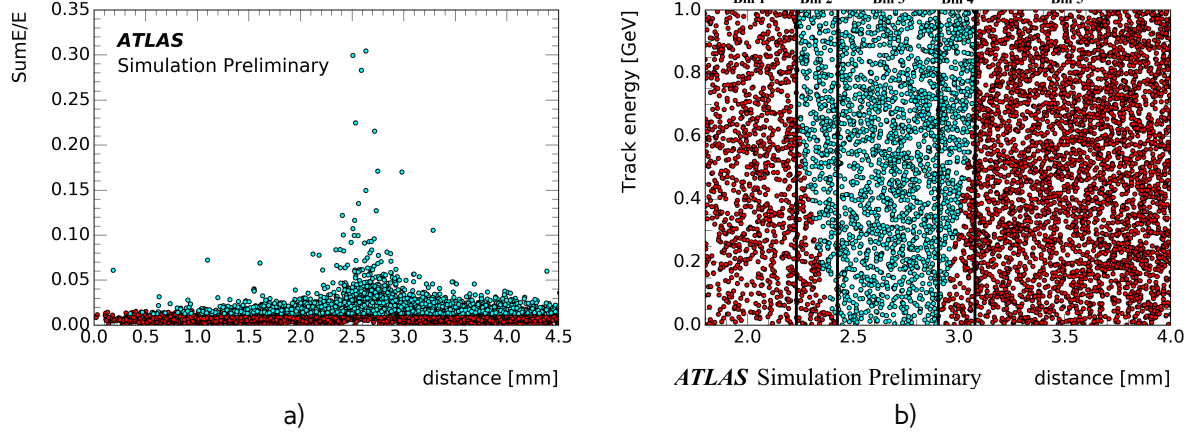
Fig. 1.10: Results of machine learning for a) first classifier b) second classifier. Cyan dots are corresponding to sensitive material showers, red - dead material showers
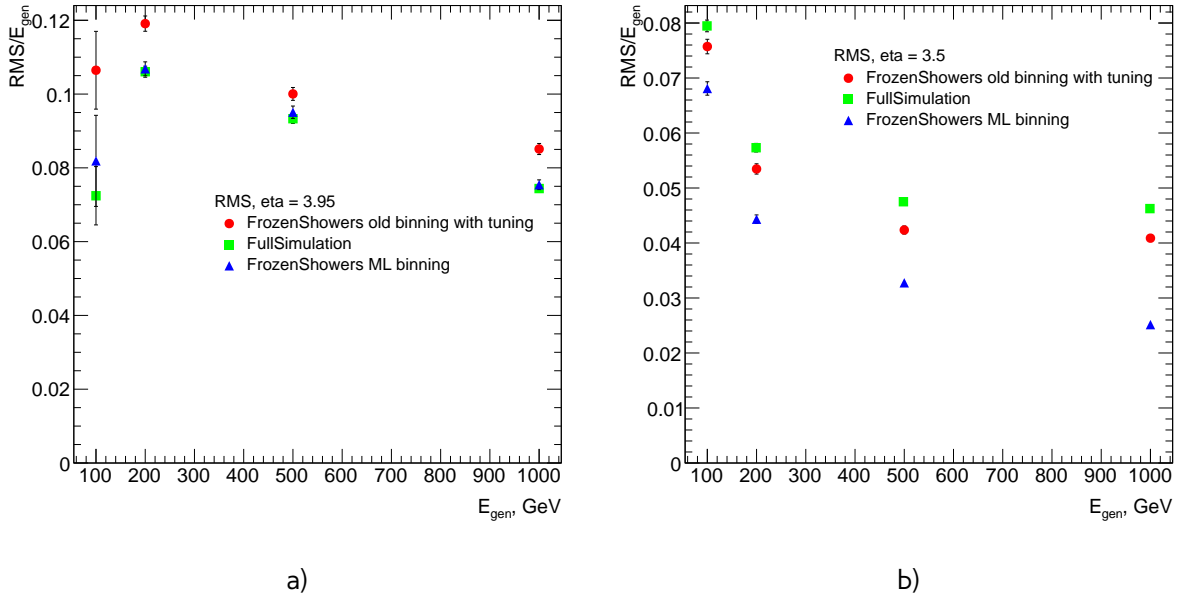

Fig. 1.11: Resolution of reconstructed electrons for full simulation, new libraries with ML binning and old tuned libraries with original binning for a) eta =3.95 b)eta=3.5