The Fast Simulation Chain in the ATLAS experiment

Martina Javurkova[1,*], on behalf of the ATLAS collaboration

[1]University of Massachusetts (US)

Abstract. The ATLAS experiment relies heavily on simulated data, requiring the production of billions of simulated proton-proton collisions every run period. As such, the simulation of collisions (events) is the single biggest CPU resource consumer. ATLAS's finite computing resources are at odds with the expected conditions during the High Luminosity LHC era, where the increase in proton-proton centre-of-mass energy and instantaneous luminosity will result in higher particle multiplicities and roughly five-fold additional interactions per bunch-crossing with respect to LHC Run-2. Therefore, significant effort within the collaboration is being focused on increasing the rate at which Monte Carlo events can be produced by designing and developing fast alternatives to the algorithms used in the standard Monte Carlo production chain.

ATL-SOFT-PROC-2021-017
25/06/2021

1 Introduction

In order to enable the ATLAS collaboration to pursue its ambitious physics research program, very large simulated collision event samples are required. The need for simulated events increases with the number of proton-proton collision data events collected from the ATLAS experiment [1] at the Large Hadron Collider (LHC) [2]. Therefore, the problem of creating samples of sufficient size will be crucial during the Run 3 data taking starting in 2022 and even more essential at the High Luminosity LHC, from 2027. These samples are produced with the ATLAS Monte Carlo (MC) chain which consists of different production steps, each providing a different output format: event generation (EVNT), detector simulation (HITS), digitization (RDO), reconstruction (AOD) and derivation (DAOD). For completeness, the data processing includes only three steps: trigger (RAW), reconstruction (AOD) and derivation (DAOD). The traditional way to model particles traversing the ATLAS detector is a very accurate simulation provided by the Geant4 framework [3, 4], which predicts the response of the complex detector taking into account various materials and particle properties. However, computing resources are limited and the requirements for large-scale MC production will increase significantly in the upcoming years. As can be seen from Figure 1, the simulation of the ATLAS detector was the biggest CPU resource consumer during the Run 2 data taking. Therefore efforts are focused on reducing CPU time spent on simulations. Once very fast detector simulation is used, digitization and reconstruction become the dominant consumers of CPU time in the MC production chain.

Full detector simulation and fast detector simulation are both currently being used by the ATLAS collaboration for physics results at approximately the same rate. The Geant4 toolkit, used in full simulation, is accurate for different types of particle interactions with

*e-mail: martina.pagacova@cern.ch

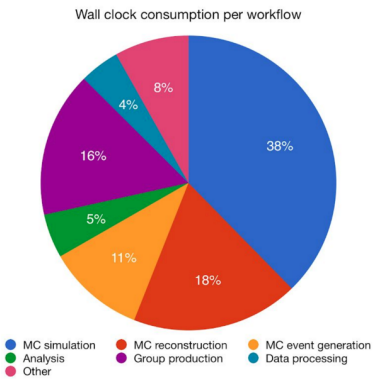


Figure 1: Relative CPU grid usage by the ATLAS experiment in 2018. The "MC reconstruction" refers to the reconstruction as well as the digitization while the "Group production" mainly to the derivations. Taken from [5].

different materials. Since the most time-consuming part is calorimeter simulation, a fast parametrized-response simulation of the calorimeters (FastCaloSim) [6] was developed to address this issue. FastCaloSim together with Geant4 (used to simulate all the detector excluding the calorimeter system) form the ATLFASTII simulation configuration, which reduces the simulation time by one order of magnitude [7]. Fatras [8] is a fast simulation of charged particle propagation in the ATLAS tracking detector with the simplified geometry. The ATLFASTIIF_G4MS simulation flavor uses Fatras for the Inner Detector, FastCaloSim for the calorimeters, and Geant4 for the Muon System. ATLFAST3 [9] is the next generation of high precision fast simulation in ATLAS. It implements two approaches of shower generation, an improved parametrization-based modeling referred to as FastCaloSim V2 [10] and a Generative Adversarial Network (GAN) based modeling referred to as FastCaloGAN [11]. A significant improvement in speed comes at the cost of modelling accuracy so these fast simulation approaches can only be used by some physics analyses.

The Fast Chain has been designed to be flexible in combining fast and full simulation tools in order to meet specific computing and modeling accuracy requirements. All faster alternatives for the components of the full chain that were developed by the collaboration in recent years are included in Fast Chain. A description of those which are developed (not only) for use in Fast Chain as well as those that require special integration to Fast Chain is given below. The last two Sections of this paper describe the tools currently used to monitor the performance of the Fast Chain and the continuous integration with the main ATLAS software framework (Athena) [12].

2 Simulation of additional proton-proton interactions

2.1 RDO-overlay

In the Run 2 reprocessing campaign and during Run 3, the ATLAS experiment will combine hard-scatter and pile-up events at the detector digitization format (RDO) level. This method, referred to here as RDO-overlay, accurately describes additional pp collisions by overlaying pre-digitized MC events [13, 14]. A large sample ($O(10^9)$) of presampled pile-up events will be produced from simulated minimum bias events during a separate digitization step, called pre-sampling, and written out as presampled (pile-up) RDO datasets. After each

simulated hard-scatter event is digitised and combined with an event from these presampled RDO datasets, the output is written to an RDO file that is then input to the reconstruction software. As Fast Chain is designed to configure and run all steps of the chain, including detector simulation, digitization and reconstruction, in one transform and sufficiently fast, the RDO-overlay has to be slightly adapted to enable running simulation and digitization in a single Athena job. These updates also reduce the disk space required as no intermediate data format needs to be stored, and the final output format used by the analyses, AOD (Analysis object data), can be produced directly.

Work is ongoing to validate this Fast Chain workflow by comparing its output with the output from the standard ATLAS full MC chain, which consists of three consecutive steps: simulation, RDO-overlay and reconstruction.

2.2 Track-overlay

The Track-overlay method is a possible faster alternative to the RDO-overlay method. It is based on a similar strategy to RDO-overlay, but in this case pile-up tracks would be reconstructed from presampled RDO events in a separate job whose output would be a special RDO file containing also pile-up tracks. Hard-scatter events would be simulated, digitized and their tracks would be reconstructed independently and then combined with the pile-up track collections. This merged track collection would be subsequently used as the input to the rest of the reconstruction.

This approach is feasible only if hard-scatter track reconstruction is not affected by surrounding pile-up events. To test the sensitivity of the reconstruction of tracks produced in high-p_{T} processes to the presence of pile-up, tracking performance has been studied using samples simulated for various physics processes, which in general give rise to a different detector response: $ggH(H \rightarrow \gamma\gamma)$, $t\bar{t}$, and boosted jets with different p_{T} of the leading jet. Each sample was then digitized and reconstructed with and without pile-up. This study has shown that in dense environments, where clusters generated from pile-up tracks merge with those inside jets, the reconstruction of tracks associated with the hard scattering is affected while otherwise it is not. The efficiencies of hard-scatter tracks reconstructed with and without pile-up are in good agreement in the $ggH(H \rightarrow \gamma\gamma)$, $t\bar{t}$, and low-p_{T} jets samples, while a difference of a few percent is observed when high-p_{T} jets are involved, as seen in Figure 2. To conclude, it has been found that the Track-overlay method can be used only for events with specific kinematics. All the other events need to be processed with the RDO-overlay. This kinematic-dependent implementation is currently being developed.

3 Integration of ACTS

ACTS (A Common Tracking Software) [15] is an experiment-independent software package developed to provide particle reconstruction in high energy experiments. The Fast Chain group has started an effort to integrate this software package into its framework with the goal of replacing the existing fast track simulation (Fatras) with ACTS fast simulation. This integration and replacement would, when combined with FastCaloSim/FastCaloGAN, allow thread-safe fast simulation of the whole of the ATLAS detector. Currently, the integration plan is to build ACTS as a standalone external software package that can be called using an interface in Athena. In this design, particles will be sent to ACTS for simulation, and ACTS will return simulated hits and any new particles to be simulated. The interface will initialize ACTS with the correct geometry and magnetic field conditions, convert particle and hit concepts between their respective implementations in Athena and ACTS, and update final

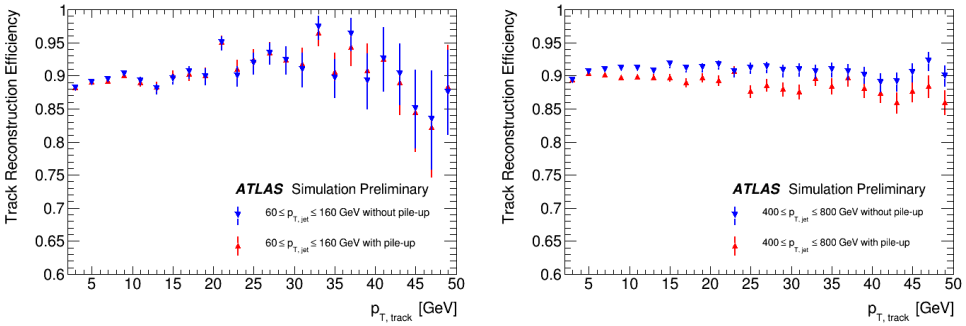


Figure 2: Track reconstruction efficiency as a function of $p_{\mathrm{T,\,track}}$ for events with low-p_{T} jets having $60 < p_{\mathrm{T,\,jet}} < 160$ (left) and high-p_{T} jets having $400 < p_{\mathrm{T,\,jet}} < 800$ (right). Hard-scatter track efficiencies reconstructed with overlaid pile-up collisions (red markers) are compared to those reconstructed without being overlaid with pile-up events (blue markers).

particle information. The plan is to have ACTS ready in Athena in time for high-luminosity data taking. At time of writing, the interface is compatible with a standalone ACTS build, can read in custom geometry, and can generate logging messages for the status information from ACTS.

4 Parametrization of nuclear interactions

Most of the nuclear interactions between the hadrons and the detector material are happening in the non-perturbative region of the theory of strong interactions, QCD. Geant4 provides various phenomenological models to describe nuclear interactions with different ranges of validity in terms of the energy. In order to reduce the CPU time consumption required for their simulation, an alternative description of hadronic showers was developed in Fatras using a parametrized approach based on properties and fits to the Geant4 simulation [8].

More recently, an updated model based on histograms simulated with Geant4 for different initial conditions (particle types and momenta) has been implemented in ACTS to achieve more reliable results. The final state multiplicity and kinematics depend on whether the initial particle survives the interaction or not. In addition, the final state kinematics depends on the number of produced particles. Therefore, the recorded histograms form a decision tree per initial state, starting with the interaction probability after a certain distance. In case of an interaction, it is evaluated if the particle survives, followed by the corresponding multiplicity. The final state kinematics is estimated from the absolute values of momenta and invariant masses. The recorded final states are then sorted in descending order of momenta and are considered to be produced in this order. When the set of momentum histograms is collected, a correlated inverse sampling method taken from FastCaloSim V2 [10] is applied and regulated by an additional histogram for total momentum. The invariant masses are evaluated with respect to the initial particle and inverse sampled with respect to their correlation. Their sum is not taken into account. Particle types are assessed from the branching probabilities derived from the recorded data under the assumption that the i^{th} particle produces the $(i+1)^{\mathrm{th}}$ particle.

In order to cover the full range of possible scenarios in the event simulation, the parametrization needs to be extended by an interpolation method. Sampling from initial states that differ from the recorded ones is based on a weighted combination of neighbouring parametrizations. Due to this interpolation procedure, deviations from the Geant4 simulation

are observed, as shown in Figure 3. While the momentum distributions produce reliable results, the invariant mass distributions are missing smaller details at low values. Moreover, a small shift of the entire invariant mass distribution towards larger values is observed. The distributions of produced particle types are dominated by the contribution of the initial particle, consequently affecting the probability of others. The impact of this effect depends on the initial momentum regime. To conclude, these artifacts depend on the interpolation distance of the input parametrizations to the momentum of the simulated particle and the applied weighting. Hence, higher density of parametrizations than the factor 2 spacing used for Figure 3 provide better results at the cost of memory and look-up time. No additional corrections are applied to adjust for these effects.

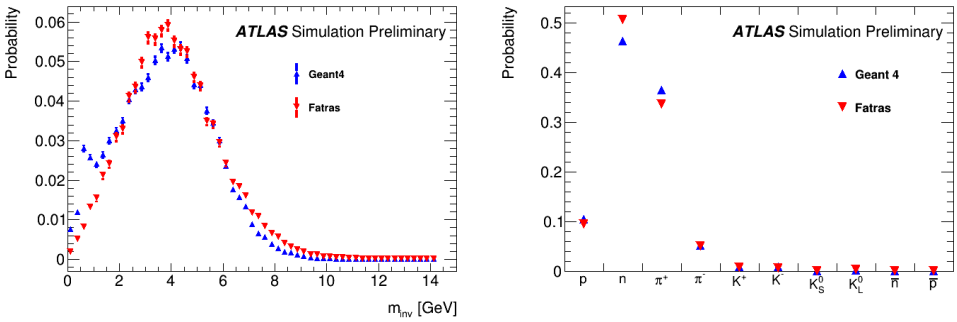


Figure 3: Invariant mass of the sum of the most common hadronic final state particles above 50 MeV (left) and a particle content of the final state (right) originated from a positively charged pion with the initial momentum of 25 GeV. The distributions from the Geant4 simulation (blue markers) and Fatras with a new parametrization method of nuclear interactions (red markers) are compared. The Fatras simulation shown here used an interpolation between parameterizations generated from Geant4 inputs at 16 GeV and 32 GeV.

5 Fast digitization of the silicon detector

Higher luminosity in future LHC runs will result in a larger number of energy deposits and a larger CPU time required to digitize the signal, together with an increased complexity in correctly assigning the energy deposits to the correct tracks. Digitization time in ATLAS is dominated by the Inner Tracker. Therefore, to speed up the digitization step, a fast approach to the digitization of the silicon detector has been developed. This is a parametric simulation of the conversion of the energy deposited in each sensor of the pixel and strip detectors into digital signals.

The inputs to the algorithm are the locations of the energy deposits. Successively, starting from these points, the geometrical trajectory of every particle inside each module is reconstructed. Each module is then divided in several cells, each corresponding to a read-out element, and the particle trajectories are split according to the number of crossed cells. The signal strength in each read-out element is proportional to the length of the trajectory in the corresponding cell. The last step of the fast digitization consists of the creation of clusters that will be the seeds for the track reconstruction step. The advantage of producing clusters at this stage is that all the cells crossed by a single trajectory can combined, without CPU intensive pattern recognition algorithms.

Since the approach followed by the algorithm is parametric, it is not expected to reach the same level of precision in the simulation as the standard digitization used in ATLAS. It is

possible, however, to tune several parameters in the fast digitization algorithm to mitigate the differences between the two approaches.

The fast digitization is currently included in various Fast Chain scenarios. It could be used in alternative to the standard digitization for analyses needing large samples but with limited accuracy or, after additional work, integrated into the presampled pile-up RDO sample production for the RDO-overlay approach.

Figure 4 and Figure 5 compare reconstructed variables (transverse momentum and pseudorapidity, η[1]) and reconstruction performance respectively for reconstruction run on pile-up only samples produced using fast digitization and the standard ATLAS digitization. As can be seen the two simulations are in good agreement with differences of the order of maximum 5% in the largest part of the distributions.

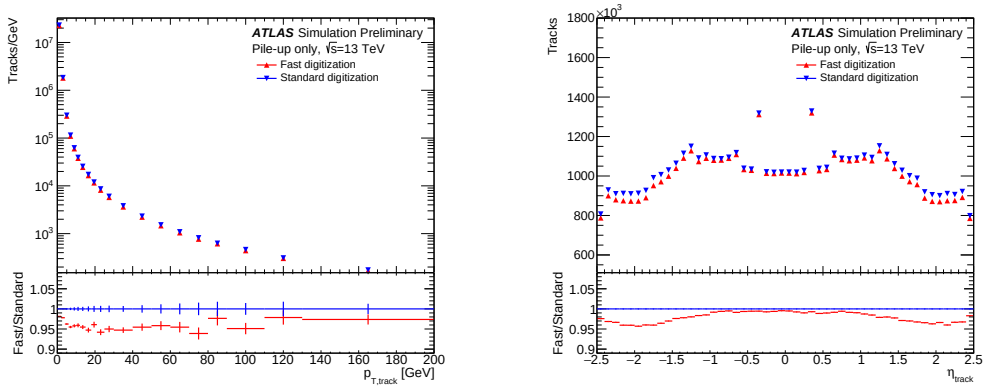


Figure 4: Transverse momentum (left) and pseudorapidity (right) of reconstructed tracks in pile-up events generated according to the ATLAS Run 2 distribution. The tracks obtained with the standard ATLAS MC production chain (blue markers) are compared to the tracks obtained replacing the standard digitization and cluster formation with a faster algorithm that parametrizes the digitization of the signal and creates cluster exploiting truth information (red markers).

6 Fast Track Reconstruction

The reconstruction step takes a significant fraction of the overall computing time in the MC production chain and will grow rapidly with increasing pile-up events. Moreover, once the fast simulation and the fast digitization techniques are used, the reconstruction time will dominate the total MC production chain time [16]. Therefore efforts are focused on reducing the reconstruction CPU time consumption. To achieve this, a new fast reconstruction approach called Truth-seeded track reconstruction is used. In Figure 6b, a comparison of the dependency of the CPU time on the average number of pile-up interactions between the standard track reconstruction and the Truth-seeded track reconstruction is shown. It is important to mention that since these studies were performed, the new tracking software has been developed and has already reduced the CPU needs for the reconstruction significantly. The gain of this new approach needs to be thus re-evaluated.

[1] ATLAS uses a right-handed Cartesian coordinate system with its origin at the nominal interaction point (IP) in the centre of the detector. The z-axis is along the beam pipe, and the x-axis points from the IP to the centre of the LHC ring. Cylindrical coordinates (r, ϕ) are used in the transverse plane, ϕ being the azimuthal angle around the beam pipe. The rapidity is defined as $y = (1/2)\ln[(E + p_z)/(E - p_z)]$, while the pseudorapidity is defined in terms of the polar angle θ as $\eta = -\ln\tan(\theta/2)$.

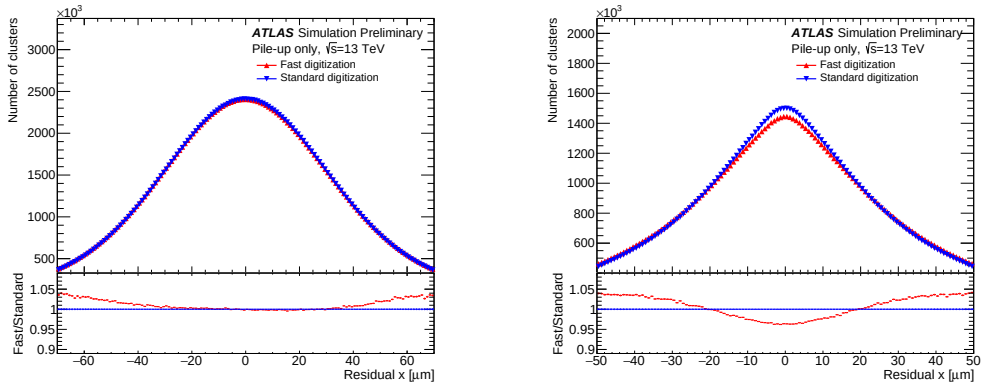


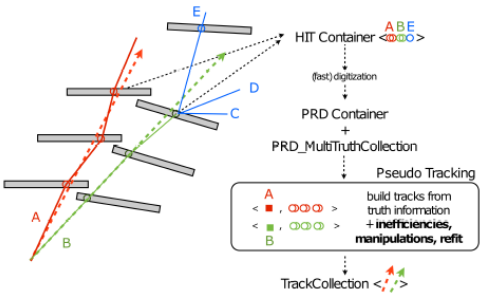
Figure 5: Residuals between the truth and reconstructed tracks along the x coordinate in the barrel ($0 < |\eta| < 1.4$) of the SCT detector (left) and of the pixel detector (right) in pile-up events generated according to the ATLAS Run 2 distribution. The tracks obtained with the standard ATLAS MC production chain (blue markers) are compared to the tracks obtained with the fast digitization approach (red markers).

The Truth-seeded tracking algorithm removes the time consuming pattern recognition, track seeding and ambiguity treatment completely and instead uses the MC truth information to assign tracking detector hits to particle tracks. In this fast approach, the tracks are manipulated by changing the hit content and efficiency, as well as by applying similar selection criteria as in the standard reconstruction, in order to mimic the effect of skipped steps [17]. A schematic illustration is shown in Figure 6a. During the reconstruction step, both hard-scatter events end pile-up events are reconstructed, so the Truth-seeded track reconstruction can be used either for the pile-up or for all tracks (hard-scatter events + pile-up events). Since the reconstruction CPU time is dominated by the pile-up tracks reconstruction, and no significant gain is observed when using this faster algorithm for hard-scatter events, only tracks coming from the pile-up will be reconstructed via this new approach. This can be realized by splitting the hard-scatter particles hits created during the digitization step from the pile-up particles hits.

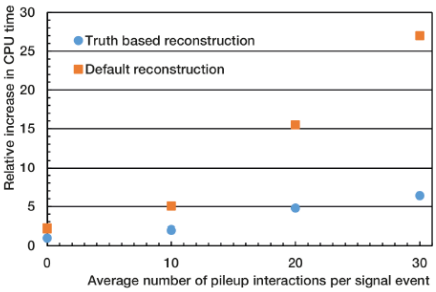
A comparison of the resolution $\sigma\left(d_0^{\text{rec}} - d_0^{\text{tru}}\right)$ on the transverse and of the resolution $\sigma\left(z_0^{\text{rec}} - z_0^{\text{tru}}\right)$ on the longitudinal impact parameters of tracks with respect to the centre of the interaction point as a function of their pseudorapidity η_{track} is shown in Figure 7. The distributions from the standard tracking are compared to those obtained using truth information in the track candidate finding step. In both cases, the inputs are samples in the raw object format, produced with fast simulations: Fatras in the tracking systems and FastCaloSim in the calorimeters. The overall agreement is good.

7 Daily ART tests

The ATLAS software, consisting of packages in a directory structure, are collected in the Athena Git repository [12]. Smaller projects can be made by combining different packages to serve for a specific purpose, such as for simulation analysis or for physics derivation. The projects for each branch are built [19] according to their schedule either every night or when there are changes in the software. The main Athena project is built from the code in the `master` branch of the Athena Git repository. Its build time is much longer than the smaller projects.



(a) Truth track creation in the Inner Detector. All hits from the simulation step are fed into the digitization. The resulting PRD (Prepared Raw Data) object is used, along with PRD MultiTruthCollection, which connects PRD contents with truth information. Tracks are built from this input, manipulators and selectors are applied, and finally tracks are refit to get the output track collection. Taken from [17].



(b) Comparison of the dependency of the CPU time on the average number of pile-up interactions μ in the event between the standard reconstruction and the truth based approach. Taken from [18].

Figure 6: Illustration of the concepts of the Truth-seeded track reconstruction (a), Comparison of the dependency of the CPU time on μ (b).

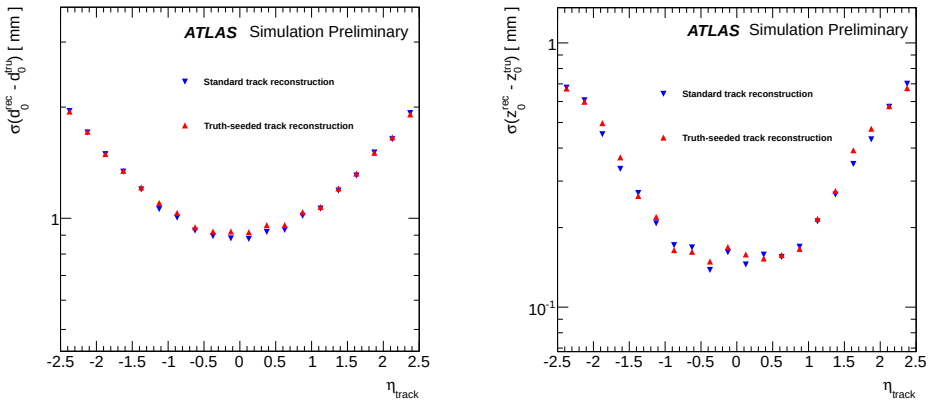


Figure 7: Transverse ($\sigma\left(\mathrm{d}_0^{\mathrm{rec}} - \mathrm{d}_0^{\mathrm{tru}}\right)$, left) and longitudinal ($\sigma\left(\mathrm{z}_0^{\mathrm{rec}} - \mathrm{z}_0^{\mathrm{tru}}\right)$, right) impact parameter resolution of tracks as a function of their pseudorapidity η_{track}. Tracks reconstructed exclusively using the standard ATLAS track reconstruction (blue markers) are compared to the tracks where those coming from pile-up events were reconstructed using the Truth-seeded approach (red markers).

The software changes which are accepted in a nightly build are validated automatically by unit tests and ART [20] tests. The ART tests currently run on Intel architecture and can be constructed in different stages. In general the first stage consists of processing input files and generating output data files. The next stages perform checks on the output data files and the parsing log files. The output files are compared with reference files or with respect to previous nightly results. Physics quantities are checked numerically or distributions of physics quanti-

ties are compared with the ones from reference files. The unexpected differences are reported and fixes can be merged back into the corresponding branch of the Athena repository.

The Fast Chain simulation workflow is monitored with a set of ART tests with about fifty events and a validation ART test with ten thousand events, in both the `master` and `21.3` nightly branches. The jobs are submitted to the grid. Various simulators such as ATLFASTII and ATLFASTIIF_G4MS, are tested with different configurations and different conditions in Fast Chain ART tests. The larger scale validation test is configured to use the ATLFASTIIF_G4MS simulator. The generated event input file is divided into ten sub-jobs with a thousand event each to run in parallel. All the output files from simulation, digitization and reconstruction in the workflow are produced. The final step is to produce distributions of physics quantities, collected in a ROOT file. In addition to checks on the physics results, the CPU and memory consumption of each test are monitored.

8 Future Plans

Before the Fast Chain workflow can be deployed for the production of official ATLAS MC samples for use in analyses, the entire chain must be validated along with the individual components. The baseline Fast Chain approach will consist of the ATLFAST3 simulator with Fatras in the Inner Detector, RDO-overlay method and standard reconstruction tools. The other fast components described in the previous sections will be included in different Fast Chain scenarios.

As Fast Chain streamlines the processing of event generator inputs to reconstructed physics objects, intermediate files are skipped. This reduces data output, and thereby saves disk space, while also permitting a more robust and easier to configure full chain production jobs. However, omission of intermediate data also means the underlying framework must handle and propagate all required information and metadata through each sub-step. Therefore, technically, the lowest level at which individual sub-detectors can be validated is at the digitization step. Any differences in digitised data – e.g., tracks or calorimeter clusters of cells energy deposition – and reconstructed physical objects – e.g., electrons, jets, etc. – would thus point to a misconfiguration in the Fast Chain transform. At the highest (analysis) level, the Fast Chain approach can be validated by comparing the yield of some analysis-based event selection strategy using the derivation data formats (Derived Analysis Data Objects, DAOD) produced using the Fast Chain and the standard simulation approaches. For this work, which is ongoing, a search for phenomena beyond the Standard Model targeting the process of production of the supersymmetric partners of the top quark has been chosen and the Fast Chain has been configured to use the ATLFASTIIF_G4MS for the simulation while the standard tools elsewhere.

9 Conclusion

Fast Chain aims to provide a faster alternative to the standard MC production chain as well as more efficient handling of I/O and CPU resources. Its configuration incorporates various fast components and combines them with more precise ones in an easily adaptable way. All fast tools need to be firstly carefully validated to ensure that their impact on the modelling accuracy is minimal. Some of them are already undergoing physics validation. Moreover, Fast Chain removes all intermediate file formats and enables to generate one single output directly from the input event generator files. Thanks to its flexibility, it can be used for all large-scale MC productions in the upcoming LHC Runs, when the computing requirements are expected to be unprecedentedly high.

References

[1] ATLAS collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider*, JINST **3**, S08003 (2008)

[2] L. Evans, P. Bryant, *LHC Machine*, Journal of Instrumentation **3**, S08001 (2008)

[3] S. Agostinelli et al., *Geant4—a simulation toolkit*, Nucl. Instrum. Meth. A **506**, 250-303 (2003)

[4] J. Allison et al., *Geant4 developments and applications*, IEEE T Nucl. Sci. **53**, 270-278 (2006)

[5] P. Calafiura, J. Catmore, D. Costanzo, A. Di Girolamo, *ATLAS HL-LHC Computing Conceptual Design Report*, CERN-LHCC-2020-015 (2020)

[6] T. Yamanaka, *The ATLAS calorimeter simulation FastCaloSim*, J. Phys. Conf. Ser. **331 032053**

[7] ATLAS collaboration, *The ATLAS Simulation Infrastructure*, Eur. Phys. J. C70:823-874 (2010) doi:10.1088/1742-6596/523/1/012035

[8] K. Edmonds, S. Fleischmann, T. Lenz, C. Magass, J. Mechnich, A. Salzburger, *The Fast ATLAS Track Simulation (FATRAS)*, ATL-SOFT-PUB-2008-001 (2008), `https://cds.cern.ch/record/1091969`

[9] ATLAS collaboration, *AtlFast3: Next Generation of Fast Simulation in ATLAS*, Computing and Software for Big Science, in press (2021)

[10] ATLAS collaboration, *The new Fast Calorimeter Simulation in ATLAS*, ATL-SOFT-PUB-2018-002 (2018), `http://cds.cern.ch/record/2630434`

[11] ATLAS collaboration, *Fast simulation of the ATLAS calorimeter system with Generative Adversarial Networks*, ATL-SOFT-PUB-2020-006 (2020), `https://cds.cern.ch/record/2746032`

[12] *Athena*, https://doi.org/10.5281/zenodo.2641997

[13] ATLAS collaboration, *Emulating the impact of additional proton-proton interactions in the ATLAS simulation by pre-sampling sets of inelastic Monte Carlo events*, arXiv:2102.09495 (2021)

[14] CMS collaboration, *Upgrades for the CMS simulation*, J. Phys.: Conf. Ser. **608** 012056 (2015)

[15] Ch. Gumpert, A. Salzburger, M. Kiehn, J. Hrdinka, N. Calace, *ACTS: from ATLAS software towards a common track reconstruction software*, J. Phys. Conf. Ser. **898** (2017) 042011

[16] E. Ritsch, *Concepts and Plans towards fast large scale MonteCarlo production for the ATLAS Experiment*, J. Phys. Conf. Ser. **523**, 012035 (2014) doi:10.1088/1742-6596/523/1/012035

[17] A. Basalaev, Z. Marshall, *The Fast Simulation Chain for ATLAS*, J. Phys. Conf. Ser. **898**, 042016 (2017) doi:10.1088/1742-6596/898/4/042016

[18] R. Jansky, *The ATLAS Fast Monte Carlo Production Chain Project*, J. Phys. Conf. Ser. **664**, 072024 (2015) doi:10.1088/1742-6596/664/7/072024

[19] E. Ritsch et al., *Modernising ATLAS Software Build Infrastructure*, J. Phys.: Conf. Ser. **1085**, 032033 (2018) doi:10.1088/1742-6596/1085/3/032033

[20] T. Cuhadar-Donszelmann, W. Lampl, G. A. Stewart, *ART ATLAS Release Tester using the Grid*, EPJ Web Conf. **245**, 05015 (2020) doi:10.1051/epjconf/202024505015