The Fast ATLAS Track Simulation (FATRAS)

K. Edmonds, S. Fleischmann, C. Magass
Universität Bonn, Germany
T. Lenz
Bergische Universität Wuppertal, Germany
J. Mechnich
Universität Freiburg, Germany
A. Salzburger*
Leopold Franzens Universität Innsbruck, Austria & CERN

March 5, 2008

Abstract

Various systematic physics and detector performance studies with the ATLAS detector require very large simulated event samples. Since the full detector simulation is a highly CPU time consuming operation, fast simulation techniques are widely used in such applications. Furthermore, the simulation of background events does, in general, not require the very detailed detector simulation and fast simulation techniques satisfy the needed accuracy.

In ATLAS, the fast simulation program ATLFAST has been extensively used for such purposes. It is, however, based on the smearing of the initial particle properties and is not capable of producing hits along the track. Tracking relevant studies that include both hit information and pattern recognition effects can not be performed when using ATLFAST. An alternative simulation program, the new *Fast ATLAS Track Simulation* (FATRAS) has been recently deployed, capable of producing full track information, including hits on track. Initially developed as a validation tool for the ATLAS offline track reconstruction, it has become a powerful engine for various use cases. In general, the CPU time determining factor of the full simulation is the tracking of the particle through the very complex detector geometry, while the event reconstruction including pattern recognition and track fitting is relatively fast. In FATRAS, the simplified reconstruction geometry is used as a simulation geometry model, which leads to a significant speed up of the simulation process. FATRAS uses furthermore mainly common offline track reconstruction code and the reconstruction event data model. It is fully embedded in the ATLAS C++ based software framework ATHENA.

ATLAS NOTE
The ATLAS Experiment, *http://www.atlas.ch*

*Corresponding author: Andreas.Salzburger@cern.ch

1 Introduction

The Monte Carlo simulation of physics events together with the according detector response is an essential technique in high energy physics. In the preparation phase of an experiment this becomes the only source to predict the sensitivity of the detector setup with respect to various event channels, while during data taking it builds the framework to test theoretical models against the real detector response. Furthermore, since most of the readout and reconstruction software is developed in parallel to the detector installation and deployment, simulated data is — besides taken data from test beam setups and commissioning runs using cosmic rays — the only input available for testing and validating the performance of the reconstruction software.

The event simulation process can be divided into two sequential parts: the primary physics event generation is usually carried out by common high energy physics libraries such as PYTHIA [1] or HERWIG [2], while the second step, the simulation of the detector response, is obviously particular to the experimental setup. Latter includes the particularities of the detector geometry and the integrated detection technologies. These components are in the most sophisticated detector simulation — in the following referred to as *full simulation* — realised through a very detailed geometry model and an accurate description of the particle interaction with the sensitive detector material, followed by a realistic clusterisation model. In fast simulation techniques based on parameter smearing, on the other hand, both components are respected intrinsically through the smearing functions that are obtained from full simulation results. These two track simulation techniques have been existing and extensively used in ATLAS: the detailed full detector simulation that is based on the well known Geant4 simulation toolkit [3], and a fast track simulation (as a part of the ATLFAST [4] program) that works on the basis of four momentum vector smearing. In the full detector simulation particles are tracked through a very complex geometry model, and interactions with the sensitive and non-sensitive detector material are simulated. Hits are generated in the sensitive parts and further processed in a digitisation[1] module that prepares the simulated hit information for the reconstruction algorithms. Many different physics processes, such as particle decay or electromagnetic and hadronic interactions of the particle with the detector material are performed; some of these processes produce new particles, which are added to the stack of particles to be processed. This procedure is iterated until the child particle falls underneath a certain energy threshold. The reconstruction software, a so-called *online* application for the event triggering and the *offline* part for the final event reconstruction and analysis, is then executed subsequently to the detector simulation and digitisation. It yields the final track resolutions and reconstruction efficiencies. For convenience, the compound of full detector simulation, digitisation and offline reconstruction will be in the following also referred to as *offline chain*.

The ATLFAST simulation bypasses the trajectory building, hit creation, digitisation and reconstruction by applying a smearing function directly on the kinematic parameters of the generated particle. The smearing approach attempts to represent the track as it is expected to be reconstructed by the offline track reconstruction; this is only valid in a purely statistical manner. The smearing functions are for this purpose obtained from track parameter resolutions that originate from the full simulation and reconstruction chain. Dedicated smearing functions have to be found for different particle types, momentum ranges and vertex radii; the parameter smearing has also to accumulate all aspects of the entire simulation and reconstruction chain (including the detector layout, the material budget, the digitisation and clustering, as well as the tracking performance); they have to be, in principle, updated if any of the involved components changes substantially. Many studies have been in the past performed using the ATLFAST simulation, however, in particular for tracking performance studies it is not suitable since no hit information is available.

Recently, a new *Fast ATLAS Track Simulation* (FATRAS) has been established that realises a full Monte Carlo simulation approach, but makes use of the simplified reconstruction geometry model rather than the simulation geometry. FATRAS is based on common reconstruction tools and uses the common tracking event data model (EDM) [5] natively[2]. FATRAS is capable of tracking the particle through the entire reconstruction geometry (in the following referred to as `TrackingGeometry` [6]), using the ATLAS extrapolation engine [7] for the transport of the track parameters and the inert

[1] The digitisation is not part of the Geant4 simulation toolkit, but carried out by a dedicated module that is integrated into the software framework of the experiment.

[2] Both, the Geant4 simulation and ATLFAST incorporate their own internal event data model that is optimised for their specific needs.

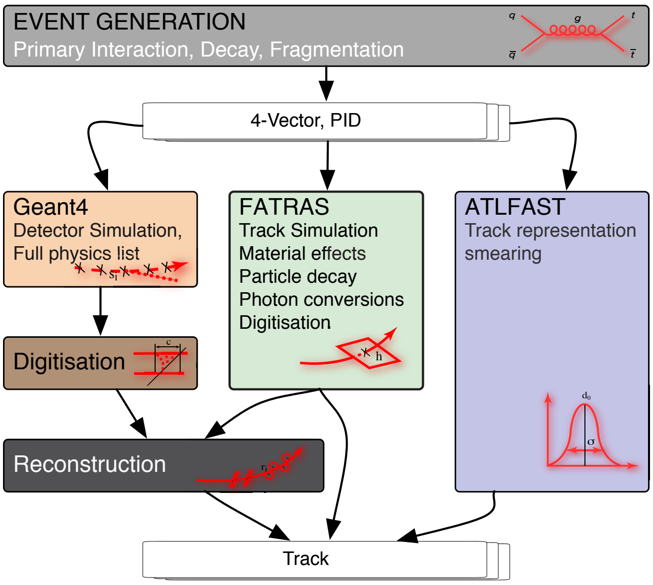


Figure 1: The track simulation techniques used in ATLAS: the event generation provides 4-vector momentum representations together with a particle identification, which builds the input to the different simulation strategies. The very precise Geant4 simulation is based on tracking the particle through the detector and simulating its interaction with the detector material. The produced hits are then fed into a digitisation module for further processing in the event reconstruction. FATRAS follows a similar approach, while using the simplified reconstruction geometry and parameterised models for the interaction of the particle with the detector material. A fast digitisation can be applied, such that the hits can either serve as input to the standard reconstruction or directly be used through the output track object. The ATLFAST simulation is based on the smearing of the generated input, aimed at representing the particle in the stage after the track reconstruction.

navigation schema of the `TrackingGeometry` for the trajectory building. Material effects are applied according to the amount of traversed material and physics processes such as bremsstrahlung, photon conversions and the decay of non-stable particles are supported. Initially developed as a validation tool that has been extensively used during the development of the track reconstruction components [8], it became a powerful tool for broader purpose including a (limited) usage in the simulation of physics events. FATRAS is able to produce hits along a track and enhances track fitting, vertex reconstruction and even the input creation for the standard offline reconstruction chain. It allows large scale tracking, vertex finding and heavy quark tagging studies while guaranteeing low execution times. Quark tagging, however, requires also the inclusion of calorimetric measurements for the jet building process. FATRAS is aimed to be executed in a final configuration together with a dedicated fast shower simulation; a very brief outlook and discussion of the current status of such a combined simulation can be found in Sec. 6.1. Figure 1 presents an overview on the track simulation techniques used in ATLAS, and Fig. 2 shows the same $t\bar{t}$ production event simulated with the full simulation and FATRAS. The tracks shown are those found by the identical offline track reconstruction; the visualisation is done with the ATLAS event display ATLANTIS [9]. FATRAS is currently limited to the ATLAS *Inner Detector* (ID), mainly because the `TrackingGeometry` description of the second tracking device, the *Muon Spectrometer*, is still in a prototype version. A future extension of FATRAS to include also the MS is one challenging part for the further development of the FATRAS project.

This document is based on ATLAS software release 13.2.0 and several attached packages, that are installed on a common group area; since this release marks the first production release that contains the close-to-final FATRAS setup, not every module of FATRAS has been tuned and calibrated to full extent. This work is expected to be integrated for the next major production release 14.0.0. A brief discussion of missing components and planed tuning and calibration work is given in Sec. 6. For the convenience of the reader, an overview of the used packages is given in the appendix A.6.

1.1 Document Structure

This document is structured as follows: Section 2 will give a high level overview of the concepts and modules that build FATRAS, concentrating on the integration of the new fast track simulation into the software framework of the ATLAS experiment and in particular the newly developed track reconstruction chain. Section 3 describes in the following the single modules of FATRAS in more detail, while already giving performance comparisons with the full simulation on the level of material

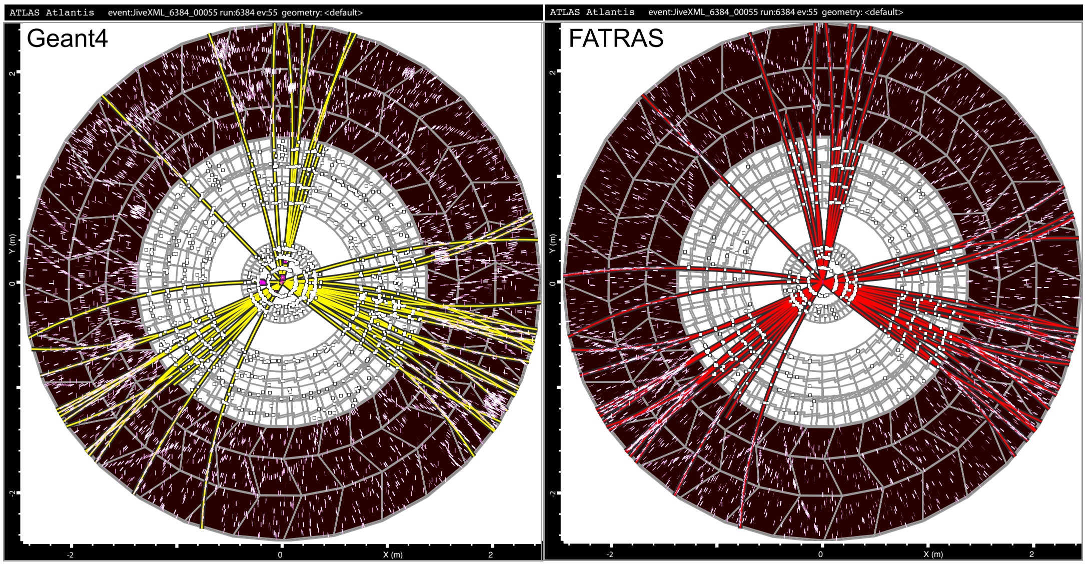

ATLAS Atlantis event:JiveXML_6384_00055 run:6384 ev:55 geometry: <default>
Geant4
Y (m)
X (m)
-2
0
2
ATLAS Atlantis event:JiveXML_6384_00055 run:6384 ev:55 geometry: <default>
FATRAS
Y (m)
X (m)
-2
0
2

Figure 2: An identical hard proton-proton scattering event leading to the production of a $t\bar{t}$ pair simulated with the full detector simulation and the FATRAS fast simulation in the ATLAS Inner Detector. In both cases, the standard ATLAS offline track reconstruction is performed, the found tracks are also illustrated.

effects integration and hit creation. Section 4 focusses on the comparison of FATRAS with the full and ATLFAST simulation mainly on single track level and includes a comparison of CPU time for the different simulation strategies, while Sec. 5 shows the usage of the new fast track simulation in some dedicated applications. Section 6 will conclude this document, but will also give an outlook on planned future expansions or modifications to the FATRAS project. The Appendix covers conventions and typesetting used within this document and gives a small introduction to the FATRAS usage. It also summarises the involved software packages and available tuning parameters for the convenience of the reader.

2 Concepts and Modules of FATRAS

FATRAS is a full Monte Carlo based track simulation that makes use of the reconstruction geometry and tools from the offline track reconstruction. It re-uses to a large extent modules and resources of the offline track reconstruction, while only few dedicated components replace standard offline algorithms and tools. The main benefits of the chosen approach — besides the pure performance aspects — can be summarised as follows:

- **maximum compatibility with the full offline chain** to guarantee client/analysis code to run independently of the chosen simulation strategy; additionally, this enhances FATRAS as a fast development alternative for future analyses, before the final analysis can be performed on fully simulated or taken data;
- **automatic adaption to changed detector conditions** through the `TrackingGeometry` and the used reconstruction modules;
- **easy expansion and modification** through the component model.

The following section will give a brief overview of the high level modules used in FATRAS, while a detailed description of the single components can be found in Sec. 3.

2.1 The ATHENA Framework and the New Tracking Approach

The new fast track simulation is fully embedded in the ATLAS C++ software framework ATHENA [10]. ATHENA is realised as a data centered software framework that follows a strict component pattern design and provides interfaces for modules at different levels of the program flow. The use of common interfaces and modules is hereby essential for the FATRAS design, since it allows the exchange of single offline reconstruction tools with modified Monte Carlo based versions that comply with the same interface definition. Many modules that are used in the FATRAS application are directly taken from the new ATLAS track reconstruction, the so-called *New Tracking* (NEWT) [11]. In particular, the tracking event data model (EDM), the reconstruction geometry description (which is used as the simulation geometry in FATRAS), and the extrapolation engine are essential parts of the FATRAS simulation. A description of these modules would go far beyond the scope of this document; the reader is, however, encouraged to find detailed documentation in [5], [6] and [7], respectively.

2.2 Module Sequence and Data Flow

The default FATRAS simulation sequence consists of six different modules, each of which realised as an ATHENA `Algorithm` class. Figure 3 illustrates the `Algorithm` sequence executed in FATRAS by a simplified UML activity diagram.

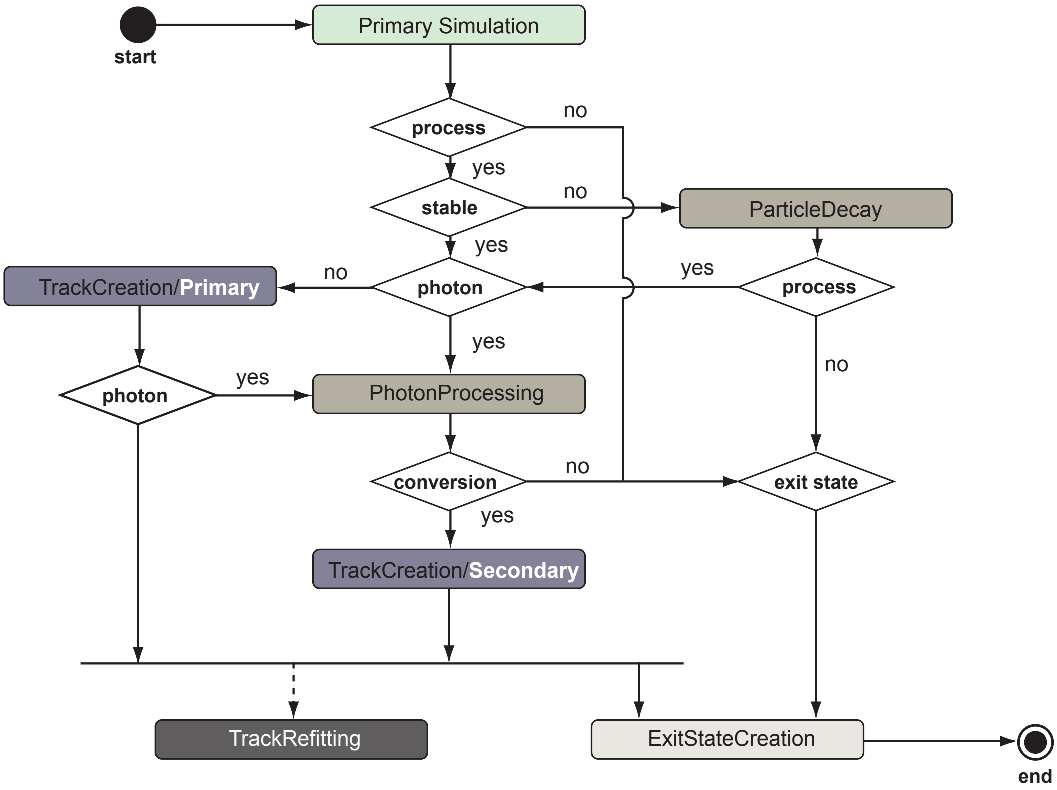


Figure 3: Simplified UML Activity diagram showing the six different modules that build the FATRAS simulation. This illustrated configuration includes only one iteration of photon processing, while this has been recently expanded with a second photon processing, particle decay and a third track creation module to further track lower energetic particles that are created in decay or conversion cascades. The component software model that has been followed throughout the FATRAS design eases this modified configuration by simple adaptions of the job configuration.

These `Algorithm` classes build the simulation chain of FATRAS and are in the simplest configuration executed in the sequence as described below. However, the component pattern design and in particular the usage of a central data store instead of direct dependencies between the acting modules, allows to modify and extend the algorithmic sequence in a flexible way, such as e.g. the insertion of an additional iteration on conversion or decay products. Latter is necessary to correctly handle secondary particles that are induced by hadronic shower interactions.

The following list presents a brief overview of the different FATRAS modules; a detailed description can be found in Sec. 3.1 to Sec. 3.7 of this document:

- **Primary Simulation:** the primary simulation module is realised either as the simple `SingleTrackSimulation` or the `GenEventSimulation Algorithm`. The `SingleTrackSimulation` provides simulated single track events mainly targeted at validating the fast track simulation itself without the — in this respect — unnecessary overhead of event generation through a dedicated generator module. It also creates a fast and convenient framework for the validation of the offline reconstruction chain. The `GenEventSimulation Algorithm`, on the other hand, is designed to process the input provided by event generators. The input collection is hereby sorted by *prompt* tracks, decaying particles, photons and non-interacting particles that are immediately scheduled to be transported to the tracker exit.

- **Particle Decay:** particles that are not flagged as stable by the generator are in FATRAS filtered into a dedicated container by the primary simulation `Algorithm`. The existence of a non-empty container of decaying particles triggers the execution of the `ParticleDecay` module. The stable decay products are added to the collection of particles that are scheduled for the (primary) track creation, photons are filled into a dedicated collection and particles that do not interact with the detector are scheduled for exiting the detector volume. Following the common FATRAS design most actions are outsourced to `AlgTool` entities, while the `Algorithm` class usually just delegates; the actual decay is performed by a dedicated `ParticleDecayer`, that exists in two flavors: a simplified version that only supports a limited number of decay channels, and a sophisticated wrapper of the particle decay modules that part of the Geant4 simulation.

- **Track Creation:** the track creation `Algorithm` is the core of FATRAS; the track creation can occur several times in the FATRAS simulation sequence to allow for an iterative treatment of secondary particles induced by hadronic shower reactions or photon conversions.

- **Photon Processing:** photons from the Monte Carlo generator (i.e. final state radiation) as well as hard photons originating from the transport of electrons through the detector are further handled by the dedicated `PhotonProcessing Algorithm`. They are extrapolated through the detector while — depending on the traversed material — the conversion probability is calculated and pair production is performed. Tracks originating from photon conversions are created by the secondary track creation and may again lead to hard photon emission. The photons are then, depending on the chosen configuration, either integrated into yet another iteration of the electron-bremsstrahlung cascade, or directly scheduled for a transport into the calorimeter.

- **Track Refitting:** the refit of the simulated track is essential for FATRAS to gain comparable track resolutions with tracks found in the standard offline reconstruction — when FATRAS is performed in the *refit* mode. The initial track parameters used for simulating the track are hereby smeared before refitting to remove the bias that is given by seeding the track fit with the true initial track parameters. Omitting this smearing step would lead to artificially narrow cluster residuals on the first detection layer[3]. Any track fitter that implements the `ITrackFitter` interface from the New Tracking realm can be used to perform the track fit. Tracks that originate from photon conversions (and are thus produced in the second track creation step) do not necessarily have to be refitted. The main focus is hereby drawn on the appropriate description of these particles at the calorimeter entrance for the shower simulation in the electromagnetic calorimeter. Hits that have been created by particles that originate from photon conversions are indeed added to the event hit collection, such that they are correctly included when FATRAS is used to feed the offline pattern recognition.

- **Exit State Creation:** the last step in the FATRAS sequence, the so-called `ExitStateCreation`, has no direct implication on the FATRAS performance itself, but prepares the input for follow up algorithms such as e.g. fast shower parameterisations of the calorimeter. Simulated tracks, neutral particles and photons that did not lead to conversions in the detector volume are extrapolated to the exit surfaces of the tracker volume.

[3] The smearing has to be kept in the linear regime of the underlying track model and is very similar to finding an appropriate track seed for the fit in the the standard offline reconstruction.

- **Post Processing:** the hit post-processing module is somehow independent from the main algorithmic sequence in FATRAS, but necessary to prepare the hit collections for the standard offline reconstruction chain. The simulated hits originating from FATRAS tracks and additionally created noise hits are filled into the dedicated hit collections that are used in the ATLAS offline reconstruction.

FATRAS incorporates the data centered blackboard architecture design of the ATHENA framework: the different `Algorithm` classes do not have any relation other than that they all read and write to the same transient event store (TES). In many cases the existence of a collection in TES indicates whether an algorithm has to run or not.

2.3 The FATRAS Event Data Model

The FATRAS event data model is to a large extent identical with the tracking EDM of the ATLAS offline reconstruction. Non-calibrated and calibrated measurements are expressed through the according offline EDM classes; the use of the extrapolation engine for the trajectory creation and the common track fitting tools ensures that the created track objects are identical to those from the offline reconstruction. This aspect has several advantages for the further event analysis; many standard validation tools, but also the ATLAS event display applications, such as ATLANTIS or VP1 [12] can directly work on FATRAS output. Only the top object collections in FATRAS are integrated in a special EDM schema, based on a generic `ParticleState` base class. This is necessary for the intrinsic truth steering in the simulation process.

EDM Extensions The polymorphic structure of the ATLAS tracking EDM allows to create generic cluster objects and use them together with the FATRAS simulation. This is in particular interesting when using FATRAS in design studies for future upgrade scenarios of the ATLAS tracker, see Sec. 5.2. A generic silicon cluster class that inherits from the common ATLAS `PrepRawData` base class and the associated calibrated version that fulfills the `RIO_OnTrack` interface can be found in the FatrasEvent package. The integration of these custom classes as direct extensions of the ATLAS tracking EDM enhances the use of common ATLAS tracking tools, such as track and vertex fitters on tracks that originate from modified detector setups — without any intervention on the standard ATLAS reconstruction chain[4]. The full extension of FATRAS to satisfy the needs of tracker upgrade studies requires also an updated detector model. This is facilitated by the design of the `TrackingGeometry` that provides generic geometry classes that are independent from any given detector technology. In FATRAS, a flexible generic geometry builder has been deployed that enhances the creation of different geometrical configurations and associated clusterisation models.

Truth Association The association of the FATRAS tracks (both, from the refit mode and the reconstruction mode) with according truth objects from the event generation and hit simulation is necessary for many validation studies. The ATLAS tracking EDM, which builds the base EDM for FATRAS, does not allow direct links between reconstruction objects and generated truth information. Thus, associative relationships have to be established to bind the simulated track to the truth information. In FATRAS — since the truth information is *a priori* known — this is enhanced by the `ParticleState` class which is used for the main FATRAS containers that are written to the transient event store. `TruthAssociation` objects can be registered to the `ParticleState` and thus a direct relation between the simulated EDM objects and the generated particle is established. When refitting tracks in FATRAS, an associative container (realised as an STL map) is created and recorded in the transient event store to keep the truth information available on this level.

There exist several dedicated truth association `Algorithm` classes that have been omitted in Sec. 2.2, where the general `Algorithm` sequence of FATRAS has been presented. This is, because these components are not part of the FATRAS simulation itself, but serve pure validation purpose. A more detailed description of these modules and their functionality can be found in Sec. 4 of this document.

[4] It is worth mentioning that this is a striking argument for the component software pattern and the polymorphic data model design that has been realised through the new track reconstruction software.

Recently, the full Monte Carlo truth tree that uses the standard `HepMC` [13] class descriptions has been deployed and concluded the effort to turn FATRAS into a look-alike pendent to the full simulation chain. The full `HepMC` truth tree has not only been necessary for the correct integration of FATRAS with a successive calorimeter simulation, but also enhanced standard validation algorithms that are widely used on offline data to cope with FATRAS simulated events.

2.4 Modes and Reconstruction Feeding

One major achievement of the FATRAS simulation is that it can serve as an input for the offline track reconstruction. This is possible, since FATRAS uses the offline tracking EDM and produces `PrepRawData` objects in the track creation process, which mark the input objects of the offline pattern recognition. A dedicated `PostProcessing` module exists that strips the created `PrepRawData` objects from the track and fills them into the standard hit container that build the input to the `SpacePointFormation`, the first module in the ATLAS New Tracking. Hereby, the noise level can be adjusted freely, which turns FATRAS into a useful application for occupancy studies and pattern recognition validation with different noise levels, see Sec. 5.1. While for a fast parameterisation of the Inner Detector targeted at providing fast but accurate input for a more detailed calorimeter shower simulation the pure track simulation is sufficient, any tracking based study has to at least use the refitted FATRAS tracks and — if pattern recognition effects are of interest for the according analysis — finally the reconstructed tracks. These different modes will be in the following referred to as *simulation*, *refit* and *reconstruction mode*, respectively[5].

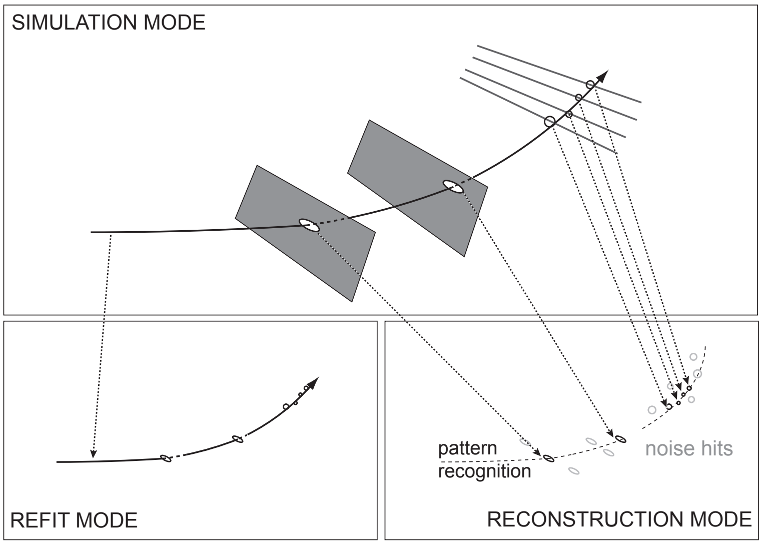


Figure 4: Simplified illustration of the three different FATRAS modes: the *simulation* mode creates only the truth tracks that are then further processed in the *refit* and *reconstruction* mode, respectively. For the *refit* mode, the simulated track is as a whole passed to a track fitting module, while for the *reconstruction* mode, only the simulated hits are — together with noise hits — filled into the standard hit collections, such that the standard track reconstruction can be performed; latter includes pattern recognition and track fitting.

3 The FATRAS Simulation Modules

In Sec. 2 of this document, a brief overview of the FATRAS concepts and modules has been presented, including a description of the main algorithmic sequence that builds the simulation chain. The following sections — Sec. 3.1 to Sec. 3.7 — will describe the single modules in more detail and will bring each of them in context with the full detector simulation.

3.1 Primary Simulation

The primary simulation is the first module in the FATRAS `Algorithm` chain. It is realised by one of two different `Algorithm` classes that can be chosen in the job configuration:

[5] The default keys to be used for retrieving these collections from the transient event store are for convenience called *FatrasTracks*, *RefittedFatrasTracks* and *ReconstructedFatrasTracks*, respectively.

- The `SingleTrackSimulation` has been mainly designed for the validation of both FATRAS and the offline reconstruction software. It offers the possibility to simulate user defined single track events with different particle types and kinematic input variables. In general, random based input parameters are generated, but the `SingleTrackSimulation` can also be executed in a scan mode and a fully configured one-event basis for debugging single tracks that show unexpected or faulty behavior in the offline reconstruction or the track fitting modules.

- The `GenEventSimulation Algorithm` interfaces the output from any given Monte Carlo generator with the fast track simulation and thus enhances FATRAS to be used in more complex physics event studies. The generated particles are hereby sorted into stable particles, decaying particles, photons or neutral particles — each type filled into a dedicated collection for the further processing in follow-up modules of the fast track simulation. Table 1 summarises the distinguished particle types and their further processing module in FATRAS.

Table 1: Particle processing in FATRAS; currently only a limited number of particle decays are supported that are mainly related to tracking relevant studies. All particles are finally processed in the `ExitStateCreation` that prepares the Monte Carlo input data for a final representation at the Inner Detector exit surface for further processing in a successive calorimeter simulation.

Particle (e.g.)	**Comments**	**FATRAS Container**	**Processing**
$\mu^{\pm}$,e$^{\pm}$,p$^{\pm}$	stable final state	TrackStates	track creation
γ	final state, or bremsstrahlung	PhotonStates	photon processing
$\pi^0, \pi^{\pm}, K_{L,S}, K^0$	non-stable	DecayingStates	particle decay
n	stable, non-interaction	ExitingStates	exit state creation

3.2 Particle Decay

In general, most of the relevant particles decays which are contributing to the final state signature of an ATLAS event happen inside the beam pipe volume and are determined by the event generator. However, a non-negligible amount of mesons will produce a decay vertex inside the ID and can therefore be a source for lepton tracks and jet fakes. These decays have to be handled by detector simulation frameworks as they give rise to additional tracks in the ATLAS Inner Detector and energy deposition in the electromagnetic calorimeter.

Taking account of this, FATRAS provides two algorithms with dedicated `AlgTools` to take care of particle decays after primary simulation (only at the time of writing). They both share the main principles of how the lifetime and therefore the path length of the decaying particle's trajectory are calculated.

Based on the lifetime τ of the particle, the decay length λ is simulated by throwing a uniformly distributed random number $\xi \in [0, 1)$, such that — when using a simple transformation method —

$$\lambda = c \cdot (\beta\gamma) \cdot (-\log \xi), \tag{1}$$

where c denotes the speed of light and $\beta\gamma$ yields the boost back into the lab system.

The trajectory of a charged particle is approximated by a helix to retrieve the decay vertex position. If it is outside the inner detector volume, the particle is added to the collection to be processed by the primary track creation algorithm later on. Neutral particles are presumed to take a non-bended path through the detector. In case they are long-lived enough to decay outside the tracker, they are handed over to the exit state creation.

Further processing of the remaining particles is performed by dedicated `AlgTools` implementing the `IParticleDecay` interface. This is where the actual decay happens and its kinematics are determined. Details are discussed in the context of the algorithms using them. All tracks are created by an `AlgTool` implementing the `ITrackCreator` interface. It is used to extrapolate the trajectory to the decay vertex and to create a `Trk::Track` from simulated detector measurements.

3.2.1 ParticleDecay

The ParticleDecay Algorithm is to be found inside the FatrasAlgs package. Only a limited number of particles and decay channels are supported, focussing on situations that are important for tracking studies (i.e. π_0, $K^{\pm}$ and $K^0_{S/L}$). The ParticleDecayCreator AlgTool residing inside the FatrasTools package is handling the decays of the already mentioned particles (see Sec. A.2 for details).

3.2.2 G4ParticleDecay

The G4ParticleDecay Algorithm is located inside the FatrasG4Algs package. It provides an interface to the Geant4 particle decay facilities which by default are able to handle all particles of the standard model. The PDGToG4Particle AlgTool is in charge of storing the information such as the mean lifetime, charge and branching ratios of decay channels which are also used by the G4ParticleDecayCreator AlgTool for calculating decay channel and kinematics.

3.3 Track Creation

The track creation in FATRAS in done in two steps: the first one marks the trajectory building and is carried out by the extrapolation engine together with the underlying reconstruction geometry. The ATLAS TrackingGeometry is characterised by a fully connective navigation model using neighboring volumes that are attached at shared surfaces. The confining surfaces of the TrackingVolume class (the main components of the TrackingGeometry) extend hereby the common surface description that is the foundation of the ATLAS event data model and can therefore be natively used with the extrapolation engine. In such a way, the trajectory can be followed through the detector, since every boundary surface — when being intersected — leads directly to the next detector volume traversed by the particle. The various volumes contain layer objects that carry a material description and/or a sub-array of sensitive detection surfaces. A simple binning scheme links the intersection with a layer to the associated detector element and consequently, the trajectory of hits can be built by intersecting one layer after the other — always guided from one volume to the next volume by the internal navigation tree. Material effects, such as ionisation loss, radiation loss or multiple coulomb scattering are applied during the trajectory building. This is enhanced by simply exchanging the stochastic material effects treatment as used in the track reconstruction with Monte Carlo based modules[6]. A detailed description of the material effects integration can be found in the following section, Sec. 3.3.1.

The second part of the track creation is the conversion of the given trajectory into a meaningful track object. This involves cluster creation on the one hand and applying efficiency tuning on the other hand. The FATRAS clusterisation model is further described in Sec. 3.3.2.

3.3.1 Material Effects Integration

The simulation of interactions between the traversing particle and the detector material is essential for any track simulation engine. This could be done on several complexity levels which quickly lead to a high calculation complexity[7]. As a general rule, a cumulative simulation is always more accurate than a fully parametric one, while the single step intervals in which the simulation is performed regulate both, the accuracy and the the CPU time. For a fast track simulation it is thus of particular interest to find a good compromise between a reasonable accurate description and the time spent for simulating it. In FATRAS, the material effects integration has been optimised to be coherent with structures in main detector components (such as silicon layers, support structures). Within such a component, the applied corrections to the trajectory are done as a single action. In the reconstruction geometry all of these detector components are described as layer objects with according material descriptions, this update mechanism is thus often called *layer-based* or *point-like*.

[6] The exchange of single modules in the program flow is facilitated by the component software model of the new offline track reconstruction.

[7] The most realistic description of these effects would require the simulation of single atom interactions with the particle and can not be carried out in any high energy physics simulation engine.

Figure 5 shows a distribution of typical path lengths expressed in terms of radiation length X_0 for the ATLAS Inner Detector and shows a comparison of the overall material distribution for the simulation geometry based on Geant4 and the `TrackingGeometry` that is used in FATRAS.

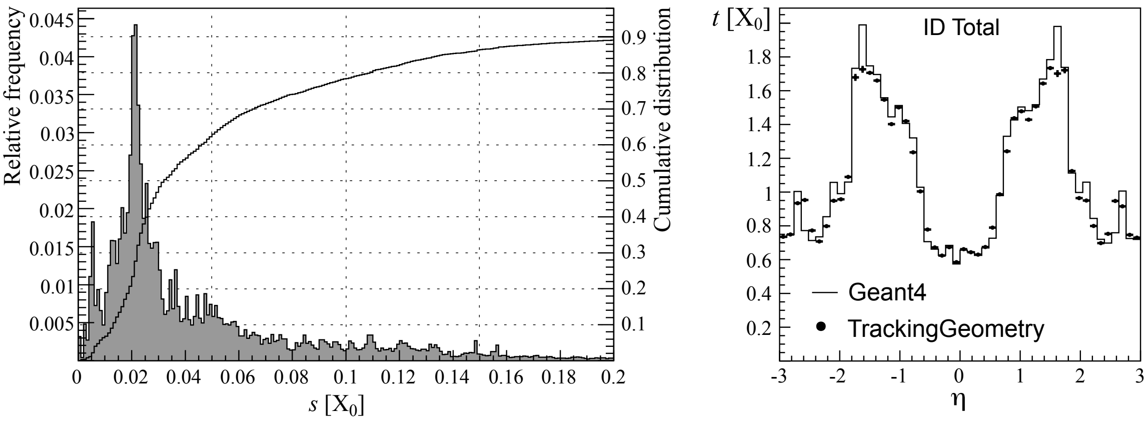


Figure 5: Left: Relative frequency and cumulative distribution of the single step lengths when traversing the Inner Detector reconstruction geometry. About 80 % of all steps through the detector material are described by less than 10 % of radiation length — a value that reaches almost 100 % for the barrel section. Steps that account for more than 10 % of the radiation length are mostly due a large incident angle of the particle in respect to the crossed layer or due to support structures in the Inner Detector. Right: Comparison of the material budget described by the Geant4 simulation geometry to the `TrackingGeometry` description in terms of total path length in units of radiation length.

Simulation of Multiple Scattering When a particle traverses detector material it undergoes multiple small angle deflections caused by the Coulomb force of the nuclei of the detector material. Following the central limit theorem, the distribution of the sum of these small deflections — the multiple scattering process — can be approximated with a Gaussian probability density function (PDF). However, single large angle scattering processes disturb the purely Gaussian character of the scattering distribution. In FATRAS, multiple scattering can be applied in two ways that are based on the two methods provided by the `MultipleScatteringUpdator` that is part of the extrapolation package: the first possibility is a purely Gaussian approximation expressed by the Highland [14] formula

$$\sigma_{ms}^{proj} = \frac{13.6\mathrm{MeV}}{\beta c p} \sqrt{t/\mathrm{X}_0}\,[1 + 0.038 \ln (t/\mathrm{X}_0\,)], \tag{2}$$

where t denotes the traversed path length and X_0 the radiation length of the material. For many applications — and in particular for the usage of a fast track simulation to validate the reconstruction software — it is of particular interest to simulate tail effects. FATRAS offers therefore a second model for the integration of multiple scattering in the trajectory building process that is able to simulate parts of the tail distribution. This is done by using a Gaussian mixture model as proposed in [15]. The root mean square of the projected scattering angle is hereby modeled by a Gaussian core distribution with a similar σ_{core} as given by the Highland formula and an additional broader Gaussian distribution to approximate the tail contribution. The PDF can be then written as

$$f(\theta_{ms}) = (1-\epsilon)\cdot g_0(\theta_{ms};0,\sigma_{core}) + \epsilon \cdot g_0(\theta_{ms};0,\sigma_{tail}), \tag{3}$$

with $g_0(x;\mu,\sigma)$ denoting a Gaussian random number distribution of width σ and mean μ. The model parameters ϵ, σ_{core} and σ_{tail} are hereby taken from the given reference. Recently [16] more elaborated models replacing the Gaussian tail description by a non-Gaussian distribution have been developed. The modular design of the ATLAS extrapolation engine hereby facilitates a future inclusion of such refined models by simply exchanging the according simulation component.

Figure 6 shows a comparison of both the Highland approach and the Gaussian mixture model for various particles and materials with the well known Geant4 simulation.

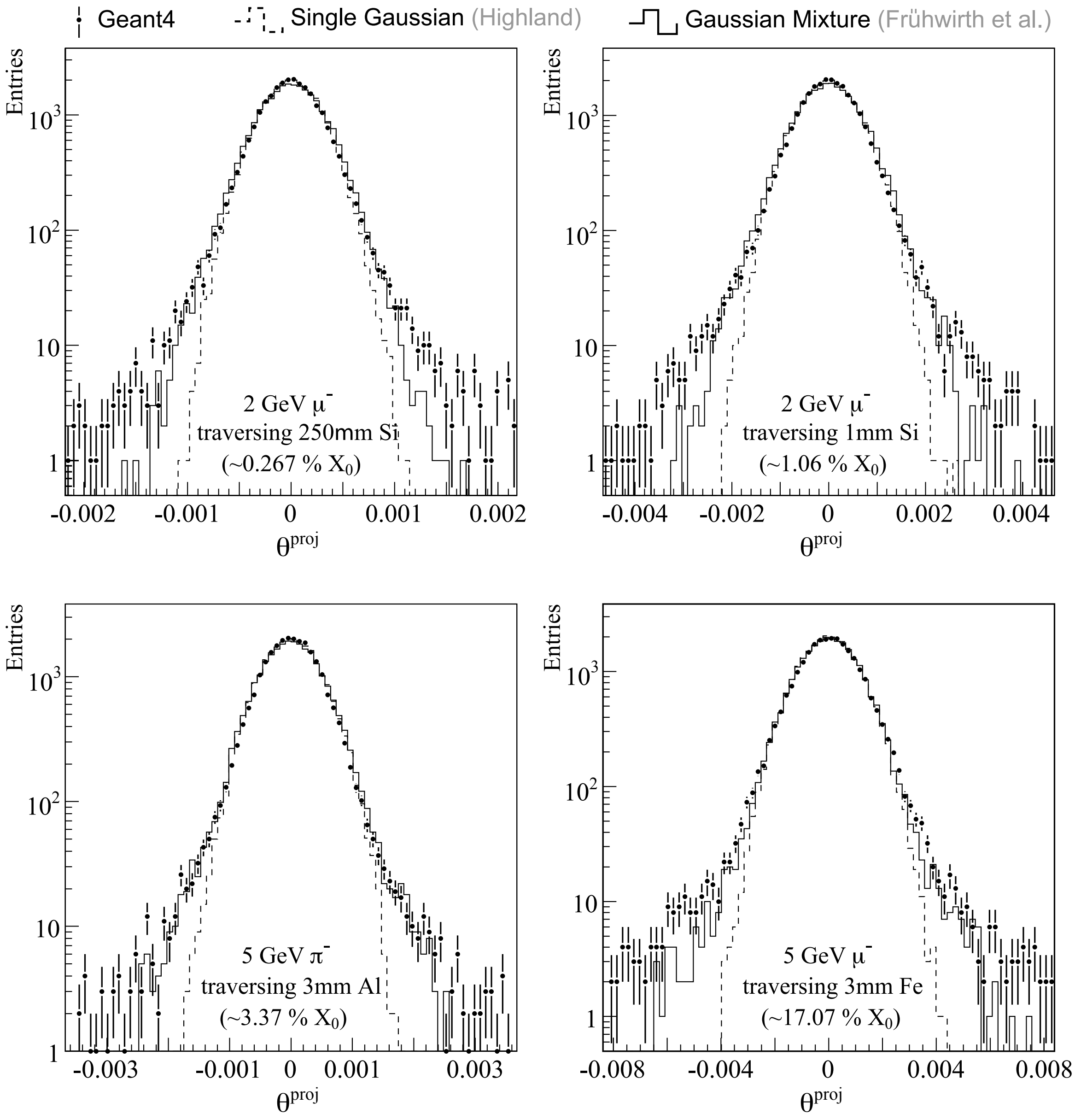


Figure 6: Comparison of the distribution of the projected scattering angle θ^{proj} for the different models available in FATRAS to the Geant4 simulation that builds the base of the ATLAS full detector simulation. Both the single Gaussian distribution following the Highland formula and the Gaussian mixture model are shown for different particle momenta, materials and layer thickness.

Both, the Highland formula and the Gaussian mixture model only provide an estimation of the projected scattering angle, the deflection of the particle due to multiple scattering requires, however, a three-dimensional modeling. The full deflection is in FATRAS simulated by taking a random number α from a uniformly distributed range $\alpha \in [0, 2\pi)$ as a deflection α in the polar direction.

Energy Loss Simulation The energy loss of particles that traverse detector material is mainly dominated by electromagnetic processes. For particles heavier than an electron, energy loss due to ionisation is the by far dominating process. It is well described by the theory of Landau [17], and defines an asymmetric probability density $\rho(\Delta)$ function around a narrow peak through an integral equation

$$\rho(\Delta) = \frac{1}{2i\pi} \int_{c-i\infty}^{c+i\infty} \exp(s \log s + \Delta E \cdot s) ds \tag{4}$$

It can be shown that this leads to a most probable energy loss

$$_L\Delta_p = \xi \left[\ln \frac{2mc^2\beta^2\gamma^2}{I} + \ln \frac{\xi}{I} + 0.2 - \beta^2 - \delta \right], \tag{5}$$

with $\xi = ZN_a \frac{k}{\beta^2} t$, when t denotes the thickness of the traversed material. Additionally, when investigating the asymptotic behavior for $\gamma \gg 1$ (i.e. $\beta \approx 1$), which is well met by all particles of interest in this context, the density correction δ can be modeled as

$$\delta \approx 4.447 - \ln \gamma^2. \tag{6}$$

This simplifies Eq. (5) to

$$_L\Delta_p = \xi \left[\ln \frac{2mc^2\gamma^2}{I} + \ln \frac{\xi}{I} - 0.8 + 4.447 \right]. \tag{7}$$

No theoretical model exists so far to determine the width of the Landau distribution analytically. In general, it is derived from cumulative sampling which would be far beyond the scope of a fast track simulation. In FATRAS, the energy loss of heavy particles is sampled as a Monte Carlo based Landau distribution using the most probable value $_L\Delta_p$ as given in Eq. (7) and a parameterised width of the distribution that has been obtained using the Geant4 simulation toolkit. Good agreement between the energy loss distribution between FATRAS compared to Geant4 could be achieved. Figure 7 shows a comparison of the energy loss distributions as given by Geant4 and FATRAS for heavy particles in different materials and varying momenta.

When traversing detector material, especially electrons lose in addition to the ionisation process a significant amount of energy due to radiation loss (*bremsstrahlung*). The resulting distribution is in general a mixture between a landau distribution due to ionisation loss and a highly asymmetric radiative addition that originates from a long tail towards high energy loss. The according theory for the bremsstrahlung loss has been developed by Bethe and Heitler [18] and is only briefly described in the following. May z in the following denote the ratio between the final energy E_f and the initial energy E_i and t describe the thickness of the traversed material in terms of radiation length X_0 . The PDF for the fraction factor $z \in (0, 1)$ can then be written as

$$\rho(z) = \frac{[-\ln z]^{c-1}}{\Gamma(c)}, \tag{8}$$

where c denotes $c = t/\ln 2$. The factor z is sampled in FATRAS, which is carried out by a dedicated energy loss module in the extrapolation engine. Again, this is only possible since the extrapolation engine follows a strict component software pattern. Two different strategies can be chosen: the default implementation is a Monte Carlo based sampling of the Bethe-Heitler distribution as given in Eq. (8), while the `GSFPDF AlgTool`, that also implements the `IEnergyLossUpdator` interface models the Bethe-Heitler distribution as a sum of single Gaussian distributions, each of which weighted through a given probability. It has been developed for validating the *Gaussian Sum Filter* (GSF) [19].

Photon Emission Significant loss of energy due to bremsstrahlung causes another aspect to be tackled: the emitted high energetic photon has to be tracked through the detector volume, since it can influence the event morphology in several ways. When interacting with the detector material, this can result in leptonic pair creation (mainly electron-positron) — in the following also referred to as *photon conversion* — and thus lead to additional tracks in the detector volume. If no conversion takes place, the additional photons still lead to different cluster signatures in the calorimeter. The dedicated handling of these photons, covering both effects — the conversion and the transport to the calorimeter volume — is in detail described in Sec. 3.4 of this document.

The FATRAS bremsstrahlung model calculates the emission of hard photons and their respective angle to the initial electron. The photon energy corresponds hereby to the energy loss fraction according to the Bethe-Heitler theory as given in Eq. (8). It can be assumed that the angle of emitted photon w.r.t. the parental electron direction is proportional to m_e/E_e; electrons that are matter of track reconstruction in the ATLAS detector have typical momentum values that are significantly higher than

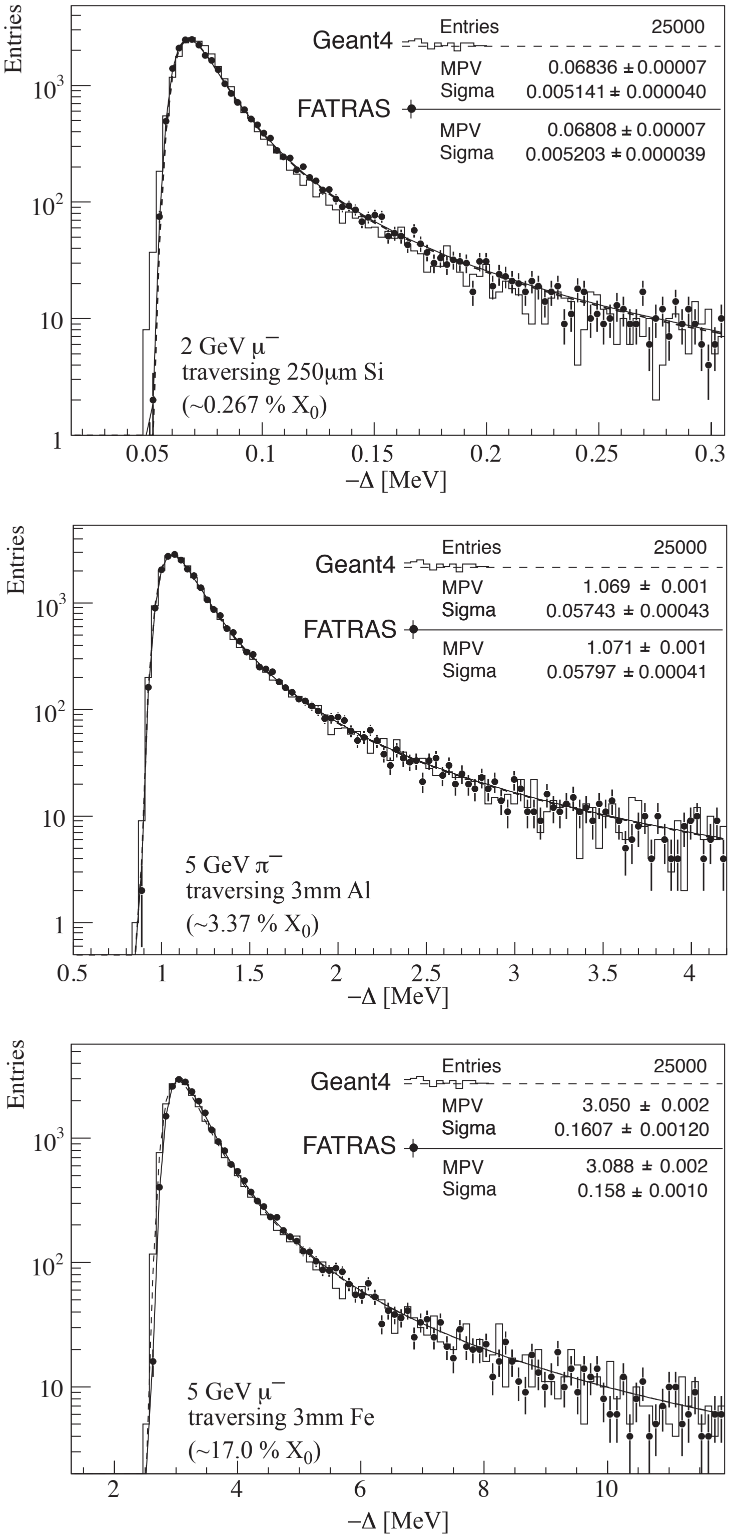


Figure 7: Energy loss distributions of heavy particles for different material layers and momentum ranges, showing the full Geant4 simulation in comparison to the FATRAS energy loss implementation. FATRAS uses the well-known Landau formula for the determination of the most probable energy loss value and a parameterised width of the distribution that has been determined from the Geant4 simulation.

the electron mass, thus, the emitted photons are almost collinear to the original electron trajectory. Figure 8 shows a comparison of emitted high-energetic photons between Geant4 and FATRAS for electrons with a transverse momentum of $p_T = 15$ GeV within the ID tracking acceptance region $|\eta| < 2.5$. The full detector simulation produces usually a higher number of emitted photons, mainly due to the fact that in FATRAS only a limited number of photon-conversion iterations are carried out, while in Geant4 a full cascade simulation is performed; this leads to an underestimation of brem photons in the low momentum spectrum. In a recent study it could be shown that Geant4 and FATRAS behave very similarly in case that both simulation engines are restricted to only one iteration in the photon-conversion cascade. A comparison of FATRAS with a modified Geant4 output is briefly discussed in the Appendix, Sec. A.3.

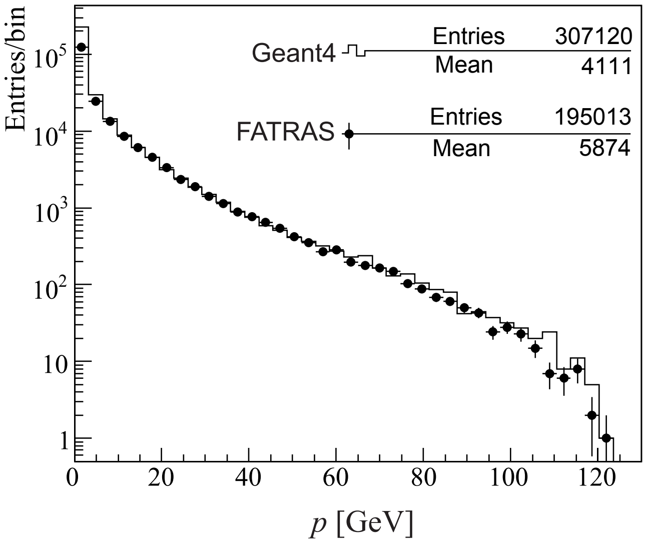


Figure 8: The momentum distribution of hard photons that are emitted from simulated electrons using Geant4 and FATRAS. An identical input sample of 50000 single electron tracks with transverse momenta of $p_T = 15$ GeV are used and restricted to $|\eta| < 2.5$. When restricting the electron energy to a momentum higher than 5 GeV and accounting only for photons with $p > 1$ GeV, the ratio of photons produced from Geant4 to FATRAS drops in the given example from about 1.5 to 1.13; the ratio of the mean value changes from 1.42 to about 1.07.

Hadronic Interactions The simulation of nuclear interaction between hadrons and the detector material is currently limited in FATRAS. The reason is, that the reconstruction geometry does not yet provide any information about the nuclear interaction probability. An updated model for this is planned to be integrated after the ATLAS software release 14.0.0. For a fast track simulation, two aspects are particularly interesting in the context of hadronic interactions; on the one hand, the dominant hadronic shower process leads very often to effectively shorter track lengths (or even no clear hadron trace in the detector at all), which influences both the track parameter resolutions and the track reconstruction efficiency. On the other hand, hadronic shower particles can penetrate into the detector and need to be followed for a successive calorimeter simulation. While the decay process ($\pi \to \mu\nu$) and the scattering process ($\pi \to \pi$) are included in the particle decay module and the multiple scattering update mechanism of FATRAS, respectively, the hadronic cascade (and hence the destruction of the initial pion) has to be taken into account in a separate module. According to the philosophy of a fast and light-weight simulation, a parameterised approach is pursued using properties and fits from the full simulation. For this purpose, a large event sample containing one charged pion (π^+ and π^-) with flat energies between 15 GeV and 50 GeV, flat η and ϕ distributions has been investigated. The hadronic shower model is carried out in a simplified way and is parameterised from data that has been simulated with Geant4. It includes several fit parameters and restricting assumptions and is thus described in more detail in this context.

First of all, the probability of a hadronic interaction cascade has to be calculated which is allowed only for charged pions, kaons and protons as stable, charged particles with energy above 1 GeV (adjustable). Since the hadronic interaction length λ is currently not accessible through the reconstruction geometry, the radiation length X_0 and the average atomic number $\bar{Z}$ are used in the following approximation, introducing an additional scaling factor s_{had}

$$\lambda = 0.37 \cdot s_{\mathrm{had}} \cdot \bar{Z} \cdot \mathrm{X}_0\,, \tag{9}$$

so that the probability p of an interaction is given by

$$p = \exp\{-p_c \cdot d/\lambda\}, \tag{10}$$

with d being the thickness of the layer and the path correction p_c for inclined passage of particles. The scaling factor is adjusted by comparing with the full simulation.

The multiplicity N of the simulated shower is extracted from a fit to the multiplicity distribution observed in the full simulation, see Fig. 10; the following fit function has been used

$$g(x) = \exp\left\{-\frac{1}{2}\left(\frac{x-p_1}{p_2} + \exp\left\{-\frac{x-p_1}{p_2}\right\}\right)\right\}., \tag{11}$$

and the obtained fit parameters p_1 and p_2 are given in the appendix, Sec. A.5.
The multiplicity is restricted to $3 \leq N \leq \min(N_{\mathrm{max}}, 34)$, with

$$N_{\mathrm{max}} = 0.25 \cdot \frac{E/\mathrm{MeV}}{1000} + 18, \tag{12}$$

whereas E denotes the energy of the incoming particle.

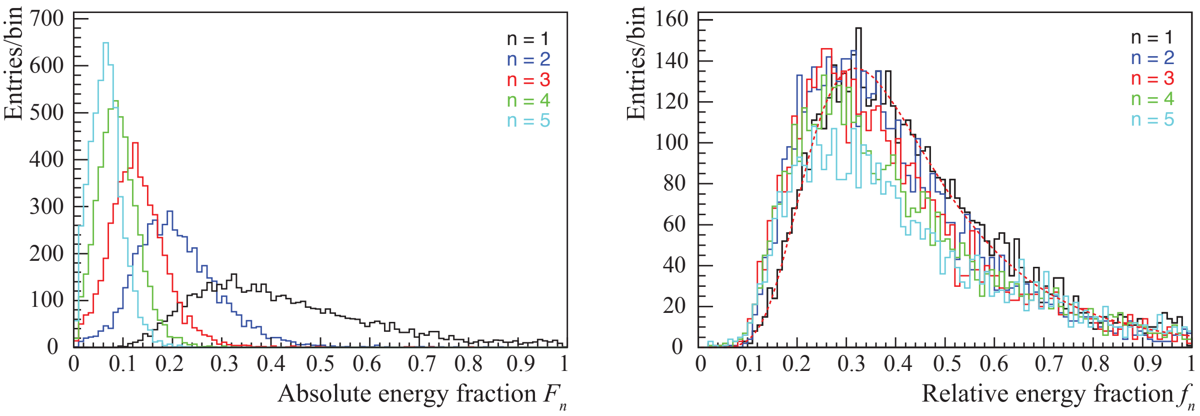


Figure 9: Absolute (w.r.t. the incoming particle) and relative (w.r.t. the rest energy) energy fractions for the five most-energetic particles in the hadronic cascade obtained with Geant4. The right plot includes the generic fit function used for calculating the relative energy fraction.

The energy of the shower particles is determined iteratively. Figure 9 (left) shows the absolute energy fraction $F_n = E_n/E$ of a particle in the shower. The relative energy fractions f_n with

$$f_n = \frac{E_n}{E_{\mathrm{rest}}} \approx f \quad \text{with} \quad E_{\mathrm{rest}} = E - \sum_{i=1}^{n-1} E_i \tag{13}$$

are shown in the right-hand plot of Fig. 9. Since the relative energy fraction f is approximately independent of n (thus the same for all shower particles), this function is fitted with Eq. (11) for various multiplicity bins and used as input for the shower simulation. The energy of a particle n $(1 < n < N)$ is calculated iteratively from the remaining energy E_{rest} with

$$E_n = f \cdot E_{\mathrm{rest}}, \tag{14}$$

whereas f denotes a random number following the distribution Eq. (11) with the parameters that can be found in the Appendix, Sec. A.5. The following constraints are applied:

- first particle: $E_1 > 2 \cdot E/N$,
- energy ordering: $E_{n+1} < E_n$,
- kinematic constraint: $F_n < F_{\mathrm{max}}$ with $F_{\mathrm{max}} = -0.033 \cdot N + 1.16$.

The last cut restricts the phase space and removes unphysically high values of the absolute fraction (high F_n at large multiplicities). The energy calculation using f ends if

- $E_n < 6 \cdot E_{\mathrm{min}}$

- or $E_{\text{rest}} - E_n < 6 \cdot E_{\min}$
- or $E_{\text{rest}} - E_n < 0.1 \cdot E$

with $E_{\min} = 200$ MeV (adjustable). In this case, the energy for the subsequent shower particles is calculated via

$$E_n = E_{\min} + \frac{1}{8}(E_{\text{rest}} - E_{\min}) + g_0(0, 200\text{ MeV}) \quad \text{if} \quad E_n > E_{\min}. \tag{15}$$

$g_0(\mu, \sigma)$ denotes a random number from a gaussian distribution with mean μ and width σ. The last particle $n = N$ always gets the remaining energy, $E_N = E_{\text{rest}}$. The creation of shower particles ends if $E_{\text{rest}} < E_{\min}$. Note, that due to this restriction the final number of particles created in the shower may be lower than the originally anticipated number of particles in the shower N (see Fig. 10).

The angle θ_n of the outgoing particle with

$$\cos\theta_n = \frac{\mathbf{p}_\pi}{|\mathbf{p}_\pi|} \cdot \frac{\mathbf{p}_n}{|\mathbf{p}_n|} \tag{16}$$

is assumed to be proportional to the inverse momentum of the particle, $\theta_n \propto 1/|\mathbf{p}_n|$. This gives a reasonable description, but is limited due to the restrictions on the energies in the shower simulation (here: $\theta_{\max} \sim 2$ corresponding to $E_{\min} = 200$ MeV). As shown in Fig. 11 (h) it is not possible to generate a sufficient number of particles at extremely large emission angles with this approach.

Finally, the particle content of the shower has to be simulated. The shower contains many different types of particles, but mainly charged and neutral pions, protons and neutrons. Because of this, only these particles are created with fractions 33% (π^0), 25% (π^+), 25% (π^+), 10% neutrons and 7% protons. These fractions are adjusted to the shower content generated by Geant4 and can be modified as tuning parameters.

Figure 10 illustrates the particle multiplicity in hadronic showers from nuclear interactions of the primary traversing particle with detector material. It includes the distribution obtained with Geant4, the generic fit function used for the parameterised level and the according distribution produced with FATRAS. Events with very low particle multiplicities that appear in the Geant4 simulation are suppressed in FATRAS due to the chosen momentum cuts that require a minimal particle energy for the application of a hadronic interaction.

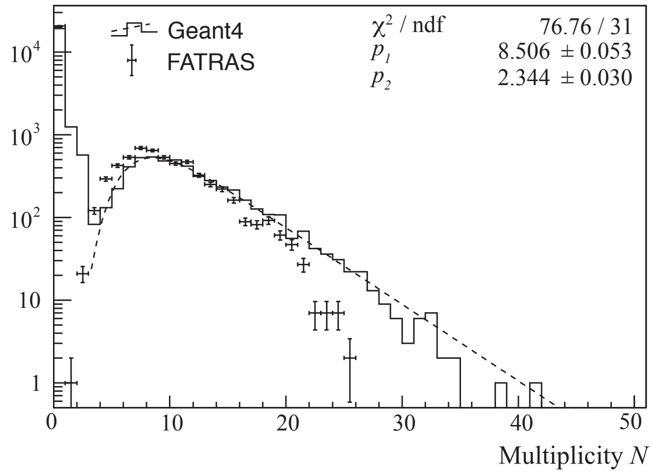


Figure 10: Particle multiplicity N in hadronic showers generated with Geant4 and FATRAS. The fit function which is basis for the FATRAS hadronic shower model is also shown in the histogram

Figures 11 (a) to (f) show comparisons of absolute and relative energy fractions for the four most energetic shower particles between the Geant4 simulation and FATRAS. It also includes the overall direction distribution of the simulated shower particles as their relative output direction to the original hadron.

In general, a good agreement in the distributions between the parameterised shower in FATRAS and the full simulation can be observed.

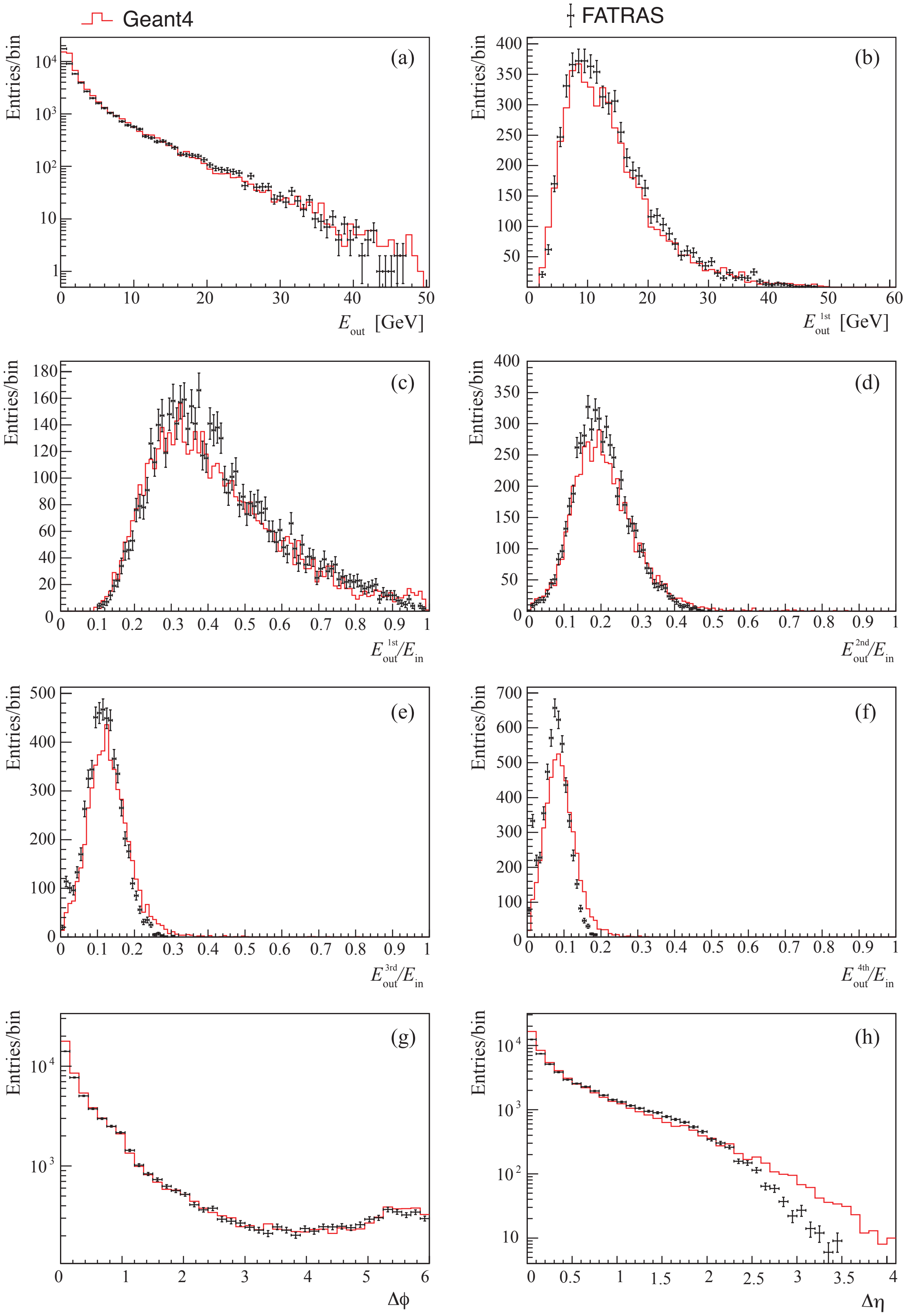


Figure 11: (a) and (b) - comparison of the absolute energy fractions of all and the most energetic shower particle in hadronic showers caused by nuclear interactions simulated with Geant4 and FATRAS, respectively. (c) to (f) show the relative energy fractions of the four most energetic shower particles, while (g) and (h) show the angular directions of the outgoing shower particles w.r.t. the incoming particle direction simulated with Geant4 and FATRAS.

3.3.2 Track Creation from a given Trajectory

The extrapolation engine is not capable of producing tracks in the ATLAS reconstruction geometry, it only provides a trajectory of track parameters on sensitive detector elements. This trajectory has to be transformed into a `Track` object of the ATLAS tracking EDM. A dedicated `AlgTool` — the `TrackCreator` — performs this task; it calls the extrapolation engine and dissolves the returned trajectory into the single surface intersections. Since the `Surface` object of the ATLAS reconstruction geometry is linked to the underlying readout element of the full ATLAS detector description, it is possible to identify the detector type and apply an appropriate clusterisation or hit smearing. The track parameter information is used to create standard ATLAS tracking EDM objects, the non-calibrated `PrepRawData` and the calibrated `RIO_OnTrack` objects representing the simulated detector measurements. Inefficiencies for the sub-detectors can hereby be applied by introducing efficiency parameters for the sub-detectors.

Clusterisation and Simplified Smearing A correct clusterisation description is essential for obtaining meaningful track parameter resolutions when being compared to tracks from the full simulation and offline reconstruction chain. It can be shown, that the impact parameter resolutions are dominated by the measurements on the innermost detection layers[8], while the momentum resolution depends on the entire hit collection of the track. The clusterisation model for the pixel detector has therefore to be the most accurate, since its influence — in particular the cluster sizes and shapes of the B layer measurement — on the track parameterisation is the most significant one.

In the ATLAS offline reconstruction, the pixel clusterisation uses an analog model that determines the cluster position with the help of the time over threshold information that is available for every readout (or simulated) pixel signal [20]. This technique yields a higher resolution than the intrinsic single pixel resolution, but requires the charge deposition distribution to be known, which is either provided through the readout when reconstructing real data or simulated in the full detector simulation. In FATRAS, the charge deposition in sensitive detector material is not simulated down to this level of detail, only energy loss is applied to the particle when traversing detector material. A geometrical clusterisation approach is used instead to recreate similar cluster sizes and shapes as given through the realistic full detector simulation and digitisation. The geometrical clusterisation is an analog clustering process that uses the path length of the track in the single intersected pixels to calculate the cluster position from the center positions of the associated pixel cells. Given n geometrically intersected pixels with s_i the according path length in pixel i, the cluster position $\mathbf{p}$ is then calculated as

$$\mathbf{p} = \frac{1}{\sum_{i=0}^{n} \Theta(s_i - s_{cut}) \cdot s_i} \sum_{i=0}^{n} \Theta(s_i - s_{cut}) \cdot s_i \cdot \mathbf{p}_i, \tag{17}$$

where the $\mathbf{p}_i$ denote the individual center positions of the intersected pixels and the Heaviside function Θ is inserted to demand a minimal path length of the track within the single silicon pixel which is equivalent to a minimal charge deposition in a single pixel; s_{cut} becomes hereby the single model parameter of the pixel clusterisation. Since FATRAS does not simulate the charge deposition in the Silicon pixels, a correction of cluster shifts accounting for the Lorentz force effects on drift electrons has to be applied. This fake correction is then cancelled by the official ATLAS clusterisation tool and may be one source for the slight discrepancies between cluster sizes in the full offline chain and FATRAS. Figure 12 shows an illustration of the geometrical clusterisation approach.

The clusterisation for the silicon strip detector is for both the offline realisation and in the simplified FATRAS model based on a digital approach, accounting for only the the intersected strips to build the final cluster position. This yields to one-, two- and three-strip clusters without using any charge deposition information. For the *Transition Radiation Tracker* (TRT), the hit creation in FATRAS is simply done by Gaussian smearing of the geometrical closest approach to the wire, since the *cluster* shapes in the TRT are in general not very complex, but roughly compatible with a Gaussian distribution of the drift radius. Since the drift radius error varies for different drift time bins, this information is retrieved from the offline drift function tools. An additional tail contribution can be added to account for non-Gaussian effects and a scaling parameter is available to regulate the momentum resolution.

[8] This can be easily shown by assuming a simplified 2 layer detector model, see [21].

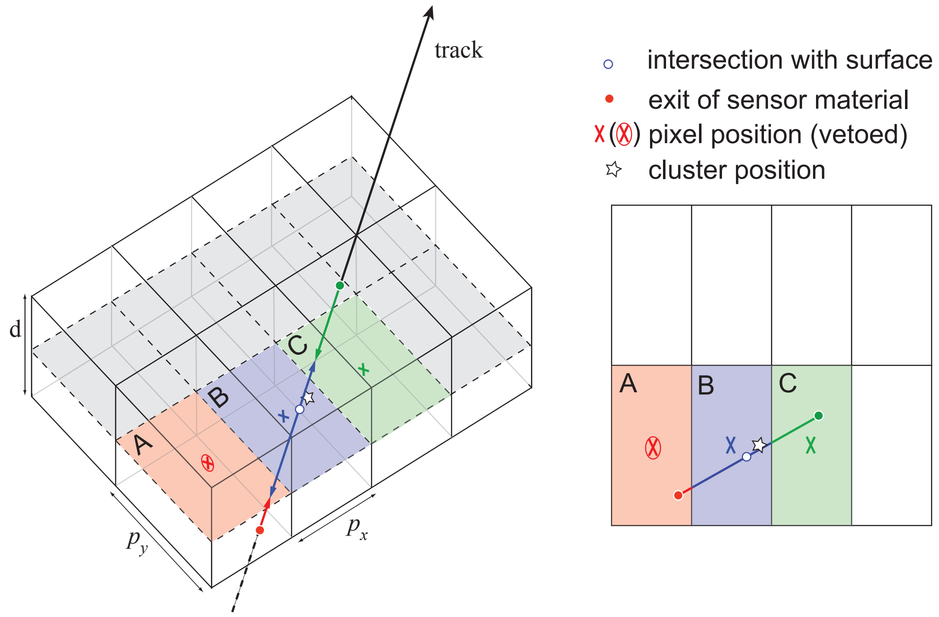

track
d
A
B
C
p_y
p_x
○ intersection with surface
● exit of sensor material
X(⊗) pixel position (vetoed)
☆ cluster position
A
B
C

Figure 12: The geometrical clusterisation model for the pixel detector that is used in the FATRAS simulation. Analog clusterisation is hereby performed by weighting the readout center positions of intersected pixels with the track distance inside the pixel. Single silicon pixels that are traversed by the track, but do not host a sufficiently long path length for the pixel to detect a signal are vetoed for the clusterisation process (pixel A).

Figure 13 shows a comparison of obtained cluster sizes and resulting hit residuals on the tracking devices between offline reconstruction and the FATRAS clusterisation model for all three tracking technologies in the ID, the double Gaussian smearing of the TRT is, however, omitted in this figure.

The hit creation — carried out by the `ClusterCreator AlgTool` — can also be performed with purely Gaussian smearing for all detector technologies. This does not lead to track parameter resolutions that are comparable to the tracks found by the offline reconstruction, but is very useful for using FATRAS as a validation tool, in particular when testing the track fitting modules.

3.4 Photon Processing

The transport of electrons through the detector causes in many cases the emission of a high-energetic photon. Since these photons create either an electromagnetic shower in the calorimeter or convert into (mainly) electron-positron pairs when interacting with detector material, it is important to follow these processes in the track simulation. In FATRAS, the photons from final state radiation or emitted by electrons — as described in Sec. 3.3.1 — are collected in one `ParticleState` container and further processed by a dedicated algorithm, the `PhotonProcessing`. Each photon is hereby propagated through the detector following a straight line model. A specially configured instance of the `Extrapolator` is used to perform this operation; a specific `IMaterialEffectsUpdator` implementation, the `McConversionCreator`, simulates conversions of the photon into an electron-positron pair depending on the amount material when a layer is crossed[9]. This process can be iteratively repeated to account for a more detailed cascade simulation.

Figure 14 shows an example photon conversion event simulated with FATRAS and displayed with the ATLANTIS event display. It presents in addition a conversion vertices map in the $r - z$ projection of the Inner Detector using Geant4 simulation and FATRAS. The simplified geometry used in FATRAS can be seen through the discrete distribution of conversion points in the TRT detector, which is modeled as a simple set of several cylinders and discs in the `TrackingGeometry`, while a continuous material distribution in the full detector geometry used by Geant4 can be identified.

[9] The simulation of $\mu^-\mu^+$ pair production is omitted in FATRAS since it is by orders of magnitudes less probable than for the electron-positron case.

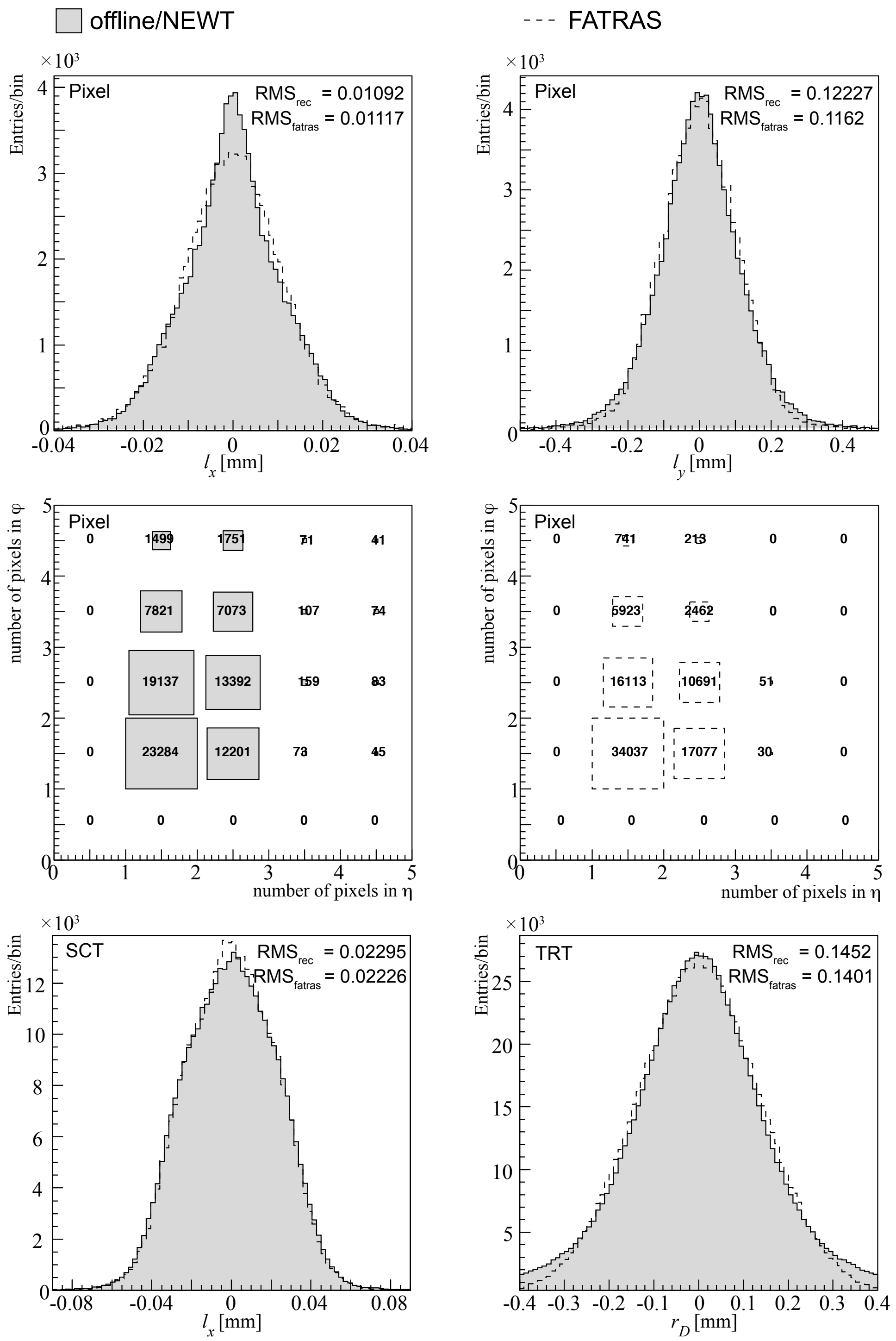


Figure 13: A comparison of cluster sizes and resulting residuals on the tracking devices between offline reconstruction and FATRAS. Most emphasis has been put on a correct description of the pixel clusterisation since it has maximum impact on the track parameter resolutions. The slightly narrower pixel residuals from the offline reconstruction may be caused by broader cluster sizes that could not fully be reproduced with the current geometrical clusterisation module in FATRAS.

The conversion probability has been parameterised and fitted to data taken from [22], where the photon conversion probability dependent on the traversed material thickness is given for aluminum and photons with energies between 0.1 and 100 GeV . The given parameterisation fits fairly well to the numbers obtained from fast simulation. However, additional scaling parameters have been inserted to allow custom tuning of FATRAS with respect to Geant4. These include also the energy and angular

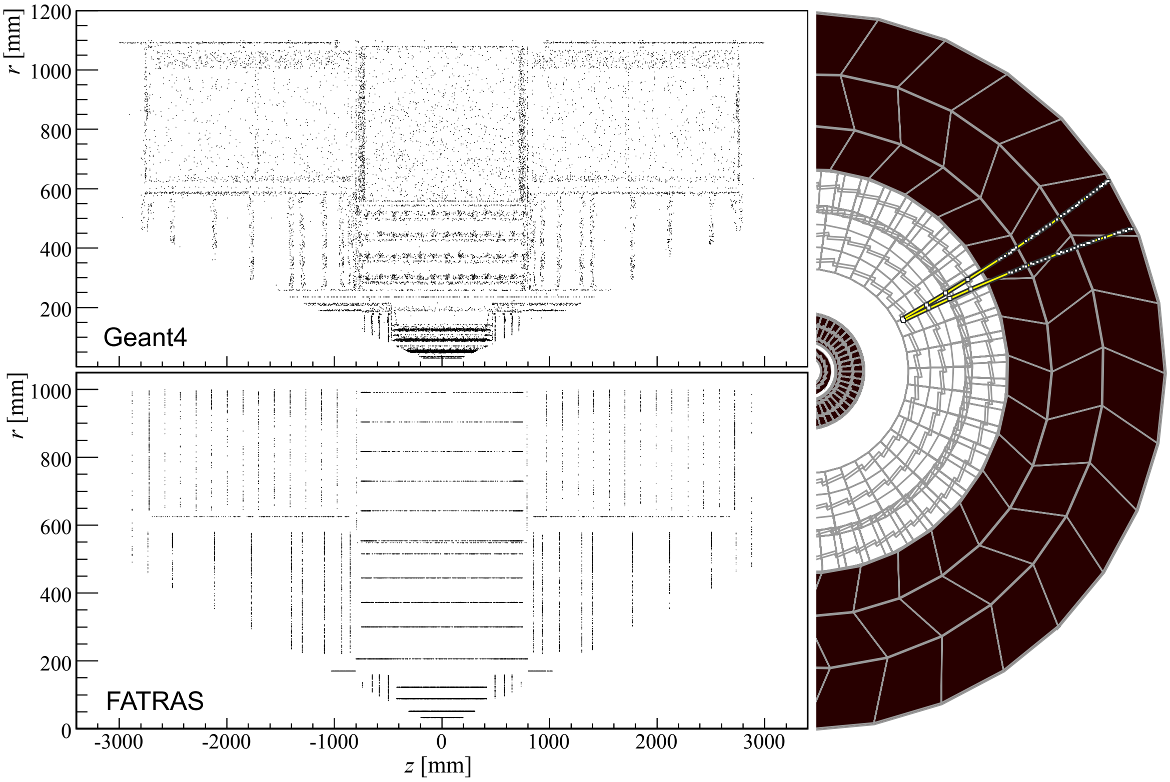


Figure 14: Photon conversions in the ATLAS Inner Detector, simulated with Geant4 and FATRAS; the simplified simulation geometry of FATRAS can be seen; it is limited to several discrete layers, while the Geant4 simulation geometry is more detailed. The picture to the right shows a photon conversion before the first SCT barrel layer simulated with FATRAS and shown with the ATLAS event display ATLANTIS.

distribution of the produced particles with FATRAS calculated using the same model [23] that is implemented in the Geant4 simulation toolkit. Since the Geant4 model provides a very accurate description of the material, while the `TrackingGeometry` represents only a strong simplification of the actual detector geometry, some tuning has to be applied. Based on large scale single photon events simulated with both, FATRAS and Geant4 the photon probability and energy spectrum of the pair products have been compared and it could be shown that only the latter requires a constant scale factor to yield a satisfactory agreement between FATRAS and the full simulation. The scaling procedure has been performed by using photon conversions that have been caused by the beam pipe material, since the beam pipe is implemented in both simulation geometries to the same detail. A detailed list of the tuning parameters can be found in the Appendix, Sec. A.5, of this document.

Figure 15 shows a comparison of simulated electron energies for particles originating from photon conversions in the ATLAS ID and gives a comparison of the mean electron energy depending on the transverse photon momentum.

When comparing a full physics event, FATRAS underestimates the number of brem photons and photon conversions for the simple reason that only a limited number of iterations of the entire cascade are carried out. For the simplest possible FATRAS configuration that only implements one iteration, this means that an electron which comes from a pair production process can indeed again radiate a high energetic photon; this photon is, however, not tracked further in terms of conversion creation; it is, together with photons that are not converted into a child pair at first place filled into the appropriate containers for the exit state creation, see Sec. 3.6. An already mentioned further investigation of this subject revealed the necessity to integrate a second iteration of the photon-conversion cascade (→ Appendix, Sec. A.3), which is planned to be fully integrated in the ATLAS release 14.0.0.

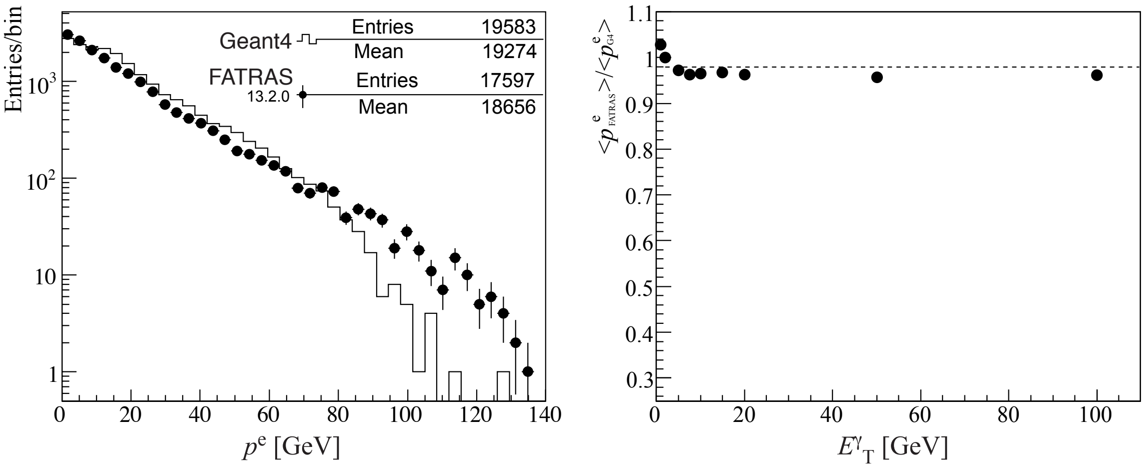


Figure 15: Comparison (Geant4/FATRAS) of the electron energy distribution originating from photon conversions in the ATLAS ID for photons with initial transverse energy of $E^{\gamma}_{T} = 15$ GeV (left). The right plot shows the ratio of the mean child electron momentum $< p^{e}_{FATRAS} > / < p^{e}_{G4} >$ from photon conversions in the ID for photon with various fixed (transverse) momenta.

3.5 Track Refitting

The track refitting module has to be only performed when FATRAS is used in *refit* mode; in this case, the refit of the simulated track is necessary to remove the truth bias and to result in an adequate track resolution compared to the full simulation and offline reconstruction chain. The simulated track is hereby kept as a reference or truth track for validation purposes. The refit of an entire track is in general performed such that the perigee representation is used as the starting point of the track. Since in FATRAS the perigee associated to a track is the true origin of the trajectory this would introduce a strong bias towards artificially high track resolution on the first measurement layer. The special `TrackRefitting Algorithm` that is contained in the FatrasAlgs package allows therefore to smear the initial perigee representation. The smearing, however, has to be kept in a regime that is compatible with the linearised measurement model that is inert to most fitting techniques; the smearing parameters for the applied Gaussian smearing of the initial perigee representation is accessible as tuning input from the job configuration.

3.6 Exit State Creation

The creation of particle representations at the exit surface of the tracking detector volume marks the last step in the FATRAS simulation sequence. It has little to do with the track simulation itself, but prepares FATRAS for future integration with both the full or fast calorimeter simulation, such as a future extension of FATRAS for the Muon Spectrometer.

In FATRAS, every particle — independent of its origin or creation process — that has not been decayed within the detector volume is transported to the boundary of the volume for further processing in a subsequent detector part. A small outlook on a potential integration of FATRAS with a fast shower application can be found in Sec. 6.1.

3.7 Post Processing and Noise level creation

An additional level of post processing is necessary in FATRAS to achieve full compatibility with the offline reconstruction. In the pure refit mode, the tracks have to be processed before refitting to simulate some pattern recognition effects such as holes on track or missed extensions from one sub-detector to another[10], see Sec. 4.1.1. When using FATRAS to prepare input for the standard

[10] The standard ATLAS reconstruction incorporates an inside-out procedure for the pattern recognition that starts at the innermost silicon layers and extends a successful silicon track segment to the TRT straw detector at larger radii.

offline reconstruction, another aspect has to be considered: the creation of *fake* hits that originate in reality from electronic noise or low energetic secondary particles that are created when the primary particles traverse the detector material. Since the pattern recognition is fully performed, a special post processing in terms of hit correction is not necessary. The different topologies that originate from effects in the pattern recognition process are automatically present in the reconstruction mode; a comparison of reconstruction efficiencies for different particles can be found in Sec. 4.2. What remains is the hit container and noise creation handling: during the hit post processing, hits are stripped from the simulated tracks and filled into the same hit collections that are used in the offline reconstruction chain. Noise hits can hereby be added independently to every sub-detector data collection and the noise levels can be adjusted freely by the user. The chosen defaults of noise levels in the three sub-detectors can be found in the Appendix, Sec. A.5. This turns FATRAS into a powerful engine for occupancy studies, see Sec. 5.1; the low execution time enables a *scan* through different noise level settings.

4 Validation and Performance

The dedicated aim of a fast simulation is to build a fast alternative to the full detector simulation (and — in the case of the ATLFAST simulation or the FATRAS refit mode — also of the reconstruction), which marks an important guideline for the validation. Thus, the main focus has to be the validation of FATRAS against results obtained with the full offline chain and not to optimise the FATRAS performance itself. As a consequence, some tuning parameters have to be introduced that allow to turn the outcome of the fast track simulation more towards the according results obtained with the full offline chain[11], sometimes this turns to the simple but non-trivial task of *how to make things worse*. Some requirements such as the correct description of material effects or a proper clusterisation model have been already presented in Sec. 3 of this document. The following sections will concentrate on hit statistics comparison and track parameter resolutions for FATRAS compare them with full simulation results, and — if applicable — with according results of the fast ATLAS simulation program ATLFAST.

Validation `Algorithm` classes Several dedicated `Algorithm` classes exist in the FATRAS repository that prepare convenient user output such as track resolution and correlation data in form of ROOT [24] tuples. Since in FATRAS the simulation trajectory is known[12] as an own `Track` object, convenient truth association can be performed in terms of associative containers that relate the simulated track to the refitted (or reconstructed, respectively) and a similar structure for the hit association. In the refit mode this can be done without any matching procedure, since the simulated and refitted track are available at once in the `TrackRefitting` module. For the reconstruction mode, a certain `TrackAssociation Algorithm` has to be performed for matching the simulated and reconstructed track collections. The matching procedure is hereby carried out in (η, ϕ) bins and is based on the lowest χ^2 of the candidate track pairs. The full integration of the MC truth chain in FATRAS that has the same identical structure as in the full offline chain allows also to include all standard validation modules that are used in the offline reconstruction.

4.1 Single Track Validation

A comparison of pure track resolutions is best performed on single track events, since pattern recognition effects or shared clusters arising from high track densities can be neglected in this situation.

Validation Samples and Quality Cuts For the following comparison, dedicated validation samples have been produced for single particle events with constant transverse momentum. 50 000 single track

[11] FATRAS leads, in general, to slightly better performance in track parameter resolutions and hit statistics. This is, since only a limited number of disturbances are simulated in the fast track simulation; the omitted effects are typically of second or third order and contribute usually mainly to the tail distribution of the track parameter resolutions.

[12] When using the Geant4 detector simulation, a truth trajectory has to be build first and matched with a reconstructed track object. For a direct comparison, this also involves the translation of the kinematic input object to the a track representation.

events have been produced for electrons, pions, muons and photons in the transverse momentum range $p_T \in [1\ \text{GeV}\,, 1\ \text{TeV}]$, spreading over a pseudo-rapidity of $|\eta| < 3.0$. The production vertex of the single particles is smeared according to standard ATLAS vertex smearing, i.e. Gaussian smearing with $\sigma_{xy} = 15\ \mu\text{m}$ in the transverse plane and $\sigma_z = 56\ \text{mm}$ in the longitudinal direction.
Tracks that have been used for the comparison passed the following generator cuts:

- *prompt particles* : generation barcode < 100000
- *tracking acceptance* : $|\eta| < 2.5$

Successfully reconstructed tracks have in addition passed the following reconstruction quality cuts:

- *prompt particles (I)* : $d_0^{rec,PV} < 2\ \text{mm}$, when d_0 is expressed to the primary vertex (PV)
- *prompt particles (II)* : $z_0^{rec,PV} < 10\ \text{mm}$, when z_0 is expressed to the PV
- *Si hits* : number of hits in the silicon detector greater than 6
- *TRT extension* (if $|\eta| < 2.0$) : number of TRT hits greater than 0

For ATLFAST, the hit cuts can not be applied, since no hit information for the track is available.

4.1.1 Hit Creation

A correct number of simulated hits per track is not in particular necessary for all tracking related studies with the ATLAS detector. The impact parameter resolution, on the one hand, is e.g. almost entirely dominated by the innermost layer measurements and missing hits further along the track are of small influence for the spatial resolution. The momentum resolution, on the other hand, improves substantially with the track length, since a longer lever arm helps to correctly determine the track curvature. In the offline reconstruction, the pattern recognition is responsible for finding the hits that are likely to belong to one single track, while the track fit is performed to test (and either verify or falsify) the track hypothesis and finally to determine the track resolution. For convenience, the track is mostly expressed with respect to the production vertex of the particle. Since FATRAS is used to feed the standard offline reconstruction algorithms, many pattern recognition effects (such as missing hits or segments that have not been found by the appropriate sub-algorithms) are directly included. Studies that are sensitive to pattern recognition effects should therefore be based on the reconstruction mode of FATRAS. In the refit mode, however, the pattern recognition is bypassed. An overview on the single CPU time contributions of the various FATRAS modules can be found in Sec. 4.3. To still keep compatibility with offline results, FATRAS offers the possibility to force the introduction of hit finding inefficiencies by losing single hits or full segments. The almost geometrical efficiency of FATRAS can thus be downscaled to meet the figures obtained by the offline reconstruction. The excluded hits can, however, be still included in the event hit collection, to guarantee a correct treatment in the pattern recognition process. Figure 16 shows a comparison of the hit numbers per track for single muon tracks with a momentum of 5 GeV in the ID for both, the calibrated and non-calibrated case in the refit mode. It can be shown that the hit finding efficiency for single muon tracks is practically at a level of 100 % in case that track segments in all sub-detectors are found. The hit numbers for the reconstruction mode are therefore omitted in this illustration, since they overlay with the ones obtained in the refit mode.

4.1.2 Single Muon Samples

Muons create the cleanest signal trace through the detector, since they cause minimal interaction of all charged particles with the detector material. This is, because muons do not undergo nuclear or hadronic interactions with the nuclei of the detector material and neither are they subject to significant energy loss through radiation. The reconstruction, i.e. the finding and successful fitting of the particle trace, of single muon tracks shows almost full efficiency over the geometrical acceptance region of the ATLAS Inner Detector. Single muon events are therefore often taken as reference tracks for the detector performance validation and have also marked the first step in the FATRAS

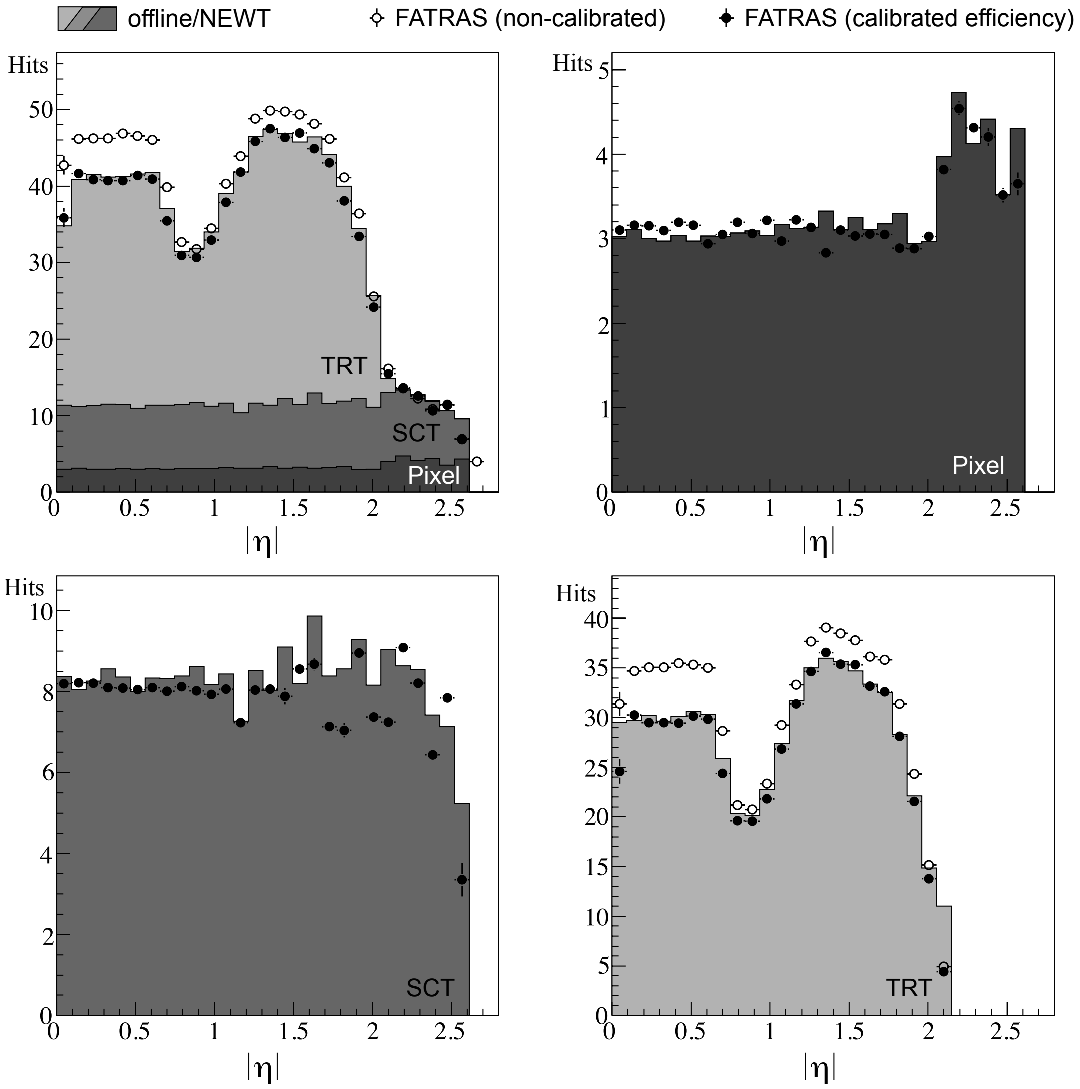


Figure 16: Comparison between the number of hits created on a FATRAS track to those from equivalent tracks simulated with the full ATLAS Geant4 detector simulation and reconstructed with New Tracking (NEWT). 5000 single muon tracks with a momentum of 5 GeV have been used for this study. In FATRAS, the nearly geometrically limited efficiency that leads in general to higher hit numbers per track can be adjusted in different eta bins.

validation against offline simulated and reconstructed events. Good agreement between tracks from offline reconstruction events and FATRAS could be met over a large momentum range. Figure 17 and Fig. 18 show comparisons of track parameter resolutions in the low and high momentum regimes for singe muons and ATLFAST, the offline chain and FATRAS, respectively. FATRAS is in comparison to the ATLFAST simulation superior in describing the tail distributions due to scattering effects at the low momentum spectrum and also in the correct cluster description at the high momentum limit. The asymmetric tail in the reconstruction of low p_T muons is due to the usage of single-charged tracks (μ^-) together with an overestimation of the ionisation loss at low energies in the reconstruction modules. This effect vanishes for FATRAS since the identical parameterisation of ionisation loss is used in both, simulation and reconstruction modules.

The $A \oplus B$ Model A general feature of the track parameter resolutions is that they degrade with lower momenta, since the contribution of multiple scattering is inverse proportional to the particle momentum, see Eq. (2). In a simplified model, the dependency of the track parameter resolutions on

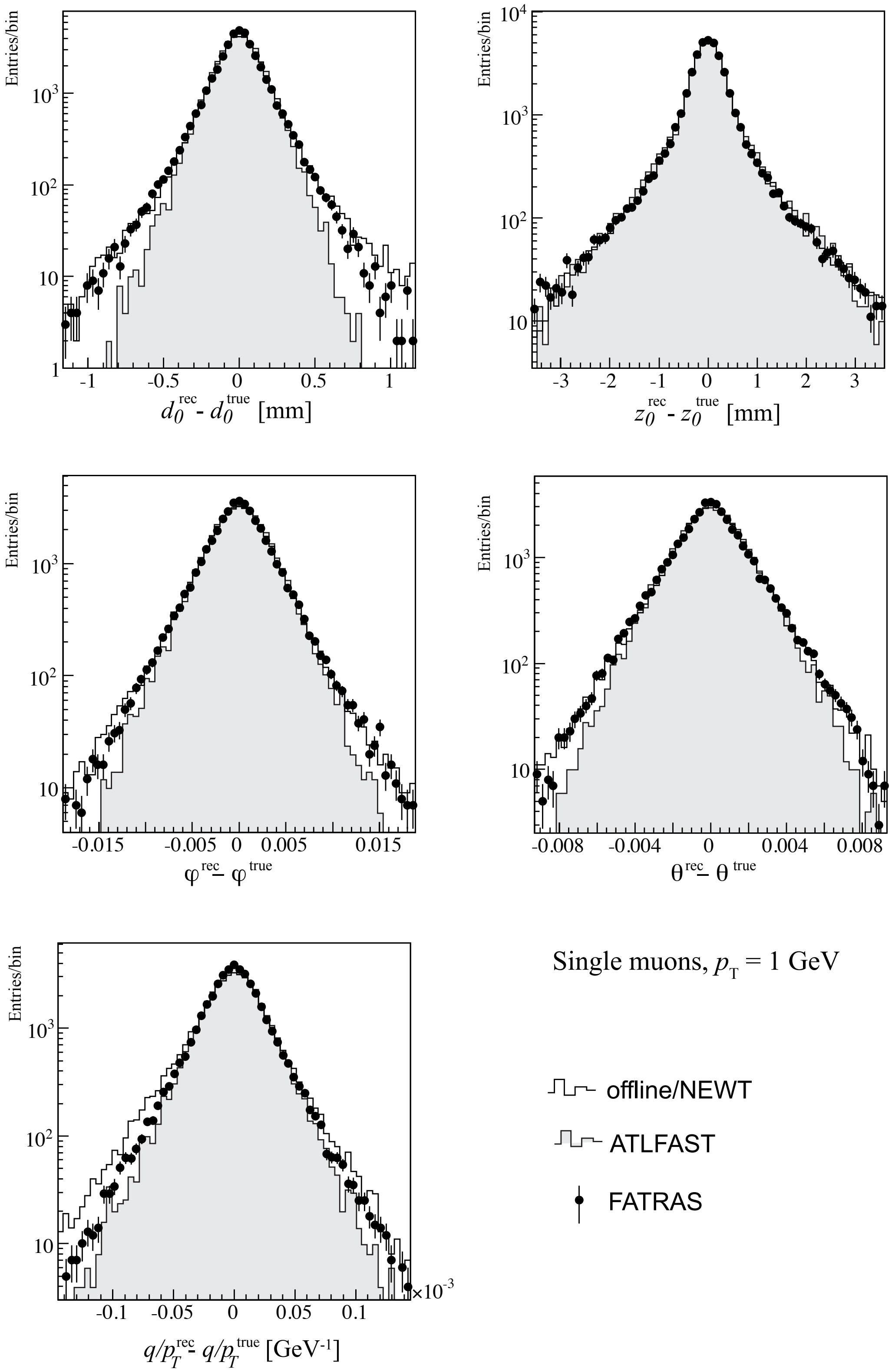


Figure 17: Track parameter resolutions for single muon tracks in the low momentum limit over the entire acceptance range of $|\eta| < 2.5$ of the ID. Results obtained from offline reconstruction are shown and compared to FATRAS tracks from the refit mode (dotted line) and results from the ATLFAST smearing (shaded area). FATRAS performs better than ATLFAST in the correct description of the tail distributions.

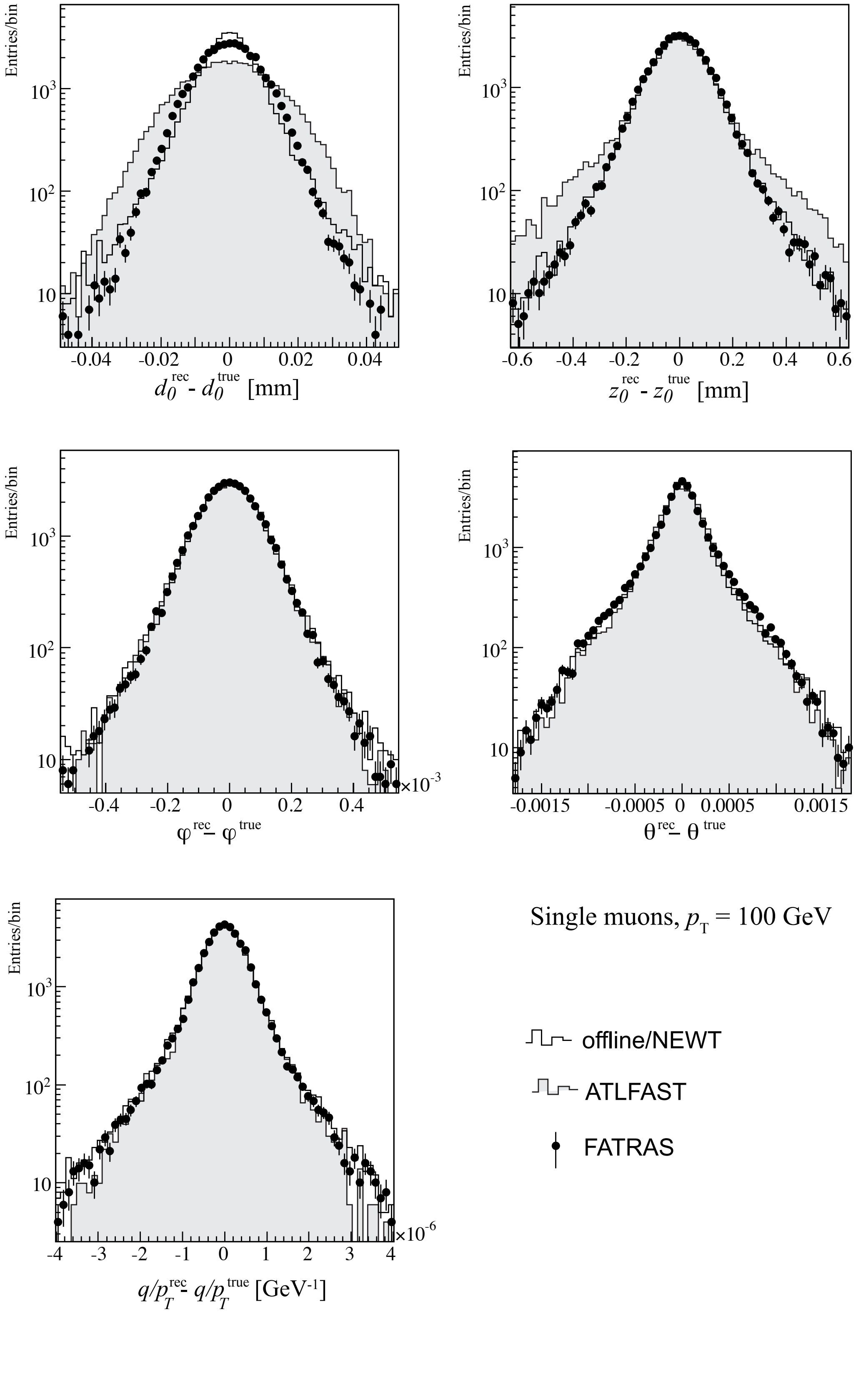


Figure 18: Track parameter resolutions for single muon tracks in the high momentum limit over the entire acceptance range of $|\eta| < 2.5$ of the ID. Results obtained from offline reconstruction are shown and compared to FATRAS tracks from the refit mode (dotted line) and results from the ATLFAST smearing (shaded area).

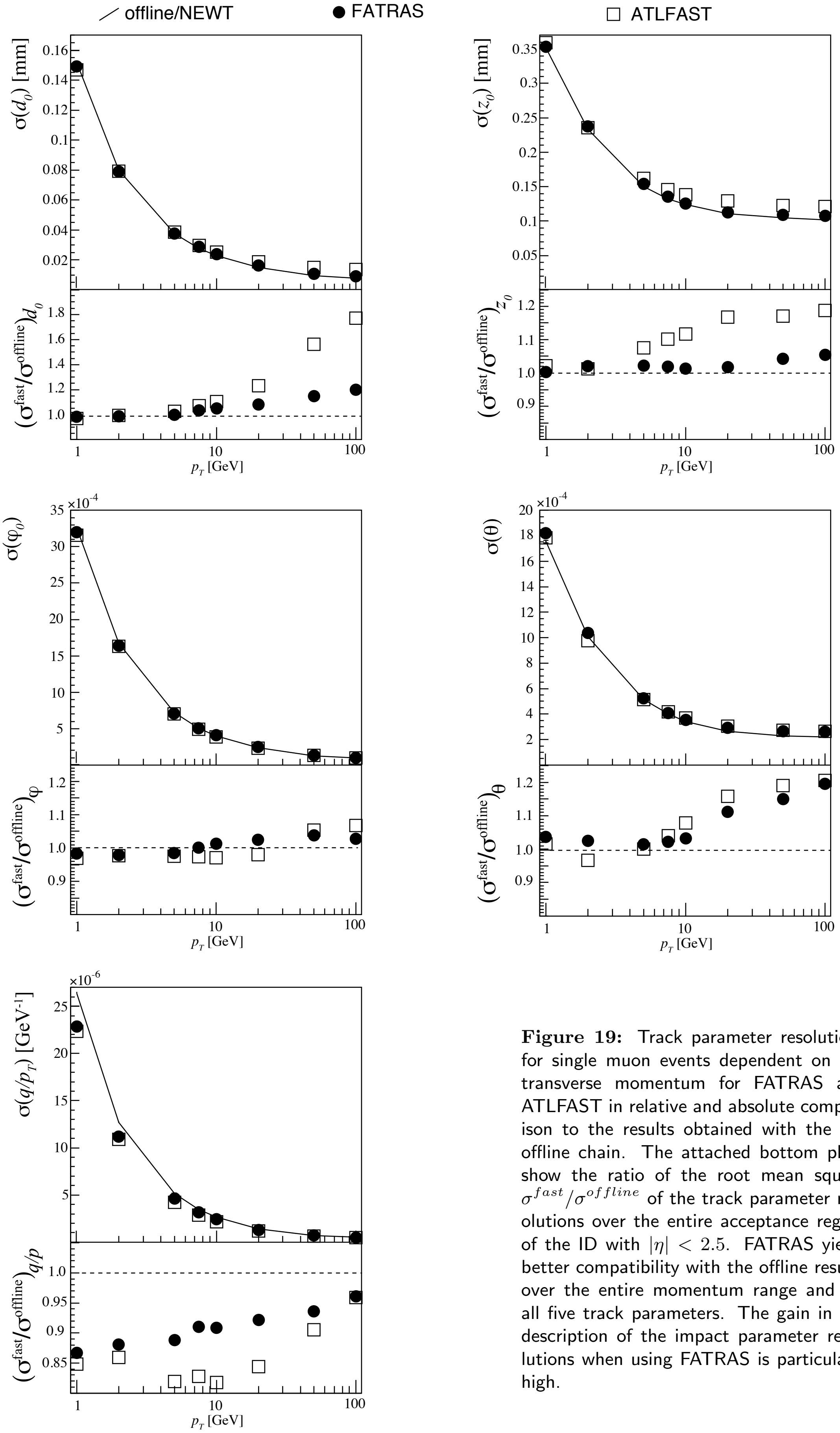


Figure 19: Track parameter resolutions for single muon events dependent on the transverse momentum for FATRAS and ATLFAST in relative and absolute comparison to the results obtained with the full offline chain. The attached bottom plots show the ratio of the root mean square $\sigma^{fast}/\sigma^{offline}$ of the track parameter resolutions over the entire acceptance region of the ID with $|\eta| < 2.5$. FATRAS yields better compatibility with the offline results over the entire momentum range and for all five track parameters. The gain in the description of the impact parameter resolutions when using FATRAS is particularly high.

the momentum can be approximated through the quadratic sum

$$\sigma_\tau = A_\tau \oplus \frac{B_\tau}{p}, \tag{18}$$

where A_τ marks the intrinsic detector resolution and B_τ includes the multiple scattering contribution for the given track parameter τ, see [21]. This approximate model — in the following referred to as $A \oplus B$ *model* — has been expanded to more sophisticated version that includes a Taylor expansion of the underlying track model [25]. For the further comparison, however, the $A \oplus B$ model will be taken, since it is intuitive and accurate enough for a simple comparison. The $A \oplus B$ model is not restricted to muon tracks, but it can be best demonstrated for minimal interacting particles, since the resulting distributions are likely to be unbiased and follow a Gaussian PDF[13]. Figure 19 shows an absolute and the relative comparison of the track parameter resolutions in dependence of the transverse momentum for ATLFAST and FATRAS with results from the full offline chain. Table 2 shows a comparison of the single parameters obtained for the $A \oplus B$ model fitted to the root mean square (RMS) of the track parameter resolutions for the entire acceptance region of the ID.

Table 2: Track parameter residuals parameterised according to the $A \oplus B$ model, comparing the offline reconstruction with FATRAS and ATLFAST. The given values mark the root mean square of the distributions that include tracks over the entire acceptance range of the ATLAS ID. The track parameter resolutions of muon tracks are very close to Gaussian curves. The RMS is therefore a good measure when overlaying the different resolutions in the various $|\eta|$ bins.

Track Parameter	Function Parameter	**offline/NEWT**	**FATRAS**	**ATLFAST**
d_0	$A_{d_0}[\mu\mathrm{m}]$	8.17	9.76	14.46
	$B_{d_0}[\mu\mathrm{m\,GeV}]$	176.4	176.5	167.4
z_0	$A_{z_0}[\mu\mathrm{m}]$	110.3	114.5	128.5
	$B_{z_0}[\mu\mathrm{m\,GeV}]$	358.0	355.2	349.2
ϕ_0	A_ϕ	$9.3 \cdot 10^{-5}$	$9.6 \cdot 10^{-5}$	$10.1 \cdot 10^{-5}$
	$B_\phi[\mathrm{GeV}]$	$3.58 \cdot 10^{-3}$	$3.56 \cdot 10^{-3}$	$3.46 \cdot 10^{-3}$
θ	A_θ	$2.36 \cdot 10^{-4}$	$2.75 \cdot 10^{-5}$	$2.82 \cdot 10^{-4}$
	$B_\theta[\mathrm{GeV}]$	$1.88 \cdot 10^{-3}$	$1.91 \cdot 10^{-3}$	$1.87 \cdot 10^{-3}$
q/p_T	$A_{q/p_T}[\mathrm{GeV}^{-1}]$	$4.63 \cdot 10^{-4}$	$4.55 \cdot 10^{-4}$	$4.62 \cdot 10^{-4}$
	B_{q/p_T}	0.026	0.023	0.021

Both ATLFAST and FATRAS show a reasonable agreement with the full reconstruction chain, although some additional tuning may have to be applied in order to fully comply with the offline results. ATLFAST shows the same features as demonstrated in Fig. 18, the impact parameter resolutions in the asymptotic high momentum region can not be reproduced. For FATRAS, in particular the good agreement of the multiple scattering term indicates that the reconstruction geometry material description is a good representation of the simulation geometry.

Dependency on the Pseudorapidity The track parameter resolutions do not only depend on the momentum, but also on the pseudorapidity. This is caused by several reasons:

- the distribution of inert detector material is far from being constant (see Fig. 5) in the full η acceptance region of the ID; in general, the material budget increases at higher pseudo-rapidity ranges, since more support structures and cable/electronics material are situated outside the central region. In addition, a geometrical effect plays another role: the incident angle of the track with respect to the intersected detector module varies with $|\eta|$ and thus results in different effective thickness of detector elements and support structures.
- when investigating particles at constant transverse momenta (which makes sense for the momentum resolution, since the transverse momentum is related to the bending power), higher $|\eta|$ values correspond to higher momenta, and have thus a lower multiple scattering contribution;

[13] For electrons, the loss of energy due to bremsstrahlung disturbs the Gaussian character of the track parameter resolutions.

- finally, the changing incident angle results in different cluster sizes; that is why the intrinsic resolution of the track parameters varies for different pseudo-rapidity values even at the high momentum limit, where the multiple scattering contribution can be neglected;

For FATRAS the track parameter resolutions versus pseudorapidity builds a stringent test for the material distribution (described by the `TrackingGeometry`) and the clusterisation module. The η-dependency of the track parameter resolutions is reproduced (without adding any dedicated tuning) when these two modules are described correctly, a feature which will be proven in the following context. In ATLFAST, on the other hand, the η dependency has to be put in *by hand*: the smearing functions are obtained from offline in certain $|\eta|$ bins and an interpolation is being performed to cover the full acceptance region. Figure 20 shows a comparison of the impact parameter resolutions for the offline chain, FATRAS, and ATLFAS when using single muon tracks of different constant transverse momenta.

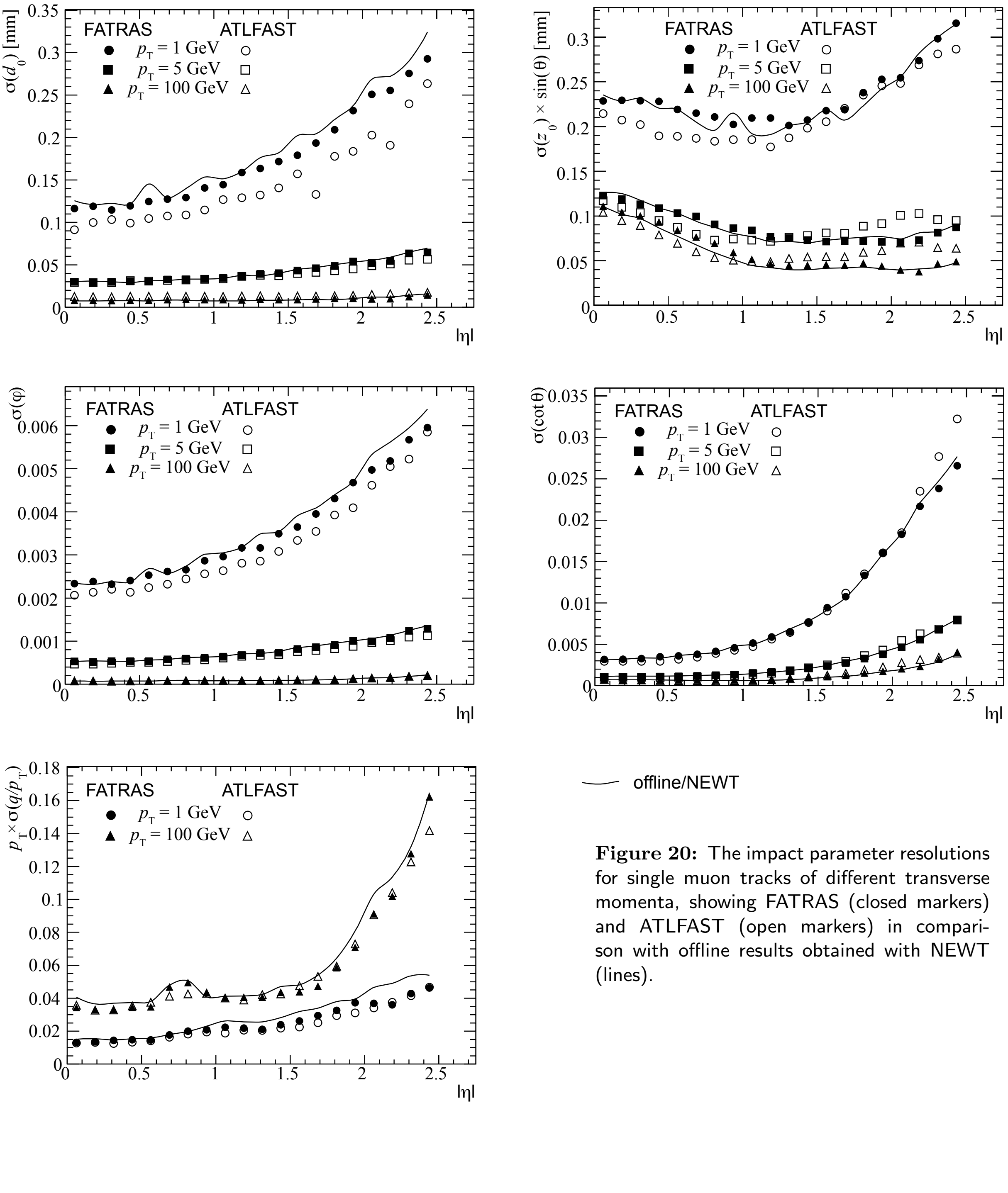


Figure 20: The impact parameter resolutions for single muon tracks of different transverse momenta, showing FATRAS (closed markers) and ATLFAST (open markers) in comparison with offline results obtained with NEWT (lines).

FATRAS shows a remarkable good agreement over the entire pseudorapidity range. The remaining differences are due to the simplified cluster creation and material effects model, but can be minimised with appropriate parameter tuning. ATLFAST follows the general trend of the offline results, but remains however at poorer level of compatibility.

Parameter Correlations One additional aspect of track reconstruction is that the track parameters are strongly correlated, i.e. an incorrect or uncertain estimation of one parameter usually corresponds to associated reconstruction errors of one or few other parameters. The correlations are hereby mainly given by the detector geometry and the magnetic field setup. In a cylindrical detector that is confined in a solenoidal magnetic field (such as the ID) the track trajectory is to a good extent described by a helical track model, which leads to four sets of strongly correlated track parameter pairs: the three transverse track parameter correlations (d_0, ϕ_0), $(d_0, q/p_T)$, $(\phi_0, q/p_T)$, and the correlation between the longitudinal parameters z_0 and θ. A correct description of the track parameter correlations is of particular interest for successive vertex fitting and b-tagging applications, since they rely on the covariance matrix of the track expression. Figure 21 shows a comparison of the four non-trivial track parameter correlations for single muon tracks with $p_T = 10$ GeV between offline, FATRAS and ATLFAST for the entire ATLAS ID. The cross-shaped correlation plot in the longitudinal sector is due to the overlap of barrel and endcap tracks that have different correlation parameters.

Since the correlations between the track parameters are mainly determined by the detector setup and magnetic field setup, it is not surprising that very similar results between FATRAS and the offline reconstruction can be achieved over the entire momentum range of reconstructed particles, see Fig. 22. The weakness of ATLFAST in this respect can probably be explained by several contributing aspects: on the one hand, the correlated smearing of the track parameters in the according ATLFAST module seems to be erroneous, in particular cases where correlation factors bigger than 1 could be identified. On the other hand, the parameterisations of the correlation factors are usually more difficult than for the according track residual parameterisations, since they usually do not follow a similarly trivial function as given through the $A \oplus B$ model[14].

The correlation factor $\rho(\tau_i, \tau_j)$ that relates the reconstruction uncertainty of the track parameter τ_i with the τ_j can be written as

$$\rho(\tau_i, \tau_j) = \frac{\mathrm{cov}(\tau_i, \tau_j)}{\sigma_i \sigma_j}, \tag{19}$$

when $\sigma_i^2 = \mathrm{cov}(\tau_i, \tau_i)$. A comparison of the obtained correlation factors for the single muon tracks over the momentum range of $p_T \in [1 \text{ GeV}, 100 \text{ GeV}]$ is presented in Fig. 22.

4.1.3 Single Electrons Samples

Electron reconstruction is usually more difficult than the reconstruction of muon or pion tracks. This is, on the one hand, mainly due to the contribution of bremsstrahlung to the energy loss of electrons, and on the other hand due to the fact that many electrons in the detector originate from photon conversions and have thus production vertices at large radii. While for the first aspect more elaborated fitting techniques can be applied, latter requires also different pattern recognition strategies[15]. The radiation loss results in a highly asymmetric energy loss distribution and the Gaussian approximation that is inert to most track fitting techniques becomes invalid to a large extend. The resulting *transverse* track parameter resolutions show therefore asymmetric tails and dedicated fitting techniques are often used to account for the non-Gaussian process noise[16]. Figure 23 shows a comparison of the transverse track parameter resolutions for single electron tracks obtained with offline, FATRAS and the ATLFAST simulation for low and high momentum single electron tracks. Both fast simulation techniques achieve asymmetric distributions for the transverse track parameters, but in particular for the momentum measurement FATRAS performs better than ATLFAST.

[14] Further investigations of the falsely calculated correlation parameters have already been started in the ATLFAST community [26].

[15] In ATLAS, a second outside-in reconstruction chain is dedicated to this purpose

[16] Any misinterpretation of the particle momentum — as long as it is within a similar regime for the multiple scattering inclusion — can only affect the track parameters that are directly affected by the momentum measurement. In the ATLAS ID, since the cylindrical geometry is aligned with a solenoidal magnetic field, the affected parameters are the so called *transverse* the parameters (d_0, ϕ) and the transverse component of the momentum p.

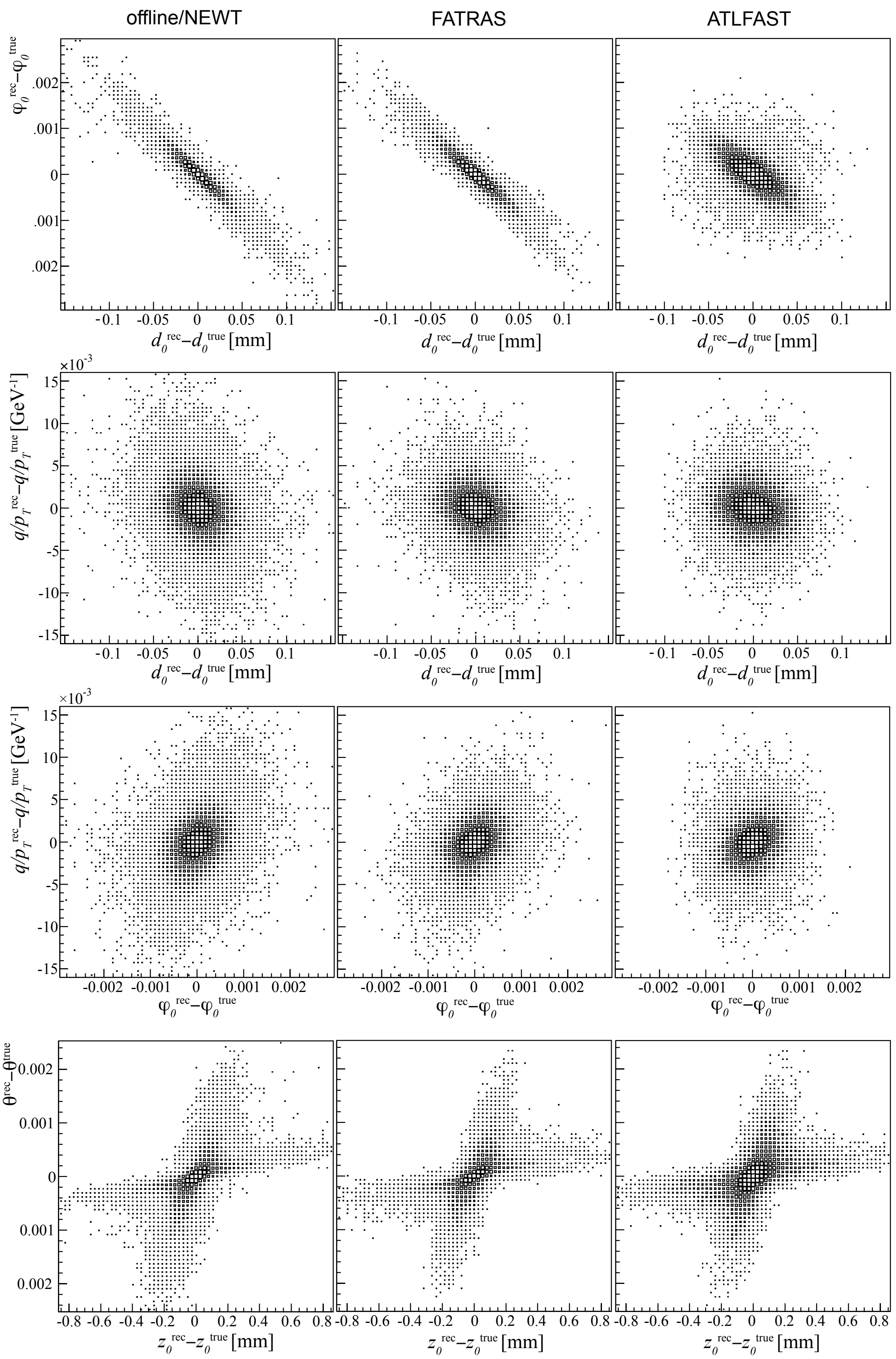


Figure 21: The four non-trivial track parameter correlations for single muon tracks of transverse momentum 10 GeV in comparison between offline, FATRAS and ATLFAST. The first column shows the correlation of the reconstruction errors for the offline reconstruction chain, the second for FATRAS in refit mode. The third column shows the according results obtained with ATLFAST that indicate a problem in the correlated smearing of the transverse track parameters.

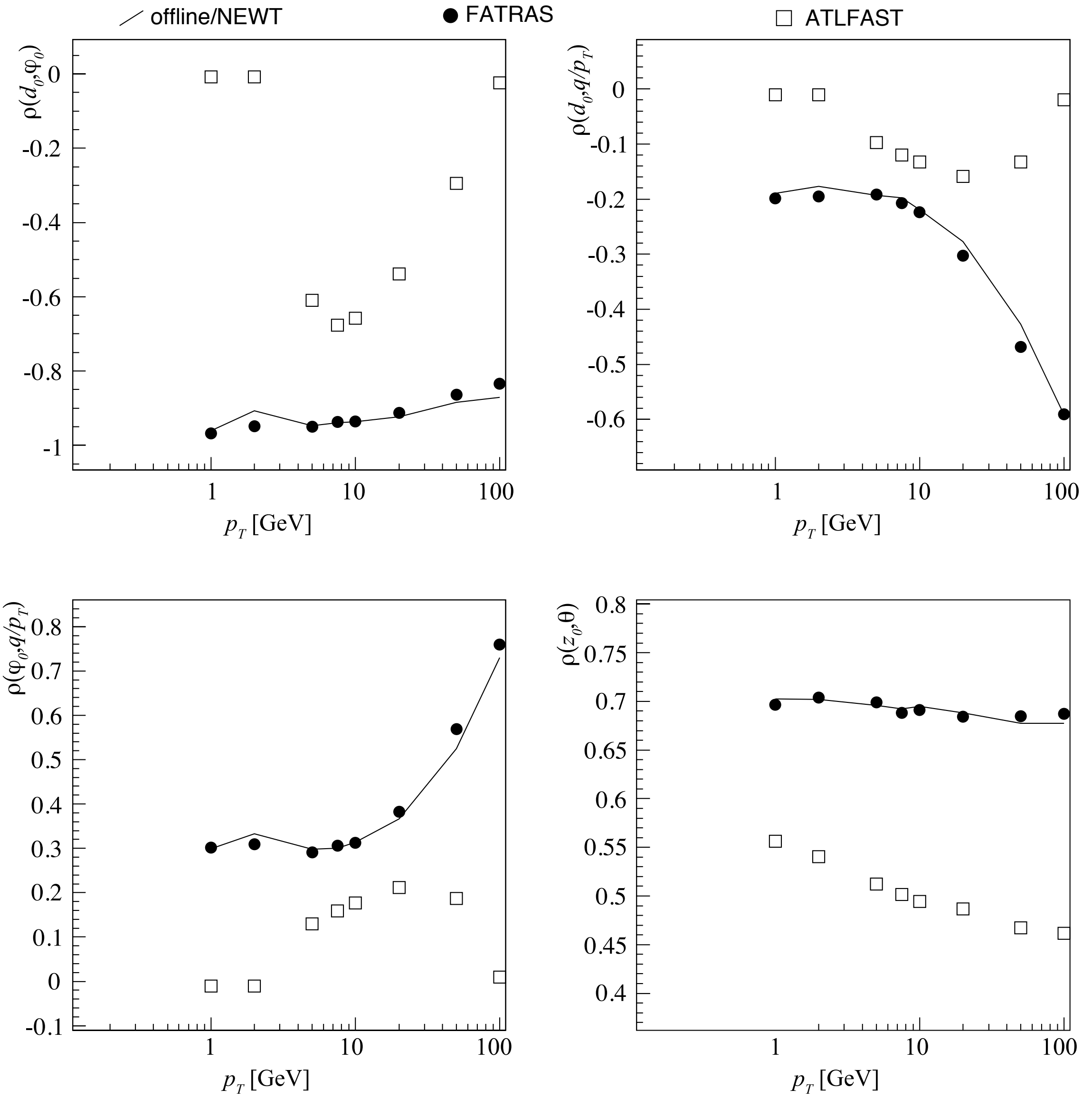


Figure 22: Correlation parameters in dependency of the transverse momentum p_T obtained for the four non-trivial track parameter pairs from offline reconstruction, ATLFAST and the FATRAS simulation.

A striking example for for the additional potential of FATRAS — in particular when comparing it with the ATLFAST track simulation is the application of dedicated electron fitting. The track fit of electron trajectories is somewhat difficult, since due to bremsstrahlung the Gaussian assumption of the included transport error caused by material interactions is not valid. Several dedicated fitting techniques exist in ATLAS that allow to include these effects in the track fit, amongst them a Gaussian Sum Filter (GSF) and a newly developed extension of the Kalman Filter that integrates a dynamic noise adjustment schema (DNA). Both fitting techniques can be applied to tracks that originate from the FATRAS simulation and show a remarkable similarity in comparison to results obtained with the full offline chain, see Fig. 24. FATRAS can thus not only be used for a very refined event analysis, but is also very powerful in helping to develop dedicated techniques for electron fitting[17].

[17] In particular for the GSF FATRAS has extensively used to validate the various component models, but also for achieving stability on large event samples.

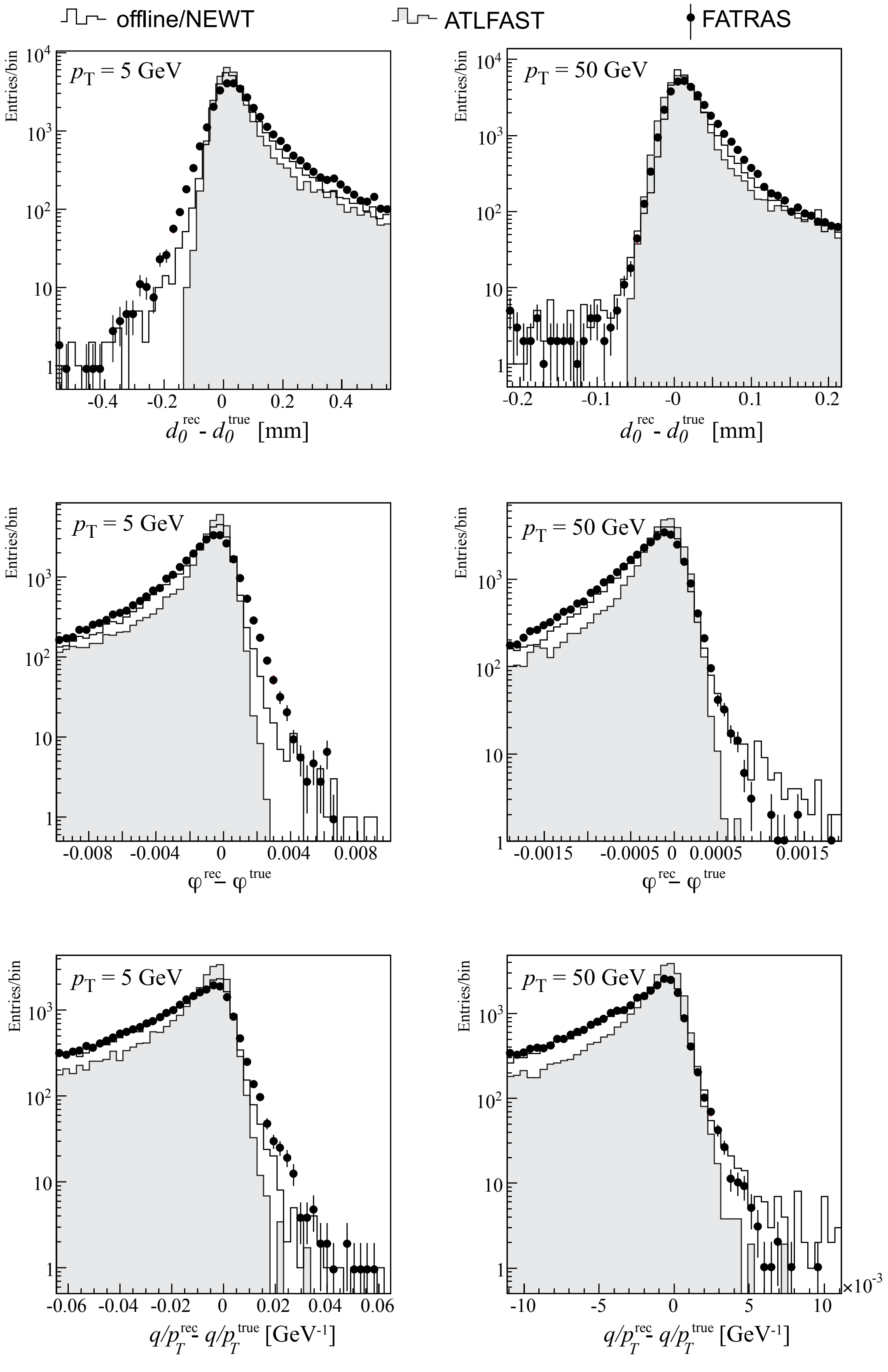


Figure 23: Transverse track parameter resolutions for the low and high (transverse) momentum electron tracks in the ATLAS Inner Detector for the full offline chain, ATLFAST and FATRAS.

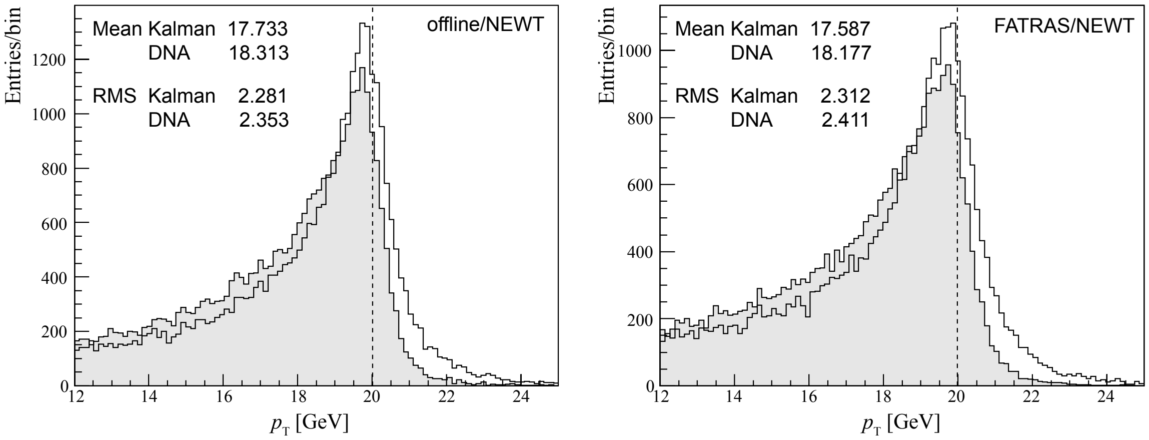


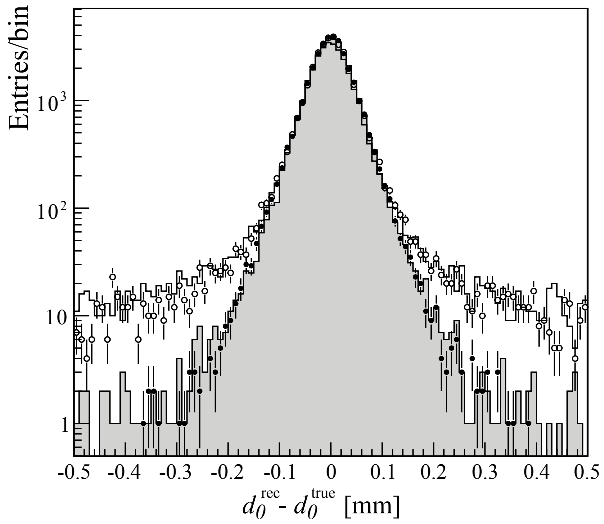
Figure 24: Single electron tracks with transverse momentum of $p_T = 20$ GeV simulated with Geant4 and FATRASand reconstructed with NEWT. In both cases, the initial track collection that has been fitted with a standard Kalman filter has been refitted with an extend Kalman fitter version, that includes dedicated dynamic noise adjustment schema for regulating large energy loss contributions (and resulting directional changes) due to bremsstrahlung effect.

4.1.4 Single Pion Samples

Pions are — similar to muons — mainly influenced by multiple scattering and ionisation loss, but also hadronic or nuclear interactions can occur with the detector material. The correct description of hadronic interactions (in particular of pions) with the detector material is an immense field of interest for the LHC simulation engines [27]. This is mainly because hadronic showers are amongst the most prominent final states to be expected with from proton-proton interactions and in particular for the simulation of the hadronic calorimetry devices the understanding of the shower developments and energy distributions is necessary. In terms of track reconstruction, a naive approach can be taken: if the pion does not undergo substantial hadronic interactions, the particle can be regarded as a minimum ionising particle (MIP) and leads to very similar track resolution as a muon. On the other hand, most of the nuclear interactions disturb the original pion in such a way that it is very unlikely to be found by the pattern recognition and thus, the particle is *lost* for the further event analysis. Latter is reflected in the overall reconstruction efficiency and will be briefly discussed in Sec. 4.2. One other aspect of pion tracks is that they often have production vertices far away from the nominal interaction point; for the ATLFAST simulation this required a special binning of the smearing functions for different vertex radii. Since FATRAS starts the track simulation at the production vertex, effects on the track resolutions originating from fewer precision hits are automatically included.

The additional hadronic interactions with the nuclei of the detector material also disturb the track parameter resolutions and result in bigger uncertainties of the impact parameter resolutions. While the core distribution remains to be similar to the ones obtained with muon tracks, the tail distribution contributes significantly; this is either through effectively shortened track lengths in case of the hadronic interactions stopping the original pion track within the detector, os by secondary shower particles that are misidentified as prompt tracks. Figure 25 shows the impact of hadronic interactions on the estimation of the transverse impact parameter d_0 in comparison for single muon and pions tracks. The agreement of FATRAS and the full offline results is hereby remarkable, although the absolute number of secondary tracks that are found by the reconstruction of FATRAS simulated events is slightly overestimated. Again, an appropriate tuning of the hadronic interaction probability is still missing in FATRAS[18]; a final integration of the nuclear interaction length to the reconstruction geometry description will hopefully account for many of these effects.

[18] In the current example, the probability of a nuclear interaction based on Eq. (9) has been scaled by an additional factor of 2.



	FATRAS	offline/NEWT
Muons	●	□
Entries	33814	33707
RMS	0.0446 mm	0.0469 mm
Pions	○	▨
Entries	37146	34539
RMS	0.070 mm	0.076 mm

Figure 25: The transverse impact parameter resolution d_0 for single muon and pion tracks of transverse momentum 5 GeV . The results of FATRAS and the full offline chain are shown without applying any quality or matching cuts to include secondary tracks caused by hadronic shower interactions.

4.2 Track Reconstruction Efficiencies

Both aspects of track reconstruction — the pattern recognition and the track resolutions — contribute to the final event typology. A dedicated aim of FATRAS is to reproduce similar reconstruction efficiencies than obtained with the full offline chain. For electrons and muons this goal is almost automatically achieved by the fact that FATRAS integrates the standard offline pattern recognition modules. A comparison of the reconstruction efficiencies for muons and pions is between offline and FATRAS can be seen in Fig. 26.

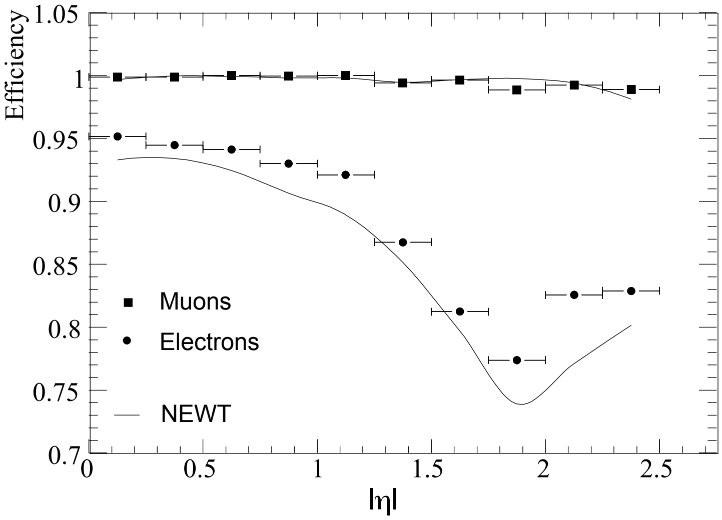


Figure 26: Comparison of reconstruction efficiencies for single electron and muon tracks with transverse momentum of $p_T = 5$ GeV that have been simulated with Geant4 and FATRAS. In both cases, the identical reconstruction chain — the ID New Tracking — has been performed.

For pions (and other hadrons), however, the situation is more difficult since the integration of hadronic interactions is currently limited in FATRAS. This is mainly due to the fact that the nuclear interaction length is not available from the `TrackingGeometry` that has been designed for reconstruction purpose. Currently, the nuclear interaction length λ is approximated by the use of the radiation length, the average atomic number Z, and a global scale factor, see Eq. (9). Figure 27 shows the reconstruction efficiencies of pion tracks, when applying two different global scale numbers in comparison with results from the offline reconstruction. While the general features and characteristics of the η dependence can be reproduced, the radiation length remains only an approximate description of the nuclear interaction length and a global scaling parameter that is valid over the entire pseudorapidity range can not be defined. To conclude the full integration of the hadronic interactions in FATRAS an updated reconstruction geometry model is inevitable. This adaption is planned to be integrated after the major ATLAS release 14.0.0.

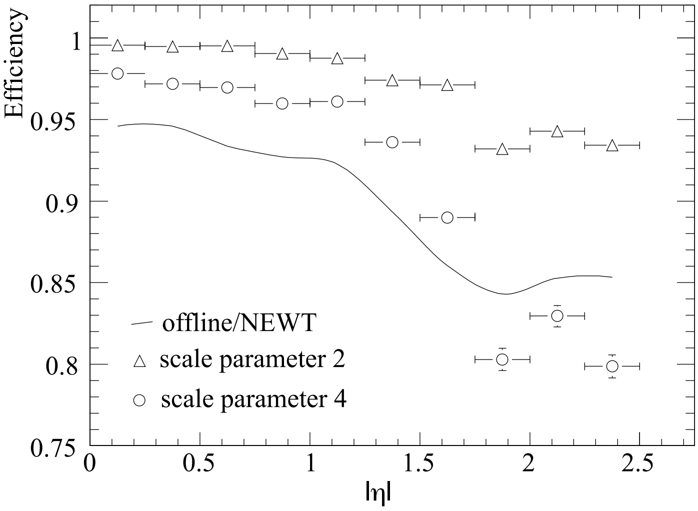


Figure 27: Comparison of reconstruction efficiencies for pion tracks with transverse momentum of $p_T = 5$ GeV that have been simulated with Geant4 and FATRAS. In both cases, the identical reconstruction chain — the ID New Tracking — has been performed.

4.3 CPU Time Consumption

Large simulated event samples are needed for many analyses with the ATLAS detector; the CPU time consumption used for the simulation and reconstruction of these events has become a stringent constraint, not only in a financial manner. Providing a fast alternative to the time consuming full detector simulation was one of the motivations of the FATRAS project. The challenging task of every fast simulation strategy is to find a good balance between the level of accuracy that can be achieved and the computing complexity that goes along with it. It is of little surprise that the ATLFAST simulation, that only performs a smearing of the kinematic variables is the by order of magnitudes fastest track simulation that is currently deployed, but has, on the other hand, the drawback that many aspects of the event reconstruction are not considered at all.

The event reconstruction, which usually only deals with a subset of the simulated information in a mainly probabilistic manner is usually by orders of magnitudes faster than the event simulation. To gain maximum compatibility between FATRAS and the offline chain, it was a necessary assessment that in FATRAS the full reconstruction part is performed. Since the simulation part also relies mainly on offline reconstruction algorithms, a similar execution time as for the event reconstruction can be achieved. In fact, since the pattern recognition imposes a high number of combinatorial track candidates that have to be processed, the pure simulation module of FATRAS is for full physics samples even faster than the event reconstruction.

The following timing tests have been obtained for repeated simulation of 100 inclusive $t\bar{t}$ events using ATLFAST, FATRAS and the full simulation. The event simulation has been hereby entirely limited to the Inner Detector and results are given in normalised CPU seconds (KSI2K) [28].

Table 3: Total simulation (and digitisation) time per inclusive $t\bar{t}$ event for the full Inner Detector, FATRAS and the ATLFAST track maker modules given in normalised CPU seconds.

Simulation Type	**Average Total Time per Event [KSI2K]**
Geant4/offline	≈ 146 $(sim) + 4.3$ $(digit) \pm 4.5$
FATRAS	≈ 2.78 $(total) \pm 0.33$
ATLFAST	≈ 0.0226 $(total)$

Table 3 summarises the normalised CPU time per event for the event simulation and digitisation. For ATLFAST, since simulation, digitisation and reconstruction can not be separated, the total time for the ATLFAST tracking is given; the contribution of event detector simulation and digitisation for the full simulation chain are given individually. The FATRAS timing contribution does not contain the track refitting, a more conclusive comparison between the processing time of FATRAS for refit or reconstruction mode is given in the following section, Sec. 4.3.1. It is demonstrated that the total simulation time spent in the FATRAS modules is lower than the digitisation step of the full offline chain. It is not very surprising, that the ATLFAST simulation has by far the lowest execution time, since only few random number samplings and trivial mathematical operations are performed to smear the generated particle parameters. For FATRAS, a factor of 50 can be easily gained without compromising the event characteristics. The overall computing time of FATRAS is dominated by the transport time of low energetic particles through the detector setup; In the default configuration, the following restrictions are applied:

- particles with $p_T < 250$ MeV are not used for track simulation
- particles with $E < 250$ MeV are not processed in the particle decay
- the track simulation is stopped, when the particle falls under an energy threshold of 50 MeV
- photons with an energy of less than 150 MeV are not further processed in the conversion module

it can be shown that by raising these threshold numbers, an event processing rate per second that is 100 times higher than the full simulation can be achieved. This optimisation, however, should be carried out in careful cooperation with the physics communities to achieve an optimal balance between the physics performance of the simulation module and the computing complexity.

4.3.1 Component Time Consumption

In the full simulation chain, there is a clear separation between the the actual detector simulation and the digitisation module. In fact, most of the time these tasks are even performed in different jobs. In FATRAS, the digitisation is performed during the simulation process, hence it is not possible to give timing figures in such a classical frame. However, the time contribution of the single FATRAS modules — described in Sec. 2 — can be measured independently and are presented with additional comments in Tab. 4.

Table 4: The execution time for the single FATRAS simulation modules and their contribution to the overall event processing time. For the FATRAS refit mode, the additional time that has to be spent in the refitting module is listed below. The time spent in the NEWT reconstruction algorithms is given for data simulated with FATRAS.

Module	**NSI2K**	Av. time [ms]	Min time [ms]	Max time [ms]
`FatrasSecondaryPhotonProcessing`	-	3 (± 1.8)	0	19
`FatrasSecondaryParticleDecay`	-	4 (± 2.5)	0	19
`FatrasCopyMcCollection`	-	6 (± 2.7)	1	18
`FatrasTertiaryTrackCreation`	0.04	19 (± 19)	0	138
`FatrasGenEventSimulation`	0.08	35 (± 19)	8	106
`FatrasPhotonProcessing`	0.11	48 (± 23)	8	164
`FatrasParticleDecay`	0.18	79 (± 58)	1	403
`FatrasHitPostProcessing`	0.39	171 (± 20)	109	285
`FatrasSecondaryTrackCreation`	0.64	282 (± 127)	36	866
`FatrasPrimaryTrackCreation`	1.36	604 (± 295)	109	2150
`FatrasTrackRefitting`	1.52	697 (± 344)	64	3202
NEWT: inside-out sequence	1.50	663 (± 135)	-	-
NEWT: outside-in sequence	0.24	108 (± 38)	-	-
NEWT: 2^{nd} stage pattern	0.56	250 (± 162)	-	-

Table 4 includes the event reconstruction for the reconstruction of FATRAS simulated events in the ID. These numbers are divided into the two main sequences, the inside-out and the outside-in track finding, respectively, and separately for the second stage pattern that includes vertex reconstruction, conversion finding and the particle candidate creation.

The large spread between minimum and maximum time spent in the track creation and refit modules is given by different event morphologies: when many low momentum tracks are being processed (such as being caused by hadronic shower interactions or multiple decay products), a huge increase in the CPU time needed for this event is observed. This is, because the track extrapolation gets immediately slower due to a rapid increase of needed steps in the numerical integration of the magnetic field. This effect can, of course, be cancelled or regulated with appropriate minimum momentum cuts. In the reconstruction modules, most of these low p_T tracks are rejected by the pattern recognition modules that have intrinsic cuts for a minimum particle momentum[19].

5 FATRAS Applications

Besides playing an important role in the validation of the track reconstruction software, FATRAS has so far also been used for applications that could not have been carried out easily with the the full simulation chain. One of which has been a large scale study of the pattern recognition stability in the TRT detector, when increasing the noise (or occupancy) level far beyond 50 %. A small review of the original study [29] is given in the next section, Sec. 5.1.

FATRAS is also used for first upgrade studies of the ATLAS detector that are — amongst other reasons — not possible with the full detector simulation, since no pattern recognition exists for the modified detector layouts. In FATRAS, however, the pattern recognition can be by-passed and first track parameter resolutions or track and hit multiplicity studies could be achieved. A more detailed description of the SLHC studies that have been performed with FATRAS will be presented in the following, Sec. 5.2.

5.1 Occupancy Study in the ATLAS TRT Detector

FATRAS was used to study the influence of the noise occupancy in the ATLAS TRT on the quality of pattern recognition and track fits. In the the ATLAS offline track reconstruction NEWT, tracks from the silicon layers are extended into the TRT by a dedicated reconstruction module. TRT measurements are collected around a cone seeded by the silicon track and local pattern recognition algorithms are used to find the correct measurement to track assignment. Additionally, the left/right ambiguity in each TRT measurement has to be solved[20]. Currently two algorithms exist for this purpose, a Kalman filter based approach and the *Deterministic Annealing Filter* (DAF). A more detailed description of both concepts can be found in [11].

The noise occupancy in the TRT can easily be set to arbitrary values in FATRAS. Noise measurements mask measurements coming from tracks in the noise simulation, such that the track measurements are not visible anymore for the pattern recognition. Therefore the pattern recognition has to cope with additional noise measurements as well as wrong measurements on the particle‘s trajectory at higher noise levels. The noise occupancy in the Pixel and SCT detector was kept at the nominal value during this study, because additional noise measurements in those detectors just increase the number of fake seeds, but do not deteriorate the TRT extended track fit significantly up to reasonable noise levels[21].

The truth matching from the FATRAS validation packages was used to calculate the residuals between simulated track parameters and reconstructed track parameters. A Gaussian fit was performed to the resulting distributions for each noise level and its width is quoted as the resolution of the track parameter. 5000 single muon events with a momentum of 5 GeV were simulated in the TRT barrel range $|\eta| < 0.5$. Figure 28 shows a small example of the mentioned study, that is laid out in more detail in [29]; the left plot shows the uncertainty of the transverse momentum estimation in including

[19] The NEWT reconstruction has, however, a dedicated mode for searching low momentum tracks. This has been designed for early data and minimum bias events. Currently, the combination of this reconstruction mode with FATRAS has not been tested.

[20] This is, because the TRT is a drift time detector, only a circle around the central drift tube is given as the *measurement*.

[21] The reconstruction breaks down due to the enormous number of seeds from combinatorics before the overall performance of the track fit degrades

the TRT barrel measurements in absence of any noise contribution, while the right plot shows the degradation of the momentum resolution with increasing noise level and compares hereby the standard Kalman fitter to the DAF extension.

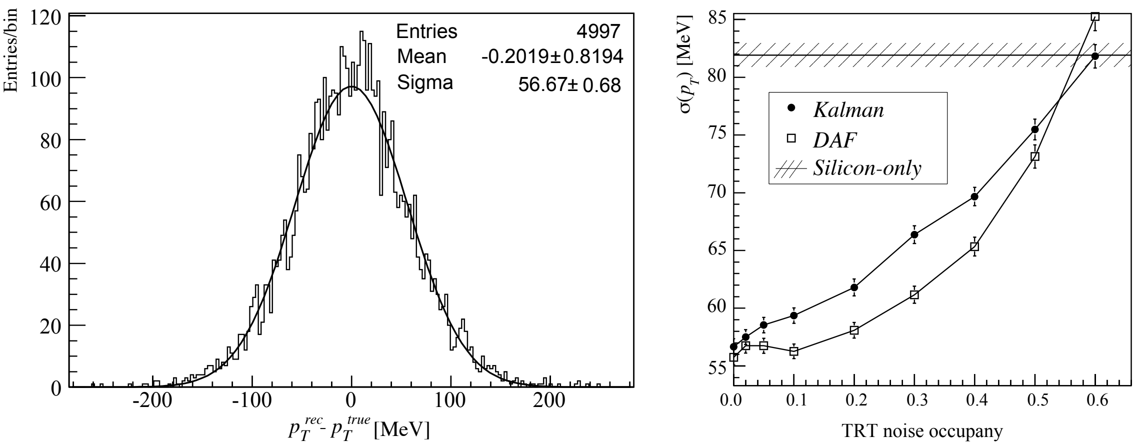


Figure 28: An example application of the FATRAS simulation in a TRT noise level study, comparing the standard Kalman filter to the Deterministic Annealing Filter; the transverse momentum resolution of 5 GeV single muon tracks is shown in the left plot, while the degradation of the momentum resolution with increasing noise level is presented to the right.

FATRAS allows to do a very detailed comparison between simulated and reconstructed tracks. In the noise study this feature was used to investigate how many noise measurements are wrongly assigned to a track. It could even be studied how often the solution of the left/right ambiguity was correct. Such a detailed study is not possible using the full simulation, since the truth information is not sufficiently available there.

5.2 Detector Upgrade Simulations

The preparation of an experiment that is of a comparable size as the ATLAS experiment spans over a long period in time and includes many studies based on Monte Carlo simulation to optimise the detector setup and to estimate the detector performance. Already during the startup phase of the ATLAS experiment, first studies are carried out to simulate different detector setups foreseen to be integrated in the first upgrade phase that is scheduled to take place after about ten years of operation. The detector upgrade of the ATLAS detector will be evoked and accomplished by an according upgrade of the LHC machine — to become the *Super Large Hadron Collider* (SLHC) — with an increased peak luminosity of $10^{35}cm^{-2}s^{-1}$. The increased collision rates, a direct consequence of the higher luminosity, require higher granulated detector devices, in particular for the TRT detector it will become almost impossible to resolve the high track density (although, Sec. 5.1 has shown, that with the use of new pattern recognition techniques, a significantly higher track multiplicity than in the ATLAS startup setup can be handled.). Several layout proposals exist in the meanwhile [30] to exchange the existing ATLAS ID detector with a silicon-only detector structure. For a final decision, however, it is important to test these proposed layouts in both feasibility and performance.

The description of such detector setups is a first requirement for the establishment of a detector simulation. In FATRAS, since the offline reconstruction geometry is used as the simulation geometry, this can realised through a dedicated `TrackingGeometry` description based on generic input parameters. FATRAS provides for this purpose generic detector builders that allow a flexible choice of cylinder and disc detector layouts with both pixel and strip detector technologies. Figure 29 shows a picture of the 3D `TrackingGeometry` for one proposed SLHC layout, illustrated with the ATLAS event display VP1. Custom clusterisation algorithms — e.g. needed for newly established developments such as the 3D-pixel detector — can be included and since the ATLAS offline EDM has been extended with appropriate generic hit and cluster classes, track simulation and refit can be performed very similarly to the ATLAS offline setup.

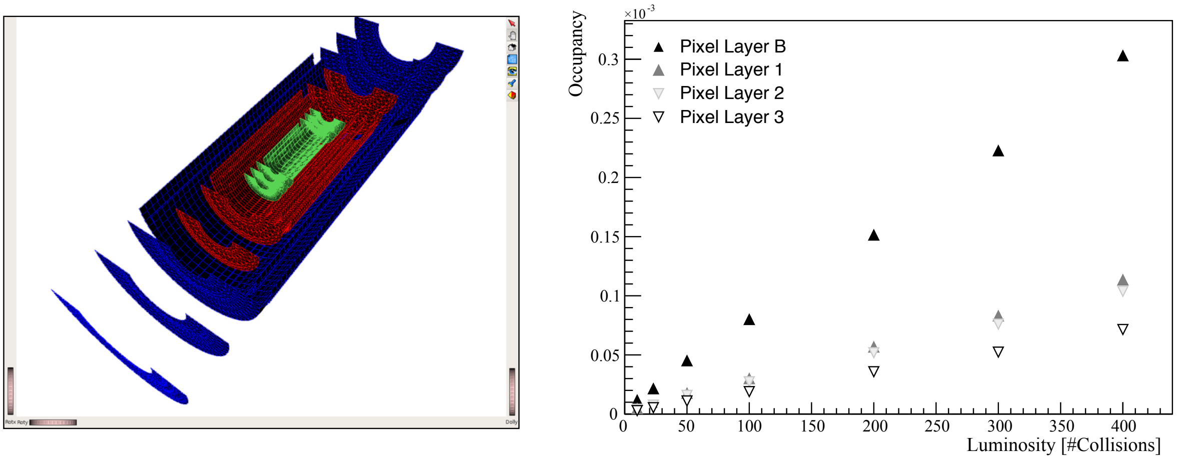


Figure 29: Left: A picture of one proposed SLHC geometry for the ATLAS ID produced with the ATLAS event display VP1. The upgrade scenario for the ID foresees the complete exchange of the ID tracker with a purely silicon based pixel vertex and strip detector in 2015. A framework for a flexible change between different layouts has been implemented in FATRAS to ease the comparison of various upgrade scenarios. Right: Example for a hit occupancy study on the pixel layers of the four pixel barrel cylinders depending on the number of pile-up events.

6 Conclusion and Outlook

We have presented the concepts, implementation and first performance figures of the new ATLAS fast track simulation application FATRAS. FATRAS is a full Monte Carlo simulation that is based on the reconstruction geometry and offline reconstruction tools and establishes a third simulation technique for ATLAS alongside to the full detector simulation and the parametric smearing approach. It agrees to a large extend with results obtained by full simulation and offline reconstruction, while using only a fraction of CPU time. The FATRAS simulation could be established to yield event processing rates of at the level of 1 Hz; together with the consecutive track reconstruction, a total event processing time of about 2.5 seconds for the ATLAS ID could be achieved.

Good agreement with the full offline chain could be met for track parameter resolutions, hit and track multiplicities but also for reconstruction efficiencies of muon and electron tracks. Latter is achieved by simply using FATRAS as an input for the standard offline track reconstruction software. FATRAS combines the possibility to use very detailed event analysis techniques with fast execution times and is thus a perfect tool for the development of a new tracking based analysis before applying it on full simulated or taken data.

6.1 Muon System and Calorimeter Fast Shower Simulation

FATRAS is in the current state limited to the the Inner Detector (and its SLHC upgrade version), mainly for the fact that a purely predictive navigation in the detector geometry that is needed for the trajectory creation has only been exiting for ID. Recently [31], first attempts have been carried out to use the more complex `TrackingGeometry` of the Muon System for a similar trajectory creation approach that is also based on the extrapolation engine. These promising results open the window to a future full fast track simulation in the ATLAS detector, since many of the existing FATRAS modules, such as the material effects integration and the clusterisation can be to a large extent re-used for the Muons System[22].

FATRAS establishes a fast alternative to the full simulation, but the time spent in tracking detectors contributes only partially to the overall CPU time consumption of the detector simulation. The simulation of the calorimetry response is still the largest fraction of all sub-detectors. This is, because

[22] In the most optimistic scenario, the integration of the Muon System requires only the creation of a dedicated track creation tool that extends the FATRAS `ITrackCreator` interface.

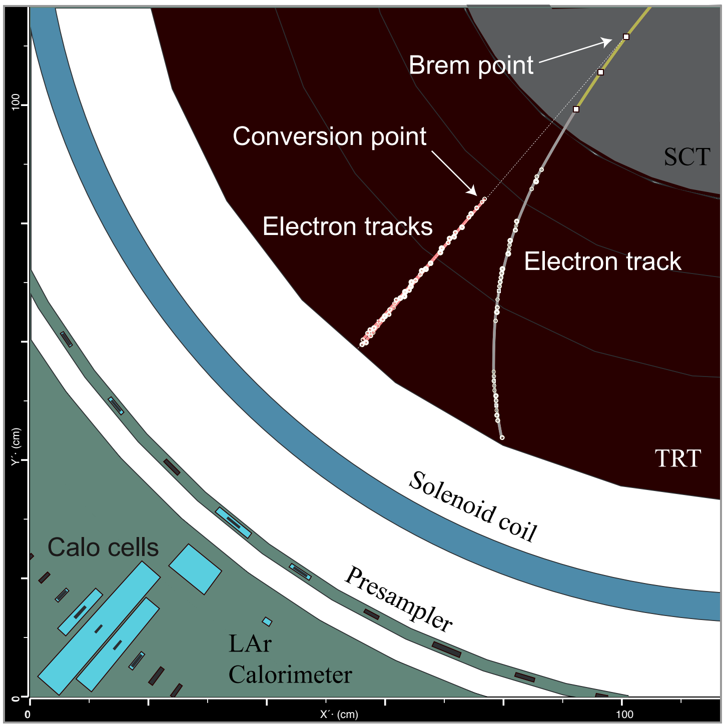


Figure 30: A detail of a simulated event shown with ATLANTIS using FATRAS for the Inner Detector simulation and the `FastCaloSim` for the simulation of the calorimetric response. The picture shows an electron track that undergoes an emission of a high energetic photon that converts into an almost collinear electron-positron pair in the TRT detector. The main part of the initial electron energy is hereby carried away by the brem photon (and then the conversion products), which leads to the cluster measurement in the electromagnetic calorimeter.

almost every particle causes shower reactions in the calorimeter and the number of particles to be tracked is multiplied by orders of magnitudes. In addition, the material distribution in the calorimeter is far denser than in tracking devices where a discrete detector setup of alternating detection layers and air- or gas-filled volumes is predominately chosen. Besides the constant optimisation of the Geant4 based simulation, recent work [32] has focussed on creating parameterised shower models that allow fast shower simulations on basis of the kinematic properties and the type of the particle. FATRAS has been expanded with the full Monte Carlo truth tree and by a dedicated module — see Sec. 3.6 — to prepare the track simulation output for a successive use in fast shower simulations. Figure 30 shows an ATLANTIS event display detail of an event that was simulated after interfaces the ID FATRAS simulation with the fast shower parameterisation[23].

The inclusion of the fast calorimeter simulation enables standard jet reconstruction and b-tagging algorithms to be performed on the combined output. A next step is thus to validate this extended usage in future event analyses. In a recent effort, a complete truth tree with an identical structure to the one produced by the full detector simulation has been established in FATRAS. This allows standard truth matching algorithms to work on FATRAS output and encapsulates the user analyses finally from the underlying used simulation technique.

A fully operational and validated FATRAS setup that includes first user feedback and a detailed parameter tuning is expected for the ATLAS offline release 14.0.0.

A Appendix

A.1 Typesetting and Nomenclature

The following type setting conventions are followed throughout this document: Software packages within the ATLAS offline software repository [33] are written in Sans-serif face, C++ or python class names are written in `Courier` face. Namespace definitions as used in the software repository are omitted in this document for readability. A exhaustive list of software packages and their location

[23] FastCaloSim

within the ATLAS software repository can be found in Sec. A.6.

A.2 Particle Decay Table

As described in Sec. 3.2, FATRAS offers two different ways to include particle decay processes in the track simulation: a simplified internal decay-model that supports only a few dedicated decay channels and the (default) full decay module that wraps the appropriate decay engines of Geant4. Latter is to a full extent described in [23]; for completeness, the supported decay modes and branching ratios of the simplified version is given in Tab. 5.

Table 5: Supported decay channels and branching ratios of the simplified FATRAS decay module.

Particle	Branching Ratio	Decay Channel
π_0	0.98798	$\pi_0 \rightarrow \gamma + \gamma$
	0.01198	$\pi_0 \rightarrow e^- + e^+ + \gamma$
$K^\pm$	0.63440	$K^\pm \rightarrow \mu^\pm \nu_\mu$
	0.20920	$K^\pm \rightarrow \pi^\pm \pi_0$
	0.05590	$K^\pm \rightarrow \pi^\pm \pi^\mp \pi^\pm$
	0.04980	$K^\pm \rightarrow \pi_0 e^\pm \nu_e$
	0.03320	$K^\pm \rightarrow \pi_0 \mu^\pm \nu_\mu$
	0.01757	$K^\pm \rightarrow \pi^\pm \pi_0 \pi_0$
K^0_S	0.69200	$K^0_S \rightarrow \pi_0 \pi_0$
	0.30690	$K^0_S \rightarrow \pi^+ \pi^-$
K^0_L	0.40530	$K^0_L \rightarrow \pi^\pm e^\mp \nu_e$
	0.27020	$K^0_L \rightarrow \pi^\pm \mu^\mp \nu_\mu$
	0.19560	$K^0_L \rightarrow \pi_0 \pi_0 \pi_0$
	0.12560	$K^0_L \rightarrow \pi^\pm \pi^\mp \pi_0$

A.3 Additional Iteration on Conversion and Brem Photon Handling

In the initial configuration, FATRAS has been limited to one iteration of photon transport through the detector and its associated conversion creation. This lead to an underestimation of low energetic photons and electron/positron tracks. Although it is not very likely that these tracks are found by the main reconstruction sequence, they are still of interest since they create patterns that are found by the second reconstruction sequence, the so-called backtracking, or simply add traces that may disturb the pattern recognition process. It could be shown that the inclusion of one additional iteration in FATRAS improves the compatibility with full simulated results significantly, while adding only a tiny addition timing fraction to the overall simulation time. Figure 31 shows a momentum comparison of radiated brem photons in FATRAS with two different Geant4 configurations: the default Geant4 setup (blank histogram) and the artificially cut configuration that stops just as FATRAS after one iteration. The remarkable good agreement of FATRAS with the also restricted Geant4 results proves the quality of the bremsstrahlung model in FATRAS alongside with the accurate material description of the reconstruction geometry.

A.4 Little User Guide and Steering

The FATRAS steering is in full coherence with the overall ATHENA job configuration schema that has been in the last year adapted to a modern job configuration framework based on python [34]. Single components of the actual reconstruction flow are entirely configured through auto-generated python configuration classes, while parameters and overall steering flags are concentrated in so-called *JobProperty* classes. The FATRAS `JobProperty` classes can be found in the FatrasExample/python directory and are listed — including some brief description — in Tab. 6.

The single property flags can be modified by the user to change the (default) behavior of the FATRAS simulation process; they include overall simulation steering (such as e.g. the run mode, the detector

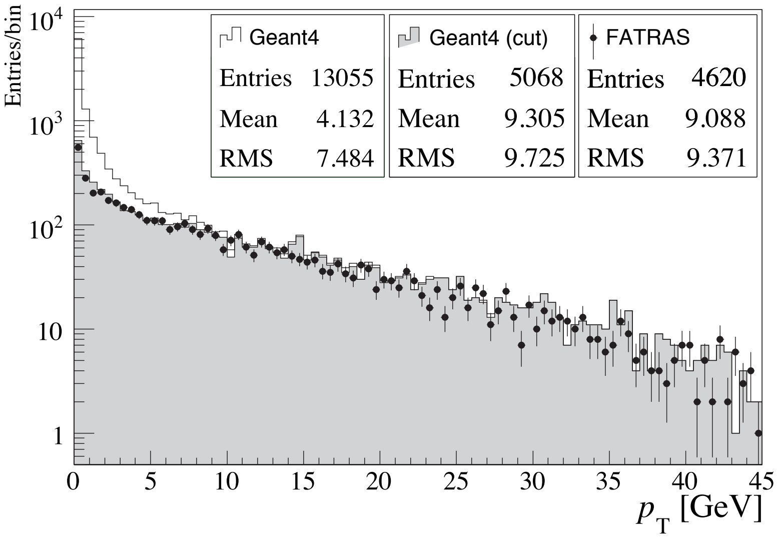


Figure 31: Photon momenta of radiated brem photons simulated with FATRAS and Geant4. The blank histogram shows the standard Geant4 setup, while the shaded area shows the Geant4 results when being restricted to one single photon/conversion iteration (i.e. the identical setup to FATRAS in this comparison).

Table 6: `JobProperty` python classes for the FATRAS job configuration.

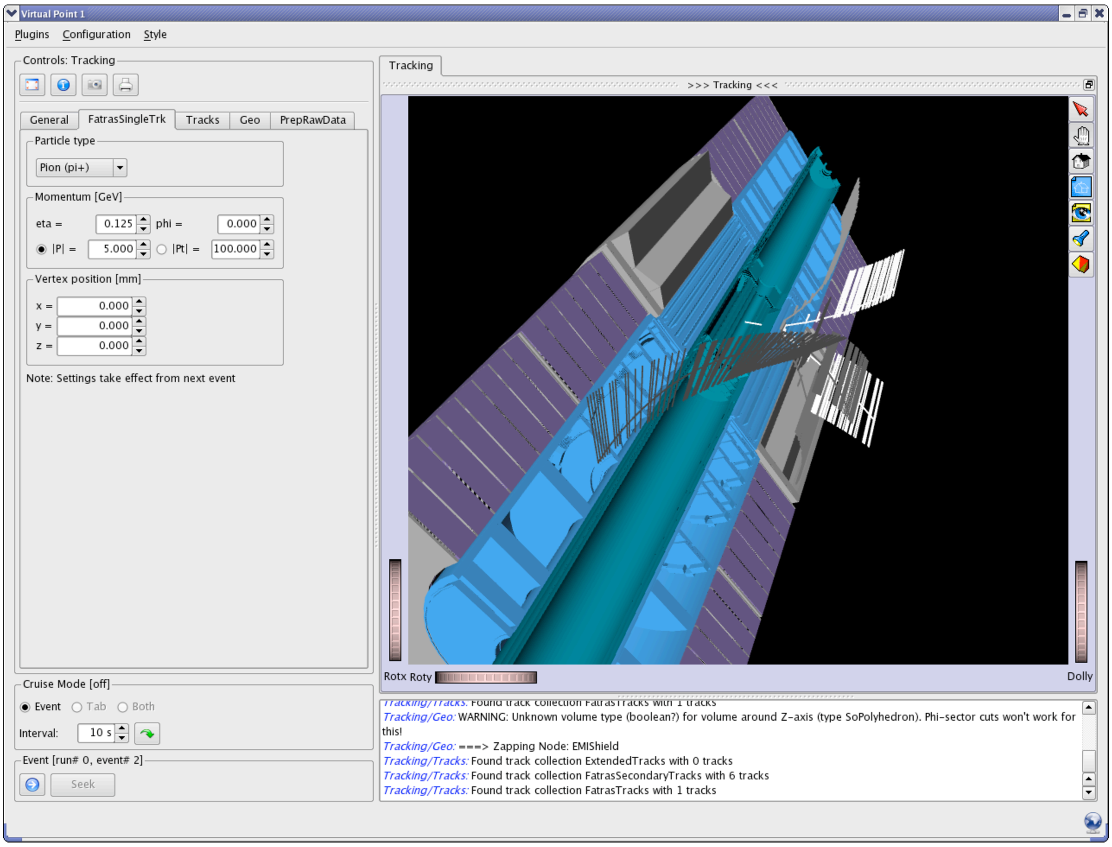
JobProperty	**Description**
`FatrasJobProperties`	general FATRAS steering: modes, geometry, auditors
`FatrasKeys`	names of FATRAS collections in the transient event store
`FatrasSingleTrackSimulation`	dedicated steering for single track simulation
`FatrasPhysicsList`	switches for physics processes
`FatrasClusterCreation`	tuning parameters for hit creation and clustering
`FatrasTuning`	remaining tuning parameters: e.g. minimum energy cuts, probability scaling for physics processes
`FatrasValidation`	definition of validation modules
`FatrasOutputLevels`	screen output steering for different FATRAS modules

setup, the physics processes to be included), but also contain tuning parameters that can be used to adjust the single FATRAS modules. A more detailed description of all the contained tuning possibilities are further described in the following, Sec. A.5. Once the `JobProperty` classes are defined, one single python script (`Fatras_jobOptions.py`) is parsed and the algorithmic sequence and the necessary tool and service classes are created according to the chosen parameters and configurations. For convenience, these tasks are sub-divided into different python setup classes: `FatrasGeometry`, `FatrasExtrapolation`, `FatrasFitting`, `FatrasTools`, `FatrasAlgs`, `FatrasReconstruction`, and `FatrasValidation`.

Stand-Alone Jobs The easiest example how to execute FATRAS is in a stand-alone job that only performs the FATRAS simulation and the default offline NEWT reconstruction. A user-friendly run configuration script can for this purpose found in the FatrasExample/share as `jobOptions.py`. It configures the default FATRAS properties and executes the `Fatras_jobOptions.py` script.

FATRAS Integration FATRAS aims to provide a fast alternative to the full offline chain that includes event simulation, digitisation and reconstruction. It is thus of great interest to provide a coherent integration into the standard ATLAS reconstruction job configurations. Taking account of this, the `Fatras_jobOptions.py` flags also include a full reconstruction and post processing setup for the Inner Detector and can be used in general ATLAS reconstruction setups replacing the standard ID reconstruction configuration (`InDetRec_jobOptions.py`)[24].

VP1 Plugin A recent development established a dedicated FATRAS plugin for the rapidly growing ATLAS 3D event display VP1. It includes a single particle generation that is coupled to the FATRAS simulation sequence and allows thus a precise single track debugging together with a convenient graphical user interface. Figure 32 shows a screenshot of the VP1 user interface including the FATRAS steering panel to the left, and the actual 3D display canvas to the right: it shows a π^+ particle with a momentum of 5 GeV that causes a hadronic shower with the detector material.



Particle type: Pion (pi+)

Momentum [GeV]

eta = 0.125 phi = 0.000

● |P| = 5.000 ○ |Pt| = 100.000

Vertex position [mm]

x = 0.000

y = 0.000

z = 0.000

Note: Settings take effect from next event

Cruise Mode [off]

● Event ○ Tab ○ Both

Interval: 10 s

Event [run# 0, event# 2]

```
Tracking/Tracks: Found track collection FatrasTracks with 1 tracks
Tracking/Geo: WARNING: Unknown volume type (boolean?) for volume around Z-axis (type SoPolyhedron). Phi-sector cuts won't work for
this!
Tracking/Geo: ===> Zapping Node: EMIShield
Tracking/Tracks: Found track collection ExtendedTracks with 0 tracks
Tracking/Tracks: Found track collection FatrasSecondaryTracks with 6 tracks
Tracking/Tracks: Found track collection FatrasTracks with 1 tracks
```

Figure 32: The VP1 FATRAS plugin for single particle event simulation. The current event shows a π^+ particle that causes a hadronic shower after interaction with the detector material.

A.5 Fit And Tuning Parameters

FATRAS offers the possibility to change its default behavior through the modification of a set of different tuning and cut parameters. Most of these parameters can be directly set through the `JobProperty` classes as described in the previous section. In the following, a complete list of the tuning parameters are given and their effect on the standard reconstruction will be described:

- the `FatrasClusterCreation` property class holds boolean flags for changing the clusterisation strategies. Simplified hit smearing can be chosen for each sub-detector, but also more complex clustering, such as the geometrical cluster creation in the silicon detectors (default) and the drift circle error description for the TRT from the offline conditions data (default). It also contains the scaling parameters for the high threshold hit simulation based on the transition radiation function used in the TRT.

[24] As a single requirement, the input collection has to be restricted to generator output, since both simulation and digitisation would have already filled event data collections that are otherwise filled by FATRAS.

- the majority of the tuning parameters can be found in the `FatrasTuning` property class. It includes the several scaling parameters for the implemented physics processes, the definitions of volumes where these processes are carried out. In addition, energy and momentum threshold cuts can be set and the noise levels in the FATRAS postprocessing modules can be adjusted. The given scaling parameters and their current default values are shown in Tab. 7.

There are certain cuts that seem to be contradictive, such as the creation of decay and conversion product at 250/150 MeV without a successive track simulation (p_T cut of 250 MeV). This is due to the fact, that when created particles would still be filled into the truth event collection. This behavior can, however, be simply switched off by adjusting the thresholds accordingly.

Table 7: `FatrasTuning` parameters for changing the default FATRAS simulation setup.

Parameter	**Default Value**
`MinimumParticlePtPrimaryTrackCreation`	250 MeV
`MinimumParticlePtSecondaryTrackCreation`	250 MeV
`MinimumParticlePtTertiaryTrackCreation`	250 MeV
`MinimumBremPhotonMomentum`	150 MeV
`BremCreationVolumeRadius`	1200 mm
`BremCreationVolumeHalfZ`	4000 mm
`BetheHeitlerScalor`	1 MeV
`ConversionMinChildEnergy`	150 MeV
`ConversionChildEnergyScalor`	2.
`ConversionProbabilityScalor`	0.98
`ConversionVolumeRadius`	1200 mm
`ConversionVolumeHalfZ`	4000 mm
`PixNoiseLevel`	10^{-6}
`SctNoiseLevel`	10^{-5}
`TrtNoiseLevel`	0.02

There are some additional model parameters that are mainly needed in the simulation of hadronic interaction, the photon conversions or in the clusterisation modules. All of which are adjustable but no public interface for the user is given since they are strongly coupled with the integrated fit or model functions; their behavior with completely random input variables can not be predicted.

The Component Software Model in FATRAS One important way of changing the FATRAS simulation has been missed out in this context: since the entire simulation chain is written in a component model software style with well-defined interfaces, several modules that include the core simulation processes in FATRAS can be exchanged with other plugins that follow completely different approaches or paradigms. The main interfaces used for FATRAS are contained in the TrkDetDescrInterfaces CVS location for the geometry description, the TrkExInterfaces concerning extrapolation and material integration and the FATRAS internal interface locations, see the following Sec. A.6. This choice of design should ease future adaptions and corrections of FATRAS, either in context of changed physics objectives or simply as an outcome of a validation at larger scale.

A.6 Package Overview

Many components of FATRAS are part of the general reconstruction software of ATLAS, in particular of the Tracking repository, while some of the hit classes are directly taken from the InnerDetector container. The central part of FATRAS that includes the `Algorithm` sequence and the steering of the simulation, on the one hand, and the dedicated Monte Carlo based components for the extrapolation engine such as the track creation engine, on the other hand, is concentrated in a dedicated container package of the ATLAS CVS repository. A complete overview of the contained packages including a brief description of their content is given in Tab. 8 for the convenience of the user.

Table 8: The JobProperty classes available in FATRAS for job configuration.

Package	**Content**
FatrasAlgs	main Algorithm classes that build the FATRAS sequence
FatrasDetDescrExample	example package for generic geometry (SLHC)
FatrasDetDescrInterfaces	an interface package for generic geometry building
FatrasDetDescrSvc	FATRAS geometry service steering
FatrasDetDescrTools	AlgTool classes for generic geometry creation
FatrasDetDescrUtils	data classes for generic geometry
FatrasEvent	extended FATRAS event data model
FatrasEventAthenaPool	persistency converter classes
FatrasExample	example package for FATRAS steering, tuning and execution
FatrasG4Algs	wrapper Algorithm for Geant4 based particle decay
FatrasG4Tools	wrapper AlgTool for Geant4 based particle decay
FatrasInterfaces	all specific interface definitions
FatrasTools	dedicated FATRAS AlgTool classes
FatrasTruth	truth data model
FatrasValidation	validation Algorithmclasses

References

[1] T. Sjörstrand, S. Mrenna and P. Skands, *PYTHIA 6.4 - Physics and Manual*, hep-ph/06013175

[2] G. Corcella et al., *HERWIG 6.5*, hep-ph/0210213

[3] Agostinelli et al., *GEANT4: A Simulation toolkit*, Nucl. Inst. & Meth., **A 506**, 2003.

[4] Atlfast homepage, *www.hep.ucl.ac.uk/atlas/atlfast*

[5] F. Akesson et al, *The ATLAS Tracking Event Data Model*, ATLAS Public Note, ATL-SOFT-PUB-2006-004, 2006.

[6] A. Salzburger, M. Wolter and S. Todorova, *The ATLAS Tracking Geometry Description*, ATLAS Note, ATL-SOFT-PUB-2007-004, 2007.

[7] A. Salzburger, *The ATLAS Extrapolation package*, ATLAS Note, ATL-SOFT-PUB-2007-005, 2007.

[8] A. Salzburger, *The new Fast ATLAS Track Simulation (FATRAS)*, Proc. of CHEP 2006, 2006.

[9] Atlantis homepage, *www.cern.ch/atlantis/*

[10] The ATLAS Collaboration, *ATLAS Computing Technical Design Report*, ATLAS TDR, CERN-LHCC-2005-022, 2005.

[11] A. Salzburger (Editor) et al, *Concepts, Design and Implementation of the ATLAS New Tracking (NEWT)*, ATLAS Note, ATL-SOFT-PUB-2007-002, 2007.

[12] VP1 homepage, *http://atlas-vp1.web.cern.ch/atlas-vp1/*

[13] HepMC homepage, *https://savannah.cern.ch/projects/hepmc/*

[14] V.L. Highland, *Some Practical Remarks on Multiple Scattering*, Nucl. Inst. & Meth. **129**, 1975, and *Erratum*, Nucl. Inst. & Meth., **161**, 1979.

[15] R. Frühwirth and M. Regler, *On the quantitative modelling of core and tails of multiple scattering by Gaussian mixtures*, Nucl. Inst. & Meth. **A 456**, 2001.

[16] R. Frühwirth and M. Liendl, *Mixture models of multiple scattering: computation and simulation*, Comp. Phys. Comm. **141**, 2001.

[17] L.D. Landau, *On the energy loss of fast particles by ionisation*, J. Phys. USSR, **8**, 1944.

[18] H. Bethe and W. Heitler, *On the stopping of fast particles and the creation of positive electrons*, Proc. Roy. Soc. **A 146**, 1934.

[19] R. Frühwirth and A. Strandlie, *Track finding and fitting with the Gaussian-sum Filter*, Proc. of CHEP 1998, 1998.

[20] T. Lari, *Study of silicon pixel sensors for the ATLAS detector*, CERN-THESIS-2001-028, 2001.

[21] E. .J Buis, R. J. Dankers, S. Haywood and A. Reichold, *Parameterisation of the Inner Detector Performance*, ATLAS Note, INDET-97-195, 1997.

[22] Yung Tsai, Rev.Mod.Particle Physics Vol. 74, No.4, October 1974

[23] Geant4 homepage, *http://www.cern.ch/geant4*

[24] ROOT homepage, *http://root.cern.ch*

[25] A. Salzburger, *A Parametrization for Fast Simulation of Muon Tracks in the ATLAS Inner Detector and Muon System*, CERN-THESIS-2004-051, 2003.

[26] A. Nairz, *private communication*

[27] Physics validation project homepage, *http://lcgapp.cern.ch/project/simu/validation/*

[28] KSI2K definition, *http://computingcourier.web.cern.ch/ComputingCourier/NL_2004Nov/Batch-Time-Units.doc*

[29] S. Fleischmann, *Track Reconstruction in the ATLAS Experiment : The Deterministic Annealing Filter*, CERN-THESIS-2007-01, 2006.

[30] ATLAS Upgrade homepage, *http://atlas.web.cern.ch/Atlas/GROUPS/UPGRADES/*

[31] S. Todorova, *private communications*

[32] FastCaloSim twiki page, *https://twiki.cern.ch/twiki/bin/view/Atlas/FastCaloSim*

[33] Atlas offline software repository, *http://atlas-sw.cern.ch/cgi-bin/viewcvs-atlas.cgi/offline/*

[34] P. Calafiura et al., *Physics-Level Job Configuration*, Proc. of CHEP 2006, 2006.