Diffusion model approach to simulating electron-proton scattering events

Peter Devlin,[1,*] Jian-Wei Qiu,[1,†] Felix Ringer,[1,2,‡] and Nobuo Sato[1,§]
[1]*Thomas Jefferson National Accelerator Facility, Newport News, Virginia 23606, USA*
[2]*Department of Physics, Old Dominion University, Norfolk, Virginia 23529, USA*

(Received 8 November 2023; accepted 27 June 2024; published 31 July 2024)

Generative artificial intelligence is a fast-growing area of research offering various avenues for exploration in high-energy nuclear physics. In this work, we explore the use of generative models for simulating electron-proton collisions relevant to experiments like the Continuous Electron Beam Accelerator Facility and the future Electron-Ion Collider (EIC). These experiments play a critical role in advancing our understanding of nucleons and nuclei in terms of quark and gluon degrees of freedom. The use of generative models for simulating collider events faces several challenges such as the sparsity of the data, the presence of global or eventwide constraints, and steeply falling particle distributions. In this work, we focus on the implementation of diffusion models for the simulation of electron-proton scattering events at EIC energies. Our results demonstrate that diffusion models can reproduce relevant observables such as momentum distributions and correlations of particles, momentum sum rules, and the leading electron kinematics, all of which are of particular interest in electron-proton collisions. Although the sampling process is relatively slow compared to other machine-learning architectures, we find diffusion models can generate high-quality samples. We foresee various applications of our work including inference for nuclear structure, interpretable generative machine learning, and searches of physics beyond the Standard Model.

DOI: 10.1103/PhysRevD.110.016030

I. INTRODUCTION

High-energy particle and nuclear collider experiments along with theoretical progress in the past decades have allowed for an increasingly sophisticated understanding of the quark and gluon dynamics at subatomic scales. Electron-proton scattering experiments including the Hadron-Electron Ring Accelerator at Deutsches Elektronen-Synchrotron, the Continuous Electron Beam Accelerator Facility (CEBAF) at JLab, and the future Electron-Ion Collider (EIC) at Brookhaven National Laboratory [1] and LHeC [2] and FCC-eh at CERN [3] play a critical role in advancing our understanding of the structure of hadrons, probing cold nuclear matter effects, and searching for physics beyond the Standard Model. In particular, the measurement of the scattered leading electron provides a clean electromagnetic probe of the inner structure of hadrons and nuclei. The experimental data has been analyzed within the framework of QCD factorization to extract the three-dimensional structure of hadrons in terms of quantum correlation functions, such as parton distribution functions. In addition, collider studies related to the emergence of hadrons and the associated neutralization of color have remained at the forefront of collider experiments. See Fig. 1 for an illustration of the average distribution of particles in high-energy electron-proton collisions, which will be discussed in more detail below.

The rapid development of artificial intelligence and machine learning in recent years has led to a wide range of applications in particle and nuclear physics [5,6]. Examples include the simulation of lattice gauge configurations [7–11], the classification of jets [12–21], the simulation of collider events [22–26], the unfolding of detector effects [27–31], data analyses with machine-learning-improved Bayesian posterior sampling [32–37], regression tasks [38–43], and searches of physics beyond the Standard Model [44–49]. See Ref. [50] for a broad overview. Several of these applications rely on generative models that can learn the structure or latent space of a dataset and generate new samples. Various types of generative models have been developed including variational autoencoders [51], autoregressive models [52], generative adversarial networks (GANs) [53], flow-based models [54], and diffusion models [55,56]. The different types of generative models each have their own advantages and disadvantages. The choice of generative models for a

*Contact author: devlin@jlab.org
†Contact author: jqiu@jlab.org
‡Contact author: fmringer@jlab.org
§Contact author: nsato@jlab.org

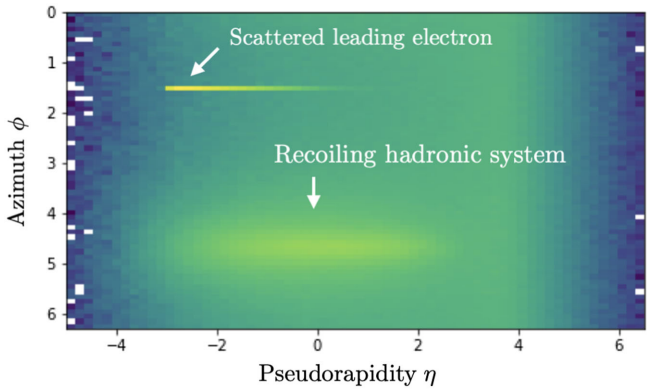

FIG. 1. Momentum distribution of the particles in the η-ϕ plane created in electron-proton scattering events in the laboratory frame at $\sqrt{s} = 105$ GeV. The events that have been generated with Pythia8 [4] are rotated such that the scattered electron is in the same azimuthal direction for all events as indicated in the figure.

particular application depends, for example, on the computational cost of training and sampling from the model, the quality of the generated samples, the scalability, and the stability of the training procedure, etc. In this work, we implement a diffusion model, which can generate samples from a data distribution by learning to reverse a stepwise noising or diffusion process; see Fig. 2. While sampling from diffusion models is generally relatively slow compared to other architectures, they have been shown to generate high-quality samples and allow for a scalable and stable training procedure. For example, in Ref. [57] it was found that diffusion models outperform GANs in image synthesis. The ability of diffusion models to generate high-quality samples is essential for the applications we foresee in the context of high-energy collider physics. In addition, Ref. [58] reported that diffusion models may cover a larger portion of the target distribution as GANs.

The development of generative models for simulating collider events or jets, collimated sprays of particles, was first initiated with GANs in Refs. [22,59,60]. Since then, different architectures such as normalizing flows [61] have been explored as well as different data representations such as point clouds instead of images have been considered [62–66]. In addition, efforts have been made to increase the interpretability of generative models [67,68]. One of the challenges for generative models is the sparsity of collider events, which is distinct from typical tasks encountered in computer vision. In addition, eventwide constraints such as momentum conservation and steeply falling momentum distributions of particles add to the complexity of the problem. Recently, diffusion and score-based generative models [69] have been developed for simulations of calorimeter showers [70–74] and jets using point clouds [62–65,75]. In this work, we will simulate full collider events with diffusion models focusing, in particular, on the unique characteristics of electron-proton (and, similarly, electron-nucleus) collisions. For example, the kinematics of the leading electron play a critical role, since it is used to determine both the photon virtuality Q^2 and the scaling variable Bjorken x of the event. In addition, particle spectra span up to 6 orders in magnitude, and, depending on the type of particle, they peak in different regions of phase space. We address this challenge by using a suitable preprocessing step before training the model on the data. In addition, we explore different loss functions and optimization procedures of diffusion models. Our findings demonstrate that diffusion models are able to generate high-quality samples indicating their potential for various future applications in high-energy nuclear and particle physics.

We foresee various applications of generative models for collider events. For example, generative models are closely related to the development of parton showers and Monte Carlo event generators. While the perturbative part is increasingly well understood from first principles in QCD [76–79], other components of Monte Carlo event generators can be simulated with generative models; see, for examples, the approach developed in Ref. [67]. Nuclear physics applications include the modification of the shower

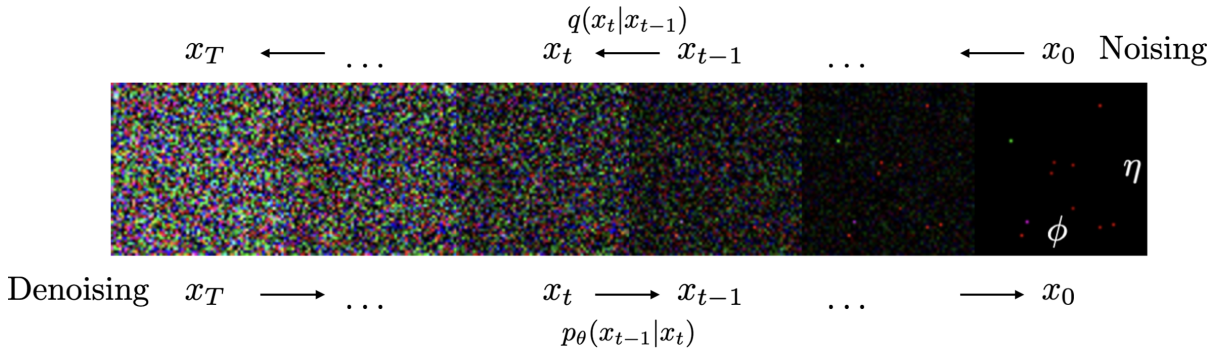

FIG. 2. Sequence of images illustrating the noising and denoising process of a diffusion model trained on Pythia8 simulations of electron-proton scattering events. Pixels colored in black are empty, and the three RGB color channels correspond to charged pions π^+, electrons e^-, and kaons K^+, respectively.

due to hot or cold nuclear matter. Moreover, anomaly detection techniques based on generative models have been developed for searches of physics beyond the Standard Model. The identification of anomalous signals requires an accurate modeling of the background distribution. Generative modeling also finds applications in hadron structure studies, which are a prominent subject of research within the Jefferson Lab 12 GeV program and the future EIC. The increasing sophistication required to extract parton-level information including transverse momentum distributions and generalized parton distributions necessitates the analysis of multidimensional phase space distributions from semi-inclusive and exclusive observables. In this context, generative modeling can be used as a generator of partonic structures for QCD global analysis, a phase space generator for particle reactions, or emulators for detector simulation [29,30]. It is also opening new avenues to integrate theory and experiment within a unified event-level analysis.

The remainder of this paper is organized as follows. In Sec. II, we provide a review of diffusion models. In Sec. III, we discuss the generation of electron-proton scattering events with Pythia8 [4], which is subsequently used as the training data, as well as the data representation. In Sec. IV, we provide details of our implementation and the training procedure. In Sec. V, we present numerical results comparing the diffusion model to Pythia8 using various metrics relevant to simulating collider events. Lastly, we conclude and present an outlook in Sec. VI.

II. DIFFUSION MODELS

Diffusion models are a class of generative machine-learning models that can learn the underlying distribution of a given dataset. The training procedure of diffusion models consists of two components—a noising and a denoising process; see Fig. 2. Starting with pixelated images of the training dataset, noise is incrementally added to the image until it is ultimately transformed into pure noise. Subsequently, the inverse denoising process can be learned by a suitably chosen machine-learning architecture. Because of the stepwise nature of the diffusion process, the results of the entire chain can be included in the loss function allowing for a scalable training process. After the training procedure is finished, we can generate new samples of the target dataset by passing noise through the trained machine-learning architecture.

Before starting the diffusion process, the pixelated images are treated as data vectors, which we label as x_0. The corresponding probability distribution of the data is $x_0 \sim q_0(x_0)$. Analogously, we denote the data vector at time step t of the diffusion process as x_t, with $t \in [0, T]$, and the corresponding probability distribution is $x_t \sim q_t(x_t)$. The stepwise forward diffusion or noising process is given by

$$q(x_1, \ldots, x_T|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}). \tag{1}$$

The probability distribution for a given time step of the noising process $x_{t-1} \to x_t$ is given by

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \mathbf{I}). \tag{2}$$

Here, $\mathcal{N}$ is a multivariate Gaussian distribution with a diagonal covariance matrix. The values of β_t are chosen according to a predefined variance schedule $\{\beta_t \in (0,1)\}_{t=1}^{T}$. We are, thus, adding a certain amount of Gaussian noise at each time step leading to a sequence of increasingly noisy samples $x_0, \ldots, x_T$, where the variance schedule and the time steps are chosen such that x_T is eventually an isotropic Gaussian distribution $q(x_T) = \mathcal{N}(x_T; 0, \mathbf{I})$. Note that Eqs. (1) and (2) describe a Markovian process, since the probability distribution at time step t depends on only the current sample at time $t-1$.

Next, we consider the reverse diffusion or denoising process. We need to train a suitable machine-learning model to approximate the probability distribution of the inverse process $q(x_{t-1}|x_t)$. The diffusion process is stochastic, which does not allow for the use of backpropagation techniques to obtain the gradient. Instead, the reparametrization trick is used to make the problem tractable and learn the parameters of a Gaussian distribution for which backpropagation can be used [51]. The denoising process $x_t \to x_{t-1}$ proceeds again in T time steps where the following Gaussians are sampled from

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)). \tag{3}$$

The mean μ_θ and covariance Σ_θ are learned by the model, where θ denotes the trainable parameters. Typically, a U-shaped convolutional neural network (U-Net) is used as a model to learn the mean and variance at each time step t [80].

The model parameters are obtained by minimizing a loss function during the training procedure. Different options have been explored in the literature. We start by considering the variational lower bound (VLB), which can be written as follows:

$$L_{\mathrm{VLB}} = L_0 + L_1 + \cdots + L_{T-1} + L_T, \tag{4}$$

$$L_0 = -\log p_\theta(x_0|x_1), \tag{5}$$

$$L_{t-1} = D_{\mathrm{KL}}(q(x_{t-1}|x_t, x_0) \| p_\theta(x_{t-1}|x_t)), \tag{6}$$

$$L_T = D_{\mathrm{KL}}(q(x_T|x_0) \| p(x_T)). \tag{7}$$

Here, L_{t-1} is used for all terms in Eq. (4) except for $t = 0, T$. Except for L_0, closed-form expressions can be found for all Kullback-Leibler (KL) divergences, since each term involves two Gaussian distributions. In Ref. [56], it was found empirically that a simplified objective function can improve the sample quality. Instead of using a neural network to predict μ_θ and Σ_θ, the network is used to predict x_0 and the noise ϵ_θ at each time step, which can be related to μ_θ; see Ref. [56] for more details. The simplified mean squared error objective function is given by

$$L_t^{\text{simple}} = E_{t,x_0,\epsilon}[\|\epsilon - \epsilon_\theta(x_t, t)\|^2], \tag{8}$$

where ϵ represents the noise of the forward diffusion process. Since this simplified objective function is sensitive only to μ_θ but not Σ_θ, a hybrid loss function was introduced in Ref. [58]:

$$L_{\text{hybrid}} = L_{\text{simple}} + \lambda L_{\text{VLB}}. \tag{9}$$

Here, λ is a hyperparameter that determines the relative importance of the two objective functions. Typically, λ is chosen to be relatively small such that μ_θ is primarily determined by L_{simple} and Σ_θ is related to L_{VLB}. In Sec. IV, we will provide more details of the setup used in this work.

After the training procedure, we can obtain new samples $q(x_0)$ from the target distribution by sampling $x_T \sim \mathcal{N}(0, \mathbf{I})$ and running the reverse process of the diffusion model. The Markovian noising and denoising processes described in Eqs. (2) and (3) are analogous to the diffusion process in nonequilibrium thermodynamics, and in the continuous-time limit, a stochastic differential equation is obtained [81]. See also Ref. [82] for a more detailed introduction to diffusion models.

III. TRAINING DATASET AND DATA REPRESENTATION

We generate the training dataset, by simulating neutral-current electron-proton scattering events with Pythia8 [4] using $\sqrt{s} = 105$ GeV as a representative center-of-mass (c.m.) energy for the future EIC [1]. Since the photoproduction region (low photon virtuality Q^2) and deep inelastic scattering (DIS) region (high Q^2) are sensitive to different physics, we choose to impose a lower cut of $Q^2 > 25\ \text{GeV}^2$ to exclude the photoproduction events. Figure 1 shows a 2D histogram of the momentum distribution of particles in electron-proton scattering events in the laboratory frame. We highlight, in particular, the kinematic region of the scattered electron. Note that the electron pseudorapidity does not extend to very low values due to the Q^2 cut. In addition, we indicate the recoiling hadronic system or jet, which is produced in the opposite azimuthal direction as the scattered electron. For each generated particle i in the event, we record the transverse momentum p_{Ti} relative to the beam axis, the pseudorapidity $\eta_i = -\ln \tan\theta/2$ with the polar angle θ_i with respect to the direction of colliding electron, the azimuthal angle ϕ_i, and the particle identification (PID_i). We impose a cut on the pseudorapidity of $|\eta_i| < 10$, which captures the vast majority of particles produced by Pythia8; see Fig. 1. We do not apply a lower cut on the transverse momentum p_{Ti} of the particles. Instead of directly incorporating the transverse momentum of each particle as a feature of the training dataset, we choose to work with a rescaled variable. A natural choice for the rescaled momentum variable is

$$\tilde{z}_i = \frac{2M_{Ti}}{\sqrt{s}} \cosh\, y_i. \tag{10}$$

Here, y_i is the rapidity and $M_{Ti}^2 = p_{Ti}^2 + m_i^2$ is the transverse mass, where m_i is the hadron mass. This variable is of great interest for simulating full collider events, since it satisfies the following eventwide momentum sum rule:

$$\Sigma \equiv \sum_{i \in \text{event}} \tilde{z}_i = 2, \tag{11}$$

where we sum over all particles in a given event. This provides an important global constraint for simulating full collider events. However, in practice, this requires having access to all particle species and fully hermetic detectors. In general, the rapidity coverage of detectors is limited, and, in this work, we also limit the diffusion model setup to simulating only three particle species. As a result, Eq. (11) is not exactly satisfied. Instead, Eq. (11) provides an upper bound on the sum over all $\tilde{z}_i$ values in each event. In the limit of massless particles, $\tilde{z}_i$ reduces to

$$z_i = \frac{2p_{Ti}}{\sqrt{s}} \cosh\, \eta_i, \tag{12}$$

where η is the pseudorapidity, as introduced above. This variable is frequently used in the perturbative QCD literature; see, for example, Refs. [79,83]. Therefore, we choose z_i as the rescaled momentum variable for this work. Equation (11) also provides an upper bound for the sum over all z_i in each event. In summary, we record the variables $(z_i, \eta_i, \phi_i, \text{PID}_i)$ for each particle in the event. When training the diffusion model, as described below, we limit ourselves to only three "color" channels for which we choose: pions π^+, kaons K^+, and electrons e^- from the full Pythia8-generated event. In particular, the (leading) scattered electron in the event plays an important role in electron-proton scattering, and we will study its kinematic distributions in detail below. Since we are primarily interested in a proof-of-concept study, we limit ourselves to modeling only a few representative particle species. In particular, the leading electron kinematics differ significantly from the rest of the hadronic system.

A convenient way to digitize collider events is to represent them as images, which is well suited for machine-learning applications. We partition the cylindrical detector around the scattering vertex into a grid of uniformly sized rectangular pixels located at regular intervals in both rapidity and azimuth. We choose the pixel intensity to be the rescaled particle momentum z_i, and the rapidity η_i and azimuthal angle ϕ_i index the location of each pixel on the cylinder. The particle type (PID_i) is stored as indices of the different image color channels similar to RGB color channels. Whenever multiple particles of the same type are in the same pixel, their z_i values are added.

In an actual experiment, the natural choice for the pixel intensity is p_{Ti}, since z_i is a derived quantity given in terms of the measured value of p_{Ti} and η_i. Therefore, combining multiple particles in a given pixel will occur at the level of p_{Ti} instead of z_i. However, as we will discuss in Sec. V below, the reconstruction of physical observables in inclusive DIS scattering experiments is strongly affected by distortions induced by the η_i pixelation. In part, this is due to the large pseudorapidity interval chosen for this work. In principle, this can be mitigated by increasing the number of pixels for the rapidity. Because of limited computing resources, we were not able to increase the number of pixels further in this work. In future work, this can be addressed by increasing the number of pixels or by changing the data representation. Here, we opt for using the z_i as the pixel intensities, which reduces pixelation effects.

We quantify the discretization effect by considering as an example the inclusive momentum distribution of electrons e^- in Pythia8-generated electron-proton scattering events, which is shown in Fig. 3. We observe a large-p_T peak due to the leading scattered electron and a continuous spectrum at intermediate to small-p_T values due to electrons generated during the shower. We compare the actual distribution with its discretized counterparts using three different choices for the number of pixels for the image: 16×16, 32×32, and 64×64. We observe that the actual distribution is increasingly well reproduced as the number of pixels is increased. Throughout this work, we use 64×64 pixels as our default choice. With larger computing resources, this can be increased until the experimental resolution of the detector is reached.

As mentioned in the introduction, different than typical tasks in computer vision, images of collider events are very sparse, especially at the relatively low energies of the CEBAF experiment at JLab and the future EIC. For a c.m. energy of $\sqrt{s} = 105$ GeV, we find that the average level of sparsity or the percentage of empty pixels is $99.95 \pm 0.02\%$ (including all particle species) for 64×64 images, which can be challenging for generative models. We address this problem by choosing a suitable data representation as discussed in the following. Besides the sparsity of the data, a significant challenge is the steeply

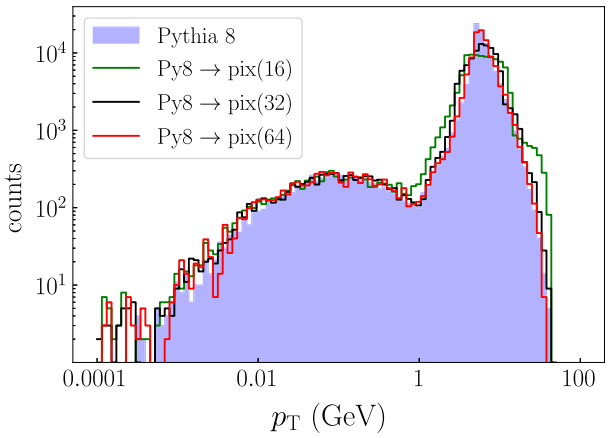

FIG. 3. Momentum distribution of electrons e^- in electron-proton scattering events for three different levels of pixelation compared to the original Pythia8 result.

falling distributions of particles. The inclusive momentum z distributions peak close to the end points $z \to 0$ for π^+, K^+, since soft hadrons have a large production cross section in QCD. Instead, for electrons e^- the distribution peaks in the region $z \to 1$; see Fig. 3. The large-z peak of the electron and positron momentum distribution is a unique feature of electron-proton scattering events. Instead, in proton-proton collisions, all distributions peak at small-z values. This feature appears due to the scattered leading electron, which plays a unique role in electron-proton collisions, since it is used to determine the virtuality of the exchanged photon Q^2 and Bjorken x. Therefore, the accurate modeling of its kinematics plays a critical role. In order to take into account the logarithmic behavior of the data near both end points $z \to 0, 1$ and to improve the training procedure, we rescale z as follows:

$$z \to S(z)E(z) + (1 - S(z))L(z). \tag{13}$$

Here, $E(z)$ is an exponential function, $L(z)$ is a logarithmic function, and $S(z)$ is a sigmoid. The three functions are defined as

$$E(z) = a_1 e^{c_1 z} + b_1, \tag{14}$$

$$L(z) = a_2 \ln(z + c_2) + b_2, \tag{15}$$

$$S(z) = \frac{1}{1 + e^{-\alpha(z-\beta)}}, \tag{16}$$

where we have introduced additional parameters that will be discussed in the following. First, we require that the parameters are chosen such that Eq. (13) is a bijective function allowing us to eventually recover to the original momentum distribution. Second, we choose the parameters such that the rescaling in Eq. (13) matches the peak structures of the z distributions near both end points.

Note that we apply the same z rescaling to all three channels. We choose the rescaling to be linear in the intermediate-z region, while near the upper (lower) end point the function approximates an exponential (logarithmic) function. This can be achieved by choosing two values $z_1 < z_2 \in]0, 1[$ with z_2 being the value where the exponential $E(z)$ smoothly becomes a linear function, i.e.,

$$E(z_2) = z_2, \qquad \frac{\mathrm{d}E(z_2)}{\mathrm{d}z} = 1. \tag{17}$$

Similarly, we require the logarithmic function $L(z)$ to become linear at z_1. These conditions along with the need to construct a bijective function fix or constrain several of the parameters in Eq. (13). We then choose the sigmoid $S(z)$ to smoothly interpolate between the exponential and logarithmic functions. The remaining parameters are chosen such that the rescaled z distribution is sufficiently smooth for the training of the diffusion model. We note that, without the preprocessing of the training data described, the results of the diffusion model can be off by several orders of magnitude near the kinematic end points.

The diffusion model takes as input values of the momentum fraction in the range $[-1, 1]$ (floating point numbers). Before pixelation and the rescaling in Eq. (13), the range of the particle momentum fractions is in the range of $z \sim [10^{-6}, 1]$. We choose a suitable range of values $z' \in [z'_{\min}, z'_{\max}]$ with $-1 < z'_{\min} < z'_{\max} < 1$ to which we map the original z values. This includes the rescaling in Eq. (13) as well as an additional linear transformation to match the targeted range. In practice, we find that $z'_{\min} = -0.76$ and $z'_{\max} = 0.86$ work well for the purposes of this work. These values are chosen to allow for an upper and lower gap from the end points at -1, 1. The lower gap allows us to train a diffusion model that can generate empty pixels. This is achieved by mapping empty pixel values in the training dataset, i.e., initially at $z = 0$, to the lower end of the allowed interval $z' = -1$. Since any finite z value is mapped to $z' > z'_{\min}$, there is a sufficiently large gap to the z' values associated with empty pixels. When generating new images by passing Gaussian noise through the denoising process as described above, the diffusion model does not need to generate pixels with exactly $z = 0$, but, instead, it is sufficient to generate a narrow peak around $z' \sim -1$. We can then apply a lower cut at $z'_{\min}$ and consider pixels with smaller z' values as empty. This allows us to generate sparse images of collider events. The upper gap associated with $z'_{\max}$ is introduced to avoid distortions of the distribution generated by the diffusion model near the upper end point. Any values produced by the diffusion model that are outside of the $z' \in [-1, 1]$ range are clipped, which would lead to artifacts near the end point without including the upper gap. Note that we do not enforce a hard upper cutoff at $z'_{\max}$. This requires the model to learn momentum conservation [see Eq. (11) above], which we quantify numerically below.

We note that one can likely choose $E(z) = 0$ and/or $S(z) = 0$ for simulating proton-proton or heavy-ion collisions, since in this case all particle spectra peak in the small-z region. We leave the exploration of this for future work. In addition, we note that an alternative approach to simulating sparse collider data is the use of point clouds as employed in Refs. [62–66] instead of the image-based data representation that we use here.

IV. IMPLEMENTATION AND TRAINING

In this section, we are going to present more details about the implementation of the diffusion model and the training procedure. As a starting point for our work, we use the implementation of the diffusion model presented in Ref. [58]. In the following, we discuss aspects of the training data, the evaluation of the loss function, and the parametrization of the inverse diffusion process.

(i) The size of the training dataset is 10^6 images of DIS events as described in Sec. III above. We choose a batch size of 8.

(ii) For the noising process $q(x_t|x_{t-1})$, we use the cosine variance schedule introduced in Ref. [58] with 500 diffusion steps. We find that the number of diffusion steps is sufficient for the 64×64 images used in this work. The cosine variance schedule adds noise relatively slowly and is well suited for the relatively low-resolution images considered here. For the denoising process $p_\theta(x_{t-1}|x_t)$, we use a diagonal covariance matrix $\Sigma_\theta = \sigma_t^2 \mathbf{I}$, where σ_t^2 are time-dependent trainable parameters.

(iii) We use the hybrid loss function L_{hybrid} given in Eq. (9) with $\lambda = 0.001$ following Ref. [58]. Since the gradient of L_{VLB} in Eq. (4) can be very noisy, we use importance sampling instead of uniform sampling of this part of the objective function as proposed in Ref. [58]. We also explored the use of the simplified loss function proposed in Ref. [56] and $\lambda = 0$, which corresponds to the L_{VLB}, which generally underperformed compared to the hybrid loss function for the purposes of this work.

(iv) To parametrize the denoising process $p_\theta(x_{t-1}, x_t)$, we use a three-layer U-Net [80] with circular or periodic convolutions. Note that this is relevant only for the azimuthal coordinate ϕ. We choose a kernel size of 3 with stride 1 and padding 1. Multihead attention layers [84] and down and up sampling blocks are included to obtain a U-shaped neural network. It takes $\mathcal{O}(\text{hours})$ to generate 10^6 samples using a single NVIDIA A100 GPU. Possible speed-ups can be achieved using, for example, the methods developed in Refs. [71,73,85].

(v) We use the AdamW optimizer [86] with a learning rate of 10^{-4}. The logarithm of the loss is shown in Fig. 4 as a function of the number of samples that the diffusion model is trained on. We observe a steep

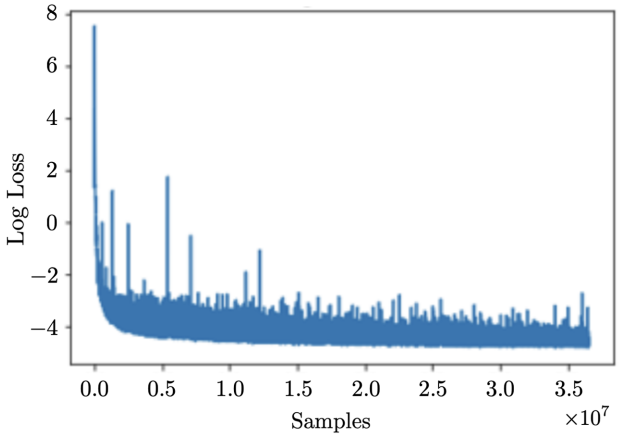

FIG. 4. The log loss of the diffusion model training procedure as a function of the number of samples that the model is trained on.

decrease of the loss at the beginning. Even though the curve flattens out later during the training process, we still observe a significant improvement of the sample quality. Despite the importance sampling mentioned above, the loss turns out to be relatively noisy.

V. NUMERICAL RESULTS AND BENCHMARKS

In this section, we will assess the quality of the trained diffusion model in simulating fully exclusive events (three particle species) in electron-proton collisions. We stress that our present work is limited in exploring the full extent of uncertainty quantification stemming from model uncertainties, limited training, and other factors. The results presented here should be viewed as an exploratory study, and we will focus on describing the qualitative overall agreement between the reconstructed synthetic phase space distributions and the training data. Dedicated studies of aleatoric and epistemic uncertainties are beyond the scope of our current work and will be addressed in the future.

We start with the inclusive momentum, rapidity, and azimuthal angle distributions for electrons e^-, pions π^+, and kaons K^+. The comparison between the diffusion model results and Pythia8 with and without pixelation is shown in Fig. 5. The systematic effects due to the pixelation are relatively small for the one-dimensional projections of the phase space. However, as we will find later on, they can become significant for other observables considered in this work. Analogous to Fig. 3, the electron z distribution peaks at $z \to 1$, whereas the pion and kaon distributions peak at small-z values; see the first column in Fig. 5. We find that our mapping in Eq. (13) allows us to train the diffusion model such that the z spectra match relatively well. This is particularly noteworthy given the fact that the distributions of the three particle species differ substantially. However, achieving an agreement within 1σ of the estimated statistical uncertainties was not attainable. The model appears to have difficulties in reproducing the distributions near the edges of the phase while performing better in regions where the distributions are relatively flat. That being said, without the use of the mapping in Eq. (13), we find that it is impossible to train the diffusion model to agree with the z spectra, with deviations as large as several orders of magnitude in several regions of phase space. We conclude that a suitable mapping of the z values is essential for training the diffusion model for electron-proton collisions. The rapidity distribution for electrons shows qualitatively different features compared to pions and kaons; see the middle column in Fig. 5. The difference is again due to the unique role of the leading electron in electron-proton scattering events. The steep drop of the rapidity distributions for electrons near $\eta \sim -3$ is due to the imposed cut on the photon virtuality Q^2; see also Fig. 1. Our model achieves a reasonable description of all three rapidity distributions with discrepancies that again tend to grow near the edges of phase space. We note that, if an additional transverse momentum cut for the electron is included, the rapidity distributions for pions and electrons would be more similar. Lastly, the right column in Fig. 5 shows the azimuthal distributions where the events have been rotated such that $\phi = 0$ corresponds to the direction opposite to the leading electron, which itself is not included in these histograms. Overall, we observe that the three bell-shaped curves of the azimuthal angular correlations are approximately reproduced by the diffusion model. We observe small differences in the kaon distributions in the tails of the distribution. This is likely due to the relatively low multiplicity of kaons. Although we did not achieve full agreement within the statistical uncertainties, being able to approximately reproduce the distributions with the help of the remapping of the z values encourages further explorations in this direction. Extending this work to include the estimation of model uncertainties is necessary to assess whether the observed disagreement stems from biases due to finite statistics or from a potential lack of model expressivity in our implementation.

Next, we consider particle multiplicity distributions $\langle N_i \rangle$. The comparison between Pythia8 with and without pixelation and the diffusion model results are shown in Fig. 6. The electron multiplicity peaks at $\langle N_e \rangle = 1$, which corresponds to the scattered leading electron and falls off steeply as the multiplicity increases. The case of $\langle N_e \rangle = 0$ corresponds to an unphysical distortion induced by the pixelation algorithm, since we are considering only neutral current DIS Pythia samples. The pion multiplicity distribution peaks at intermediate values $\langle N_{\pi^+} \rangle \sim 4$ and exhibits a long tail extending up to ~20 pions per event. The kaon multiplicity distribution also declines rapidly toward larger values, with many events having no kaons.

In contrast to the situation encountered for the momentum distributions, the multiplicity distributions are rather well reproduced and within 1σ of the statistical uncertainties

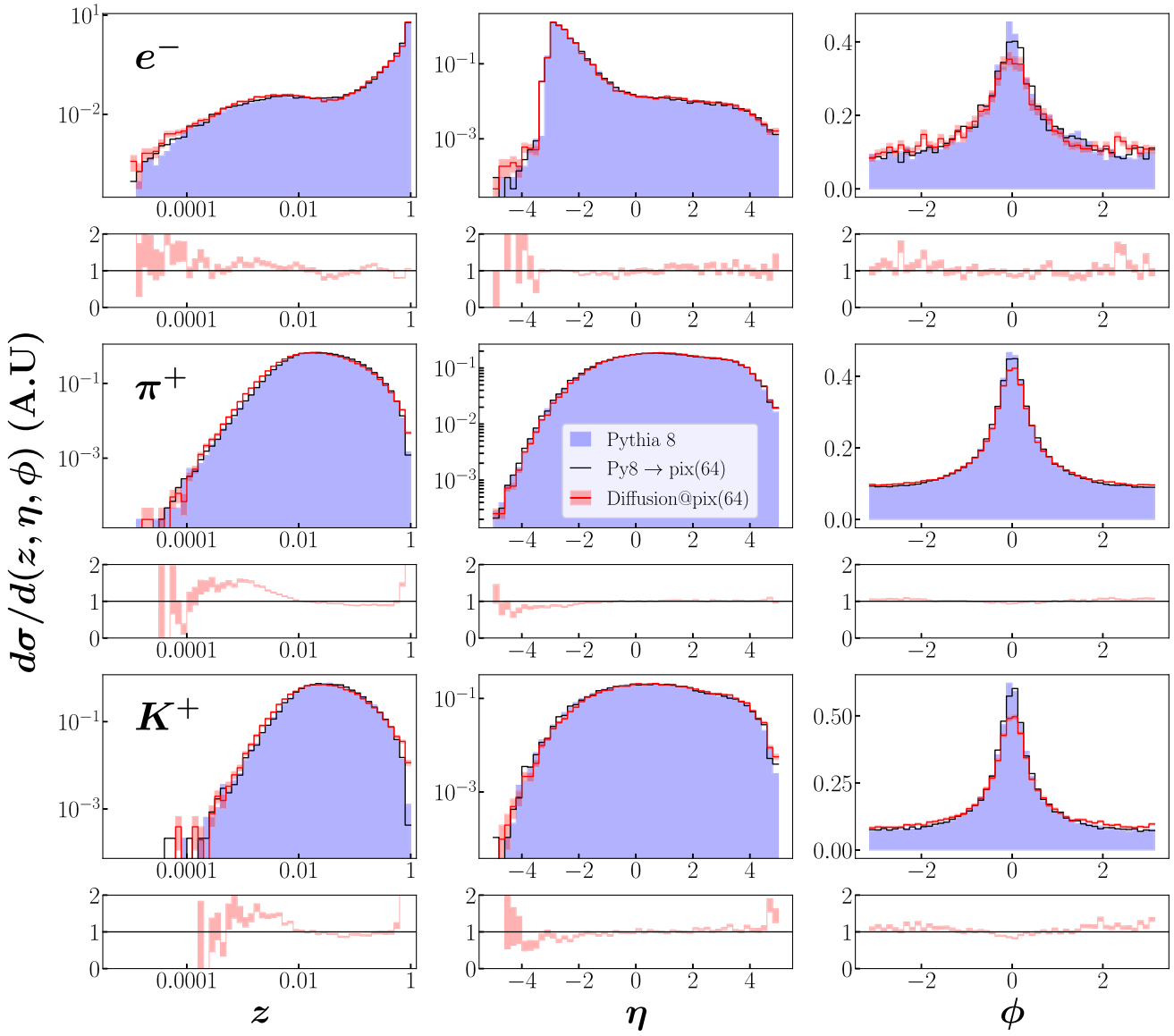

FIG. 5. Differential cross sections in arbitrary units (arb. units) of the rescaled momentum variable z, pseudorapidity η, and azimuthal angle ϕ (left to right). The results are shown separately for electrons e^-, charged pions π^+, and kaons K^+ (top to bottom). We show the diffusion model results (red) including statistical uncertainties and compare them to Pythia8 with (black) and without (purple) pixelation effects. Under each panel, we include cross sections ratios between the diffusion model and Pythia8 with pixelation.

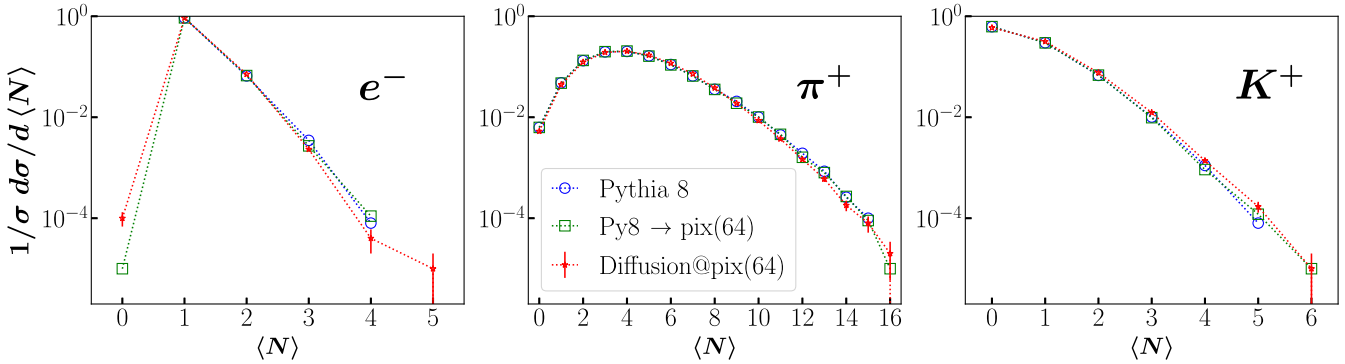

FIG. 6. Comparison of the (normalized) particle multiplicities $\langle N_i\rangle$ for electrons e^-, pions π^+, and kaons K^+ (from left to right). For the diffusion model results, we include statistical uncertainties.

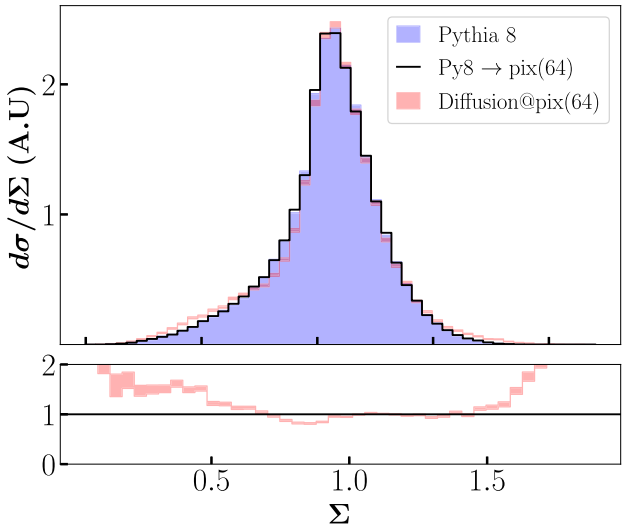

FIG. 7. Comparison of Pythia8 and diffusion model results for the sum over the rescaled momentum fractions in the entire event; see Eqs. (11) and (12).

across most of the available phase space for the three particles. Only in the electron case do we find that the diffusion model generates nonvanishing $\langle N_e \rangle = 5$, and it systematically deviates in the unphysical case of $\langle N_e \rangle = 0$. However, these events are very rare, occurring at a rate of 10^{-4} relative to the peak value.

Next, we consider the sum over the rescaled momentum fractions; see Eq. (11) above. This provides an important test of the global characteristics of the events generated by the diffusion model. The diffusion model result compared to Pythia8 is shown in Fig. 7. The distribution peaks near $\sum_i z_i \sim 1$ and falls off steeply toward the end points, where the upper limit results from momentum conservation; see Eq. (11) above. It is worth noting that while we simulate pions π^+ and kaons K^+, which correspond to some of the most frequently produced particles in the events, the distributions here are significantly shifted to the left compared to the upper boundary due to the omission of other particle species (e.g., π^-, π^0) in our current implementation. Our diffusion model is able to approximately reproduce this distribution. While the peak around $\Sigma \sim 1$ is well reproduced, the deviations grow toward the edges of phase space. We find that the model does not generate unphysical events with $\Sigma > 2$. We conclude that the model is able to approximately learn momentum conservation, i.e., without it being imposed directly as an additional constraint.

Next, we consider a set of multihadron correlations, which serve as an important benchmark for evaluating the ability of the diffusion model to capture correlated features beyond the one-dimensional projections shown earlier. The two-dimensional histograms of the momentum fractions of leading versus subleading pions (upper row) and kaons (lower row) are shown in Fig. 8. The diffusion model clearly learns the correlations between the leading and

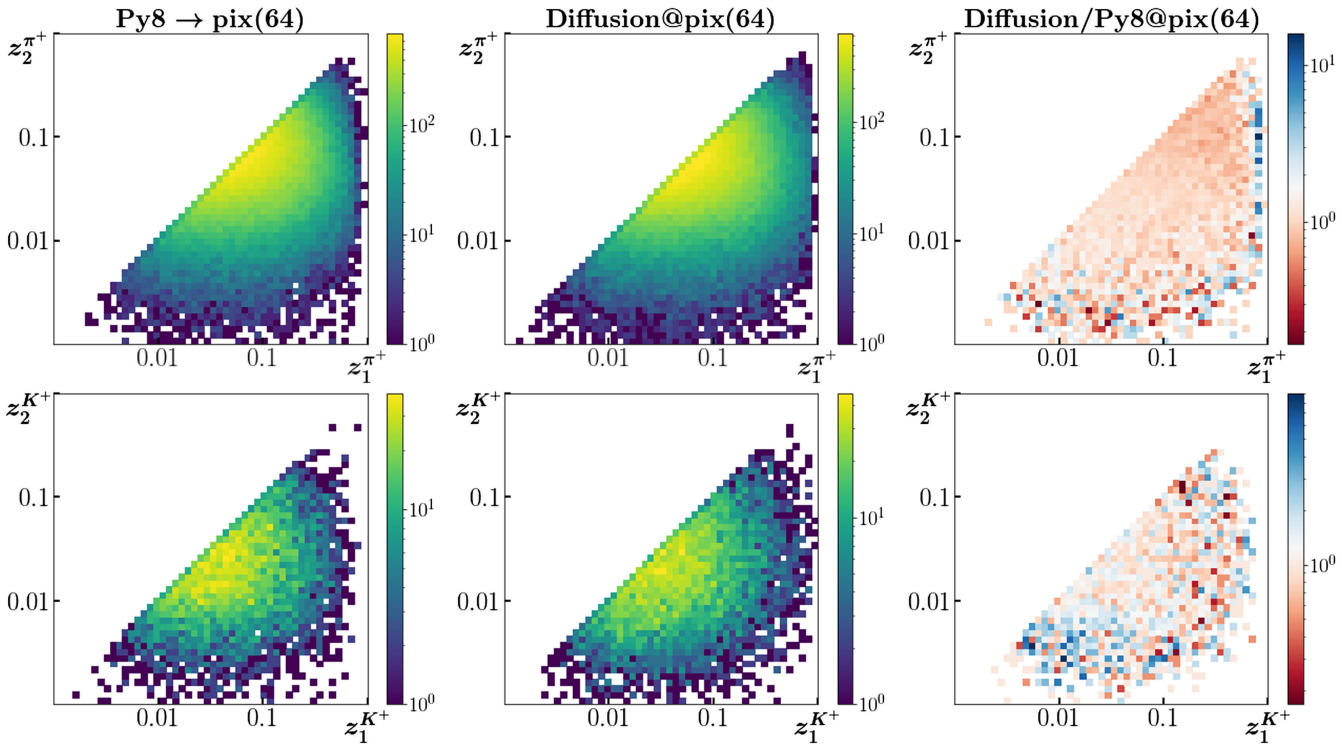

FIG. 8. Dihadron correlations comparing Pythia8 (left), the diffusion model (middle), and ratio to Pythia8 (right): leading and subleading pions π^+ (upper row) and kaons K^+ (lower row).

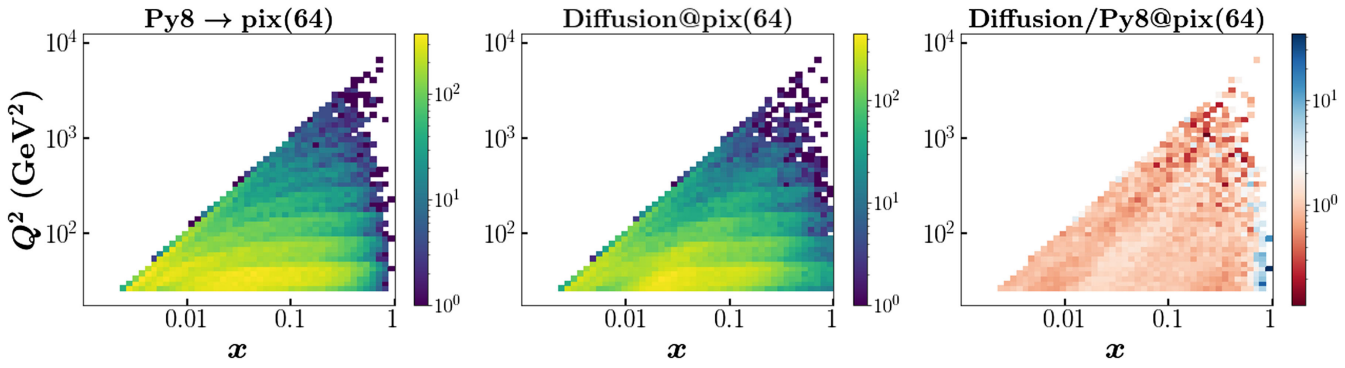

FIG. 9. Correlations of the DIS variables Bjorken x and the photon virtuality Q^2 comparing Pythia8 events (left) to results from the diffusion model (middle) and ratio to Pythia8 (right).

subleading particles that vary within 2 orders of magnitude in phase space regions with large statistics and loses its ability to reproduce the correlations toward the edges of phase space. As mentioned before, quantifying the model uncertainty is required to better assess the degree of agreement. However, we conclude that our results in Fig. 8 encourage future explorations in this direction.

We are now going to evaluate the diffusion model's performance in describing inclusive DIS reactions characterized by the virtuality of the exchanged photon $Q^2 = -q^2 = -(l - l')^2$ and the Bjorken scaling variable, defined as $x = Q^2/2p \cdot q \leq 1$. Here, l and l' denote the incoming and outgoing four-momenta of the scattered electron, respectively, and p is the momentum of the incoming proton. Assessing the diffusion model's ability to reproduce the distribution of these variables is a stringent test that allows us to gauge its ability to capture the correlations between the outgoing electron phase space and the initial state momenta. In Fig. 9, we display the reconstructed 2D density in the x, Q^2 space. Overall, we find good agreement between the diffusion model results and Pythia8 with pixelation effects with some noticeable differences only in the $x \to 1$ region. The presence of stripes in the 2D density plots can be attributed to pixelation effects, which can be mitigated by increasing the number of pixels used to represent the events, analogous to the momentum distributions shown in Fig. 3. We defer improvements to future work due to limitations of computational resources. Nevertheless, within the described limitations, we conclude that the diffusion model is able to approximately reproduce the correlations of the DIS phase space.

As discussed above, we use the variable z as the pixel intensity rather than the transverse momentum p_T. Here, we discuss why the pixelation of p_T induces larger systematic uncertainties compared to z when reconstructing the DIS kinematic variables x and Q^2. The photon virtuality can be expressed as $Q^2 = 2l_0 l'_T e^{-y_{l'}}$, where l_0 is the incident lepton energy and l'_T and $y_{l'}$ are the outgoing lepton transverse momentum and rapidity in the lab frame, respectively. Focusing on the uncertainty induced by the rapidity pixelation, we find that $\delta Q^2 \sim Q^2 \delta y_{l'}$ for fixed values of l'_T. This implies that uncertainties on Q^2 from the rapidity pixelation are amplified by a factor of Q^2, which is typically required to be large for phenomenological applications. Instead, when using the variable z_i, we find that the photon virtuality is given by $Q^2 = l_0 \sqrt{s} z (e^{-y_{l'}}/\cosh y_{l'})$, where $\sqrt{s}$ is the c.m. energy. In this case, the uncertainty induced by the rapidity pixelation for fixed values of z is $\delta Q^2 = Q^2 (e^{y_{l'}}/\cosh y_{l'}) \delta y_{l'}$; i.e., there is an additional factor $K = (e^{y_{l'}}/\cosh y_{l'})$ relative to the p_T pixelation case. In the c.m. frame of the electron-proton reaction, with the incoming beam of electrons moving along the z axis, the rapidity of the outgoing electrons is mostly negative, and typical values for K are around 0.01, which significantly suppresses systematic errors.

One of the salient features of the DIS process is the presence of scaling in the variable Q^2 which is interpreted as the evidence of pointlike constituents inside the proton. The so-called DIS neutral current (NC) reduced cross section is defined as

$$\sigma_{\text{red NC}}^{ep \to e'+X} = \frac{d\sigma_{\text{NC}}^{ep \to e'+X}}{dx dQ^2} \frac{Q^4 x}{2\pi\alpha^2 Y_+} = F_2(x, Q^2) - \frac{y^2}{Y_+} F_L(x, Q^2) + \frac{Y_-}{Y_+} x F_3(x, Q^2). \tag{18}$$

Here, we defined $Y_\pm = 1 \pm (1 - y)^2$, with the inelasticity $y = Q^2/(s - M^2)/x$. M is the proton mass, and α corresponds to the electromagnetic fine-structure constant. The structure functions $F_{2,L,3}$ are independent of Q^2 up to logarithmic corrections that can be predicted within perturbative QCD, provided that Q^2 is sufficiently large relative to any other hadronic scale. In this regime, the F_2 structure function is the dominant contribution to the reduced cross section; hence, it is approximately invariant under changes in Q^2. In Fig. 10, we compare the reconstructed cross section $\sigma_{\text{red NC}}^{ep \to e'+X}$ from diffusion-model-generated events

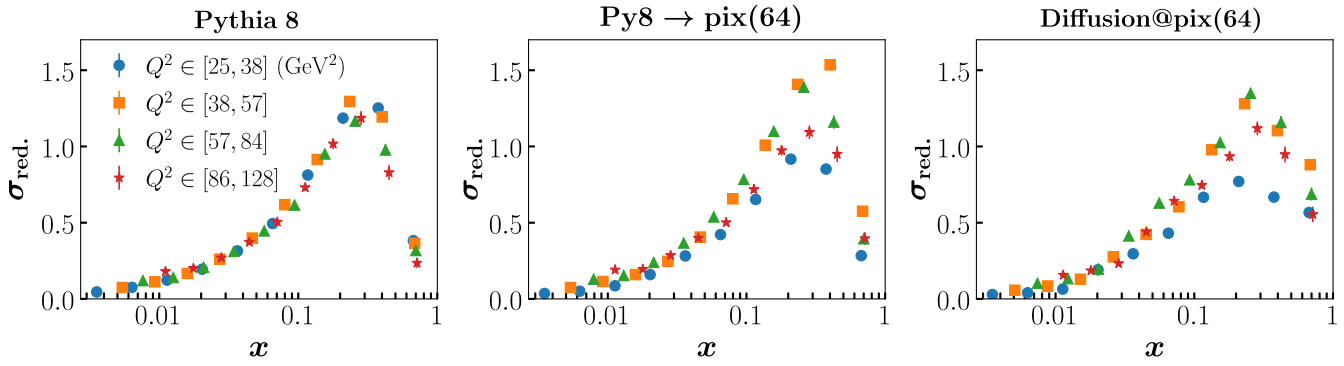

FIG. 10. Photon virtuality Q^2 scaling of the DIS reduced cross sections as a function of Bjorken x. The three panels show the Pythia8 results without and with pixelation and the diffusion model results (from left to right).

and the Pythia8 samples. The reconstructed reduced cross sections from Pythia8 show the expected scaling behavior. In contrast, after pixelation, distortions on the scaling behavior are induced mostly in the large-x valence region. We stress that this can be mitigated by enlarging the number of pixels. Taking into account these systematic scaling violations induced by the pixelation, the reconstructed reduced cross sections from the diffusion model are qualitatively in agreement with the pixelized version of Pythia8, albeit the diffusion model exhibits deviations at high values of x compared to Pythia8. These deviations may be associated with the epistemic uncertainties of the diffusion model. We highlight that achieving a faithful representation of DIS events serves as the starting point for studying cross sections such as semi-inclusive DIS that are differential in up to approximately ten variables, which will play a critical role at the future EIC. We note that further improvements of our results may be achieved by using diffusion models based on point clouds [64] and adapting methods developed in Ref. [87] in the context of emulating hard-scattering events.

VI. CONCLUSIONS

In this work, we presented simulations of electron-proton scattering events using diffusion models. Our results are relevant for simulations at CEBAF, the future Electron-Ion Collider, and LHeC/FCC-eh. The diffusion model is based on a noising schedule that sequentially turns the images from the training dataset into Gaussian noise. The stochastic reverse process is learned by a U-Net architecture based on convolutional layers with small filters. We trained the diffusion model on Pythia8 simulations of electron-proton scattering events at EIC energies and observed that it can generate high-quality sparse samples of collider events. We achieved good agreement near the kinematic endpoints by rescaling the particle momenta with a mixed exponential-logarithmic function, which accounts for the unique role that the scattered electron plays in electron-proton collisions. We employed an image-based representation of the training data and explored associated pixelation effects. As a first step, we limited ourselves in this work to electrons, pions, and kaons, which are represented by different "color" channels. Overall, we found good agreement for various observables and their correlations as well as eventwide constraints such as momentum conservation. We foresee various applications of our work in the context of generative modeling for collider physics including event-level analysis of hadron structure, data storage, studies of hadronization, exclusive processes like deeply virtual Compton scattering, and searches of new physics, which will be addressed in future work.

ACKNOWLEDGMENTS

We thank Jack Araz, Prerit Jaiswal, Vinicius Mikuni, James Mulligan, Mateusz Ploskon, and Fernando Torales-Acosta for helpful discussions. P. D., J.-W. Q., F. R., and N. S. are supported by the U.S. Department of Energy (DOE), Office of Science, Contract No. DE-AC05-06OR23177, under which Jefferson Science Associates, LLC operates Jefferson Lab. F. R. is supported in part by the DOE, Office of Science, Office of Nuclear Physics, Early Career Program under Contract No. DE-SC0024358. N. S. is supported by the DOE, Office of Science, Office of Nuclear Physics in the Early Career Program.

[1] R. Abdul Khalek *et al.*, Science requirements and detector concepts for the electron-ion collider: EIC Yellow Report, Nucl. Phys. **A1026**, 122447 (2022).

[2] P. Agostini *et al.* (LHeC, FCC-he Study Group), The Large Hadron–Electron Collider at the HL-LHC, J. Phys. G **48**, 110501 (2021).

[3] A. Abada *et al.* (FCC Collaboration), FCC physics opportunities: Future circular collider conceptual design report volume 1, Eur. Phys. J. C **79**, 474 (2019).

[4] T. Sjöstrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C. O. Rasmussen, and P. Z. Skands, An introduction to Pythia 8.2, Comput. Phys. Commun. **191**, 159 (2015).

[5] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, Machine learning and the physical sciences, Rev. Mod. Phys. **91**, 045002 (2019).

[6] A. Boehnlein *et al.*, Colloquium: Machine learning in nuclear physics, Rev. Mod. Phys. **94**, 031003 (2022).

[7] M. S. Albergo, G. Kanwar, and P. E. Shanahan, Flow-based generative models for Markov chain Monte Carlo in lattice field theory, Phys. Rev. D **100**, 034515 (2019).

[8] K. A. Nicoli, C. J. Anders, L. Funcke, T. Hartung, K. Jansen, P. Kessel, S. Nakajima, and P. Stornati, Estimation of thermodynamic observables in lattice field theories with deep generative models, Phys. Rev. Lett. **126**, 032001 (2021).

[9] S. Lawrence and Y. Yamauchi, Normalizing flows and the real-time sign problem, Phys. Rev. D **103**, 114509 (2021).

[10] R. Abbott *et al.*, Normalizing flows for lattice gauge theory in arbitrary space-time dimension, arXiv:2305.02402.

[11] M. Demirtas, J. Halverson, A. Maiti, M. D. Schwartz, and K. Stoner, Neural network field theories: Non-Gaussianity, actions, and locality, Mach. Learn. **5**, 015002 (2024).

[12] L. de Oliveira, M. Kagan, L. Mackey, B. Nachman, and A. Schwartzman, Jet-images—deep learning edition, J. High Energy Phys. 07 (2016) 069.

[13] G. Kasieczka, T. Plehn, M. Russell, and T. Schell, Deep-learning top taggers or the end of QCD?, J. High Energy Phys. 05 (2017) 006.

[14] K. Datta and A. Larkoski, How much information is in a jet?, J. High Energy Phys. 06 (2017) 073.

[15] P. T. Komiske, E. M. Metodiev, and J. Thaler, Energy flow networks: Deep sets for particle jets, J. High Energy Phys. 01 (2019) 121.

[16] T. Heimel, G. Kasieczka, T. Plehn, and J. M. Thompson, QCD or what?, SciPost Phys. **6**, 030 (2019).

[17] F. A. Dreyer, G. Soyez, and A. Takacs, Quarks and gluons in the Lund plane, J. High Energy Phys. 08 (2022) 177.

[18] T. Cai, J. Cheng, K. Craig, and N. Craig, Which metric on the space of collider events?, Phys. Rev. D **105**, 076003 (2022).

[19] K. Lee, J. Mulligan, M. Płoskoń, F. Ringer, and 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.

[20] J. Y. Araz and M. Spannowsky, Classical versus quantum: Comparing tensor-network-based quantum circuits on Large Hadron Collider data, Phys. Rev. A **106**, 062423 (2022).

[21] D. Athanasakos, A. J. Larkoski, J. Mulligan, M. Płoskoń, and F. Ringer, Is infrared-collinear safe information all you need for jet classification?, arXiv:2305.08979.

[22] L. de Oliveira, M. Paganini, and B. Nachman, Learning particle physics by example: Location-aware generative adversarial networks for physics synthesis, Comput. Software Big Sci. **1**, 4 (2017).

[23] A. Butter, T. Plehn, and R. Winterhalder, How to GAN LHC events, SciPost Phys. **7**, 075 (2019).

[24] C. Gao, S. Höche, J. Isaacson, C. Krause, and H. Schulz, Event generation with normalizing flows, Phys. Rev. D **101**, 076002 (2020).

[25] K. Danziger, T. Janßen, S. Schumann, and F. Siegert, Accelerating Monte Carlo event generation—rejection sampling using neural network event-weight estimates, SciPost Phys. **12**, 164 (2022).

[26] S. Badger *et al.*, Machine learning and LHC event generation, SciPost Phys. **14**, 079 (2023).

[27] M. Bellagente, A. Butter, G. Kasieczka, T. Plehn, and R. Winterhalder, How to GAN away detector effects, SciPost Phys. **8**, 070 (2020).

[28] A. Andreassen, P. T. Komiske, E. M. Metodiev, B. Nachman, and J. Thaler, OmniFold: A method to simultaneously unfold all observables, Phys. Rev. Lett. **124**, 182001 (2020).

[29] Y. Alanazi *et al.*, Machine learning-based event generator for electron-proton scattering, Phys. Rev. D **106**, 096002 (2022).

[30] T. Alghamdi *et al.*, Toward a generative modeling analysis of CLAS exclusive 2π photoproduction, Phys. Rev. D **108**, 094030 (2023).

[31] Y. Huang, D. Torbunov, B. Viren, H. Yu, J. Huang, M. Lin, and Y. Ren, Unsupervised domain transfer for science: Exploring deep learning methods for translation between LArTPC detector simulations with differing response models, arXiv:2304.12858.

[32] Y. He, L.-G. Pang, and X.-N. Wang, Bayesian extraction of jet energy loss distributions in heavy-ion collisions, Phys. Rev. Lett. **122**, 252302 (2019).

[33] M. Gabrié, G. M. Rotskoff, and E. Vanden-Eijnden, Efficient Bayesian sampling using normalizing flows to assist Markov chain Monte Carlo methods, arXiv:2107.08001.

[34] D. Everett *et al.* (JETSCAPE Collaboration), Multisystem Bayesian constraints on the transport coefficients of QCD matter, Phys. Rev. C **103**, 054904 (2021).

[35] M. Gabrié, G. M. Rotskoff, and E. Vanden-Eijnden, Adaptive Monte Carlo augmented with normalizing flows, Proc. Natl. Acad. Sci. U.S.A. **119**, e2109420119 (2022).

[36] N. T. Hunt-Smith, W. Melnitchouk, F. Ringer, N. Sato, A. W. Thomas, and M. J. White, Accelerating Markov chain Monte Carlo sampling with diffusion models, Comput. Phys. Commun. **296**, 109059 (2024).

[37] Y. Yamauchi, L. Buskirk, P. Giuliani, and K. Godbey, Normalizing flows for Bayesian posteriors: Reproducibility and deployment, arXiv:2310.04635.

[38] R. D. Ball *et al.* (NNPDF Collaboration), Parton distributions from high-precision collider data, Eur. Phys. J. C **77**, 663 (2017).

[39] J. Grigsby, B. Kriesten, J. Hoskins, S. Liuti, P. Alonzi, and M. Burkardt, Deep learning analysis of deeply virtual

exclusive photoproduction, Phys. Rev. D **104**, 016001 (2021).
[40] L. Del Debbio, T. Giani, J. Karpie, K. Orginos, A. Radyushkin, and S. Zafeiropoulos, Neural-network analysis of parton distribution functions from Ioffe-time pseudodistributions, J. High Energy Phys. 02 (2021) 138.
[41] M. Arratia, D. Britzger, O. Long, and B. Nachman, Reconstructing the kinematics of deep inelastic scattering with deep learning, Nucl. Instrum. Methods Phys. Res., Sect. A **1025**, 166164 (2022).
[42] M. Diefenthaler, A. Farhat, A. Verbytskyi, and Y. Xu, Deeply learning deep inelastic scattering kinematics, Eur. Phys. J. C **82**, 1064 (2022).
[43] C. Fanelli and J. Giroux, ELUQuant: Event-level uncertainty quantification in deep inelastic scattering, Mach. Learn. Sci. Tech. **5**, 015017 (2024).
[44] B. Nachman and D. Shih, Anomaly detection with density estimation, Phys. Rev. D **101**, 075042 (2020).
[45] A. Andreassen, B. Nachman, and D. Shih, Simulation assisted likelihood-free anomaly detection, Phys. Rev. D **101**, 095004 (2020).
[46] T. Finke, M. Krämer, A. Morandini, A. Mück, and I. Oleksiyuk, Autoencoders for unsupervised anomaly detection in high energy physics, J. High Energy Phys. 06 (2021) 161.
[47] K. Fraser, S. Homiller, R. K. Mishra, B. Ostdiek, and M. D. Schwartz, Challenges for unsupervised anomaly detection in particle physics, J. High Energy Phys. 03 (2022) 066.
[48] J. Y. Araz and M. Spannowsky, Quantum-probabilistic Hamiltonian learning for generative modelling & anomaly detection, Phys. Rev. A **108**, 062422 (2023).
[49] A. Morandini, T. Ferber, and F. Kahlhoefer, Reconstructing axion-like particles from beam dumps with simulation-based inference, Eur. Phys. J. C **84**, 200 (2024).
[50] M. Feickert and B. Nachman, A living review of machine learning for particle physics, arXiv:2102.02770.
[51] D. P. Kingma and M. Welling, Auto-encoding variational bayes, arXiv:1312.6114.
[52] A. Razavi, A. van den Oord, and O. Vinyals, Generating diverse high-fidelity images with VQ-VAE-2, arXiv:1906.00446.
[53] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, Generative adversarial nets, in *Proceedings of NIPS'14* (Curran Associates, Inc., Cambridge, MA, 2014), pp. 2672–2680.
[54] E. Tabak and C. Turner, A family of nonparametric density estimation algorithms, Commun. Pure Appl. Math. **66**, 145 (2013).
[55] J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, and S. Ganguli, Deep unsupervised learning using nonequilibrium thermodynamics, arXiv:1503.03585.
[56] J. Ho, A. Jain, and P. Abbeel, Denoising diffusion probabilistic models, arXiv:2006.11239.
[57] P. Dhariwal and A. Nichol, Diffusion models beat GANs on image synthesis, arXiv:2105.05233.
[58] A. Nichol and P. Dhariwal, Improved denoising diffusion probabilistic models, arXiv:2102.09672.
[59] M. Paganini, L. de Oliveira, and B. Nachman, CaloGAN: Simulating 3D high energy particle showers in multilayer electromagnetic calorimeters with generative adversarial networks, Phys. Rev. D **97**, 014021 (2018).
[60] Y. Alanazi *et al.*, Simulation of electron-proton scattering events by a Feature-Augmented and Transformed Generative Adversarial Network (FAT-GAN), in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)* (2021), arXiv:2001.11103.
[61] C. Krause and D. Shih, CaloFlow: Fast and accurate generation of calorimeter showers with normalizing flows, Phys. Rev. D **107**, 113003 (2023).
[62] M. Leigh, D. Sengupta, G. Quétant, J. A. Raine, K. Zoch, and T. Golling, PC-JeDi: Diffusion for particle cloud generation in high energy physics, SciPost Phys. **16**, 018 (2024).
[63] A. Butter, N. Huetsch, S. P. Schweitzer, T. Plehn, P. Sorrenson, and J. Spinner, Jet diffusion versus JetGPT—Modern networks for the LHC, arXiv:2305.10475.
[64] F. T. Acosta, V. Mikuni, B. Nachman, M. Arratia, K. Barish, B. Karki, R. Milton, P. Karande, and A. Angerami, Comparison of point cloud and image-based models for calorimeter fast simulation, J. Instrum. **19**, P05003 (2024).
[65] M. Leigh, D. Sengupta, J. A. Raine, G. Quétant, and T. Golling, PC-Droid: Faster diffusion and improved quality for particle cloud generation, Phys. Rev. D **109**, 012010 (2024).
[66] E. Buhmann, G. Kasieczka, and J. Thaler, EPiC-GAN: Equivariant point cloud generation for particle jets, SciPost Phys. **15**, 130 (2023).
[67] Y. S. Lai, D. Neill, M. Płoskoń, and F. Ringer, Explainable machine learning of the underlying physics of high-energy particle collisions, Phys. Lett. B **829**, 137055 (2022).
[68] S. Bieringer, A. Butter, T. Heimel, S. Höche, U. Köthe, T. Plehn, and S. T. Radev, Measuring QCD splittings with invertible networks, SciPost Phys. **10**, 126 (2021).
[69] Y. Song and S. Ermon, Generative modeling by estimating gradients of the data distribution, arXiv:1907.05600.
[70] V. Mikuni and B. Nachman, Score-based generative models for calorimeter shower simulation, Phys. Rev. D **106**, 092009 (2022).
[71] V. Mikuni, B. Nachman, and M. Pettee, Fast point cloud generation with diffusion models in high energy physics, Phys. Rev. D **108**, 036025 (2023).
[72] O. Amram and K. Pedro, Denoising diffusion models with geometry adaptation for high fidelity calorimeter simulation, Phys. Rev. D **108**, 072014 (2023).
[73] E. Buhmann, C. Ewen, D. A. Faroughy, T. Golling, G. Kasieczka, M. Leigh, G. Quétant, J. A. Raine, D. Sengupta, and D. Shih, EPiC-ly fast particle cloud generation with flow-matching and diffusion, arXiv:2310.00049.
[74] Z. Imani, S. Aeron, and T. Wongjirad, Score-based diffusion models for generating liquid argon time projection chamber images, Phys. Rev. D **109**, 072011 (2024).
[75] V. Mikuni and B. Nachman, CaloScore v2: Single-shot calorimeter shower simulation with diffusion models, J. Instrum. **19**, P02001 (2024).
[76] M. Dasgupta, F. A. Dreyer, K. Hamilton, P. F. Monni, G. P. Salam, and G. Soyez, Parton showers beyond leading logarithmic accuracy, Phys. Rev. Lett. **125**, 052002 (2020).

[77] J. R. Forshaw, J. Holguin, and S. Plätzer, Building a consistent parton shower, J. High Energy Phys. 09 (2020) 014.
[78] F. Herren, S. Höche, F. Krauss, D. Reichelt, and M. Schoenherr, A new approach to color-coherent parton evolution, J. High Energy Phys. 10 (2023) 091.
[79] D. Neill, F. Ringer, and N. Sato, Leading jets and energy loss, J. High Energy Phys. 07 (2021) 041.
[80] O. Ronneberger, P. Fischer, and T. Brox, U-net: Convolutional networks for biomedical image segmentation, arXiv:1505.04597.
[81] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, Score-based generative modeling through stochastic differential equations, in *International Conference on Learning Representations* (2021), https://openreview.net/forum?id=PxTIG12RRHS.
[82] L. Weng, What are diffusion models?, lilianweng.github.io (2021), https://lilianweng.github.io/posts/2021-07-11-diffusion-models/.
[83] T. Kaufmann, A. Mukherjee, and W. Vogelsang, Hadron fragmentation inside jets in hadronic collisions, Phys. Rev. D **92**, 054015 (2015).
[84] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, Attention is all you need, arXiv:1706.03762.
[85] Z. Wu, P. Zhou, K. Kawaguchi, and H. Zhang, Fast diffusion model, arXiv:2306.06991.
[86] I. Loshchilov and F. Hutter, Fixing weight decay regularization in Adam, arXiv:1711.05101.
[87] B. Nachman and R. Winterhalder, ELSA—Enhanced latent spaces for improved collider simulations, Eur. Phys. J. C **83**, 843 (2023).