Generative Adversarial Networks for LHCb Fast Simulation

Fedor Ratnikov[1],[*]

[1]National Research University Higher School of Economics, Moscow, Russia

Abstract. LHCb is one of the major experiments operating at the Large Hadron Collider at CERN. The richness of the physics program and the increasing precision of the measurements in LHCb lead to the need of ever larger simulated samples. This need will increase further when the upgraded LHCb detector will start collecting data in the LHC Run 3. Given the computing resources pledged for the production of Monte Carlo simulated events in the next years, the use of fast simulation techniques will be mandatory to cope with the expected dataset size. Generative models, which are nowadays widely used for computer vision and image processing, are being investigated in LHCb to accelerate generation of showers in the calorimeter and high-level responses of Cherenkov detector. We demonstrate that this approach provides high-fidelity results and discuss possible implications of these results. We also present an implementation of this algorithm into LHCb simulation software and validation tests.

1 Introduction

Detailed simulation of the detector response on different types of physics events is a vital component of every experiment in high energy physics. Without such simulation it is virtually impossible to infer a physics result from the experimental observations. Detailed simulation however requires significant computing resources. Moreover, simulation is the primary consumer of computing resources: about 80% of the total computing is used by HEP experiment for simulation.

A significant increase in total event rate is expected due to upgrades to LHC machine and detectors [1]. The simulation rate will need to be increased accordingly. However, we can not expect a significant increase of computational power for computing hardware. Because computation constraints make it impossible to work harder, we have to work smarter to accommodate the challenge of simulation.

Using surrogate generative models is one of the possible approaches to this challenge. It is driven by the observation, that if the physics detectors has a granularity significantly coarser than the level of the corresponding `Geant4` simulation, the surrogate model can aggregate micro-level simulation effects into the required macro-level response.

2 Generative Model for Calorimeter Response Simulation

The simulation of particle showers in the electromagnetic calorimeter is the most computationally expensive component of the Monte Carlo event simulation for the LHCb detector.
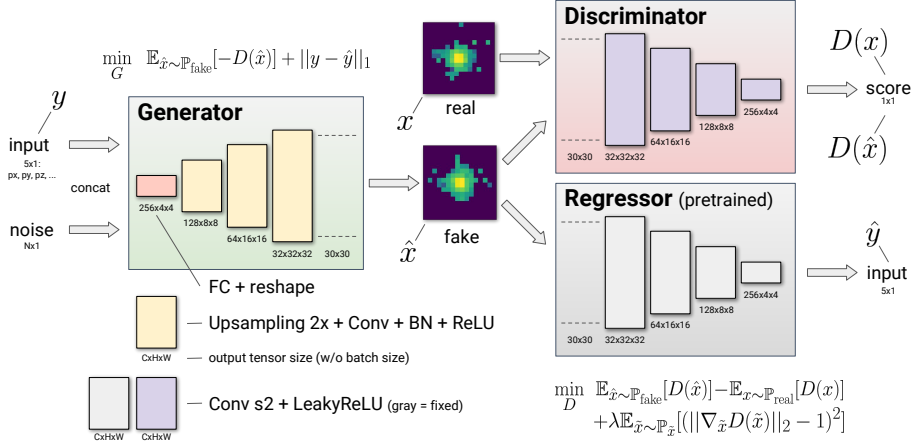
---

[*]e-mail: fedor.ratnikov@cern.ch

Figure 1: Model architecture. Pre-trained regressor for the particle parameters prediction makes our model conditional. Thanks to building up the information from the pre-trained regressor into the discriminator gradient we learn G to produce a specific calorimeter response.

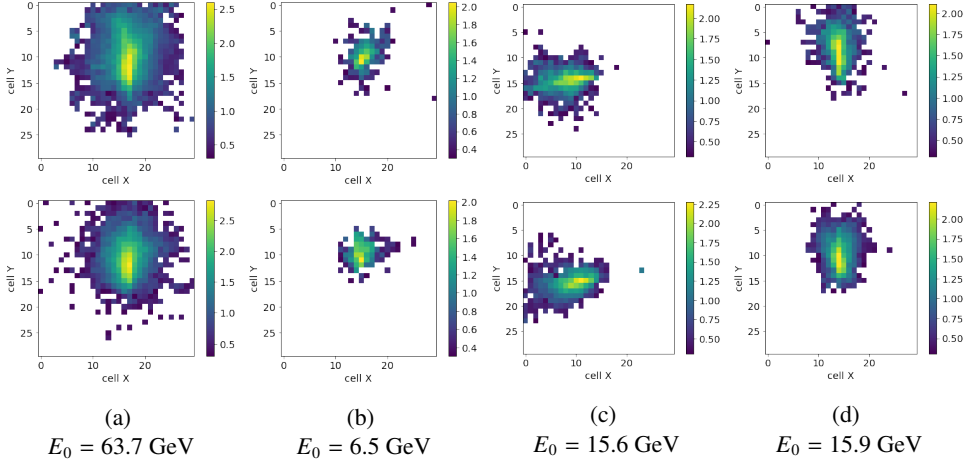| (a)	(b)	(c)	(d)
$E_0 = 63.7$ GeV	$E_0 = 6.5$ GeV	$E_0 = 15.6$ GeV	$E_0 = 15.9$ GeV

Figure 2: Visual comparison of generated calorimeter showers. Showers generated with `Geant4` (first row) and the showers, simulated with our model (second row) for four different sets of input parameters. Colour represents $log_{10}(\frac{E}{MeV})$ for every cell.

The relatively coarse 2D granularity of the calorimeter allows the use of surrogate generative models to be built on top of the detailed `Geant4` simulation. Thus, this approach seems to be promising to speed up the calorimeter simulation. Wasserstein GAN with gradient penalty [2] is considered to be a state-of-the-art technique for image production. We use Wasserstein GAN as a model for generating calorimeter responses. The architecture of this Neural Network and details of training the generative model are presented in Ref. [3].
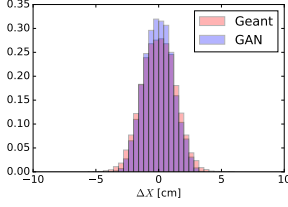
After the generative model is built and trained, we compare the original clusters produced by full `Geant4` simulation with the clusters generated by the trained model for the same pa-
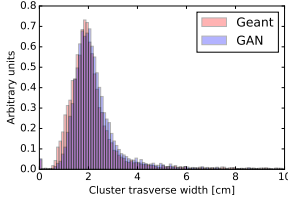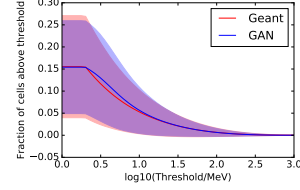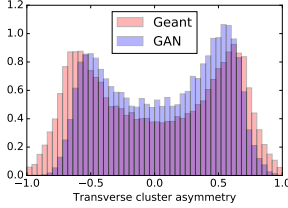
(a) The transverse width of clusters.
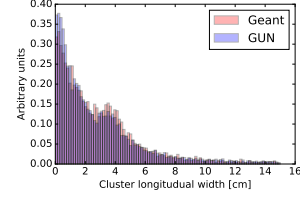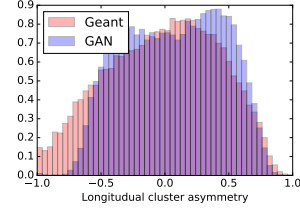

(b) The longitudinal width of clusters.

(c) ΔX between cluster centre of mass and the true particle coordinate.

(d) The sparsity in the 30x30 cells matrix containing clusters.

(e) The transverse asymmetry of clusters.

(f) The longitudinal asymmetry of clusters.

Figure 3: Comparing physics characteristics for `Geant4` simulated (red) and generated (blue) clusters.

rameters of the incident particles: the same energy, the same direction, and the same position on the calorimeter face. Corresponding images for the four arbitrary parameter sets are presented in Fig. 2. These images demonstrate very good visual similarity between simulated and generated clusters.

Then, the quantitative evaluation of the proposed simulation method is performed. While generic evaluation methods for generative models exist, the evaluation is based on physics-driven similarity metrics. A few cluster properties, which essentially drive the cluster properties used in the reconstruction of the calorimeter objects and following physics analysis, are selected. If the initial particle direction is not perpendicular to the calorimeter face, the produced cluster is elongated in that direction. Therefore, cluster widths in the direction of the initial particle and in the transverse direction are considered separately. Spatial resolution, which is the distance between the centre of mass of the cluster and the initial track projection to the shower max depth, is another important characteristic affecting physics properties of the cluster. Cluster sparsity, which is the fraction of cells with energies above some threshold, reflects marginal low energy properties of the generated clusters. These characteristics are presented in Fig. 3.

Data

DLLx, P, ETA, nTracks

Preprocessing

Quantile Transformer (Normal)

Latent space

Normal, 64

DLLx, P, ETA, nTracks

P, ETA, nTracks

Concatenate

Critic

Dense, 128, ReLU

Dense, 128, ReLU

⋮

Dense, 128, ReLU

Dense, 256, Linear

x10

Generator

Dense, 128, ReLU

Dense, 128, ReLU

⋮

Dense, 128, ReLU

Dense, 5, Linear
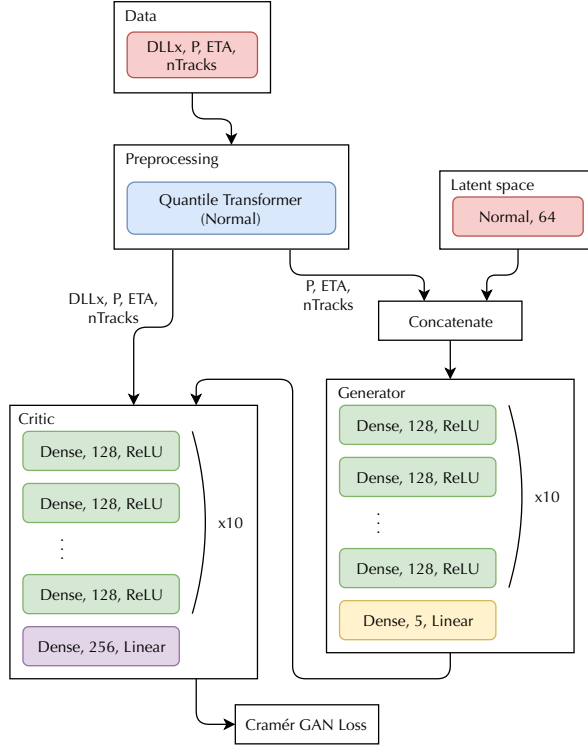
x10

Cramér GAN Loss

Figure 4: RICH GAN model architecture. Generator is trained to produce five particle identification responses for the RICH detector for three input parameters: particle momentum, pseudorapidity, and the total event track multiplicity.

3 Generative Model for the RICH Particle Identification

The appropriate generation of the LHCb RICH detector [4] response requires detailed simulation of the Cherenkov photon production in the body of the detector, its transport to the photodetector, including reflection and refraction on the way, its registering by the photodetectors, providing good angular precision for the Cherenkov photons. Collected signals are used then for testing each charged particle candidate reconstructed in the tracker against six possible mass hypotheses: electron, muon, pion, kaon, proton, "below threshold". The logarithm of the ratio of the likelihood for each hypothesis, except the pion one, to the likelihood of the pion hypothesis, `RichDLL*`, with '*' standing for 'e', 'mu', 'k', 'p', and 'bt' respectively is associated with every reconstructed charged track and is used for the particle identification in the following physics analyses.

Taking into account symmetry around the beam axis, this chain in fact converts kinematics of the track, momentum p and pseudorapidity η into five likelihoods for different hypothesis. This gives a possibility to substitute the directly calculated transfer function which includes micro-level detector simulation and detector reconstruction for the effective surrogate model. `RichDLL*` values also depend on the multiplicity in the event, since high track density might lead to a reconstruction algorithm confusion. The proxy variable to a total multiplicity is the number of reconstructed tracks in the event. Thus, the full surrogate model may have three input parameters: (p, η, N_{tracks}) where the latter is a total number of tracks in the event.
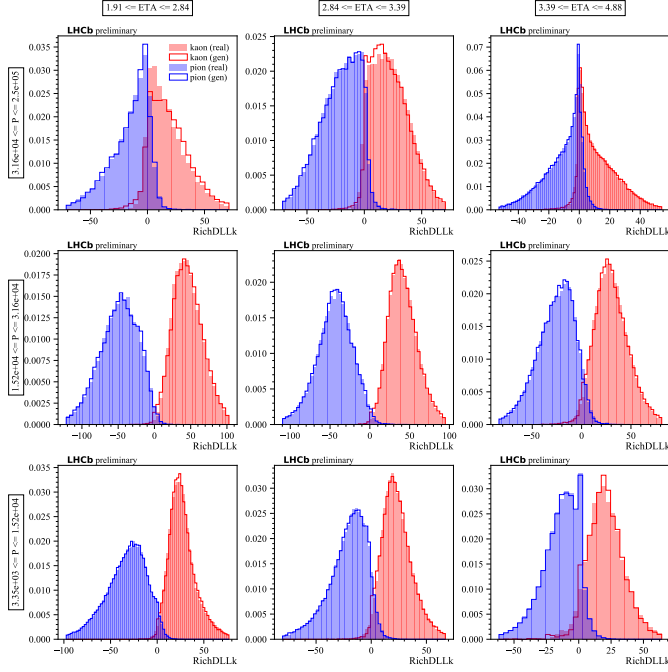
Figure 5: Visual comparison of RICH GAN quality. Weighted real data (filled histograms) and generated (open histograms) distributions of `RichDLLk` for kaon (red) and pion (blue) track candidates in bins of pseudorapidity (ETA) and momentum (P) over full phase-space.

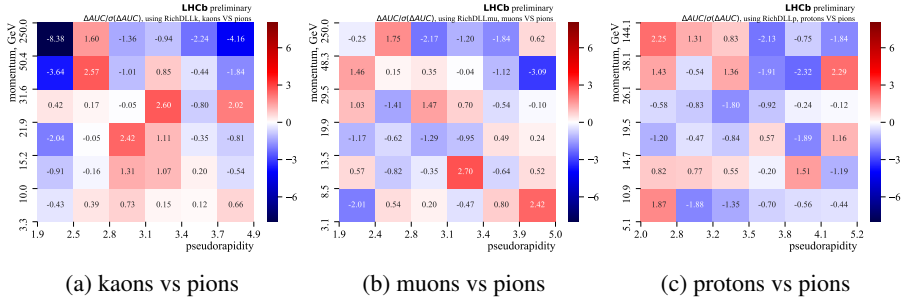| (a) kaons vs pions | (b) muons vs pions | (c) protons vs pions |

Figure 6: Separation power comparison. Differences between ROC AUC for discriminating pions from kaons, muons and protons classifiedd with the `RichDLLk`, `RichDLLmu` and `RichDLLp` variables, respectively for real calibration samples and using RICH GAN generated values, divided by the ROC AUC uncertainties, in bins of momentum and pseudorapidity.

Technical details of this approach are described in Ref. [5]. The Cramer GAN approach [6] build using fully connected neural network layers presented in Fig. 4 is used to build and train the surrogate generative model. The model was trained using calibration sample of well identified decays in real data [7]. While true id of the particles is stochastic, thus unknown, the usage of identified decays provides the information of sWeights [8], which is used in the training process.
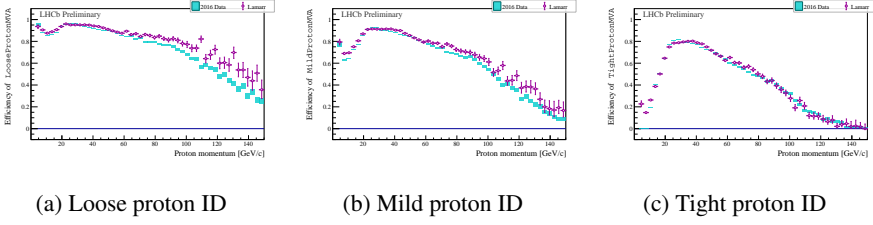
(a) Loose proton ID	(b) Mild proton ID	(c) Tight proton ID

Figure 7: Comparing proton identification efficiency. Efficiency of three different require-
ments on the output of a Multi-Layer Perceptron trained to identify protons as evaluated on
protons tagged through the decay $\Lambda_b^0 \rightarrow \Lambda_c^+ \mu \bar{\nu}_\mu$ with $\Lambda_c^+ \rightarrow pK^-\pi^+$. The efficiency is com-
pared, in bins of the proton momentum, for a dataset selected without introducing bias on
the particle identification of the proton (cyan shaded area), and a Fast Simulation sample
where the rich and the calorimeter responses are modeled through a Generative Adversarial
Network trained using protons from $\Lambda^0 \rightarrow p\pi^-$ decays, only (purple markers).

The distributions for `RichDLLk` values obtained for pions and kaons directly from the
corresponding calibration data samples and generated by the surrogate model for MC pions
and kaons for different regions in (p, η) phase space are presented in Fig. 5.

The primary requirement for the surrogate model is to properly reproduce a discrimina-
tion power of corresponding hypothesis estimators `RichDLL*`. Thus, Fig. 6 represents the
difference between separation power, ROC AUC, of original and surrogate `RichDLL*` values,
relative to the uncertainty of the ROC AUC calculation. This comparison is presented in
different bins in momentum and pseudorapidity, and demonstrates that the deviation in most
cases does not exceed 1-2 standard deviations, and is essentially unbiased for different bins.

As the results of simulation are used in the subsequent PID algorithm, the ultimate met-
ric ultimate quality metrics for the surrogate `RichDLL*` model is the correct reproduction
of identification power. Fig. 7 presents such a comparison for the proton identification ef-
ficiency for different proton ID requirements. Demonstrated consistency, especially for the
tight requirements, confirms the feasibility of this approach.

4 Conclusions

In this paper we demonstrated two approaches to significant speedup of the simulation of
two most expansive computationally components of the LHCb detector: electromagnetic
calorimeter and RICH. In the first case the surrogate generative model is built on top of
the highly detailed `Geant4` response. The surrogate model for the RICH based particle iden-
tification is built on top of real calibration data samples, thus bypassing the simulation and
digitization steps of the MC event production for this detector completely.

5 Acknowledgements

References

[1] LHCb collaboration, CERN-LHCC-2018-007, LHCb-TDR-017.

[2] T.C. Wang, M.Y. Liu, J.Y. Zhu, G. Liu, A. Tao, J. Kautz, B. Catanzaro, arXiv preprint arXiv:1808.06601 (2018).

[3] V.Chekalina et al., EPJ Web Conf. **214** 02034 (2019).

[4] Alves Jr A A et al. (LHCb), JINST **3**, 8005 (2008).

[5] A Maevskiy et al., Fast Data-Driven Simulation of Cherenkov Detectors Using Generative Adversarial Networks, 19th International Workshop on Advanced Computing and Analysis Techniques in Physics Research (in materials), arXiv preprint arXiv:1905.11825 (2019).

[6] M. G. Bellemare et al., arXiv preprint arXiv:1705.10743 (2017).

[7] R. Aaij et al., EPJ Tech.Instrum. **6** no.1, 1 (2019)

[8] M. Pivk and F. R. Le Diberder, Nucl. Instrum. Meth. **A555**, 356 (2005)