Phys. Lett. B 868 (2025) 139694

Contents lists available at ScienceDirect

Physics Letters B

journal homepage: www.elsevier.com/locate/physletb

Letter

Point cloud-based diffusion models for the Electron-Ion Collider

Jack Y. Araz [a,b,c], Vinicius Mikuni [d], Felix Ringer [a,b,c], Nobuo Sato [b], Fernando Torales Acosta [e,*], Richard Whitehill [b]

[a] Center for Nuclear Theory, Department of Physics and Astronomy, Stony Brook University, 11794, NY, Stony Brook, USA
[b] Thomas Jefferson National Accelerator Facility, Newport News, 23606, VA, USA
[c] Department of Physics, Old Dominion University, 23529, VA, Norfolk, USA
[d] National Energy Research Scientific Computing Center, Berkeley Lab, Berkeley, 94720, CA, USA
[e] Physics Division, Lawrence Berkeley National Laboratory, Berkeley, 94720, CA, USA

ARTICLE INFO

Editor: H Gao

ABSTRACT

At high-energy collider experiments, generative models can be used for a wide range of tasks, including fast detector simulations, unfolding, searches of physics beyond the Standard Model, and inference tasks. In particular, it has been demonstrated that score-based diffusion models can generate high-fidelity and accurate samples of jets or collider events. This work expands on previous generative models in three distinct ways. First, our model is trained to generate entire collider events, including all particle species with complete kinematic information. We quantify how well the model learns event-wide constraints such as the conservation of momentum and discrete quantum numbers. We focus on the events at the future Electron-Ion Collider, but we expect that our results can be extended to proton-proton and heavy-ion collisions. Second, previous generative models often relied on image-based techniques. The sparsity of the data can negatively affect the fidelity and sampling time of the model. We address these issues using point clouds and a novel architecture combining edge creation with transformer modules called Point Edge Transformers. Third, we adapt the foundation model OmniLearn, to generate full collider events. This approach may indicate a transition toward adapting and fine-tuning foundation models for downstream tasks instead of training new models from scratch.

1. Introduction

High-energy collider experiments offer unique opportunities to probe the internal dynamics of protons and nuclei, study emergent phenomena such as hadronization, and search for physics beyond the Standard Model of particle physics. By analyzing the particles observed in detectors centered around the scattering vertex, it is possible to infer the dynamics of particles at subatomic scales. The next-generation experiment will be the future Electron-Ion Collider (EIC) [1], where high-luminosity electron-proton/nucleus scattering will be studied at center-of-mass (CM) energies up to $\sqrt{s} = 140$ GeV. Analyzing vast amounts of recorded collider data is a challenging task where machine learning tools are expected to have a significant impact on the experimental and theoretical workflows. Example applications include detector design, inference tasks, searches of beyond standard model (BSM) physics, fast detector simulations, jet classification, and unfolding. Similar considerations apply to proton-proton and heavy-ion collisions at RHIC and the LHC. For recent results, see Refs. [2–32] and references therein.

Some of the key tools to advance different areas of collider phenomenology are generative models that can be trained to generate full collider events. Various architectures have been trained in the past to generate collider events or jets including GANs [34–38], variational autoencoders [39], normalizing flows [40,41] and score-based diffusion models [42–48,38,49–51]. In particular, diffusion models have been demonstrated to produce high-fidelity samples. While the sampling time is typically relatively slow compared to GANs, it has been improved significantly using techniques such as distillation [52,44]. Score-based diffusion models learn an approximation of the score function or the gradients of the logarithm of the data probability. This approximation is then used during sampling to transform a simple distribution, such as a Gaussian distribution, into complex collider data. In Ref. [53], a first diffusion-based model was developed for EIC events based on pixelated images. Since > 99% of the pixels are empty and the distributions of dif-

* Corresponding author.
E-mail addresses: jack.araz@stonybrook.edu (J.Y. Araz), vmikuni@lbl.gov (V. Mikuni), felix.ringer@stonybrook.edu (F. Ringer), nsato@jlab.org (N. Sato), ftoralesacosta@lbl.gov (F. Torales Acosta), rwhit058@odu.edu (R. Whitehill).

https://doi.org/10.1016/j.physletb.2025.139694
Received 5 November 2024; Received in revised form 19 February 2025; Accepted 23 June 2025
Available online 27 June 2025

ferent observables can fall steeply toward their kinematic endpoints, a suitable remapping of the input variables was required. Due to its relevance in Deep Inelastic Scattering (DIS), the modeling of the scattered electron kinematics plays a critical role in electron-proton collisions that requires special attention. See also Ref. [54] where an image-based diffusion model was developed for heavy-ion collisions. Generative models at the event-level can avoid loss of information in summary statistics such as cross sections and spin asymmetries. Such a model can be used in simulation based inference for many physical analyzes relevant at the EIC, including studies of hadron structure and hadronization, and BSM searches.

In this work, we extend previous results by developing a point cloud-based diffusion model for EIC events. By using point clouds and a novel architecture that combines edge creation with transformer modules, termed Point Edge Transformers (PET), we achieve significant improvements compared to the diffusion model of Ref. [53]. In particular, we focus on success metrics such as the shape of different kinematic distributions as well as the event-wide conservation of momentum and discrete quantum numbers. We adapt the pre-existing foundation model OmniLearn [33] to generate full EIC events. OmniLearn was initially developed for both classification and generation tasks in the context of jet physics at the LHC. To generate EIC events, including full Particle IDentification (PID), we use a two-step generation process. As a first step, the scattered electron kinematics are generated. Second, the remaining particles in the event are conditioned on the electron kinematics. We expect similar multi-step generative processes may also improve the generation of full events in different collision systems. While we train the model developed here from scratch instead of fine-tuning the foundation model, our approach is closely related to OmniLearn. Our results may, therefore, point toward a transition toward adapting foundation models for different downstream tasks at collider experiments.

The remainder of this paper is organized as follows. In section 2, we describe the score-based diffusion model for EIC events developed in this work employing a point cloud data representation and the PET architecture. In section 3, we consider several metrics to evaluate the performance of the diffusion model. We consider different particle distributions and observables as well as event-wide constraints such as momentum and baryon number conservation. We conclude and present an outlook in section 4.

2. Point cloud-based diffusion models

We start by reviewing the generation of EIC events used to train the diffusion model. We then describe the model architecture and the two-step diffusion process used to generate electron-proton events.

2.1. Event generation and data representation

We generate electron-proton scattering events using PYTHIA8 [55] at a representative CM energy for electron-nucleus collisions at the EIC $\sqrt{s} = 105$ GeV. We avoid the low-Q^2 photoproduction region by imposing a cut of $Q^2 > 25$ GeV^2. We include the following list of stable particles in the data set

$$e^{\pm}, \mu^{\pm}, \nu + \bar{\nu}, \pi^{\pm}, \pi^0, K^{\pm}, K_L^0, p, \bar{p}, n + \bar{n}, \gamma\,. \tag{1}$$

We include all particles in the rapidity range $|y| < 5$, and we do not impose a lower cut on the transverse momentum. Note that here, $\nu + \bar{\nu}$ and $n + \bar{n}$ are combined due to experimental limitations in distinguishing them.

For each particle i in the event, we record its transverse momentum p_{Ti}, rapidity y_i, azimuthal angle ϕ_i, and PID$_i$. In addition, we consider the dimensionless quantity

$$\tilde{z}_i = \frac{2M_{Ti}}{\sqrt{s}} \cosh y_i\,. \tag{2}$$

Here, $M_{Ti}^2 = p_{Ti}^2 + m_i^2$ is the transverse mass, and m_i is the mass of the particle. This variable is of particular interest as it satisfies

$$\sum_{i \in \text{event}} \tilde{z}_i = 2\,, \tag{3}$$

in the CM frame due to event-wide momentum conservation. In the limit of massless particles $\tilde{z}_i$ reduces to

$$z_i = \frac{2p_{Ti}}{\sqrt{s}} \cosh \eta_i\,, \tag{4}$$

where $\eta_i = -\ln \tan \theta_i/2$ is the i^{th} particle's pseudorapidity. While the relation between a massive particle's rapidity and pseudorapidity is somewhat intricate, one can convert between the variables $\tilde{z}_i$ and z_i using the relation $\tilde{z}_i = \sqrt{z_i^2 + 4m_i^2/s}$.

2.2. Model architecture: Point Edge Transformer

This work extends the generalized machine learning model, OmniLearn [33], designed for analyzing data from particle physics experiments. The model processes inputs consisting of particles and event-level information such as the particle multiplicity and is conditioned on a diffusion time parameter t that determines the perturbation level applied to the data. In particular, for time t we apply a perturbation to data x such that $x(t) = \alpha(t)x + \sigma(t)\epsilon$, with $\epsilon \sim \mathcal{N}(0, 1)$ and perturbation parameter $\alpha(t) = \cos(\pi t/2)$ and $\sigma(t) = \sin(\pi t/2)$. The role of the network is then to predict a velocity parameter $v(t) = \alpha(t)\epsilon - \sigma(t)x$ by receiving as inputs the perturbed data, the time value, and any additional event-level information available. The time information for the diffusion process, as done in previous diffusion models for collider physics [43,44,52], is encoded to a higher dimensional space using a time embedding layer. This embedding layer utilizes Fourier features [56] and is further processed by two multi-layer perceptrons (MLPs) employing a GELU activation function [57].

The generative model designed to produce the scattered electron kinematic information is based on a fully-connected architecture incorporating multiple skip connections. Specifically, the model employs three RESNET [58] blocks, where each residual layer is connected to the output of a two-layer network through a skip connection. The model is designed to generate particles and then integrates the time-related data with particle-specific information, which includes both the kinematics of each particle and their PID after the perturbation. These inputs are transformed into a higher dimensional space using a feature embedding composed of two MLP layers. Prior to the transformer block – which is responsible for processing data in a manner that considers the relationships between particles – we insert a positional token. This token encodes the geometric context surrounding each particle in the event, aiding the transformer in understanding local particle arrangements. Although transformers are capable of capturing broad correlations among particles, adding local geometric data typically enhances the model's performance by creating a latent representation aware of particle distances [59]. The local encoding is constructed using dynamic graph convolutional network (DGCNN) [60] layers, which define each particle's neighborhood through a k-nearest neighbor algorithm, set to include ten neighbors in this work. The distances between these neighbors are measured in the specific rapidity-azimuthal angle space. For each of the k-neighbors, edge features are defined by concatenating the particle features with the subtraction between those features and the features of each respective neighbor. These edge features are then processed by a multi-layer perception (MLP), followed by an average pooling operation performed across the dimensions of the neighbors.

2.3. Two-step diffusion and training

We adopt the two-model strategy implemented in Ref. [44]. See Fig. 1 for an illustration of the model architecture developed here. The

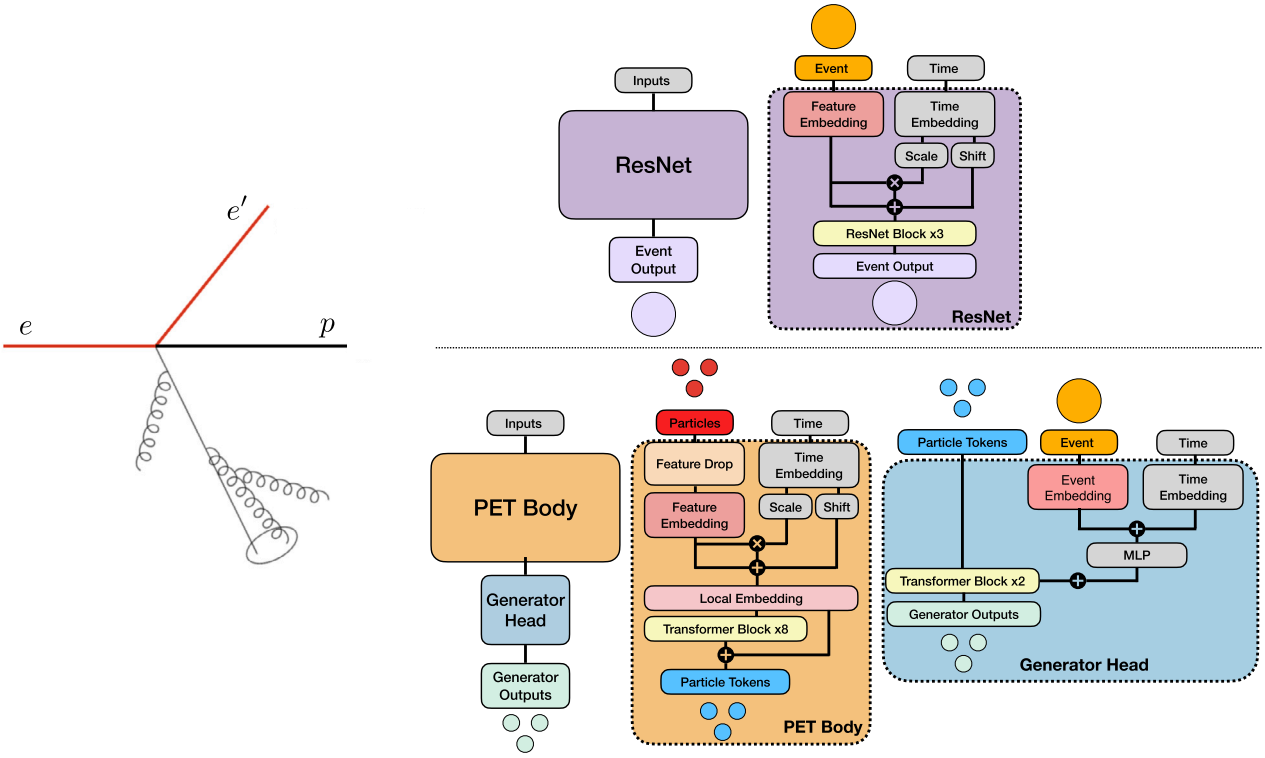

Fig. 1. Left: Electron-proton scattering event $e + p \to e' + X$. Right: Model architecture adapted from the foundation model OmniLearn [33]. The final model is composed of two diffusion models: One that generates the scattered electron and the event properties, such as the multiplicity (top), and a second model that generates all other particles in the event with their kinematics (bottom).

first model is trained to exclusively learn event-level features that are then utilized as conditional information for a second diffusion model that processes particles as inputs. Most important for this process is the total number of particles in the event, N, which is learned by the first diffusion model. N is then shared with the second diffusion model that then generates N particles and all their features for that event. Up to 50 particles are saved per event to be used during training, the maximum of all PYTHIA8 events in the training sample.

In addition to the global event variables such as multiplicity, the first model is also tasked to learn the kinematic distribution of the scattered electron in the event. This information is then used to generate the particle candidates: instead of generating the full four-momentum of each particle i we use the electron e^- to generate particles in relative coordinates, learning instead $\phi_i - \phi_e$, $y_i + y_e$, and p_{Ti}/p_{Te}. This choice of coordinates is invariant under rotations in the $y-\phi$ plane and improves the model generalization. The set of particle features learned by the second diffusion model is:

$$\log_{10}(p_{Ti}/p_{Te}),\ y_i + y_e,\ \phi_i - \phi_e,\ \log_{10}(\tilde{z}_i),\ C_i,\ \text{PID}_i, \tag{5}$$

where C_i is the particle charge, and $\tilde{z}_i$ is given in Eq. (2). The ranges of p_{Ti}/p_{Te} and $\tilde{z}_i$ still span several orders of magnitude, so the logarithm is taken for better normalization.

By conditioning the full event distributions on the dominant flow of momentum, we expect that the multi-step approach developed here can be extended to other collision systems such as eA, e^+e^-, pp, and heavy-ion collisions. We leave a more detailed exploration for future work.

The training is carried out on the Perlmutter Supercomputer [61] using 128 GPUs simultaneously with the Horovod [62] package for data distributed training. A local batch of size 256 is used with model training up to 200 epochs. OmniLearn is implemented in TENSORFLOW [63] with KERAS [64] backend. The cosine learning rate schedule [65] is used with an initial learning rate of 3×10^{-5}, increasing to $3\sqrt{128}\times10^{-5}$ after three epochs and decreasing to 10^{-6} until the end of the training. The LION optimizer [66] is used with parameters $\beta_1 = 0.95$ and $\beta_2 = 0.99$. The PET body model has 1.3M trainable weights, while the generator head has 416k trainable parameters.

3. Numerical results

In this section, we consider different kinematic distributions as benchmarks to assess the performance of the point cloud-based diffusion model. In addition, we consider several event-wide constraints and we quantitatively assess the improvement compared to the image-based diffusion model for electron-proton scattering events presented in Ref. [53].

3.1. Kinematic distributions with full PID

We start by analyzing the average particle multiplicities per event. In the left panel of Fig. 2, we show the results from the diffusion model compared to PYTHIA8 for all particle species. Overall, the diffusion model performs better for particles with higher average multiplicity. For several of the most frequently produced particles, the average yield from the diffusion model agrees with PYTHIA8 within the statistical uncertainties. For muons, which have the lowest average yield of the particles considered here, we observe differences of $\lesssim 20\%$. This can be attributed to the fact that the muon yield is three orders of magnitude lower than, for example, the photon multiplicity. Instead of considering only the average multiplicities, we plot the particle multiplicity distributions for four representative examples in the right panel of Fig. 2. Overall, we observe good agreement between the diffusion model and the PYTHIA8 results. The distributions fall over multiple orders of magnitude for large multiplicities and we observe small differences only in the tails of the distributions.

As a next step, we consider the distribution of different kinematic variables. Since the distributions exhibit rather different features, we choose electrons e^- and pions π^+ as representative examples. The results

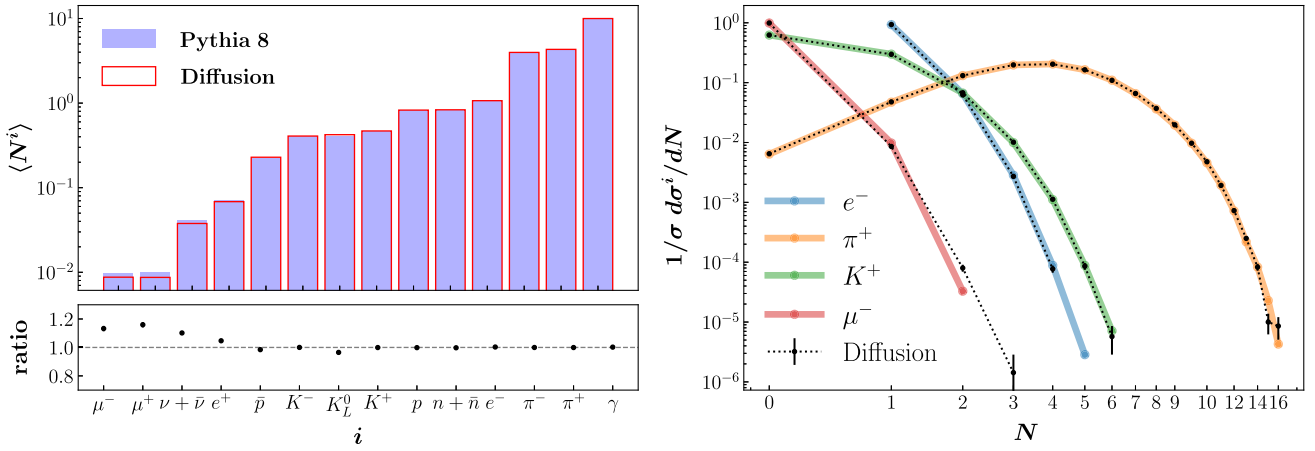

Fig. 2. Left: Average particle multiplicities produced by the diffusion model compared to the PYTHIA8 training data. Right: Comparison of the event-wide particle multiplicity distributions for electrons e^-, pions π^+, kaons K^+, and muons μ^-.

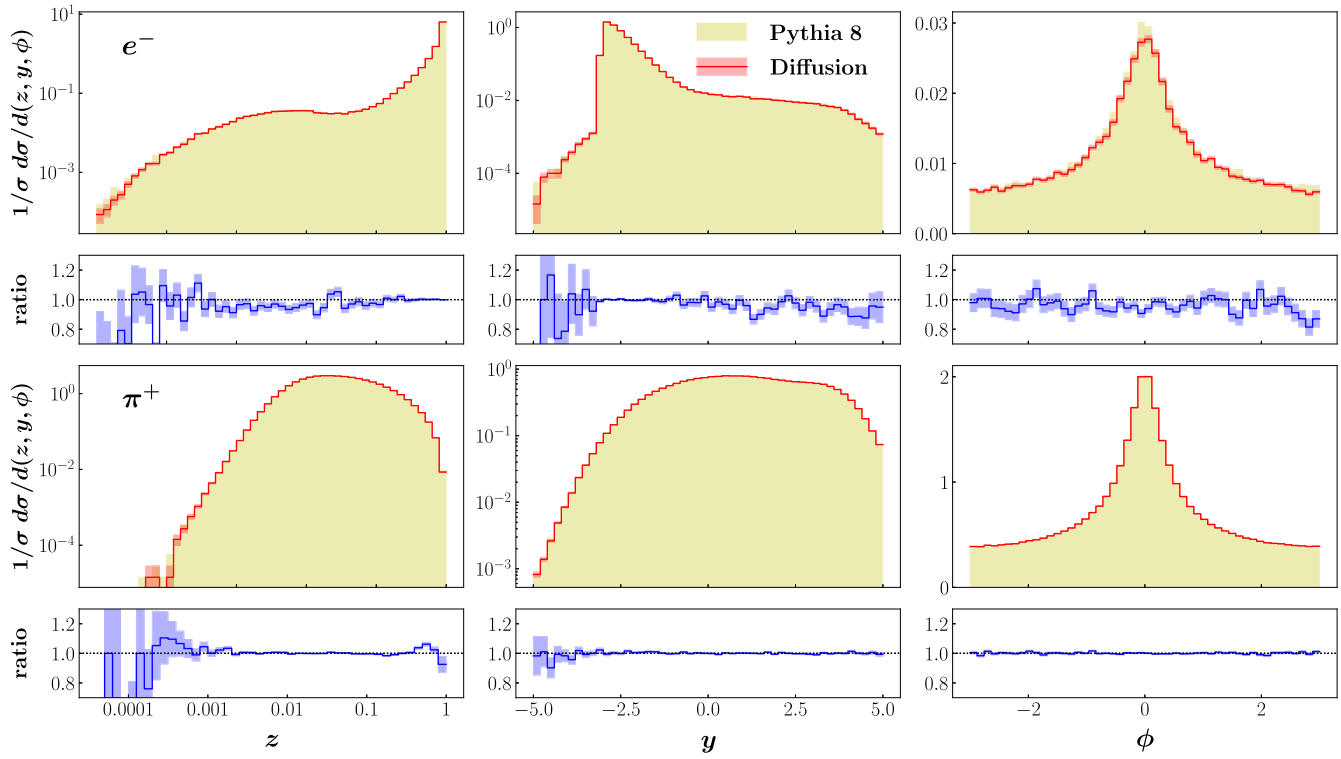

Fig. 3. Left to right: Kinematic distributions for the rescaled momentum variable z, rapidity y, and azimuthal angle ϕ (relative to the scattered leading electron). We show the diffusion model results along with the PYTHIA8 training data for electrons (top row) and pions (bottom row). The shaded red uncertainties show the statistical errors of the diffusion model. The blue error bands in the ratio plots include the statistical uncertainties from the diffusion model and PYTHIA8.

from the diffusion model compared to PYTHIA8 are shown in Fig. 3. We show histograms for the rescaled momentum fraction z, the rapidity y, and the azimuthal angle ϕ. For both particle species, the azimuthal angle is considered relative to the scattered leading electron. We observe good agreement in the bulk of the distribution and smaller deviations toward the endpoints where low statistics lead to larger uncertainties that are, however, statistically distributed around the target result. We note that these results constitute a significant improvement compared to the diffusion model for electron-proton events presented in Ref. [53]. To further evaluate the performance of the diffusion model, we consider the kinematic distributions of muons and neutrinos, which are the particles with the lowest average event multiplicities, see Fig. 2. As an example, we show a comparison of the z-distributions in Fig. 4. As expected, while the agreement between the diffusion model and PYTHIA8 is slightly worse compared to the distributions for electrons and pions in Fig. 3, we find overall satisfactory results.

Next, we consider kinematic variables that are particularly relevant for the analysis of electron-proton scattering data. First, we consider the Deep Inelastic Scattering (DIS) cross-section, which is differential in the scaling variable Bjorken x and the photon virtuality

$$x = \frac{Q^2}{2P\cdot q}, \quad Q^2 = -q^2 = -\left(k-k'\right)^2 . \tag{6}$$

Here k, k' are the four momenta of the incoming and outgoing electron, respectively, and P denotes the incoming proton momentum. Second, we consider Semi-Inclusive DIS (SIDIS), where the following two additional variables are typically defined

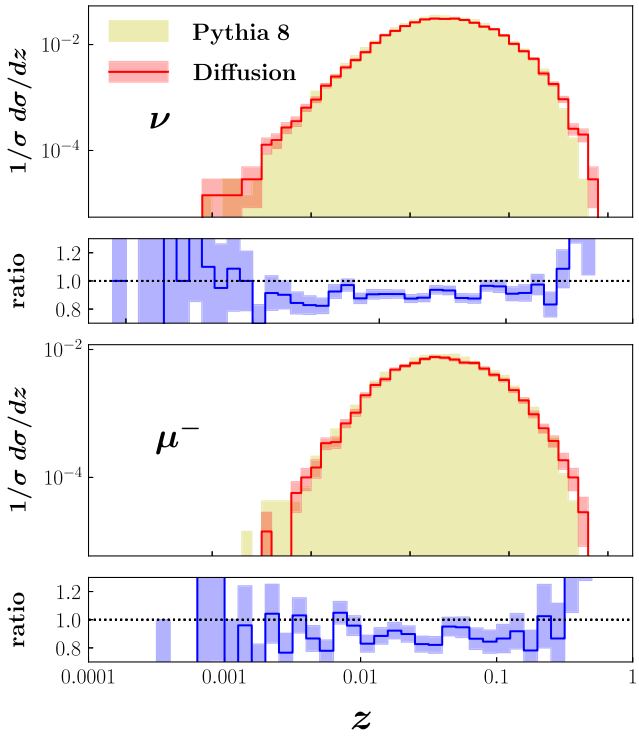

Fig. 4. Distributions of the rescaled momentum variable z for neutrinos (top) and muons (bottom) from the diffusion model and PYTHIA8.

$$z_h = \frac{P \cdot P_h}{P \cdot q}, \quad q_T = \frac{p_{Th}}{z_h}. \tag{7}$$

Here, P_h is the four-momentum of an observed final-state hadron, and p_{Th} is its transverse momentum in the Breit frame. In the target rest frame, z_h is the hadron energy over the photon virtuality. See Ref. [67] for frame-independent definitions of the relevant variables listed above.

In the upper left panel of Fig. 5, we show the results from the diffusion model for the DIS variables in Eq. (6) as a two-dimensional histogram. In the upper right panel, we show a comparison of the diffusion model results for the DIS variables relative to PYTHIA8. We observe good agreement over the entire kinematic range. Minor deviations are noticeable only near the kinematic endpoints. Similar to the results for the kinematic distributions for the leading electron in Fig. 5, the deviations near the endpoint are likely due to statistical effects. In the lower two panels of Fig. 5, we show the analogous results for the SIDIS variables given in Eq. (7) for pions. Again, we find good agreement indicating the suitability of our model for different applications at the future EIC.

Generative models of full collider events need to satisfy global constraints such as momentum conservation; see Eq. (3) above. In addition, discrete quantum numbers such as the total baryon and lepton numbers need to be conserved. For each electron-proton event, we expect to have $\sum_{i\in\text{event}} L_i = 1$ and $\sum_{i\in\text{event}} B_i = 1$, where L_i and B_i are the lepton and

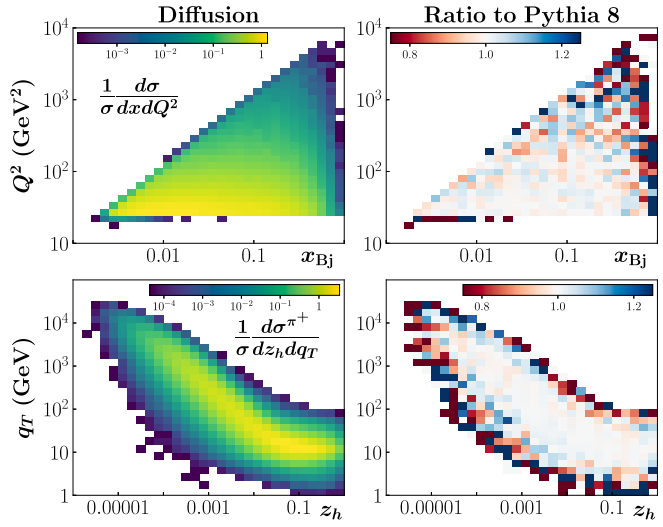

Fig. 5. Top row: Diffusion model results and comparison to PYTHIA8 for the DIS variables Bjorken x and photon virtuality Q^2. Bottom row: Analogous comparison for the energy z_h of pions and transverse momentum q_T in the Breit frame relevant for SIDIS.

baryon numbers of the i^{th} particle in the considered event, respectively. For electrons, muons, and neutrinos, we assign $L = +1$ and $L = -1$ for their antiparticles. All other particles are assigned $L = 0$. Analogous considerations apply to the baryon number. Due to experimental considerations, we combine neutrinos and anti-neutrinos as well as neutrons and anti-neutrons and exclude them when evaluating the event-wide lepton and baryon number. To assess the agreement between the diffusion model and PYTHIA8, we consider the ratio between the two for the momentum sum rule as well as the discrete quantum numbers within a 1-σ confidence level:

$$\text{Momentum: } 0.999(3),$$

$$\text{Baryon number: } 0.995(2),$$

$$\text{Lepton number: } 1.001(2).$$

Overall, we observe very good agreement with deviations limited to the subpercent level.

The extent to which these conservation laws need to be satisfied depends on the specific application of the diffusion model. Alternatively, additional constraints could be incorporated into the training process, where violations are penalized, or the conservation laws can be strictly enforced on an event-by-event basis. We leave a quantitative comparison of different approaches for future work.

3.2. *Learning event-wide constraints*

Table 1
Metrics quantifying the performance of the image- and point cloud-based diffusion models compared to PYTHIA8. Small values are preferred for each metric except for the coverage.

	Image-based diffusion model, Ref. [53]			Point cloud-based diffusion model (this work)		
	e^-	K^+	π^+	e^-	K^+	π^+
$W_1^P(\eta)$	63.167 ± 0.035	36.669 ± 0.029	57.887 ± 0.062	0.266 ± 0.009	0.041 ± 0.003	0.310 ± 0.016
$W_1^P(\phi)$	18.910 ± 0.054	18.736 ± 0.048	18.789 ± 0.030	0.015 ± 0.004	0.025 ± 0.009	0.158 ± 0.003
$W_1^P(p_T)$	5.917 ± 0.005	0.323 ± 0.002	0.820 ± 0.007	0.251 ± 0.004	0.129 ± 0.005	0.464 ± 0.007
Cov	0.011	0.017	0.010	0.546	0.518	0.473
MMD	1.266	2.160	1.945	0.166	0.382	0.595
KPD	$7 \times 10^7 \pm 1 \times 10^7$	20.576 ± 26.608	$4.6 \times 10^3 \pm 1.5 \times 10^3$	0.0023 ± 0.0003	0	0.0062 ± 0.0009

3.3. Image vs. Point cloud-based diffusion models

To better evaluate the improvements achieved in this work compared to the image-based diffusion model from Ref. [53] for electron-proton scattering events, we present the values for several quantitative metrics in Table 1. We evaluate the models using the Wasserstein distance for the particle transverse momentum, rapidity, and azimuthal angle along with coverage (Cov), Maximum Mean Discrepancy (MMD) using the energy mover's distance, and kernel physics distance (KPD). See Ref. [68] for more details. We only focus on the comparison for electrons, kaons, and pions, as the image-based diffusion model in Ref. [53] was limited to these three particle species. Lower values of the different metrics indicate better results except for the coverage, where higher values are preferred. While some metrics show a more significant improvement than others, the point cloud-based model presented here consistently outperforms the image-based diffusion model of Ref. [53]. This can be attributed to both its more advanced architecture and the loss of granularity in the image-based model due to pixelation.

4. Conclusions and outlook

In this work, we introduced a diffusion model to generate full events at the future Electron-Ion Collider. Expanding on previous work, we developed a point cloud-based model combining edge creation with transformer modules to generate all particle species in the event. We evaluated the model's performance using different metrics and kinematic distributions, observing significant improvements across all metrics compared to earlier results. The model approximately learned event-wide momentum conservation, as well as the conservation of discrete quantum numbers such as baryon and lepton numbers. We expect that a similar multi-step generative process employed here could be applied to generate full events in other collision systems. While estimating the statistical power of generative models is a difficult task, there are some works trying to address this question [69,70]. By adopting the foundation model OmniLearn, our work may indicate a transition toward adapting foundation models for downstream tasks in fundamental particle and nuclear physics. In future work, we will explore different applications of the diffusion model developed here in the context of collider phenomenology, including fast simulations, inference tasks, and anomaly detection.

Code availability

The code for this paper can be found at https://github.com/ViniciusMikuni/OmniLearn while the data generated to train the model is available at https://zenodo.org/records/14027110.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We would like to thank Kaori Fuyuto, Chris Lee, Emanuele Mereghetti and Benjamin Nachman for helpful discussions. JYA, FR, and NS were supported by the U.S. Department of Energy Office of Science, Contract No. DE-AC05-06OR23177, under which Jefferson Science Associates, LLC operates Jefferson Lab. JYA and FR were supported in part by the DOE, Office of Science, Office of Nuclear Physics, Early Career Program under contract No DE-SC0024358. NS and RW are supported by the DOE, Office of Science, Office of Nuclear Physics in the Early Career Program. VM and FT are supported by the U.S. Department of Energy (DOE Office of Science under contract DE-AC02-05CH11231. This research used resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility using NERSC award NERSC DDR-ERCAP0030239. This research was supported in part by the Quark-Gluon Tomography (QGT) Topical Collaboration, under contract no. DE-SC0023646. We thank the Institute for Nuclear Theory at the University of Washington for its kind hospitality and stimulating research environment. This research was supported in part by the INT's U.S. Department of Energy grant No. DE-FG02-00ER41132.

Data availability

Data will be made available on request.

References

[1] R. Abdul Khalek, et al., Science requirements and detector concepts for the electron-ion collider: EIC yellow report, arXiv:2103.05419 [physics.ins-det].
[2] D.P. Kingma, M. Welling, Auto-encoding variational Bayes, arXiv:1312.6114 [stat.ML], Dec. 2013.
[3] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Proceedings of NIPS'14, Cambridge, MA, USA, 2014, pp. 2672–2680, http://dl.acm.org/citation.cfm?id=2969033.2969125.
[4] J. Sohl-Dickstein, E.A. Weiss, N. Maheswaranathan, S. Ganguli, Deep unsupervised learning using nonequilibrium thermodynamics, CoRR, arXiv:1503.03585, 2015.
[5] J. Ho, A. Jain, P. Abbeel, Denoising diffusion probabilistic models, CoRR, arXiv:2006.11239, 2020.
[6] G. Kasieczka, T. Plehn, M. Russell, T. Schell, Deep-learning top taggers or the end of QCD?, J. High Energy Phys. 05 (2017) 006, arXiv:1701.08784 [hep-ph].
[7] T. Cai, J. Cheng, K. Craig, N. Craig, Which metric on the space of collider events?, Phys. Rev. D 105 (7) (2022) 076003, arXiv:2111.03670 [hep-ph].
[8] K. Datta, A. Larkoski, How much information is in a jet?, J. High Energy Phys. 06 (2017) 073, arXiv:1704.08249 [hep-ph].
[9] P.T. Komiske, E.M. Metodiev, J. Thaler, Energy flow networks: deep sets for particle jets, J. High Energy Phys. 01 (2019) 121, arXiv:1810.05165 [hep-ph].
[10] T. Heimel, G. Kasieczka, T. Plehn, J.M. Thompson, QCD or what?, SciPost Phys. 6 (3) (2019) 030, arXiv:1808.08979 [hep-ph].
[11] F.A. Dreyer, G. Soyez, A. Takacs, Quarks and gluons in the Lund plane, J. High Energy Phys. 08 (2022) 177, arXiv:2112.09140 [hep-ph].
[12] M. Bellagente, A. Butter, G. Kasieczka, T. Plehn, R. Winterhalder, How to GAN away detector effects, SciPost Phys. 8 (4) (2020) 070, arXiv:1912.00477 [hep-ph].
[13] A. Andreassen, P.T. Komiske, E.M. Metodiev, B. Nachman, J. Thaler, OmniFold: a method to simultaneously unfold all observables, Phys. Rev. Lett. 124 (18) (2020) 182001, arXiv:1911.09107 [hep-ph].
[14] Y. Alanazi, et al., Machine learning-based event generator for electron-proton scattering, Phys. Rev. D 106 (9) (2022) 096002, arXiv:2008.03151 [hep-ph].
[15] T. Alghamdi, et al., Toward a generative modeling analysis of CLAS exclusive 2 π photoproduction, Phys. Rev. D 108 (9) (2023) 094030, arXiv:2307.04450 [hep-ph].
[16] Y. Huang, D. Torbunov, B. Viren, H. Yu, J. Huang, M. Lin, Y. Ren, Unsupervised domain transfer for science: exploring deep learning methods for translation between LArTPC detector simulations with differing response models, arXiv:2304.12858 [hep-ex].
[17] K. Lee, J. Mulligan, M. Płoskoń, F. Ringer, F. Yuan, Machine learning-based jet and event classification at the electron-ion collider with applications to hadron structure and spin physics, J. High Energy Phys. 03 (2023) 085, arXiv:2210.06450 [hep-ph].
[18] A. Butter, T. Plehn, R. Winterhalder, How to GAN LHC events, SciPost Phys. 7 (6) (2019) 075, arXiv:1907.03764 [hep-ph].
[19] C. Gao, S. Höche, J. Isaacson, C. Krause, H. Schulz, Event generation with normalizing flows, Phys. Rev. D 101 (7) (2020) 076002, arXiv:2001.10028 [hep-ph].
[20] K. Danziger, T. Janßen, S. Schumann, F. Siegert, Accelerating Monte Carlo event generation – rejection sampling using neural network event-weight estimates, SciPost Phys. 12 (2022) 164, arXiv:2109.11964 [hep-ph].
[21] S. Badger, et al., Machine learning and LHC event generation, SciPost Phys. 14 (4) (2023) 079, arXiv:2203.07460 [hep-ph].
[22] V. Cirigliano, K. Fuyuto, C. Lee, E. Mereghetti, B. Yan, Charged lepton flavor violation at the EIC, J. High Energy Phys. 03 (2021) 256, arXiv:2102.06176 [hep-ph].
[23] B. Nachman, D. Shih, Anomaly detection with density estimation, Phys. Rev. D 101 (2020) 075042, arXiv:2001.04990 [hep-ph].
[24] O. Atkinson, A. Bhardwaj, C. Englert, P. Konar, V.S. Ngairangbam, M. Spannowsky, IRC-safe graph autoencoder for unsupervised anomaly detection, Front. Artif. Intell. 5 (2022) 943135, arXiv:2204.12231 [hep-ph].
[25] A. Scheinker, CDVAE: multimodal generative conditional diffusion guided by variational autoencoder latent embedding for virtual 6D phase space diagnostics, arXiv:2407.20218 [physics.acc-ph].
[26] A. Andreassen, B. Nachman, D. Shih, Simulation assisted likelihood-free anomaly detection, Phys. Rev. D 101 (9) (2020) 095004, arXiv:2001.05001 [hep-ph].
[27] T. Finke, M. Krämer, A. Morandini, A. Mück, I. Oleksiyuk, Autoencoders for unsupervised anomaly detection in high energy physics, J. High Energy Phys. 06 (2021) 161, arXiv:2104.09051 [hep-ph].

[28] K. Fraser, S. Homiller, R.K. Mishra, B. Ostdiek, M.D. Schwartz, Challenges for unsupervised anomaly detection in particle physics, J. High Energy Phys. 03 (2022) 066, arXiv:2110.06948 [hep-ph].

[29] J.Y. Araz, M. Spannowsky, Quantum-probabilistic Hamiltonian learning for generative modelling & anomaly detection, arXiv:2211.03803 [quant-ph].

[30] D. Sengupta, M. Leigh, J.A. Raine, S. Klein, T. Golling, Improving new physics searches with diffusion models for event observables and jet constituents, J. High Energy Phys. 04 (2024) 109, arXiv:2312.10130 [physics.data-an].

[31] A. Morandini, T. Ferber, F. Kahlhoefer, Reconstructing axion-like particles from beam dumps with simulation-based inference, arXiv:2308.01353 [hep-ph].

[32] J. Birk, A. Hallin, G. Kasieczka, OmniJet-α: the first cross-task foundation model for particle physics, Mach. Learn.: Sci. Technol. 5 (3) (2024) 035031, arXiv:2403.05618 [hep-ph].

[33] V. Mikuni, B. Nachman, OmniLearn: a method to simultaneously facilitate all jet physics tasks, arXiv:2404.16091 [hep-ph].

[34] L. de Oliveira, M. Paganini, B. Nachman, Learning particle physics by example: location-aware generative adversarial networks for physics synthesis, Comput. Softw. Big Sci. 1 (1) (2017) 4, arXiv:1701.05927 [stat.ML].

[35] M. Paganini, L. de Oliveira, B. Nachman, CaloGAN: simulating 3D high energy particle showers in multilayer electromagnetic calorimeters with generative adversarial networks, Phys. Rev. D 97 (1) (2018) 014021, arXiv:1712.10321 [hep-ex].

[36] Y. Alanazi, et al., Simulation of electron-proton scattering events by a feature-augmented and transformed generative adversarial network (FAT-GAN), arXiv:2001.11103 [hep-ph].

[37] R. Kansal, J. Duarte, H. Su, B. Orzari, T. Tomei, M. Pierini, M. Touranakou, J.-R. Vlimant, D. Gunopulos, Particle cloud generation with message passing generative adversarial networks, arXiv:2106.11535 [cs.LG].

[38] E. Buhmann, G. Kasieczka, J. Thaler, EPiC-GAN: equivariant point cloud generation for particle jets, arXiv:2301.08128 [hep-ph].

[39] M. Touranakou, N. Chernyavskaya, J. Duarte, D. Gunopulos, R. Kansal, B. Orzari, M. Pierini, T. Tomei, J.-R. Vlimant, Particle-based fast jet simulation at the LHC with variational autoencoders, Mach. Learn.: Sci. Technol. 3 (3) (2022) 035003, arXiv:2203.00520 [physics.comp-ph].

[40] B. Käch, D. Krücker, I. Melzer-Pellmann, M. Scham, S. Schnake, A. Verney-Provatas, JetFlow: generating jets with conditioned and mass constrained normalising flows, arXiv:2211.13630 [hep-ex].

[41] R. Verheyen, Event generation and density estimation with surjective normalizing flows, SciPost Phys. 13 (5) (2022) 047, arXiv:2205.01697 [hep-ph].

[42] Y. Song, S. Ermon, Generative modeling by estimating gradients of the data distribution, CoRR, arXiv:1907.05600, 2019.

[43] V. Mikuni, B. Nachman, Score-based generative models for calorimeter shower simulation, Phys. Rev. D 106 (9) (2022) 092009, arXiv:2206.11898 [hep-ph].

[44] V. Mikuni, B. Nachman, M. Pettee, Fast point cloud generation with diffusion models in high energy physics, arXiv:2304.01266 [hep-ph].

[45] M. Leigh, D. Sengupta, G. Quétant, J.A. Raine, K. Zoch, T. Golling, PC-JeDi: diffusion for particle cloud generation in high energy physics, arXiv:2303.05376 [hep-ph].

[46] A. Butter, N. Huetsch, S.P. Schweitzer, T. Plehn, P. Sorrenson, J. Spinner, Jet diffusion versus JetGPT – modern networks for the LHC, arXiv:2305.10475 [hep-ph].

[47] F.T. Acosta, V. Mikuni, B. Nachman, M. Arratia, K. Barish, B. Karki, R. Milton, P. Karande, A. Angerami, Comparison of point cloud and image-based models for calorimeter fast simulation, arXiv:2307.04780 [cs.LG].

[48] M. Leigh, D. Sengupta, J.A. Raine, G. Quétant, T. Golling, PC-droid: faster diffusion and improved quality for particle cloud generation, arXiv:2307.06836 [hep-ex].

[49] O. Amram, K. Pedro, Denoising diffusion models with geometry adaptation for high fidelity calorimeter simulation, arXiv:2308.03876 [physics.ins-det].

[50] E. Buhmann, C. Ewen, D.A. Faroughy, T. Golling, G. Kasieczka, M. Leigh, G. Quétant, J.A. Raine, D. Sengupta, D. Shih, EPiC-ly fast particle cloud generation with flow-matching and diffusion, arXiv:2310.00049 [hep-ph].

[51] Z. Imani, S. Aeron, T. Wongjirad, Score-based diffusion models for generating liquid argon time projection chamber images, arXiv:2307.13687 [hep-ex].

[52] V. Mikuni, B. Nachman, CaloScore v2: single-shot calorimeter shower simulation with diffusion models, arXiv:2308.03847 [hep-ph].

[53] P. Devlin, J.-W. Qiu, F. Ringer, N. Sato, Diffusion model approach to simulating electron-proton scattering events, Phys. Rev. D 110 (1) (2024) 016030, arXiv:2310.16308 [hep-ph].

[54] Y. Go, D. Torbunov, T. Rinn, Y. Huang, H. Yu, B. Viren, M. Lin, Y. Ren, J. Huang, Effectiveness of denoising diffusion probabilistic models for fast and high-fidelity whole-event simulation in high-energy heavy-ion experiments, arXiv:2406.01602 [physics.data-an].

[55] T. Sjöstrand, S. Ask, J.R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C.O. Rasmussen, P.Z. Skands, An introduction to Pythia 8.2, Comput. Phys. Commun. 191 (2015) 159–177, arXiv:1410.3012 [hep-ph].

[56] M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, R. Ng, Fourier features let networks learn high frequency functions in low dimensional domains, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, vol. 33, Curran Associates, Inc., 2020, pp. 7537–7547, https://proceedings.neurips.cc/paper_files/paper/2020/file/55053683268957697aa39fba6f231c68-Paper.pdf.

[57] D. Hendrycks, K. Gimpel, Gaussian error linear units (gelus), arXiv preprint, arXiv:1606.08415, 2016.

[58] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[59] V. Mikuni, F. Canelli, Point cloud transformers applied to collider physics, Mach. Learn.: Sci. Technol. 2 (3) (2021) 035027, arXiv:2102.05073 [physics.data-an].

[60] Y. Wang, Y. Sun, Z. Liu, S.E. Sarma, M.M. Bronstein, J.M. Solomon, Dynamic graph CNN for learning on point clouds, CoRR, arXiv:1801.07829, 2018.

[61] Perlmutter system, https://docs.nersc.gov/systems/perlmutter/system_details/. (Accessed 4 May 2022).

[62] A. Sergeev, M.D. Balso, Horovod: fast and easy distributed deep learning in TensorFlow, arXiv preprint, arXiv:1802.05799, 2018.

[63] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., Tensorflow: a system for large-scale machine learning, in: OSDI, vol. 16, 2016, pp. 265–283.

[64] F. Chollet, Keras, https://github.com/fchollet/keras, 2017.

[65] I. Loshchilov, F. Hutter, SGDR: stochastic gradient descent with restarts, CoRR, arXiv:1608.03983, 2016.

[66] X. Chen, C. Liang, D. Huang, E. Real, K. Wang, H. Pham, X. Dong, T. Luong, C.-J. Hsieh, Y. Lu, et al., Symbolic discovery of optimization algorithms, Adv. Neural Inf. Process. Syst. 36 (2024).

[67] A. Bacchetta, M. Diehl, K. Goeke, A. Metz, P.J. Mulders, M. Schlegel, Semi-inclusive deep inelastic scattering at small transverse momentum, J. High Energy Phys. 02 (2007) 093, arXiv:hep-ph/0611265.

[68] R. Kansal, A. Li, J. Duarte, N. Chernyavskaya, M. Pierini, B. Orzari, T. Tomei, Evaluating generative models in high energy physics, Phys. Rev. D 107 (7) (2023) 076017, arXiv:2211.10295 [hep-ex].

[69] A. Butter, S. Diefenbacher, G. Kasieczka, B. Nachman, T. Plehn, GANplifying event samples, SciPost Phys. 10 (6) (2021) 139, arXiv:2008.06545 [hep-ph].

[70] S. Bieringer, S. Diefenbacher, G. Kasieczka, M. Trabs, Calibrating Bayesian generative machine learning for bayesiamplification, Mach. Learn.: Sci. Technol. 5 (4) (2024) 045044, arXiv:2408.00838 [cs.LG].