

## Fast calorimeter simulation in LHCb

*Thursday 5 July 2018 15:30 (15 minutes)*

In HEP experiments CPU resources required by MC simulations are constantly growing and become a very large fraction of the total computing power (greater than 75%). At the same time the pace of performance improvements from technology is slowing down, so the only solution is a more efficient use of resources. Efforts are ongoing in the LHC experiments to provide multiple options for simulating events in a faster way when higher statistics is needed. A key of the success for this strategy is the possibility of enabling fast simulation options in a common framework with minimal action by the final user. In this talk we will describe the solution adopted in Gauss, the LHCb simulation software framework, to selectively exclude particles from being simulated by the Geant4 toolkit and to insert the corresponding hits generated in a faster way. The approach, integrated within the Geant4 toolkit, has been applied to the LHCb calorimeter but it could also be used for other subdetectors. The hits generation can be carried out by any external tool, e.g. by a static library of showers or more complex machine-learning techniques. In LHCb generative models, which are nowadays widely used for computer vision and image processing are being investigated in order to accelerate the generation of showers in the calorimeter. These models are based on maximizing the likelihood between reference samples and those produced by a generator. The two main approaches are Generative Adversarial Networks (GAN), that takes into account an explicit description of the reference, and Variational Autoencoders (VAE), that uses latent variables to describe them. We will present how both approaches can be applied to the LHCb calorimeter simulation, their advantages as well as their drawbacks.

**Primary author:** RATNIKOV, Fedor (Yandex School of Data Analysis (RU))

**Presenter:** RATNIKOV, Fedor (Yandex School of Data Analysis (RU))

**Session Classification:** Computing and Data Handling

**Track Classification:** Computing and Data Handling