

Deep learning for jet modification in the presence of the quark gluon plasma background

Ran Li ¹, Yi-Lun Du ^{2,*} and Shanshan Cao ^{1,†}

¹*Institute of Frontier and Interdisciplinary Science, Shandong University, Qingdao 266237, China*

²*Shandong Institute of Advanced Technology, Jinan 250100, China*



(Received 6 September 2025; accepted 29 January 2026; published 20 February 2026)

Jet interactions with the color-deconfined quantum chromodynamics medium in relativistic heavy-ion collisions are conventionally assessed by measuring the modification of the distributions of jet observables with respect to their baselines in proton-proton collisions. Deep learning methods enable per-jet evaluation of these modifications, enhancing the use of jets as precision probes of the nuclear medium. In this work, we predict the jet-by-jet fractional energy loss χ for jets evolving through a quark-gluon plasma (QGP) medium using a linear Boltzmann transport model. To approximate realistic experimental conditions, we embed medium-modified jets in a thermal background and apply constituent subtraction for background removal. Two network architectures are studied: convolutional neural networks (CNNs) using jet images, and dynamic graph convolutional neural networks (DGCNNs) using particle clouds. We find that CNNs achieve accurate predictions for background-free jets but degrade in the presence of the QGP background and remain below the background-free baseline even after background subtraction. In contrast, DGCNNs applied to background-subtracted particle clouds maintain high accuracy across the entire χ range, demonstrating the advantage of point-cloud-based graph neural networks that exploit full jet structure under realistic conditions.

DOI: [10.1103/469v-zbdd](https://doi.org/10.1103/469v-zbdd)

I. INTRODUCTION

Quark-gluon plasma (QGP) is a novel state of matter formed at extremely high temperature and/or density [1–4]. In this state, partons are no longer confined within hadrons and can move freely over distances significantly larger than typical hadronic sizes. This state can be created in relativistic heavy-ion collisions, providing a controlled environment for studying strongly interacting matter under extreme conditions. Hard processes occurring in the early stage of relativistic heavy-ion collisions produce high-transverse-momentum partons, which undergo showering and fragmentation to form nearly collimated streams of final-state particles, known as jets [5–7]. When these energetic partons traverse the QGP medium, they interact with the medium’s constituents, leading to energy loss or even complete absorption into the medium. This phenomenon is referred to as jet quenching, which has been extensively studied experimentally to probe the properties of the QGP. For instance, the observed suppression of jet spectra [8–13] in nucleus-nucleus collisions reflects energy loss induced by quenching effects. Furthermore, comprehensive investigations on modifications of jet substructures

[14–26] in nuclear collisions provide critical insight into the influence of quenching on the internal dynamics of jets.

Despite rapid progress, jet quenching studies continue to face challenges. The steeply falling jet spectra with respect to transverse momentum (p_T) make jet modification measurements highly susceptible to selection bias [27–32]: when comparing jet samples from proton-proton and nucleus-nucleus collisions within the same p_T interval, jets that have lost less energy are more likely to be selected. To minimize the selection bias, an ideal approach is to estimate the energy loss on a jet-by-jet basis. However, the quenching process experienced by jets in the medium is highly complex: the path length traversed by each jet varies, and local medium properties such as temperature and flow velocity further influence the energy loss. Moreover, the large and fluctuating background inherent in heavy-ion environments poses additional difficulties for jet studies.

Among different approaches proposed to address the challenge of selection bias, machine learning (ML) offers a unique capability to characterize the energy loss process at the single-jet level. ML technologies have brought significant progress to jet physics. They have been successfully applied in various topics, including quark/gluon jet classification [33,34], quenched jet identification [35–39], jet production vertex localization [40], and background subtraction [41–45]. In parallel, diverse jet representations have been explored, including image-based [46–49], sequence-based [50,51], tree-based [52,53], and point-cloud representations [54–59].

In previous studies [28,29], per-jet energy loss was predicted using machine learning methods based on a hybrid strong/weak coupling model [60], without including effects of

*Contact author: yilun.du@iat.cn

†Contact author: shanshan.cao@sdu.edu.cn

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article’s title, journal citation, and DOI. Funded by SCOAP³.

the QGP background particles. A crucial step toward applying such methods to real experimental data, however, is to assess their robustness in realistic heavy-ion environments where jets are embedded in a large fluctuating background. In this work, we explicitly introduce thermal background particles on top of the simulated jets to emulate experimental conditions, and employ constituent subtraction to mitigate their impact. In addition, we employ the linear Boltzmann transport (LBT) model [61,62] to simulate jet evolution in the medium, thereby not only extending the predictive framework to more realistic scenarios but also testing its applicability across different jet quenching models. Using this setup, we demonstrate that ML techniques, specifically convolutional neural networks (CNNs) and dynamic graph convolutional neural networks (DGCNNs) [63], can effectively predict the jet-by-jet energy loss ratio, offering a promising path to mitigate the selection bias in jet quenching studies.

The remainder of this paper is structured as follows. In Sec. II, we describe the simulation frameworks: PYTHIA for proton-proton collisions, LBT for jet-medium interactions, and a thermal toy model for the QGP background, as well as a jet matching procedure used to define the energy loss ratio. Section III details the constituent subtraction method for background removal. Section IV introduces the ML methodologies, where CNNs process jet images and DGCNNs operate on particle-cloud representations, dynamically learning the graph structure of jet constituents. In Sec. V, we present the prediction performance of these ML methods: CNNs achieve accurate predictions for clean jets, while under background conditions, DGCNNs consistently outperform CNNs across the entire energy loss ratio range. Finally, Sec. VI summarizes our findings and discusses future directions.

II. SIMULATION MODELS

We use PYTHIA [64,65] to generate jet partons produced in proton-proton ($p + p$) collisions. They also serve as the initial-state partons in nucleus-nucleus ($A + A$) collisions prior to scatterings through the QGP. The LBT model [61,62] is employed to simulate jet interactions with the QGP. In order to study impacts of the QGP background on jet reconstruction, we use a toy thermal model [66] to mimic thermal (soft) particle production from the QGP. Below, we briefly summarize the key features of these models and their parameter setups in the present study.

PYTHIA is a well-established event generator of high-energy particle collisions. We use it to simulate the production of energetic (hard) partons in nucleon-nucleon collisions, and the subsequent splittings of these partons, known as parton showers, at their high-virtuality stage. By default, splittings of a parton cease when it reaches a scale of 0.5 GeV. In this work, we generate 140k $p + p$ collision events at a center-of-mass energy per nucleon pair of $\sqrt{s_{NN}} = 5.02$ TeV using PYTHIA 8 with the Monash 2013 tune [67]. Triggers of momentum exchange of hard collisions are uniformly sampled within [300, 1000] GeV. We apply the FastJet 3.3.1 toolkit [68] on the final state partons from the PYTHIA shower to reconstruct them into jets for $p + p$ collisions. Meanwhile, these partons can

further interact with the QGP before they are reconstructed into quenched jets for $A + A$ collisions.

The LBT model simulates elastic and inelastic scatterings between low-virtuality jet partons and thermal partons inside the QGP medium based on the Boltzmann equation. The elastic scattering rates of jet partons are calculated by convoluting the $2 \rightarrow 2$ matrix elements at the leading order with thermal distributions of the medium partons at a given temperature, with all partonic scattering channels taken into account [69]; and the inelastic scattering rates are related to the number of medium-induced gluons per unit time, which are taken from the higher-twist energy loss calculations [70–73]. According to these scattering rates, the LBT model not only updates the kinematic information of jet partons as they plough through the medium, but also sample “recoil” partons scattered out of the medium by jets, leaving energy holes (named as “negative” partons) inside the medium. Recoil partons are treated equally to jet partons originating from the PYTHIA shower, and can undergo further scatterings with the medium. Contributions from negative partons to jet observables need to be subtracted from that from regular jet partons and recoil partons. These recoil and negative partons constitute jet-induced medium excitation (or medium response) in the LBT model, which is essential to guarantee the energy-momentum conservation of the jet + medium system and improve our quantitative understanding of jet observables in relativistic heavy-ion collisions. The medium information, including its local temperature and flow velocity, is provided by the $(3 + 1)$ -dimensional viscous hydrodynamic model CLVisc [74,75].

In this work, we simulate the environment created in Pb + Pb collisions at $\sqrt{s_{NN}} = 5.02$ TeV within the 0–10 % centrality bin. Positions of hard nucleon-nucleon collisions are sampled using the Monte Carlo Glauber model. Each jet parton produced by PYTHIA is assumed to stream freely until both its formation time, defined as the sum of the splitting times of its ancestors in the PYTHIA shower [76], and the thermalization time of the QGP (0.6 fm). Then, jet partons interact with the hydrodynamic medium until they exit the QGP, i.e., when the local temperature of the medium falls below 165 MeV. The strong coupling constant is set as $\alpha_s = 0.3$ in LBT. The output partons from LBT are then clustered into jets for $A + A$ collisions. Due to the lack of a reliable hadronization scheme for jet partons in $A + A$ collisions, we use partons from PYTHIA and LBT, instead of hadrons, to reconstruct jets for both $p + p$ and $A + A$ collisions. This will be improved in our upcoming efforts.

To study the performance of ML models in realistic experimental environments, we also employ a thermal toy model [66] to sample soft particles emitted by the QGP background. For each $A + A$ collision event, the thermal model is tuned to generate $\pi^\pm$ particles with momenta following a Boltzmann distribution that gives a total multiplicity of 1538 and an average transverse momentum of $\langle p_T \rangle = 0.5696$ GeV within the rapidity range of $|y| < 1$, corresponding to the soft hadron spectra created in 0–10 % Pb + Pb collisions at $\sqrt{s_{NN}} = 5.02$ TeV [77].

We will embed jet partons from LBT into these background particles and reconstruct jets with and without background subtraction. This will allow us to investigate how the QGP

background affects the ML models' accuracy in predicting the jet energy loss. For convenience, we refer to jets reconstructed without the QGP background as "LBT-only jets" or "background-free LBT jets", those reconstructed with the QGP background as "LBT jets within background", and those reconstructed after background subtraction as "LBT jets with background subtracted". For all samples, jets are reconstructed using the anti- k_T algorithm [78] with a radius parameter of $R = 0.4$. For analysis, we select jets with transverse momenta $p_T > 100$ GeV and pseudorapidity within $|\eta| < 1 - R$.

The primary objective of this work is to determine, on a jet-by-jet basis, the amount of energy lost by jets as they propagate through the QGP medium. The energy loss ratio is quantified by a variable [28,29]

$$\chi \equiv \frac{p_T^f}{p_T^i}, \quad (1)$$

where p_T^f denotes the transverse momentum of a jet after traversing the medium, simulated using the LBT model, and p_T^i corresponds to the transverse momentum of the *same* jet in the absence of medium effects, simulated using PYTHIA. For LBT jets embedded in the QGP background, the p_T^f used in the definition of χ includes contributions only from partons generated by LBT, excluding contributions from the background particles.

Establishing this definition requires a dedicated matching between a quenched LBT jet (j_{LBT}) and its unquenched PYTHIA counterpart (j_{PYTHIA}). For each j_{LBT} , whether embedded in the background, background-subtracted, or not, we search for its counterpart j_{PYTHIA} from its corresponding PYTHIA event. The distance between j_{LBT} and j_{PYTHIA} is required to satisfy $\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2} < 0.4$. If multiple PYTHIA jets meet this criterion, the hardest one is chosen as the match. Conversely, if no PYTHIA jet satisfies the condition, the j_{LBT} candidate is excluded from both training and analysis. Starting with the 140k PYTHIA events, we obtain a final dataset of about 170k medium-modified jets for this study.

III. CONSTITUENT SUBTRACTION METHOD

To mitigate possible degradation of the prediction performance of ML models on the jet energy loss ratio due to the contamination from the QGP background, a background subtraction procedure may be essential to recover or improve the performance. We employ the constituent subtraction (CS) method [66,79], which operates directly at the particle level. Instead of correcting jets after clustering, this method iteratively modifies or removes the momenta of individual particles before reconstruction. Such an approach is particularly well suited for heavy-ion environments, where the abundance of soft particles can distort clustering and bias jet observables [80].

The procedure begins with an estimate of the average background momentum density ρ . This is obtained as the median of $p_{T,i}/A_i$ over clusters reconstructed with the k_T algorithm [81,82], excluding the few hardest clusters (two in this

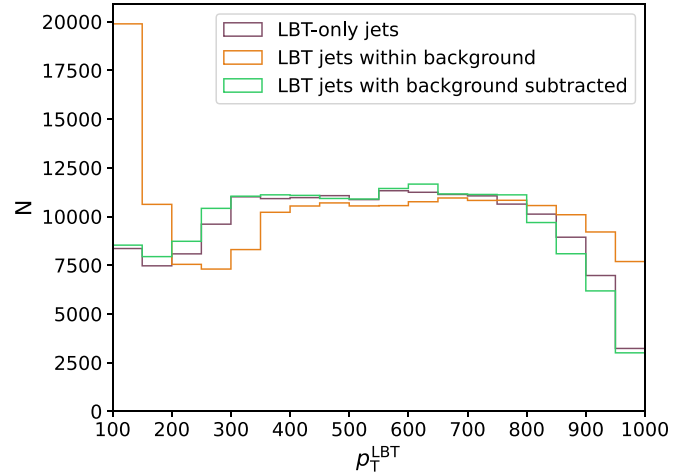


FIG. 1. Transverse momentum distributions of the LBT-only jets, the LBT jets within background, and the LBT jets with background subtracted.

work) to avoid bias from real jets. Here, $p_{T,i}$ is the transverse momentum of the i th cluster and A_i its geometric area, determined via the ghost-particle method. To mimic the underlying event, a dense set of ghost particles is distributed uniformly across the rapidity-azimuthal-angle (y - ϕ) plane. Each ghost is assigned a transverse momentum $p_T^g = \rho A^g$, where $A^g = 0.001$ denotes its effective area, so that the ensemble of ghosts reproduces the overall background contribution.

Pairs are then formed between each real particle i and ghost k , with their distance defined as

$$\Delta R_{i,k} = p_{T,i}^\alpha \sqrt{(y_i - y_k^g)^2 + (\phi_i - \phi_k^g)^2}, \quad (2)$$

where α is a tunable parameter, set to 1 in this work. This definition incorporates the p_T weighting, favoring the removal of softer particles. Distances are ordered from the smallest to the largest, and momentum subtraction is carried out pair by pair:

$$\begin{aligned} p_{T,i} &\rightarrow p_{T,i} - p_{T,k}^g, & p_{T,k}^g &\rightarrow 0, & \text{if } p_{T,i} \geq p_{T,k}^g; \\ p_{T,k}^g &\rightarrow p_{T,k}^g - p_{T,i}, & p_{T,i} &\rightarrow 0, & \text{otherwise.} \end{aligned}$$

The procedure continues until the distance between pairs exceeds ΔR^{max} , which sets the subtraction range. We use $\Delta R^{\text{max}} = \infty$ in our analysis. After subtraction, jets are reconstructed with the anti- k_T algorithm using only particles that retain nonzero momentum.

By locally correcting each particle according to the estimated background density, constituent subtraction succeeds in removing a large fraction of the soft background while preserving the essential jet kinematics and substructure. This makes it particularly effective for heavy-ion jet analyses.

To scrutinize the performance of the background subtraction, in Fig. 1, we compare the transverse momentum distributions between three jet categories: the LBT-only (background-free) jets, jets within background, and jets with background subtracted via constituent subtraction, demonstrating that the subtraction procedure effectively recovers the p_T distributions of the LBT-only jets. On the jet-by-jet

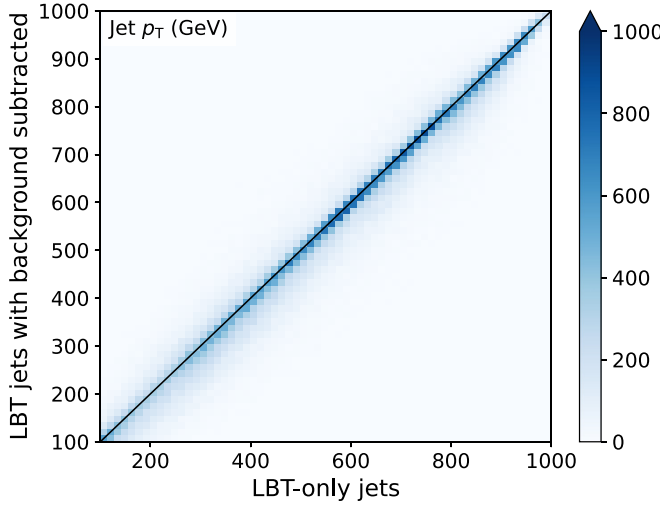


FIG. 2. Joint p_T distribution of the LBT-only jets and the background-subtracted LBT jets.

level, Fig. 2 presents the joint p_T distribution of the LBT-only jets and the background-subtracted LBT jets. The matching procedure introduced at the end of Sec. II is used to pair the background-subtracted LBT jet sample with the LBT-only jet sample. The close alignment between the two samples here further confirms the accuracy of the constituent subtraction method in recovering the jet p_T in the presence of the QGP background. In Fig. 3, we compare the particle number distributions of jets between the same three jet categories. One can clearly observe the clustering of background particles into jets when jet partons are embedded into the QGP background. This can be effectively corrected using the constituent subtraction method, though a slight oversubtraction in the particle multiplicity is hinted.

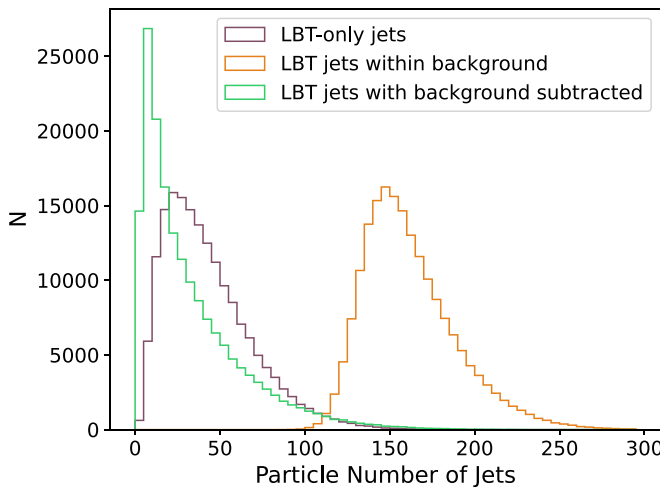


FIG. 3. Particle number distributions of jets, compared between the LBT-only jets, the LBT jets within background, and the LBT jets with background subtracted.

IV. MACHINE LEARNING METHODS

In this work we use two machine learning architectures in order to extract the fractional energy loss χ on a jet-by-jet basis. These are the convolutional neural network (CNN) and the dynamic graph convolutional neural network (DGCNN).

A. Convolutional neural network

We adopt the same CNN architecture as in Ref. [28] (see also Refs. [83,84] for technical details). The network consists of three convolutional layers (16, 16, and 32 filters of sizes 8×8 , 7×7 , and 6×6) followed by a fully connected layer with 128 neurons, and a single output neuron predicting the energy loss ratio χ . All convolutional and dense layers use “He normal” initialization [85], L2 regularization [86], batch normalization [87], PReLU activation [85], and Dropout (0.2 after convolutions, 0.5 after the dense layer). Average pooling (2×2) follows the first and third convolutional layers.

The network is trained as a regression model using the log-cosh loss and optimized with AdaMax [88] at a learning rate of 10^{-4} . Training uses batches of 1024 jet images, reshuffled each epoch, for up to 400 epochs with checkpointing at the best validation loss.

The input jet image $J(\eta, \phi)$ is a 33×33 matrix of the total p_T deposited per pixel, covering translated pseudorapidity $|\eta| \leq 0.4$, and translated azimuthal angle $|\phi| \leq 0.4$ for a fixed jet radius $R = 0.4$. Preprocessing follows Refs. [47,48]. We first translate the jet images so the hardest groomed subjet is at (0,0). Then we rotate it so the second-hardest subjet is at $-\pi/2$ (or align the first principal component if absent). Finally we apply a parity flip so the right half has larger total intensity.

B. Dynamic graph convolutional neural network

To improve prediction accuracy, we adopt the ParticleNet architecture [56], which is based on the dynamic graph convolutional neural network (DGCNN) [63]. This model operates directly on jet constituents represented as an unordered point cloud, stacking multiple EdgeConv modules and dynamically updating neighborhood graphs between layers using the k -nearest neighbors (KNN) algorithm [89,90].

In ParticleNet, the EdgeConv module processes two types of input information for each particle: its “coordinates” in a chosen feature space and its associated high-dimensional feature vector. For each particle, the k nearest neighbors are identified based on Euclidean distances in the coordinate space ($k = 16$ in this work). The features of each particle and its neighbors are combined to form edge features that encode both local geometry and particle-level information. These edge features are transformed by edge functions and subsequently aggregated to update the representation of each particle.

The architecture consists of three EdgeConv modules, each parametrized by a multilayer perceptron (MLP) with channel dimensions (64, 64, 64), (128, 128, 128), and (256, 256, 256), respectively. The first module computes distances in the $(\Delta\eta, \Delta\phi)$ plane relative to the jet axis, using the particle four-momentum as input features. The subsequent modules

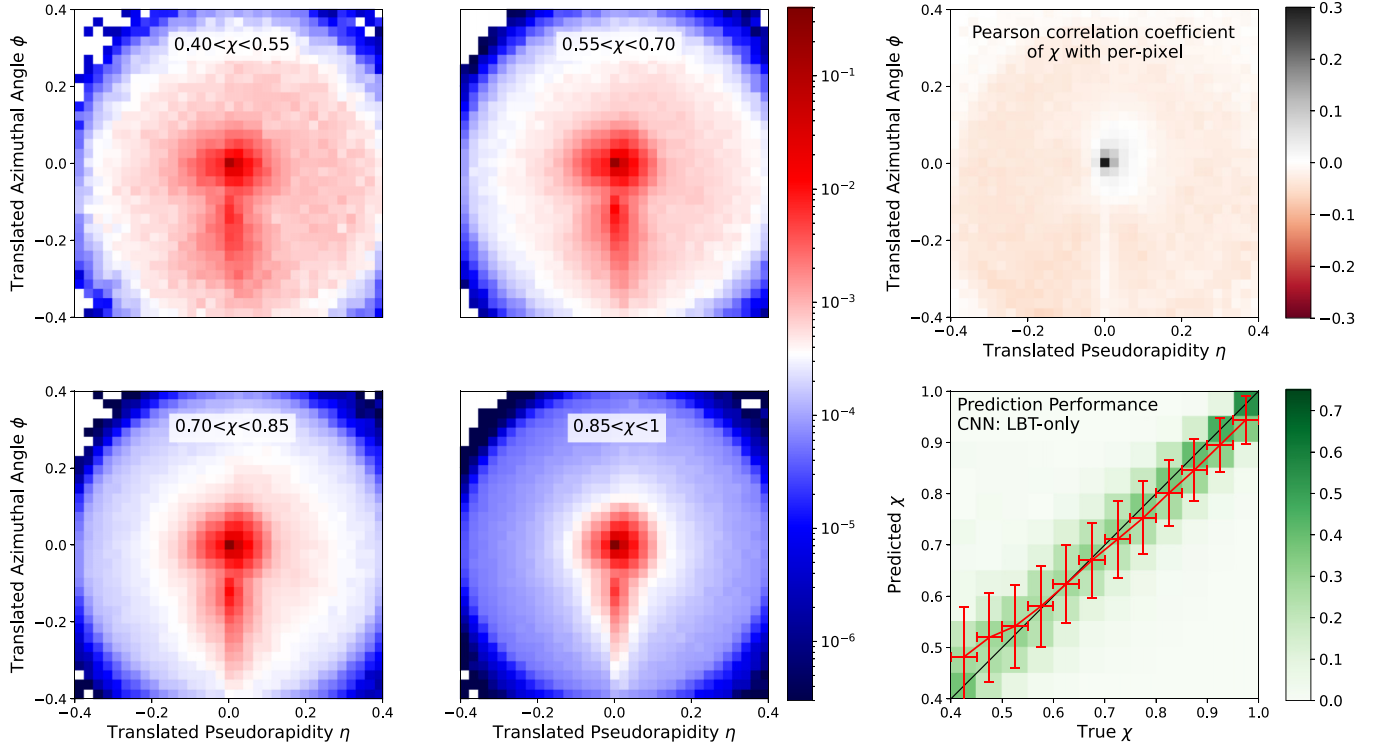


FIG. 4. The left four panels show the average images from self-normalized energy distributions inside the LBT jets reconstructed without introducing the QGP background particles, grouped into four energy loss ratio (χ) bins. The top right panel presents the Pearson correlation coefficient between χ and value at each pixel of unnormalized jet image. The bottom right panel presents the CNN prediction performance on χ obtained using unnormalized jet images.

take the learned features from the previous stage as both new coordinates and features. After the EdgeConv layers, a global average pooling operation aggregates the features into a representation of size 256, which is passed through a fully connected layer with 256 neurons and ReLU activation [91]. To mitigate overfitting, a Dropout layer with rate 0.1 is applied. The final output layer has two units with Softmax normalization, yielding probabilities that sum to one, with one component interpreted as the predicted energy-loss ratio.

The network is trained using the AdamW optimizer [92] with a learning rate of 10^{-4} . Training is performed with batches of 1024 jet point clouds, reshuffled at each epoch, for up to 10 epochs, with model checkpointing based on the best validation loss.

The architectural advantage of DGCNN lies in its ability to capture fine-grained substructure patterns that are inherently lost in image-based representations. While CNN’s pixelization process blurs these precise particle-level relationships by compressing continuous kinematic information into discrete bins, DGCNN’s EdgeConv operations preserve geometric fidelity through direct computation of feature differences between neighboring particles. Furthermore, the dynamic graph update mechanism allows the network to hierarchically integrate local constituent information, enabling it to learn both detailed particle-level correlations and larger-scale jet substructure patterns with higher fidelity.

Since the DGCNN involves dynamic k -nearest-neighbor searches and multiple EdgeConv operations, it requires more computational time than the CNN used in this work. On a

single NVIDIA T4 GPU, the training times of our tasks are about 8 h for the CNN and 33 h for the DGCNN. The per-jet inference times are about 0.05 ms for the CNN and 0.5 ms for the DGCNN.

V. PREDICTION PERFORMANCE

We first study the performance of the CNN on predicting jet energy loss with images of energy distribution inside jets as input. To facilitate feature extraction by the CNN, we identify leading and subleading subjets inside each jet using the Soft Drop algorithm [93] with parameters $z_{\text{cut}} = 0.1$ and $\beta = 0$, and translate the leading subjet to the center of the η - ϕ plane, rotate the subleading one to the $-\pi/2$ direction, and reflect the image if necessary to ensure that its right half owns higher intensity than its left half. Shown in the left four panels of Fig. 4 are images of jets within different ranges of the energy loss ratio χ with smaller χ representing stronger jet energy loss. Here, jets are reconstructed using the output particles from the LBT simulation, without contribution from the background particles of the QGP, i.e., the “LBT-only” or “background-free” setup defined in Sec. II. To demonstrate the features of quenched jets visually, we average self-normalized jet images in each bin of χ per panel, where the normalization prevents high- p_T jets from dominating the results. From these average jet images, one can observe the emergence of soft particles at large angles relative to the hard core of a jet, represented by its leading subjet, as the jet scatters through the QGP. The jet energy appears more

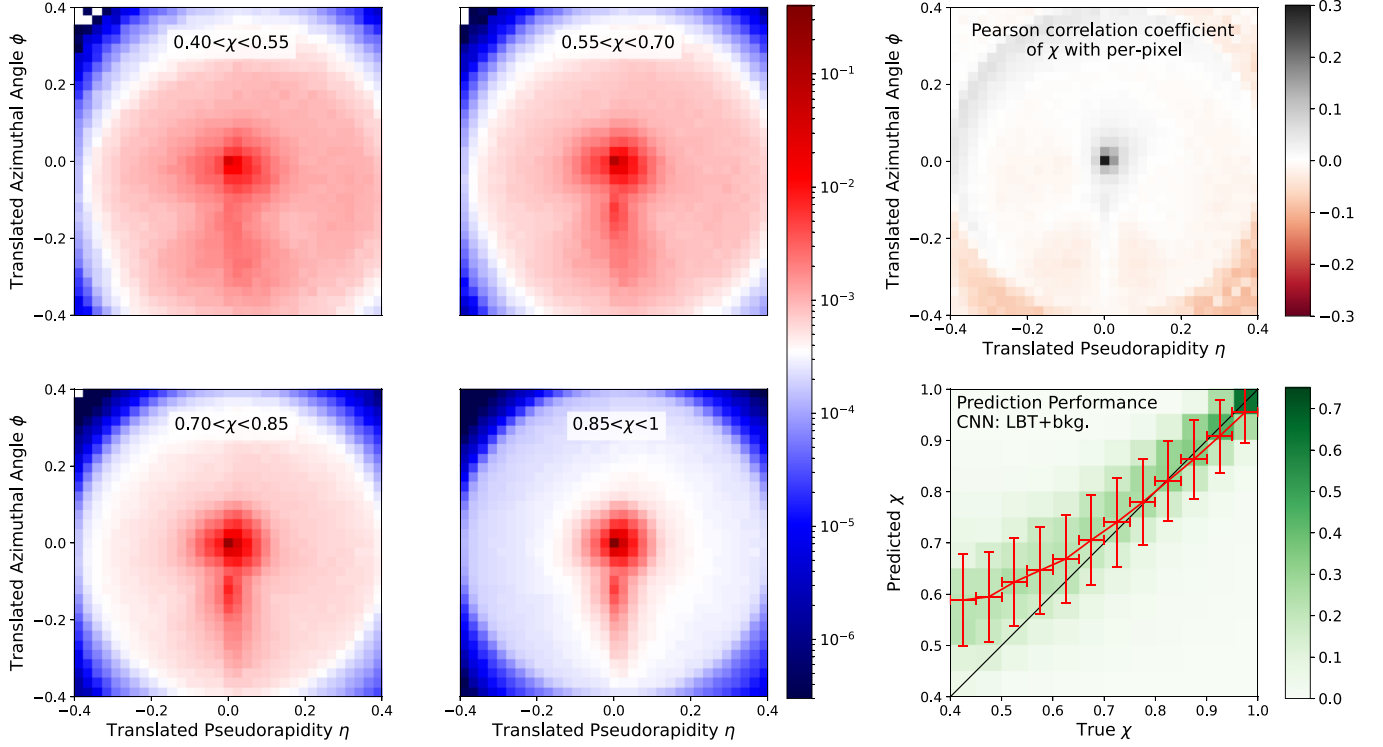


FIG. 5. Same as Fig. 4, except that jets are reconstructed together with the QGP background particles.

sparsely distributed as it loses more energy, or χ is smaller. This could be a distinct correlation between jet energy loss and jet image grasped by the CNN. To quantify this correlation, in the top right panel of Fig. 4 we present the Pearson correlation coefficient between χ and the value at each pixel of unnormalized jet image. The Pearson coefficient between samples X and Y with n population is defined as

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}. \quad (3)$$

This coefficient quantifies the linear correlation between two quantities, ranging from -1 (perfect anticorrelation) to $+1$ (perfect correlation), with 0 indicating no linear correlation. Large positive values of the coefficient are observed near the jet core, reflecting a positive correlation between χ and the remaining energy of the hard core of a jet. In contrast, negative values are observed at large angles, reflecting stronger energy transport from the jet core towards large angles for more quenched jets (with smaller χ). This further illustrates the potential of using a CNN to infer the amount of jet energy loss from their images.

Using the unnormalized background-free jet images as input, we train the CNN model as described in Sec. IV to predict the jet energy loss ratio χ . The bottom right panel of Fig. 4 shows the prediction performance, with the horizontal axis representing the true value of χ and the vertical axis representing the CNN prediction. The green shading shows the conditional probability distribution of the predicted χ given a true value, normalized per true- χ bin. The red line with error bars shows the mean and standard deviation of predictions in

each true- χ bin. The mean prediction closely follows the diagonal, indicating that the CNN effectively learns the mapping from jet image to the energy loss ratio χ for the LBT jets in the absence of the QGP background.

To assess the prediction performance under more realistic experimental conditions, we embed the LBT output particles in a soft-particle background before jet reconstruction. The background particles are generated using a thermal model, as described in Sec. II, and this setup is referred to as “LBT jets within background” (“LBT + bkg.”). The corresponding average jet images are shown in the left four panels of Fig. 5. Compared to the background-free case shown above, a large number of background particles are clustered into jets, as evident in Fig. 3. This contamination becomes more significant when jets lose more energy. This is because for more quenched jets, more recoil particles are generated at large angles, which can be hardly distinguished from background particles in jet reconstruction. Therefore, compared to the left four panels of Fig. 4, we see more sparsely distributed jet energy in Fig. 5 when jet energy loss is strong. For significantly quenched jets, e.g., in the range of $0.4 < \chi < 0.7$, the variation of jet images with respect to χ can be barely observed. The weaker correlation between χ and jet image is also reflected in the Pearson coefficients in the top right panel, which get closer to zero inside the jet cone ($R < 0.4$) compared to those in Fig. 4. Interestingly, negative Pearson coefficients emerge outside the $R = 0.4$ cone here, indicating considerable numbers of background particles outside the cone are clustered into jets when jet-medium interaction is strong. When these jet images are used as input to the CNN, the prediction performance, shown in the bottom right panel of

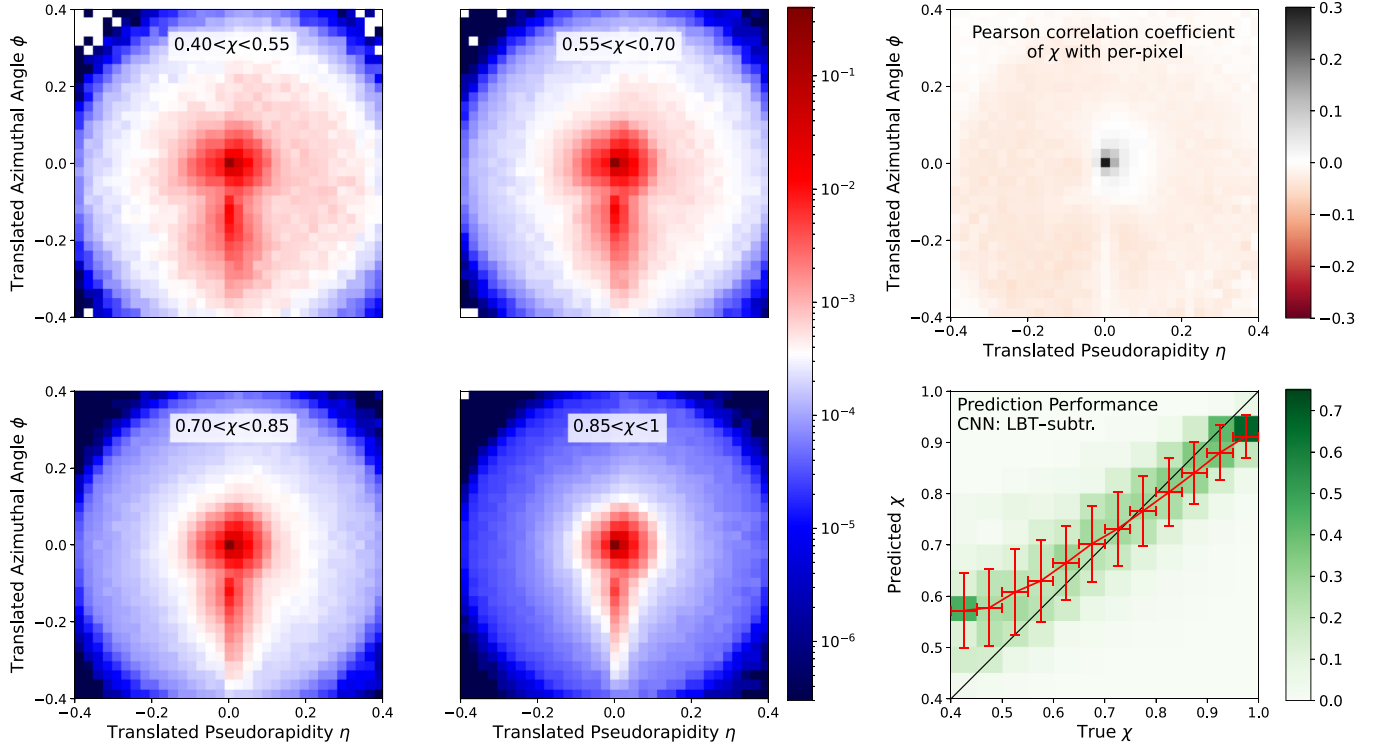


FIG. 6. Same as Figs. 4 and 5, except that the QGP background particles are introduced and then subtracted before jet reconstruction.

Fig. 5, exhibits clear deviations from the diagonal for $\chi \lesssim 0.7$. Thus, the degradation of the image-level features caused by background contamination directly translates into the reduced predictive power of the CNN. These results highlight the necessity of background subtraction before feeding jet images to machine learning models.

Now we apply background subtraction to the LBT jets embedded in the background using the constituent subtraction method as described in Sec. III. The corresponding results are labeled as “LBT jets with background subtracted” (“LBT-subtr.”). As shown in Fig. 6, the average jet images (left four panels) largely recover the χ -dependent features of jet quenching seen in Fig. 4, and negative values of Pearson coefficients re-emerge clearly inside the jet cone (top right panel). The mean CNN predictions for these images (bottom right panel) remain close to the diagonal, though slight deviations persist across the entire χ range.

For a quantitative comparison between different model setups, we show in Fig. 7 their prediction errors $\sqrt{\langle(\Delta\chi)^2\rangle}$ as functions of the true value of χ . The error is defined for each true- χ bin as the root-mean-square deviation between the true and predicted values of χ : $\sqrt{\langle(\Delta\chi)^2\rangle} = \sqrt{\sum_{i \in \text{bin}} (\chi_i^{\text{true}} - \chi_i^{\text{pred}})^2 / N_{\text{bin}}}$, where N_{bin} is the number of jets in the bin. The prediction accuracy based on the LBT jets with background subtracted improves for $\chi \lesssim 0.90$ but degrades for $\chi > 0.95$ compared to the case based on that within background. Nevertheless, the overall performance remains below that of the background-free (LBT-only) case, likely due to an over-subtraction of soft particles by the constituent subtraction method, as illustrated in Fig. 3 and the comparison

between jet images in Figs. 4 and 6. These results indicate that although background subtraction partially restores the predictive power of CNNs, significant limitations persist, motivating the need for alternative representations beyond jet images.

To further improve the prediction performance, we replace the CNN with a DGCNN. As discussed in Sec. IV B, for each background-subtracted LBT jet, we generate a point-cloud representation and use it as input to the DGCNN. Figure 8 shows the corresponding prediction results. Within each true-

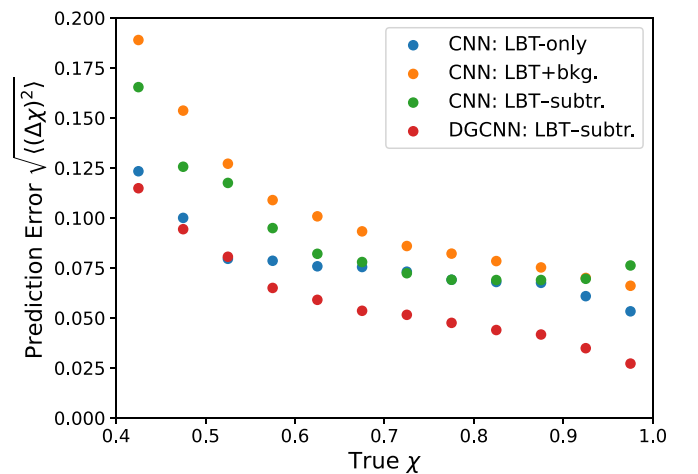


FIG. 7. Prediction errors $\sqrt{\langle(\Delta\chi)^2\rangle}$ as functions of the true value of χ , compared between the CNNs trained using three jet samples (LBT-only, LBT + bkg., LBT-subtr.) and the DGCNN trained using the LBT-subtr. jets.

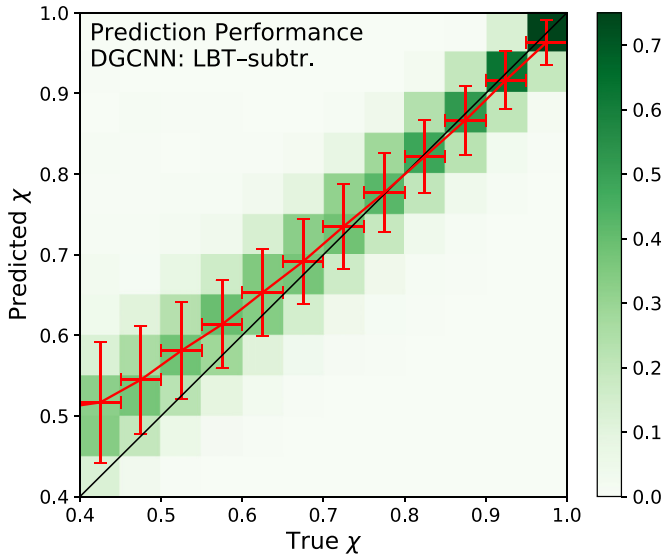


FIG. 8. The prediction performance of the DGCNN with the LBT jets with background subtracted as input. The green shading represents the probability of the predicted χ along the y axis given a true value of χ . Each column is normalized here. The red line and error bars quantify the mean and standard deviation of the predicted χ , respectively, given the true χ .

χ bin, the predicted values are more tightly clustered around the mean, and the mean line lies closer to the diagonal compared to the previous CNN results, indicating a more accurate per-bin prediction. The prediction error as a function of the true χ , quantified in Fig. 7, further demonstrates the advantage of the DGCNN: the root-mean-square deviation $\sqrt{\langle(\Delta\chi)^2\rangle}$ is smaller across the full χ range compared to the CNN results, even relative to the background-free case. We note that although the performance of the CNN could potentially be improved by further refining the background subtraction procedure, our main conclusion regarding the superior performance of the DGCNN remains unchanged. In particular, the “LBT-subtr.” setup of the DGCNN outperforms the “LBT-only” setup of the CNN, with the latter corresponding to an idealized scenario of perfect background subtraction that is not achievable in realistic experiments. This demonstrates that the advantage of the DGCNN is robust even in the presence of realistic fluctuations introduced by background contamination and subtraction.

It is important to understand why the point-cloud-based DGCNN architecture achieves a higher prediction accuracy than the image-based CNN. A previous study [28] showed that a fully connected network trained on a set of jet-shape variables, fragmentation functions, and soft-drop substructure observables can reach a performance comparable to CNNs trained on jet images. This indicates that CNNs are indeed capable of learning features closely related to these physically motivated observables, even though such information is partly blurred by the pixelization of jet images.

In contrast, the point-cloud representation used by DGCNN preserves the full per-particle kinematic information without discretization, enabling the network to access more

detailed correlations among jet constituents. Through the EdgeConv operations, the DGCNN directly processes local geometric relationships between neighboring particles, which allows it to capture fine-grained structures such as total particle multiplicity and its angular distribution that are strongly correlated with medium-induced jet modifications. While it is inherently difficult to determine which specific jet features are learned by a neural network, the improved performance of the DGCNN suggests that models operating on minimally processed particle-level data can more faithfully encode the jet-medium interaction patterns that govern the energy-loss fraction χ .

VI. CONCLUSIONS

We implemented two neural-network architectures, convolutional neural networks (CNNs) and dynamic graph convolutional neural networks (DGCNNs), to predict the jet-by-jet energy loss ratio χ in LBT simulations, aiming to reduce the selection bias in jet quenching studies. Extending earlier work [28] based on hybrid strong/weak coupling models [60] for background-free jets, we demonstrated that these machine-learning approaches remain effective for jets quenched in the LBT model, even in the presence of the QGP background. Each jet can be represented in two ways: as an energy-deposition image for the CNN, and as a particle cloud for the DGCNN. The jet images intuitively reveal that stronger quenching corresponds to enhanced soft-particle activity at large angles, while the particle-cloud representation retains full constituent-level information.

For the background-free LBT jets, the CNN effectively predicts the true χ values. To approximate realistic experimental conditions, we embedded jets in a soft-particle background, and then applied background subtraction before training. CNN performance degrades in the presence of background and remains below the background-free baseline even after the background removal with the constituent subtraction method. In contrast, the DGCNN applied to background-subtracted particle clouds achieves consistently higher accuracy across the entire χ range (0.4–1.0), demonstrating that point-cloud-based methods, by leveraging the full jet structure, retain superior predictive power for jet energy loss even after background contamination and removal.

To reliably apply our framework for per-jet energy loss predictions under realistic experimental conditions, future studies must address five key directions. First, it is necessary to incorporate hadronization into our jet modeling so that the trained neural networks can be applied more reliably to real data. While hadronization is not expected to significantly affect the jet images used in the CNN, it does influence the per-particle information that the DGCNN relies on. Second, incorporating more realistic background particles into the simulations is essential for faithfully reproducing experimental environments. This includes sampling soft hadrons from hydrodynamic simulations of the QGP and embedding nonjet particles from hard nucleon-nucleon scatterings to model the underlying event. In addition, future studies should consider pile-up and detector effects to more accurately reflect realistic analyses. Third, developing background subtraction techniques that better

preserve jet substructures will be crucial to mitigate distortions from the soft QGP particles. Fourth, it is important to validate, and if necessary, enhance the generalizability of machine learning approaches across different jet quenching models. Although we expect the ML models trained using the LBT data should retain certain predictive power when applied to jets generated from other quenching models, as long as those models contain proper descriptions of jet-medium interactions and capture the key correlation between jet energy loss and jet structure, a more reliable generalizability would require training the networks on a combined dataset from multiple jet models. Last but not least, Fig. 7 shows that the prediction errors generally increase for jets that experience larger energy loss, regardless of which ML model is used and whether background contamination is present or subtracted. This trend reflects the fact that strongly quenched jets tend to exhibit more complex internal structures in heavy-ion collisions. As confirmed in an earlier study [28], the prediction errors grow for jets with large groomed jet radius R_g , large Soft Drop multiplicity n_{SD} , and (groomed) jet mass around 40 GeV. These observations motivate the exploration of more advanced architectures or alternative ML techniques capable of capturing the subtler features of medium-modified jet structure, thereby further improving both the precision and robustness of per-jet energy-loss predictions.

Finally, we comment on the definition of the jet energy-loss fraction used throughout this work. Our study employs the inclusive definition $\chi = p_T^{\text{jet,med}}/p_T^{\text{jet,vac}}$, based on matched jets with $\Delta R < 0.4$, which naturally incorporates possible modifications of the jet axis and constituent composition during in-medium evolution. This definition provides a direct and physically transparent measure of the total energy

loss in theoretical jet-quenching studies. Nonetheless, jets in realistic heavy-ion environments can experience significant background contamination and medium response, motivating the exploration of more differential and potentially more robust quenching measures. Examples include groomed-jet p_T fractions, leading-subjet p_T fractions, and jet core energy fractions, which emphasize the hardest branches or the jet core and therefore reduce sensitivity to soft background activity. Although such alternatives were not investigated in the present work, systematically examining groomed-based or subjet-based definitions of the energy-loss fraction thus constitutes an interesting and promising direction for future research.

ACKNOWLEDGMENTS

We are grateful for helpful discussions with Xiao-Zhi Bai. This work is supported in part by the National Natural Science Foundation of China (NSFC) under Grants No. 12175122, No. 2021-867, No. 12321005 (R.L. and S.C.), and in part by the Taishan Scholars Program under Grant No. tsqz20221162, and Shandong Excellent Young Scientists Fund Program (Overseas) under Grant No. 2023HWYQ-106 (Y.D.).

DATA AVAILABILITY

The data that support the findings of this article are not publicly available upon publication because it is not technically feasible and/or the cost of preparing, depositing, and hosting the data would be prohibitive within the terms of this research project. The data are available from the authors upon reasonable request.

-
- [1] M. Gyulassy and L. McLerran, *Nucl. Phys. A* **750**, 30 (2005).
 - [2] P. Jacobs and X.-N. Wang, *Prog. Part. Nucl. Phys.* **54**, 443 (2005).
 - [3] W. Busza, K. Rajagopal, and W. van der Schee, *Annu. Rev. Nucl. Part. Sci.* **68**, 339 (2018).
 - [4] H. Elfner and B. Müller, *J. Phys. G: Nucl. Part. Phys.* **50**, 103001 (2023).
 - [5] X.-N. Wang and M. Gyulassy, *Phys. Rev. Lett.* **68**, 1480 (1992).
 - [6] G.-Y. Qin and X.-N. Wang, *Int. J. Mod. Phys. E* **24**, 1530014 (2015).
 - [7] S. Cao and X.-N. Wang, *Rep. Prog. Phys.* **84**, 024301 (2021).
 - [8] B. Abelev, *et al.* (The ALICE Collaboration), *J. High Energy Phys.* **03** (2014) 013.
 - [9] J. Adam, *et al.* (ALICE Collaboration), *Phys. Lett. B* **746**, 1 (2015).
 - [10] G. Aad, *et al.* (ATLAS Collaboration), *Phys. Rev. Lett.* **114**, 072302 (2015).
 - [11] M. Aaboud, *et al.* (The ATLAS Collaboration), *Phys. Lett. B* **774**, 379 (2017).
 - [12] M. Aaboud Jr., *et al.* (The ATLAS Collaboration), *Phys. Lett. B* **790**, 108 (2019).
 - [13] S. Acharya, *et al.* (ALICE Collaboration), *Phys. Rev. C* **101**, 034911 (2020).
 - [14] L. Apolinário, Y.-T. Chien, and L. Cunqueiro Mendez, *Int. J. Mod. Phys. E* **33**, 2430003 (2024).
 - [15] J. Casalderrey-Solana, D. Gulhan, G. Milhano, D. Pablos, and K. Rajagopal, *J. High Energy Phys.* **03** (2017) 135.
 - [16] Y. Tachibana, N.-B. Chang, and G.-Y. Qin, *Phys. Rev. C* **95**, 044909 (2017).
 - [17] R. Kunawalkam Elayavalli and K. C. Zapp, *J. High Energy Phys.* **07** (2017) 141.
 - [18] G. Milhano, U. A. Wiedemann, and K. C. Zapp, *Phys. Lett. B* **779**, 409 (2018).
 - [19] W. Chen, S. Cao, T. Luo, L.-G. Pang, and X.-N. Wang, *Phys. Lett. B* **810**, 135783 (2020).
 - [20] C. Park, S. Jeon, and C. Gale, *Nucl. Phys. A* **982**, 643 (2019).
 - [21] T. Luo, S. Cao, Y. He, and X.-N. Wang, *Phys. Lett. B* **782**, 707 (2018).
 - [22] J. Casalderrey-Solana, G. Milhano, D. Pablos, and K. Rajagopal, *J. High Energy Phys.* **01** (2020) 044.
 - [23] N.-B. Chang, Y. Tachibana, and G.-Y. Qin, *Phys. Lett. B* **801**, 135181 (2020).
 - [24] Y. Tachibana, C. Shen, and A. Majumder, *Phys. Rev. C* **106**, L021902 (2022).
 - [25] Z. Yang, Y. He, I. Moul, and X.-N. Wang, *Phys. Rev. Lett.* **132**, 011901 (2024).

- [26] W.-J. Xing, S. Cao, G.-Y. Qin, and X.-N. Wang, *Phys. Rev. Lett.* **134**, 052301 (2025).
- [27] J. Brewer, J. G. Milhano, and J. Thaler, *Phys. Rev. Lett.* **122**, 222301 (2019).
- [28] Y.-L. Du, D. Pablos, and K. Tywoniuk, *J. High Energy Phys.* **03** (2021) 206.
- [29] Y.-L. Du, D. Pablos, and K. Tywoniuk, *Phys. Rev. Lett.* **128**, 012301 (2022).
- [30] A. Takaacs and K. Tywoniuk, *J. High Energy Phys.* **10** (2021) 038.
- [31] J. Brewer, Q. Brodsky, and K. Rajagopal, *J. High Energy Phys.* **02** (2022) 175.
- [32] C. Andres, J. Holguin, R. Kunawalkam Elayavalli, and J. Viinikainen, *Phys. Rev. Lett.* **134**, 082303 (2025).
- [33] Y.-T. Chien and R. Kunawalkam Elayavalli, *arXiv:1803.03589*.
- [34] Y.-L. Du, D. Pablos, and K. Tywoniuk, *PoS PANIC2021*, 224 (2022).
- [35] L. Apolinário, N. F. Castro, M. Crispim Romão, J. G. Milhano, R. Pedro, and F. C. R. Peres, *J. High Energy Phys.* **11** (2021) 219.
- [36] L. Liu, J. Velkovska, Y. Wu, and M. Verweij, *J. High Energy Phys.* **04** (2023) 140.
- [37] Y. S. Lai, J. Mulligan, M. Płoskoń, and F. Ringer, *J. High Energy Phys.* **10** (2022) 011.
- [38] M. Crispim Romão, J. G. Milhano, and M. van Leeuwen, *SciPost Phys.* **16**, 015 (2024).
- [39] U. S. Qureshi and R. K. Elayavalli, *arXiv:2411.19389*.
- [40] Z. Yang, Y. He, W. Chen, W.-Y. Ke, L.-G. Pang, and X.-N. Wang, *Eur. Phys. J. C* **83**, 652 (2023).
- [41] R. Haake and C. Loizides, *Phys. Rev. C* **99**, 064904 (2019).
- [42] S. Acharya, *et al.* (ALICE Collaboration), *Phys. Lett. B* **849**, 138412 (2024).
- [43] R. Li, Y.-L. Du, and S. Cao, *Phys. Lett. B* **870**, 139940 (2025).
- [44] D. Stewart and J. Putschke, *Phys. Rev. C* **111**, 054902 (2025).
- [45] U. S. Qureshi and R. K. Elayavalli, *Phys. Rev. C* **112**, 064906 (2025).
- [46] P. Baldi, K. Bauer, C. Eng, P. Sadowski, and D. Whiteson, *Phys. Rev. D* **93**, 094034 (2016).
- [47] L. de Oliveira, M. Kagan, L. Mackey, B. Nachman, and A. Schwartzman, *J. High Energy Phys.* **07** (2016) 069.
- [48] P. T. Komiske, E. M. Metodiev, and M. D. Schwartz, *J. High Energy Phys.* **01** (2017) 110.
- [49] G. Kasieczka, T. Plehn, M. Russell, and T. Schell, *J. High Energy Phys.* **05** (2017) 006.
- [50] D. Guest, J. Collado, P. Baldi, S.-C. Hsu, G. Urban, and D. Whiteson, *Phys. Rev. D* **94**, 112002 (2016).
- [51] F. A. Dreyer, G. P. Salam, and G. Soyez, *J. High Energy Phys.* **12** (2018) 064.
- [52] G. Louppe, K. Cho, C. Becot, and K. Cranmer, *J. High Energy Phys.* **01** (2019) 057.
- [53] T. Cheng, *Comput. Softw. Big Sci.* **2**, 3 (2018).
- [54] P. T. Komiske, E. M. Metodiev, and J. Thaler, *J. High Energy Phys.* **01** (2019) 121.
- [55] M. J. Dolan and A. Ore, *Phys. Rev. D* **103**, 074022 (2021).
- [56] H. Qu and L. Gouskos, *Phys. Rev. D* **101**, 056019 (2020).
- [57] J. Guo, J. Li, T. Li, and R. Zhang, *Phys. Rev. D* **103**, 116025 (2021).
- [58] X. Ju and B. Nachman, *Phys. Rev. D* **102**, 075014 (2020).
- [59] S. Gong, Q. Meng, J. Zhang, H. Qu, C. Li, S. Qian, W. Du, Z.-M. Ma, and T.-Y. Liu, *J. High Energy Phys.* **07** (2022) 030.
- [60] J. Casallerrey-Solana, D. C. Gulhan, J. G. Milhano, D. Pablos, and K. Rajagopal, *J. High Energy Phys.* **09** (2015) 175.
- [61] S. Cao, T. Luo, G.-Y. Qin, and X.-N. Wang, *Phys. Rev. C* **94**, 014909 (2016).
- [62] T. Luo, Y. He, S. Cao, and X.-N. Wang, *Phys. Rev. C* **109**, 034919 (2024).
- [63] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, *ACM Trans. Graphics* **38**, 1 (2019).
- [64] T. Sjöstrand, S. Mrenna, and P. Z. Skands, *J. High Energy Phys.* **05** (2006) 026.
- [65] T. Sjöstrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C. O. Rasmussen, and P. Z. Skands, *Comput. Phys. Commun.* **191**, 159 (2015).
- [66] JetToyHI: Framework to study jet substructure in heavy ion environment, <https://github.com/mverwe/JetToyHI>.
- [67] P. Skands, S. Carrazza, and J. Rojo, *Eur. Phys. J. C* **74**, 3024 (2014).
- [68] M. Cacciari, G. P. Salam, and G. Soyez, *Eur. Phys. J. C* **72**, 1896 (2012).
- [69] Y. He, T. Luo, X.-N. Wang, and Y. Zhu, *Phys. Rev. C* **91**, 054908 (2015); **97**, 019902(E) (2018).
- [70] X.-F. Guo and X.-N. Wang, *Phys. Rev. Lett.* **85**, 3591 (2000).
- [71] X.-N. Wang and X.-F. Guo, *Nucl. Phys. A* **696**, 788 (2001).
- [72] B.-W. Zhang, E. Wang, and X.-N. Wang, *Phys. Rev. Lett.* **93**, 072301 (2004).
- [73] A. Majumder, *Phys. Rev. D* **85**, 014023 (2012).
- [74] L.-G. Pang, H. Petersen, and X.-N. Wang, *Phys. Rev. C* **97**, 064918 (2018).
- [75] X.-Y. Wu, G.-Y. Qin, L.-G. Pang, and X.-N. Wang, *Phys. Rev. C* **105**, 034909 (2022).
- [76] M. Zhang, Y. He, S. Cao, and L. Yi, *Chin. Phys. C* **47**, 024106 (2023).
- [77] S. Acharya, *et al.* (ALICE Collaboration), *Phys. Rev. C* **101**, 044907 (2020).
- [78] M. Cacciari, G. P. Salam, and G. Soyez, *J. High Energy Phys.* **04** (2008) 063.
- [79] P. Berta, L. Masetti, D. W. Miller, and M. Spousta, *J. High Energy Phys.* **08** (2019) 175.
- [80] M. Cacciari, G. P. Salam, and G. Soyez, *J. High Energy Phys.* **04** (2008) 005.
- [81] S. Catani, Y. L. Dokshitzer, M. H. Seymour, and B. R. Webber, *Nucl. Phys. B* **406**, 187 (1993).
- [82] S. D. Ellis and D. E. Soper, *Phys. Rev. D* **48**, 3160 (1993).
- [83] L.-G. Pang, K. Zhou, N. Su, H. Petersen, H. Stöcker, and X.-N. Wang, *Nat. Commun.* **9**, 210 (2018).
- [84] Y.-L. Du, K. Zhou, J. Steinheimer, L.-G. Pang, A. Motornenko, H.-S. Zong, X.-N. Wang, and H. Stöcker, *Eur. Phys. J. C* **80**, 516 (2020).
- [85] K. He, X. Zhang, S. Ren, and J. Sun, in *Proceedings of the IEEE International Conference on Computer Vision* (IEEE, Santiago, Chile, 2015), pp. 1026–1034.
- [86] A. Y. Ng, in *Proceedings of the Twenty-First International Conference on Machine Learning* (ACM, New York, 2004), p. 78.
- [87] S. Ioffe and C. Szegedy, in *Proceedings of the 32nd International Conference on Machine Learning* (PMLR, Lille, 2015), pp. 448–456.
- [88] D. P. Kingma and J. Ba, in *3rd International Conference for Learning Representations* (ICLR, San Diego, 2015).

- [89] E. Fix and J. L. Hodges, Jr., Discriminatory analysis-nonparametric discrimination: Small sample performance, Technical Report, USAF, Project No. 21-49-004, Report No. 11 (1952).
- [90] T. Cover and P. Hart, [IEEE Trans. Inf. Theory](#) **13**, 21 (1967).
- [91] X. Glorot, A. Bordes, and Y. Bengio, in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, JMLR Workshop and Conference Proceedings (JMLR, Fort Lauderdale, 2011), pp. 315–323.
- [92] I. Loshchilov and F. Hutter, in *Proceedings of Seventh International Conference on Learning Representations (ICLR, Vancouver, 2019)*.
- [93] A. J. Larkoski, S. Marzani, G. Soyez, and J. Thaler, [J. High Energy Phys.](#) **05** (2014) 146.