PROCEEDINGS OF SCIENCE

LAMARR: the ultra-fast simulation option for the LHCb experiment

Lucio Anderlini,[a,*] **Matteo Barbetti,**[a,b] **Gloria Corti,**[c] **Adam Davis,**[d] **Denis Derkach,**[e] **Nikita Kazeev,**[e] **Artem Maevskiy,**[e] **Sergei Mokonenko,**[e] **Benedetto Siddi**[f] **and Zehua Xu**[g]

[a] *Istituto Nazionale di Fisica Nucleare, Sezione di Firenze, via G. Sansonse 1, Sesto Fiorentino (FI), Italy*
[b] *Department of Information Engineering, University of Florence, via Santa Marta 3, Firenze, Italy*
[c] *European Organization for Nuclear Research, CERN, Esplanade des Particules, Meyrin, Switzerland*
[d] *Schuster Laboratory, University of Manchester, Oxford Rd, Manchester, M13 9PL, United Kingdom*
[e] *High School of Economics (HSE) University, 20 Myasnitskaya st., Moscow 101000, Russia*
[f] *Physics Department, University of Ferrara, Via G. Saragat 1, Ferrara (FE), Italy*
[g] *Laboratoire de Physique de Clermont, LPC, Université Clermont Auvergne, 4 Avenue Blaise Pascal, Clermont, France*

E-mail: Lucio.Anderlini@fi.infn.it

During Run 2 of the Large Hadron Collider at CERN, the LHCb experiment has spent more than 80% of the pledged CPU time to produce simulated data samples. The upgraded LHCb detector, being commissioned now, will be able to collect much larger data samples, requiring many more simulated events to analyze the collected data. Simulation is a key necessity of analysis to interpret signal, reject background and measure efficiencies. The needed simulation will exceed the pledged resources, requiring an evolution in technologies and techniques to produce these simulated samples. In this contribution, we discuss LAMARR, a GAUDI-based framework to deliver simulated samples parametrizing both the detector response and the reconstruction algorithms. Generative Models powered by several algorithms and strategies are employed to effectively parametrize the high-level response of the multiple components of the LHCb detector, encoding within neural networks the experimental errors and uncertainties introduced in the detection and reconstruction process. Where possible, models are trained directly on real data, leading to a simulation process completely independent of the detailed simulation.

41st International Conference on High Energy physics - ICHEP2022
6-13 July, 2022
Bologna, Italy

*Speaker

PoS(ICHEP2022)233

1. Introduction

The simulation of high-energy collisions, of the decays of the generated particles, and the interaction of decay products with the detector provide crucial tools for interpretation of physics data, by disentangling detector and reconstruction effects from genuine physical phenomena. The detailed simulation of the dynamics of the hadron collision and of the interaction of all primary and secondary particles with the detector material, however, is extremely expensive in terms of CPU time and therefore unsustainable already for experiments taking data in the nearest future.

The LHCb detector is a single-arm forward spectrometer covering the pseudorapidity range $2 < \eta < 5$, designed for the study of particles containing b and c quarks produced at the Large Hadron Collider (LHC), at CERN [1]. The detector includes a high-precision tracking system, interleaved with a dipole magnet with a bending power of about 4 Tm, providing a measurement of the momentum, p, of charged particles. The minimum distance of a track to a primary vertex (PV), the impact parameter (IP), is measured with high resolution and employed to distinguish particles produced in the decay of short-lived heavy hadrons. Different types of charged hadrons are distinguished using information from two two ring-imaging Cherenkov (RICH) detectors. Photons, electrons and hadrons are identified by a calorimeter system consisting of scintillating-pad and preshower detectors, an electromagnetic and a hadronic calorimeter. Muons are identified by a system (MUON) composed of alternating layers of iron and multiwire proportional chambers.

The simulation software of the LHCb experiment is composed of two projects, Gauss and Boole [2]. The former combines a *Generation* phase in which the high-energy collisions are simulated with Monte Carlo generators such as Pythia8 [3] and EvtGen [4], and a *Simulation* phase in which particles are propagated through the LHCb spectrometer, relying on Geant4 [5] to simulate the radiation-matter interaction with the detector material to compute the energy deposited in the active volumes. The energy deposits are then converted into raw data mimicking the data format used in the Data Acquisition pipeline in a step named *Digitization* implemented in the Boole application. Both Gauss and Boole are based on the Gaudi framework [6].

The same data processing procedure is adopted for acquired and digitized simulated data. A preliminary selection based on information from the calorimeter and muon systems is followed by a high-level selection stage relying on partial event reconstruction. Selected events are fully reconstructed and grouped in streams according to their relevance for specific physics analyses, possibly discarding part of the reconstructed objects whenever irrelevant for the target physics studies.

During the LHC Run 2 (years 2015 – 2018), the LHCb experiment has spent more than 80% of the pledged CPU time to produce simulated data samples. The foreseen needs for Run 3 will far exceed the CPU resources made available to the LHCb Collaboration. Hence, a profound transformation of the LHCb simulation software stack is needed, combining code optimization and modernization (discussed for example in Ref. [7]), with the development of novel and faster simulation options.

Several *fast* simulation options based on resampling techniques [8] or parametrizations of the energy deposits [9, 10] are being developed to achieve a speed-up of the Simulation phase relative to *detailed* simulation up to a factor 20. Fast simulation options do not modify the data processing flow discussed above, as depicted in Figure 1 (top). Further improvements can be obtained by

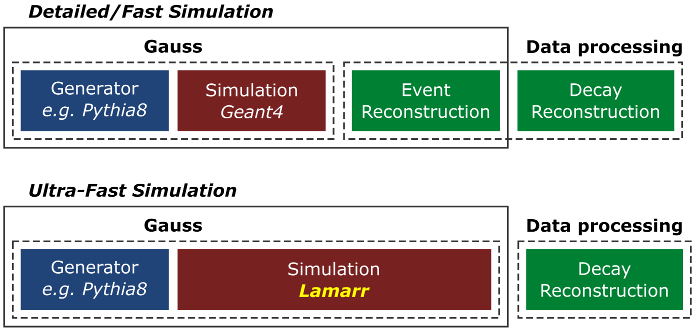

Figure 1: Schematic representation of the data processing flow in detailed and fast simulation (top), and in ultra-fast simulation (bottom).

parametrizing the response of the detector in terms of reconstructed physics objects to particles obtained from physics generators, as represented in Figure 1 (bottom). Such parametrizations can be built using *machine learning* generative models trained on special simulated samples or calibration datasets [11]. The terms *ultra-fast*, *flash* and *parametric* simulation are used in the literature to refer to this second approach.

The remainder of this document is devoted to discuss the LAMARR application, the software framework developed by the LHCb Collaboration to implement the *ultra-fast* simulation within its software stack for the benefit of its physics program (Section 2) and its validation on $\Lambda_b^0 \to \Lambda_c^+ \mu^- \bar{\nu}_\mu$ decays (Section 3).

2. The LAMARR framework

The LAMARR framework is conceived as a pipeline of parametrizations taking as input the particles generated by the event generator and providing as an output the LHCb representation of the particles successfully reconstructed. LAMARR is part of GAUSS and provides a dedicated interface to the physics generators to identify and select those particles that need to be propagated through the detector, splitting them into charged and neutral particles. Two different pipelines are then applied to the two streams. Charged particles are propagated through the magnetic field following a trajectory approximated as two rectilinear segments with a single point of deflection (*single p_{T} kick* approximation), then the tracking acceptance and reconstruction efficiency is computed from a trainable parametrization taking as input geometrical and kinematic features of the track. Tracks are randomly sampled as *reconstructed* accordingly. Gradient Boosting Decision Trees (GBDTs) are trained on a sample of tracks obtained from a dedicated simulation of b-hadron decays to predict the fraction of tracks reconstructed as traversing the whole detector, or as segments involving only the tracking stations upstream the magnetic field. Generative Adversarial Networks (GANs) are used to predict the smearing effect of phenomena such as multiple scattering on the reconstructed momentum and IP, without neglecting correlation effects. The correlation matrix obtained from the Kalman filter used in the reconstruction algorithm to define the position, slope and curvature of each track is necessary at physics analysis level to assess the consistency with other tracks or vertices. In LAMARR it is parametrized with a GAN-based model.

Particle Identification (PID) information is then attached to the reconstructed tracks using GANs trained to reproduce the response of the RICH and MUON systems. Global PID classifier, obtained in real data by combining RICH and MUON response with information from the Calorimeter system and features of the reconstructed tracks, result from GANs taking GAN-generated RICH and MUON responses as inputs. The efficiency of a binary muon-identification criterion, available since the earliest stage of data processing via a hardware implementation on FPGAs is parametrized with a GBDT. GANs and GBDTs are trained on simulated data to validate ultra-fast simulation by comparing the datasets generated with LAMARR and detailed simulation. An extension of the GAN training algorithm is adopted to train the models directly on background-contaminated calibration datasets acquired with dedicated trigger selections originally designed to provide data-driven corrections to the simulated PID efficiencies.

Neutral objects are propagated to the calorimeters to predict the pattern of energy deposits which are then smeared and used to simulate a reconstructed value for the particle energy.

The pipelines for charged and neutral objects converge at the persistency step, converting the parametrized variables into a data format as similar as possible to the one adopted for reconstructed data.

The modular structure of LAMARR enables a variety of studies and developments on the single parametrizations providing a unique and shared infrastructure for validation and performance measurements. In addition to the interface to the generators and the persistency service, the LAMARR framework provides common solutions to distribute the parametrizations through the LHCb Computing Grid via the CERN VM filesystem (`cvmfs`) [12], a GAUDI-based configuration system to select and tune the parametrizations via Python option files, and a low-latency interface to models trained with `scikit-learn` and `Keras`, relying on the `scikinC` transcompilation approach [13].

3. Validation using $\Lambda_b^0 \to \Lambda_c^+ \mu^- \bar{\nu}_\mu$ decays

Among the many validation comparisons performed during the development of LAMARR, we report here on the comparison of simulated $\Lambda_b^0 \to \Lambda_c^+ \mu^- \bar{\nu}_\mu$ decays with $\Lambda_c^+ \to p K^- \pi^+$. Semileptonic Λ_b^0 decay channels are being widely studied in LHCb, at the point that $\Lambda_b^0 \to \Lambda_c^+ \mu^- \bar{\nu}_\mu$ is part of the calibration samples for the particle identification properties of the proton. Interestingly, it includes in its final state the four charged particle types parametrized in the current version of LAMARR. The decay model describing the semileptonic Λ_b^0 decay is not trivial, providing a clear example of the importance of interfacing to dedicated generators, in this case EVTGEN. Unfortunately, the generation of Λ_b^0 hadrons with PYTHIA8 is several orders of magnitude slower than the subsequent LAMARR-based Simulation phase. Since the effects on the detector performance due to the occupancy are enclosed in the parametrizations, very similar results are produced by LAMARR running on Λ_b^0 hadrons generated without the underlying event, with parametrized momentum and rapidity spectra. This generator is indicated in the following as *particle gun*.

We compare three simulated datasets for the chosen Λ_b^0 decay obtained with PYTHIA8 followed by EVTGEN and LAMARR (estimated CPU cost: $\sim$500 HS06 $\cdot$ s per event), with a particle gun followed by EVTGEN and LAMARR ($\sim$1 HS06 $\cdot$ s per event), and with PYTHIA8 followed by EVTGEN and GEANT4 ($\sim$2.5 kHS06 $\cdot$ s per event), representing the reference sample we aim at reproducing with LAMARR. Figure 2 reports the comparison of the distributions of the proton impact parameter χ^2 (IPχ^2), a measure of the inconsistency of the proton track with the PV obtained executing the same analysis algorithm on the three datasets combining information from the reconstructed position, slope and momentum of the tracks, of the PV, and the respective uncertainties, including

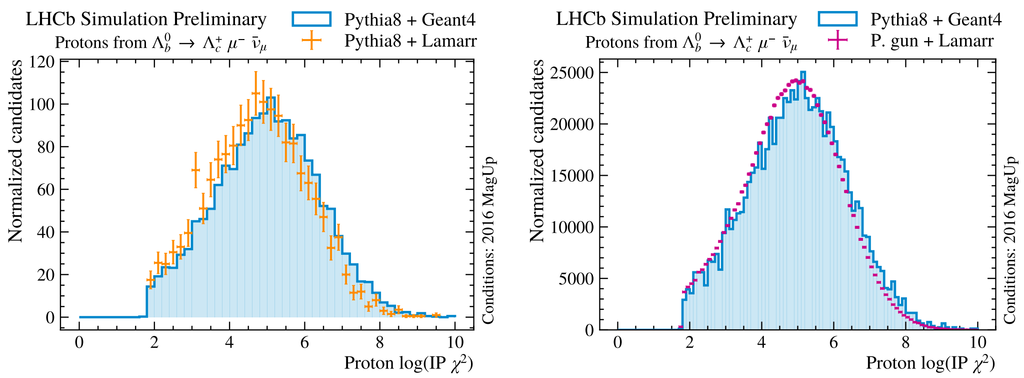

Figure 2: Distribution of the IP χ^2 for protons produced in $\Lambda_b^0 \to \Lambda_c^+ \mu^- \bar{\nu}_\mu$ decays as simulated with Pythia8, EvtGen and Lamarr (on the left) and with a particle gun, EvtGen and Lamarr (on the right). Both distributions are compared with a sample obtained with *detailed simulation* relying on Pythia8, EvtGen and Geant4. Reproduced from Ref. [14].

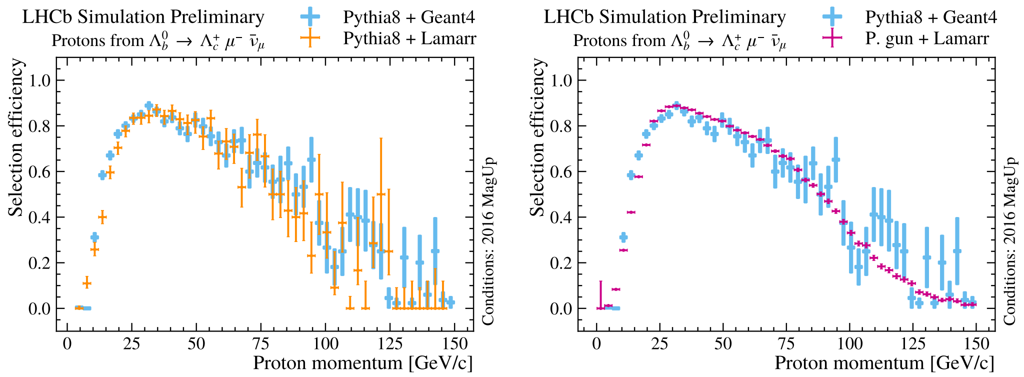

Figure 3: Selection efficiency of protons applying a tight PID requirement on a multivariate classifier combining the responses of most the subsystems of the LHCb detector. Protons are obtained from $\Lambda_b^0 \to \Lambda_c^+ \mu^- \bar{\nu}_\mu$ decays as simulated with Pythia8, EvtGen and Lamarr (on the left) and with a particle gun, EvtGen and Lamarr (on the right). Both distributions are compared with a sample obtained with *detailed simulation* relying on Pythia8, EvtGen and Geant4. Reproduced from Ref. [14].

correlations. As an example of the parametrization of the Particle Identification detectors, we report in Figure 3 the efficiency of the selection of protons with a tight requirement on a multivariate classifier combining the responses of most the subsystems of the LHCb detector [11]. A more complete set of comparisons has been released by the Collaboration in Ref. [14].

4. Conclusion

Supporting physics data analysis for the Run 3 of the LHCb experiment with simulated samples requires investigations and developments towards faster simulation technologies and techniques. The most radical option consists of parametrizing, possibly using Generative Models, both the detector response and the reconstruction effects. Such parametrizations are deployed in the LHCb software stack via the novel Lamarr framework, in which statistical models for tracking and charged particle identification have been deployed and validated with satisfactory results on $\Lambda_b^0 \to \Lambda_c^+ \mu^- \bar{\nu}_\mu$

decays. Improvements on the quality of the parametrizations, extensions to a larger number of features, and code optimization represent the main ongoing developments with the purpose of enhancing the variety of physics analyses relying on LAMARR.

References

[1] LHCb collaboration, *LHCb detector performance*, *Int. J. Mod. Phys.* **A30** (2015) 1530022 LHCB-DP-2014-002, CERN-PH-EP-2014-290, [1412.6352].

[2] M. Clemencic et al., *The* LHCb *simulation application, Gauss: Design, evolution and experience*, *J. Phys. Conf. Ser.* **331** (2011) 032023.

[3] T. Sjöstrand, S. Mrenna and P. Skands, *A brief introduction to PYTHIA 8.1*, *Comput. Phys. Commun.* **178** (2008) 852 [0710.3820].

[4] D.J. Lange, *The EvtGen particle decay simulation package*, *Nucl. Instrum. Meth.* **A462** (2001) 152.

[5] GEANT4 collaboration, *Geant4 developments and applications*, *IEEE Trans.Nucl.Sci.* **53** (2006) 270.

[6] G. Barrand et al., *Gaudi – a software architecture and framework for building HEP data processing applications*, *Comput. Phys. Commun.* **140** (2001) 45.

[7] I. Belyaev et al., *Handling of the generation of primary events in Gauss, the LHCb simulation framework*, *J. Phys. Conf. Ser.* **331** (2011) 032047.

[8] D. Müller, M. Clemencic, G. Corti and M. Gersabeck, *ReDecay: A novel approach to speed up the simulation at LHCb*, *Eur. Phys. J. C* **78** (2018) 1009. 9 p [1810.10362].

[9] M. Rama and G. Vitali, *Calorimeter fast simulation based on hit libraries LHCb Gauss framework*, *EPJ Web Conf.* **214** (2019) 02040. 7 p.

[10] V. Chekalina, E. Orlova, F. Ratnikov, D. Ulyanov, A. Ustyuzhanin and E. Zakharov, *Generative Models for Fast Calorimeter Simulation: the LHCb case*, *EPJ Web Conf.* **214** (2019) 02034. 8 p [1812.01319].

[11] R. Aaij et al., *Selection and processing of calibration samples to measure the particle identification performance of the LHCb experiment in Run 2*, *Eur. Phys. J. Tech. Instr.* **6** (2018) 1 LHCb-DP-2018-001, [1803.00824].

[12] P. Buncic, C. Aguado Sanchez, J. Blomer, L. Franco, A. Harutyunian, P. Mato et al., *CernVM: A virtual software appliance for LHC applications*, *J. Phys. Conf. Ser.* **219** (2010) 042003.

[13] L. Anderlini and M. Barbetti, *scikinC: a tool for deploying machine learning as binaries*, *PoS* **CompTools2021** (2022) 034.

[14] LHCb collaboration, “Validation of the LAMARR framework with $\Lambda_b^0 \to \Lambda_c^+ \mu^- X$ decays.” LHCb-FIGURE-2022-014, Jun, 2022.