Generative models for fast simulation

S.Vallecorsa[1,2]

[1] CERN, Geneva, Switzerland
[2] Gangneung-Wonju National University, Gangneung, South Korea

E-mail: sofia.vallecorsa@cern.ch

Abstract. Machine Learning techniques have been used in different applications by the HEP community: in this talk, we discuss the case of detector simulation. The need for simulated events, expected in the future for LHC experiments and their High Luminosity upgrades, is increasing dramatically and requires new fast simulation solutions. We will present results of several studies on the application of computer vision techniques to the simulation of detectors, such as calorimeters. We will also describe a new R&D activity, within the *GeantV* project, aimed at providing a configurable tool capable of training a neural network to reproduce the detector response and replace standard Monte Carlo simulation. This represents a generic approach in the sense that such a network could be designed and trained to simulate any kind of detector and, eventually, the whole data processing chain in order to get, directly in one step, the final reconstructed quantities, in just a small fraction of time. We will present the first three-dimensional images of energy showers in a high granularity calorimeter, obtained using Generative Adversarial Networks.

1. Introduction

Particle transport simulation is a building block of any physics experiment from detector design and R&D to the final steps of analysis and comparison to theoretical models. The traditional approach is based on Monte Carlo methods, which is a time consuming procedure requiring a tremendous amount of computation. For the past few years Monte Carlo simulations have represented more than 50% of the WLCG (LHC Computing Grid) workload and the simulation demands expected for the High Luminosity LHC runs in 2025 will increase significantly [1]. Unfortunately, WLCG is expected to hit the power consumption limits in several of the current centres but, at the same time, it is foreseeable that its budget will stay flat at best, or even decrease. All considered, the LHC experiments expectation is that the exploitation of the High Luminosity LHC will require a factor of 100 improvement in simulation throughput with the same computing resources available today [1].

Several initiatives have been trying to approach this problem from different perspectives. On one hand, techniques that try to reduce the quantity of data that needs to be simulated are being studied by the experiment. One example is the overlay technique that attempts to reduce the amount of simulated events needed to correctly describe physics quantities in high pile-up environment while improving data-simulation comparison. The idea is to mix data and simulation in order to reduce CPU time and memory usage [2].

On the other hand, different level of optimisations are being studied, in order to speed-up the existing simulation software and make it better suited to leverage on the latest hardware advancement. Multi-threading and a task based approach are being introduced in different

frameworks, for example in GaudiMP [3]. Event level parallelism and multi-threading are being implemented in Geant4 [4]) to improve throughput.

A new prototype for particle transport simulation, *GeantV*, is being developed to improve physics accuracy and performance, in particular on modern architectures, such as the Intel Xeon Phi and (GP)GPUs [5]. It is aimed at introducing fine grained parallelism to achieve a factor 5 speedup with respect to *Geant4* by exploiting vectorisation, concurrency and locality. The project has raised large interest within different LHC experiments: improved geometry algorithms such as the VecGeom library [6] and a new SIMD library (VecCore [7]) have been developed within *GeantV* and are now used by a much larger community.

It is clear, however that, in order to increase the performance of simulation applications by two orders of magnitude, new methods and algorithms are to be studied as a complement to the basic work of parallelizing and optimizing the detailed simulation.

Fast simulation approaches are already used by the HEP community to reduce computation time, typically for cases in which it is possible to trade-off accuracy for speed (e.g. searches, upgrade studies, ...). Both the ATLAS and CMS experiments, for example, use pre-simulated EM-showers libraries to replace the detailed simulation of their forward calorimeters [9] [10]. GFlash [11] simulates electromagnetic and hadronic showers using parametrizations for the longitudinal and lateral profile. Another example is Delphes [8]: a C++-based framework for fast simulation of general purpose experiments, capable of simulating tracking systems (in magnetic field), calorimeters and muon systems. Each of these solutions can reach different performance in terms of speed improvements (x10 - x1000) and different levels of accuracy (generally around 10% with respect to full simulation). See, for example the figure 1 (left), comparing energy shower shapes in the FCC electromagnetic calorimeter, simulated using *Geant4* and GFlash.

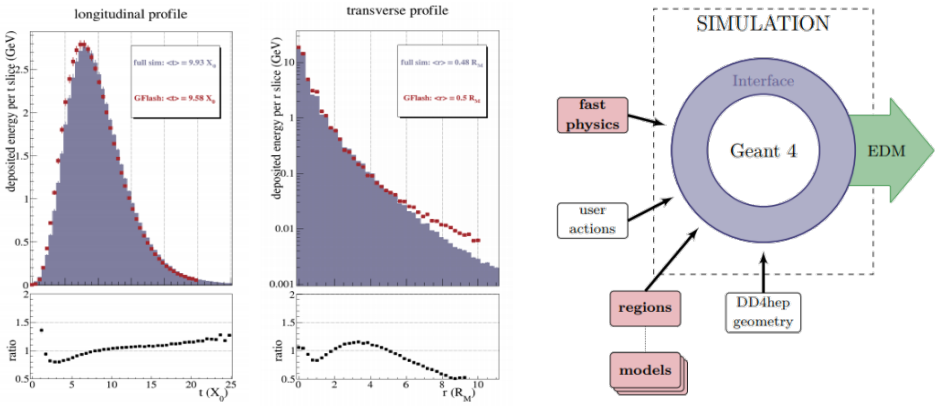

Figure 1. (left) FCC calorimeter showers GFlash results and comparison to full simulation. (right) Gaudi framework for integrating fast and full simulation [11]

The shortcoming of most of the fast simulation approaches used so far is the fact that they remain very specific to individual experiments. Instead, a generic coherent simulation framework, capable of combining different simulation types, is essential to facilitate and speedup data analysis and the comparison of the data. The idea behind such a simulation framework would be to use a single input for the detector geometries needed by the different simulation types and an identical analysis event data model [13]. Figure 1 shows, on the right, an example of a generic simulation framework that is being developed for the FCC communities [13], using the Gaudi architecture [12]. The ATLAS Integrated Simulation Framework is another example of this approach [16]. *Geant4* also has a mechanism sending particles through different simulations defined in certain detector regions.

A new R&D activity has recently started within *GeantV* to develop a generic, fully customizable, fast simulation framework: the structure of *GeantV* simulation steering allows to attach different simulation techniques to specific particle types, or energy ranges or single sub-detectors. This results in the possibility of seamlessly switching from detailed simulation to fast simulation in order to balance accuracy and speed. *GeantV* fast simulation framework is intended to make a varied palette of tools available to the user. Among those tools, a new deep learning based simulation is being developed. The recent developments in deep learning, coupled to the advancements of computing hardware, provide, in fact, the opportunity to replace complex algorithms with deep neural networks capable of reproducing the same results at a much higher speed.

2. Machine Learning in High Energy Physics

The use of machine learning methodologies in High Energy Physics is not a new approach [17, 18] and a lot of work has been done in that regard. The highly stochastic and non deterministic nature of High Energy Physics combined with complex and intricate nature of interactions makes the traditional approaches very time consuming [19]. One of the pioneering work in the quest for the elusive Higgs boson through machine learning was undertaken at Fermilab [20]. Further work proved that deep learning replaced the tedious feature engineering procedures and provided similar performance using low level features [21] as well as discovering novel high level features providing higher classification accuracy [22]. Image recognition networks employing convolutional and fully connected layers to classify between jets from single hadronic particles and overlapping jets from pairs of collimated hadronic particles resulted in modest performance improvement [23]. Moreover, the work of machine learning for HEP is not just limited to discrimination but also regression [24] and triggering [25]. A recent work even discussed analogies between quantum wave function and deep convolutional arithmetic circuits (convolutional layers with linear activation and pooling layers) [26]. A new frontier is that of physics simulation: only recently machine learning has been used to replace simulation [24], but the interest in these techniques is rapidly spreading through the experimental community.

3. Generative Models and Generative Adversarial Networks

Generative models like Generative Stochastic Networks [27], Variational AutoEncoders [28] and Generative Adversarial Networks [29] seem particularly suited to replace Monte Carlo simulation. The production of realistic samples is straightforward, as these techniques can model complicated probability function and deal with multi-modal output. Their ability to perform interpolation and to recover missing data has been proven several times [30, 31], two features that are particularly useful when dealing with particle physics simulations.

Introduced by I. Goodfellow in 2014, Generative Adversarial Networks consist of two networks, a generator and a discriminator, competing against each other [29]. Following the original implementation, a mathematical formulation of the problem can be expressed as follows. The generator is designed to learn some data distribution, starting from a prior on input noise variables $p_z(z)$: it can be represented as a differentiable function mapping z to the data space as $G(z;\theta_g)$ with parameters θ_g. The discriminator network is represented by a second differentiable function $D(x;\theta_d)$ that outputs a single scalar: it represents the probability that x came from the data rather than p_z. The value function, V, is defined as:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_g(x)}[logD(x)] + \mathbb{E}_{z\sim p_z(z)}[log(1-D(G(z)))] \quad (1)$$

The first term on the right hand side in the above equation denotes the loss for real data while the second term is the loss for generated data. The discriminator tries to maximize V, while generator tries to minimize it and the problem translates in finding the equilibrium saddle point.

Initially GANs were introduced as a Multi Layer Perceptron (MLP) networks but these networks suffered from high rate of non-convergence [29]. GAN shortcomings were further investigated and improvements were suggested to meet them. LAPGAN presented the idea of employing convolutional neural networks with the adversarial training [32]. Another version of GAN, called Deep Convolutional GAN or DCGAN, was also presented exploring unsupervised learning with Convolutional Neural Networks for discriminator and generator [33]. DCGAN model was explored in detail and architectural guidelines, like using strided convolutions instead of upsampling, or deconvolution for avoiding pooling layers, were presented. Batch normalization was suggested and fully connected layers were completely removed from the networks except the first layer in Generator and the last in the Discriminator. They also advocated Rectified Linear Units as most suitable activation functions for Generators and leaky Rectified Linear Units for Discriminators. Image generation was also conditioned to a label in a number of ways. An MLP implementation incorporated class labels as one hot vectors mapped to additional hidden dense layer before being combined with hidden layer generated by latent noise [34]. Another application of GAN employed Convolutional Neural Networks while both the Generator and Discriminator were conditioned on the class label implemented as embedding layer [35]. SGAN demonstrated semi-supervised approach by training the discriminator to predict labels for images and then the Generator utilizing these labels for generation [37]. Image generation and classification through GAN was further improved through techniques like feature matching, minibatch discrimination, historical averaging, label smoothing and batch normalization [36]. ACGAN adopts the semi-supervised approach and demonstrates that introduction of a label results in faster convergence and stable performance [38].

4. The LAGAN and CaloGAN applications

In High Energy Physics, some detector outputs can be also be interpreted in the form of images, thus the same techniques that are used for image recognition can be employed for detector output analysis. The LAGAN or Location Aware GAN [41] is one of the first applications of this approach, successfully reconstructing two-dimensional representations of jet images in calorimeters. To account for the specificity of jet images with respect to typical image reconstruction problems, namely their sparsity and highly non-linearly location-dependent data, the LAGAN networks make use of two-dimensional convolutional layers, locally connected layers and leaky rectified linear units, following work previously done on jet classification [42]. CaloGAN networks [43] extend the LAGAN work to generate images of energy deposited by particles travelling through a simplified detector geometry resembling the ATLAS LAr calorimeter: three instrumented layers in the radial (z) direction, with different thicknesses and different transverse segmentations. As an example, figure 4 represent the three transverse energy showers, one for each of the layers, created by a 10 GeV e^+ incident perpendicular to the center of the detector. The three images corresponds to the three calorimeter layers.

The CaloGAN architecture implements a LAGAN unit per each calorimeter layer plus a trainable transfer unit to preserve layer correlations. The result is a concatenation of two-dimensional images that reproduce the full three-dimensional picture. Figure 3 (left) shows a comparison of different energy shower shape variables and other event level variables as simulated by *Geant4* and CaloGAN. Different levels of agreement are reached depending on the variables and particle types and, although not perfect, the results prove that the CaloGAN network can reproduce successfully the main shower features.

The images are also conditioned on energy: the discriminator calculates the reconstructed energy per layer as well as the total energy. These quantities are used to build a specific loss components introduced to penalize the absolute deviation between the nominal energy and the reconstructed energy. Figure 3 (right), shows the energy response corresponding to different primary energy values. It is interesting to note that the system was trained over the $[0, 100]$

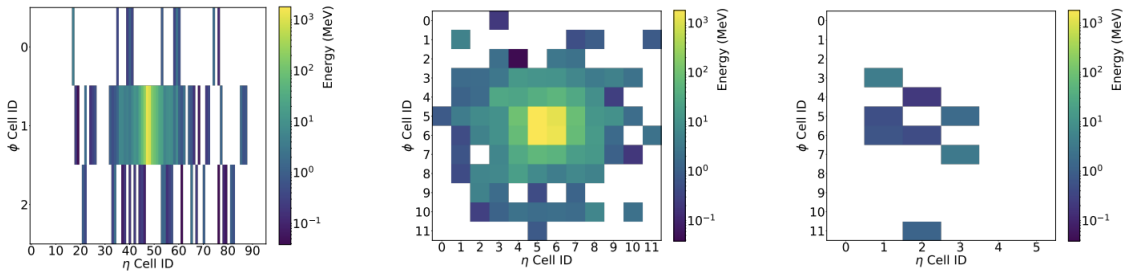

Figure 2. The three two-dimensional images, one for each calorimeter layer, represent together a 10 GeV e^+ incident perpendicular to the center of the detector.

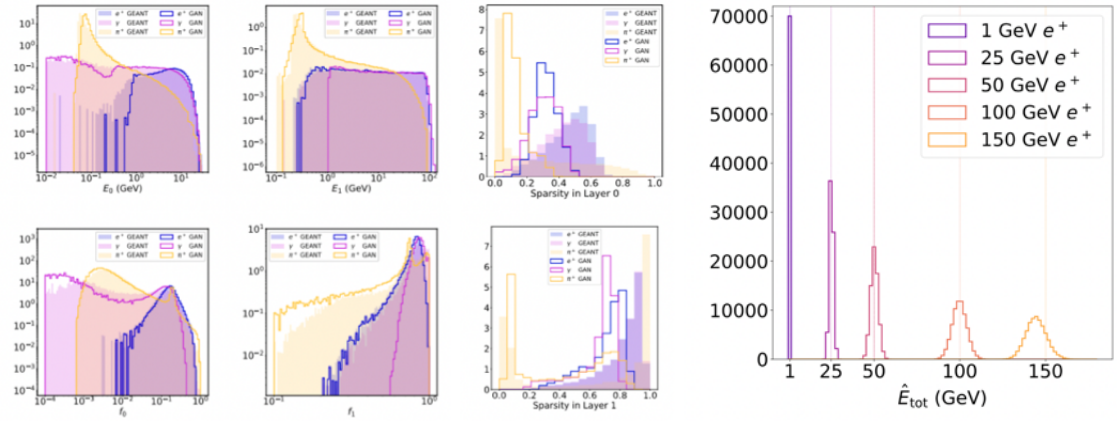

Figure 3. (left) Comparison of typical energy shower shape variables for the Geant and CaloGAN datasets for electrons, photons and pions: total deposited energy, fraction of measured energy and sparsity in the first and second layer. (right) CaloGAN energy response for primary particle energies of 1 GeV, 25 GeV, 50 GeV, 100 GeV, and 150 GeV.

GeV energy range, but it was still capable of generating images with energies lying outside the training range (150 GeV), although with broader width and shifted towards the training domain. This fact represent a first hint of the network extrapolation capability although the authors do no investigate the level at the which the extrapolated samples show the expected shower distributions. At the time of this writing, additional work is needed to fully validate the quality of the generated images with respect to detailed simulation or existing fast simulation methods. Nevertheless, CaloGAN represents a milestone in the development of deep learning application to HEP simulation.

5. The three-dimensional GAN

The *GeantV* three-dimensional GAN application steps further simulating three-dimensional calorimeter showers as a whole. It is intended as a first proof of concept to understand the level of accuracy achievable and to what extend the approach can be generalised to different detectors. It represent, in fact, the first step towards a fully integrated fast simulation tool configurable and trainable according to the different user needs in terms of physics and the nature of the detector.

The CLIC electromagnetic calorimeter design (ECAL) is chosen as an example of future high granularity detector [44]. The data is simulated using the DD4hep software framework [45]: it consists of energy showers produced by incoming particles travelling through the detector. They are generated with *Geant4*: each calorimeter cell is characterized by the energy recorded in it and three indices (iX, iY, iZ), identifying the position of the cell. The ECAL is made of 25 instrumented layers and for each of them, a 25x25 array of cells is considered. The energy of the incoming particle is sampled from a uniform [0-500 GeV] spectrum, a much larger energy range than the one tested by CaloGAN. Fig. 4 shows, on the right, an example energy shower, produced by a 100 GeV electron inside ECAL.

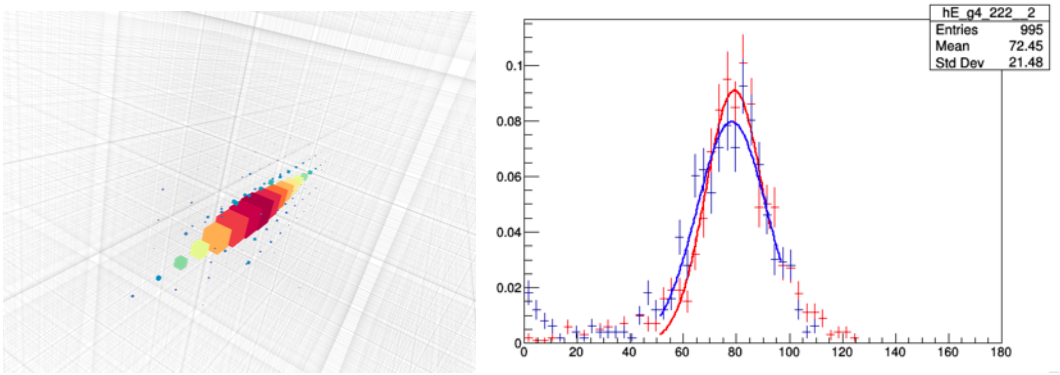

Figure 4. Typical energy showers produced by a 100 GeV electron inside the electromagnetic calorimeter. The grey lines represent identify the calorimeter cell. The color and hit dimensions correspond to the amount of energy deposited in the cell.

The *GeantV* GAN performs three-dimensional image reconstruction: the discriminator as well as generator exploit three-dimensional convolutional layers: the convolution parameters are adjusted to match the energy shower shapes. Usage of three-dimensional layers leads to a large number of training parameters, and therefore we keep the network as simple as possible. The model has been developed using Keras [47] and Tensorflow [48]. There are four convolutional layers in both the generator and discriminator with leaky rectified linear units activation functions. Batch normalization layers are added after the convolutional layers for improved performance [49]. The input image is mapped to two categorical outputs obtained by the final flattened layers. The first output denotes whether images are real or generated while the second output is an auxiliary classifier. Using this basic architecture two different experiments were conducted. In the first implementation the auxiliary classifier is the particle type (e.g. electron or photon) and a sigmoidal activation function generated the discriminator output. In the second version the auxiliary output is the energy of the incoming particle (a rectified linear unit constitutes the discriminator output). The generator uses a latent noise vector initialized to a gaussian probability distribution as well as the desired class of the image to generate the primary particle energy. To assess the performance of the networks a detailed study of the generated calorimeter response is performed. Figure 4 for example, shows the typical single cell energy response: the mean and width of the cell energy deposition are compared to *Geant4* prediction. The result is generally better for cells receiving large amount of energy. The average transverse and longitudinal shower shapes are also compared against the detailed simulation (see figure 5.) Figure 6 shows instead the discriminator energy response corresponding to 100, 150 and 300 GeV input energies: the network energy prediction is good both in terms of central value and the width of the distribution, with average offset and spread well below 10%. Even more promising, from the point of view of designing a reliable simulation engine,

is the fact that the network is capable of correctly describing how the energy shower shape distributions change depending on the primary particle energy: an example is shown in figure 7: the longitudinal shower shapes generated by GAN are compared to the detailed simulation predictions for different primary particle energies. Further work is currently ongoing to compare the three-dimensional GAN results to different fast simulation tools and to study the possibility to generalise the tool to different detectors via algorithm meta-optimisation and hyper-parameter scans.

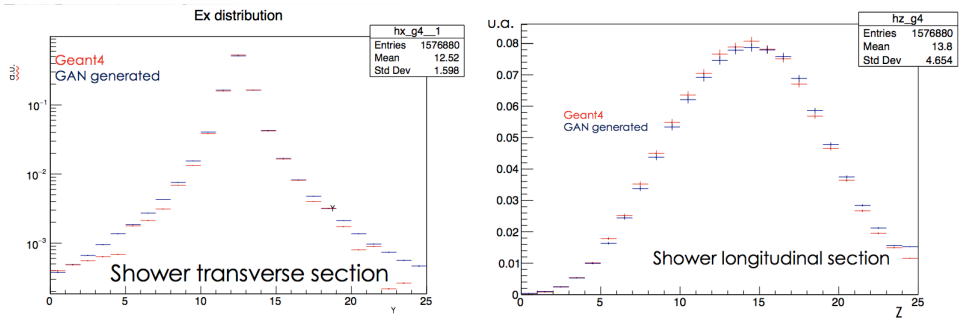

Figure 5. The transverse and longitudinal shower shapes generated for 100 GeV electrons.

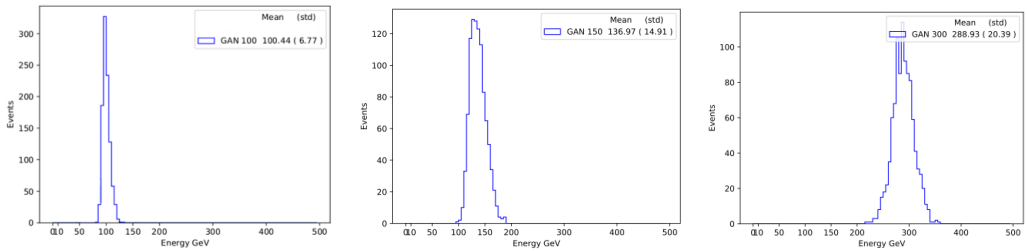

Figure 6. The discriminator energy prediction corresponding to 100, 150 and 300 GeV primary energies.

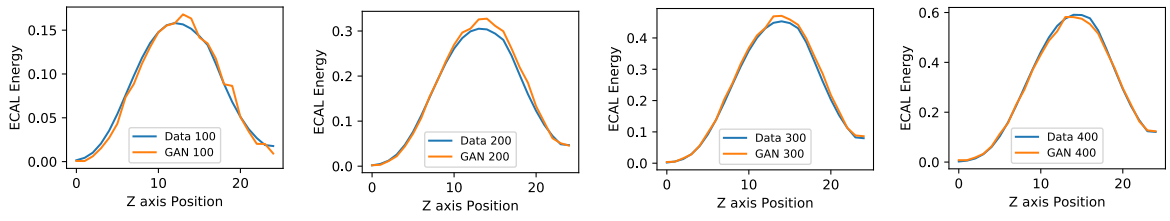

Figure 7. The longitudinal shower shapes generated by three-dimensional GAN for different primary particle energies (100, 200, 300, 400) GeV (orange). The GAN shape is compared to the corresponding detailed simulation results (blue).

6. Conclusion

Monte Carlo production has been so far a major fraction of WLCG computing workload and the High Luminosity LHC experiments needs will scale up orders of magnitude. A generic

framework with common fast simulation algorithm and strategies for mixing detailed and fast simulation could bring great benefit to HEP and also to those smaller communities that could not afford to develop their own simulation framework. In this context generative models, relying on the possibility to interpret detector response as images, seem natural candidates to speedup simulation. Generative Adversarial Networks, in particular, require relatively small amount of data to train and are the subject of many ongoing studies. Their performance as imaging tools for calorimeter simulation is very promising, and from a computing resources perspective, the gain in time needed to generate a shower is huge: for example, using the three-dimensional GAN takes only $O(10^{-3})$ ms compared to the typical one minute time needed to generate it using the detailed approach. This corresponds to a speedup of more than 6 orders of magnitude. The accent should therefore be on optimising the computing resources needed to train the networks, studying parallelisation on clusters and cross-platform development.

7. Acknowledgement

The authors wish to acknowledge the contribution of Intel to the *GeantV* project through the Intel Performance Computing Centre (IPCC) program. We also want to acknowledge the very useful technical contributions of the CERN openlab.

References

[1] Ian Bird. Workshop Introduction, Context of the Workshop: Half-way through Run2; Preparing for Run3, Run4. (2016). https://indico.cern.ch/event/ 555063/contributions/2236372/ WLCG Workshop.
[2] The ATLAS collaboration. arXiv:1603.02934 (2016).
[3] . M. Clemencic at al. Journal of Physics: Conference Series, Volume 608, conference 1
[4] GEANT4 Collaboration, Nuclear Instruments and Methods in Physics Research A 506, 250 (2003).
[5] G Amadio et al 2016 J. Phys.: Conf. Ser. 762 012019.
[6] J Apostolakis et al 2015 J. Phys.: Conf. Ser. 608 012023
[7] https://github.com/root-project/veccore
[8] The DELPHES 3 collaboration, de Favereau, J., Delaere, C. et al. J. High Energ. Phys. (2014) 2014: 57.
[9] The ATLAS collaboration, ATL-SOFT-PROC-2017-005
[10] Iammanco, Andrea. (2014). Journal of Physics: Conference Series. 513. 022012. 10.1088/1742-6596/513/2/022012.
[11] G. Grindhammer and M. Rudowicz and S. Peters, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 290-2, 1990, 469-488.
[12] Gaudi webpage, http://cern.ch/gaudi
[13] J. Hrdinka, A. Zaborowska, A. Salzburger, B. Hegner. PoS(EPS-HEP2015)248
[14] G. Aad et al. (ATLAS), Phys. Lett. B716, 1 (2012), arXiv:1207.7214 [hep-ex].
[15] S. Chatrchyan et al. (CMS), Phys. Lett. B716, 30 (2012), arXiv:1207.7235 [hep-ex].
[16] E. Ritsch, ATLAS Detector Simulation in the Integrated Simulation Framework applied to the W Boson Mass Measurement, Universitaet Innsbruck, 2014, http://physik.uibk.ac.at/hephy/theses/diss_er.pdf
[17] P. Vannerem et al. 1999, hep-ex/9905027.
[18] R.K. Bock et al. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 516,2511-528,2004
[19] Bhat, Pushpalatha C, FERMILAB-PUB-10-425-PPD.
[20] Engineering Applications of Artificial Intelligence, 22, 8, 1203 - 1217, 2009
[21] P. Baldi, P. Sadowski and D. Whiteson, arXiv: 1402.4735
[22] Sadowski, Peter and Collado, Julian and Whiteson, Daniel and Baldi, Pierre, Proceedings of the 2014 International Conference on High-Energy Physics and Machine Learning - Volume 42, 2014
[23] D. Whiteson et al. arXiv:1603.09349
[24] A. Vitek at al. 2013 5th International Conference on Intelligent Networking and Collaborative Systems. 121-126
[25] V V Gligorov and M Williams, Journal of Instrumentation, 8, 02, 2013
[26] Yoav Levine et al., arxiv.org:1704.01552, 2017
[27] Yoshua Bengio et al., arxiv:1306.1091, 2013
[28] D. Kingma et al., arxv:1312.6114, 2013.
[29] I.J.Goodfellow et al. 2014. arXiv:stat.ML/1406.2661.
[30] , Scott E. Reed et al., arxiv:1605.05396, 2016.

[31] Ranzato, Susskind, Mnih, Hinton, IEEE CVPR 2011.
[32] EmilyL.Denton,SoumithChintala,ArthurSzlam,andRobertFergus.2015.Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks. CoRR abs/1506.05751 (2015). http://arxiv.org/abs/1506.05751.
[33] A. Radford et al. 2015.Arxiv e-prints: 1511.06434
[34] M. Mirza and S. Osindero. 2014. Conditional Generative Adversarial Nets. Arxiv e-prints: 1411.1784
[35] J. Gauthier. 2014. Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition, Winter semester 2014, 5 (2014), 2
[36] T. Salimans et al. 2016. Improved techniques for training gans. In Advances in Neural Information Processing Systems. 2234?2242.
[37] A.Odena.2016.Semi-SupervisedLearningwithGenerativeAdversarialNetworks. ArXiv e-prints (June 2016). arXiv:stat.ML/1606.01583
[38] A. Odena, C. Olah, and J. Shlens. 2016. Conditional Image Synthesis With Auxiliary Classi er GANs. ArXiv e-prints (Oct. 2016). arXiv:stat.ML/1610.09585
[39] JiajunWu,ChengkaiZhang,TianfanXue,BillFreeman,andJoshTenenbaum. 2016. Learning a Probabilistic Latent Space of Object Shapes via 3D Generative- Adversarial Modeling. In Advances In Neural Information Processing Systems. 82?90.
[40] M. Pierini. 2016. The LCD dataset. (2016). https://indico.hep.caltech.edu/ indico/contributionDisplay.py?sessionId=9&contribId=16&confId=102 DSHEP at the Simons Foundation.
[41] M. Paganini, L. de Oliveira, B. Nachman.arXiv preprint. arXiv:1701.05927.(2017).
[42] J. Barnard, E. N. Dawe, M. J. Dolan and N. Rajcic, arXiv preprint. arXiv:1609.00607
[43] M. Paganini, L. de Oliveira, and B. Nachman. arXiv preprint. arXiv:1705.02355 (2017).
[44] The CLIC collaboration. http://cds.cern.ch/record/2254048
[45] M Frank et al 2014 J. Phys.: Conf. Ser. 513 022010
[46] M. Pierini. DS@HEP at the Simons Foundation (2016).
[47] F. Chollet et al. 2015. Keras. https://github.com/fchollet/keras. (2015).
[48] M. Abad et al., TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org
[49] TimSalimans,IanGoodfellow,WojciechZaremba,VickiCheung,AlecRadford, and Xi Chen. 2016. Improved techniques for training gans. In Advances in Neural Information Processing Systems. 2234?2242.