2-2024

Artificial Intelligence for the Electron Ion Collider (AI4EIC)

C. Allaire

...

Cristiano Fanelli
William & Mary, cfanelli@wm.edu

James Giroux
William & Mary

Joey Niestroy
William & Mary

See next page for additional authors

Recommended Citation

Allaire, C.; ...; Fanelli, Cristiano; Giroux, James; Niestroy, Joey; Stevens, Justin R.; Stone, Patrick; Suarez, L.; Suresh, K.; Walter, Eric; and et al., Artificial Intelligence for the Electron Ion Collider (AI4EIC) (2024). *Computing and Software for Big Science*, 8(5).
https://doi.org/10.1007/s41781-024-00113-4

Computing and Software for Big Science (2024) 8:5
https://doi.org/10.1007/s41781-024-00113-4

REVIEW

Artificial Intelligence for the Electron Ion Collider (AI4EIC)

C. Allaire[60] · R. Ammendola[22] · E.-C. Aschenauer[3] · M. Balandat[33] · M. Battaglieri[36] · J. Bernauer[6,46] · M. Bondì[35] · N. Branson[14,32] · T. Britton[27] · A. Butter[28] · I. Chahrour[55] · P. Chatagnon[27] · E. Cisbani[37] · E. W. Cline[46] · S. Dash[23] · C. Dean[31] · W. Deconinck[54] · A. Deshpande[3,6] · M. Diefenthaler[27] · R. Ent[27] · C. Fanelli[27,64] · M. Finger[10] · M. Finger Jr.[10] · E. Fol[5] · S. Furletov[27] · Y. Gao[3] · J. Giroux[56,64] · N. C. Gunawardhana Waduge[58] · O. Hassan[54,57] · P. L. Hegde[9] · R. J. Hernández-Pinto[16] · A. Hiller Blin[25] · T. Horn[47] · J. Huang[3] · A. Jalotra[53] · D. Jayakodige[21,27] · B. Joo[39] · M. Junaid[56] · N. Kalantarians[62] · P. Karande[30] · B. Kriesten[8] · R. Kunnawalkam Elayavalli[61] · Y. Li[41] · M. Lin[3] · F. Liu[39] · S. Liuti[58] · G. Matousek[15] · M. McEneaney[15] · D. McSpadden[27] · T. Menzo[51] · T. Miceli[17] · V. Mikuni[65] · R. Montgomery[52] · B. Nachman[29,48] · R. R. Nair[34] · J. Niestroy[64] · S. A. Ochoa Oregon[16] · J. Oleniacz[63] · J. D. Osborn[3] · C. Paudel[18] · C. Pecar[15] · C. Peng[1] · G. N. Perdue[17] · W. Phelps[11,27] · M. L. Purschke[3] · H. Rajendran[9] · K. Rajput[27] · Y. Ren[29] · D. F. Renteria-Estrada[16] · D. Richford[2] · B. J. Roy[38] · D. Roy[45] · A. Saini[17] · N. Sato[27] · T. Satogata[27,40] · G. Sborlini[12,20] · M. Schram[27] · D. Shih[44] · J. Singh[43] · R. Singh[4,7] · A. Siodmok[26] · J. Stevens[64] · P. Stone[64] · L. Suarez[64] · K. Suresh[56,64] · A.-N. Tawfik[19] · F. Torales Acosta[29] · N. Tran[17] · R. Trotta[47] · F. J. Twagirayezu[50] · R. Tyson[52] · S. Volkova[42] · A. Vossen[15,27] · E. Walter[64] · D. Whiteson[49] · M. Williams[31] · S. Wu[54] · N. Zachariou[59] · P. Zurita[13,24]

Received: 19 July 2023 / Accepted: 12 January 2024

Abstract
The Electron-Ion Collider (EIC), a state-of-the-art facility for studying the strong force, is expected to begin commissioning its first experiments in 2028. This is an opportune time for artificial intelligence (AI) to be included from the start at this facility and in all phases that lead up to the experiments. The second annual workshop organized by the AI4EIC working group, which recently took place, centered on exploring all current and prospective application areas of AI for the EIC. This workshop is not only beneficial for the EIC, but also provides valuable insights for the newly established ePIC collaboration at EIC. This paper summarizes the different activities and R&D projects covered across the sessions of the workshop and provides an overview of the goals, approaches and strategies regarding AI/ML in the EIC community, as well as cutting-edge techniques currently studied in other experiments.

Keywords Artificial Intelligence · Deep learning · EIC · ePIC · Machine learning · QCD · Physics

Abbreviations

ACTS	A common tracking software
ADWIN	Adaptive windowing
AI	Artificial Intelligence
AI4EIC	Artificial Intelligence for the Electron Ion Collider
ASICs	Application-specific integrated circuit
AWS	Amazon web services
BNL	Brookhaven National Laboratory
CDC	Central drift chamber
cMAF	Conditional masked autoregressive flow
cAE	Conditional autoencoder
CNN	Convolutional neural network
CPU	Central processing unit
DAQ	Data acquisition
DIRC	Detection of internally reflected Cherenkov light
DIS	Deep inelastic scattering
DL	Deep learning
DM	Diffusion model
dRICH	dual-radiator Ring Imaging CHerenkov
DVCS	Deeply Virtual Compton Scattering
FPGA	Field Programmable Gate Array
GAN	Generative adversarial network
GIN	Graph isomorphism networks
GNN	Graph neural network
GPD	Generalized parton distribution
GPU	Graphics processing unit
HEP	High energy physics
JLab	Jefferson Lab

Extended author information available on the last page of the article

Published online: 15 February 2024

LHC	Large Hadron Collider
LSTM	Long-short term memory
MARS	Modified multivariate value-at-risk approximation based on random scalarizations
MC	Monte Carlo
MCEG	Monte Carlo event generator
ML	Machine learning
MLP	Multi-layer perceptron
MLOps	Machine learning operations
MOBO	Multi-objective Bayesian optimization
MOEA	Multi-objective evolutionary algorithm
MOGA	Multi-objective genetic algorithm
MOO	Multi-objective optimization
MORBO	Multi-objective trust-region Bayesian optimization
NF	Normalizing flow
NN	Neural network
NP	Nuclear physics
ODD	Open data detector
PDF	Parton distribution function
PID	Particle identification
pQCD	Perturbative quantum chromodynamics
QCD	Quantum chromodynamics
QCF	Quantum correlation function
SI-DIS	Semi-inclusive deep inelastic scattering
SRF	Superconducting radio frequency
SRO	Streaming readout
sWAE	Sliced-Wasserstein auto-encoder
TCS	Timelike Compton scattering
VAE	Variational auto-encoder
VLAD	Vectors of locally aggregated descriptors

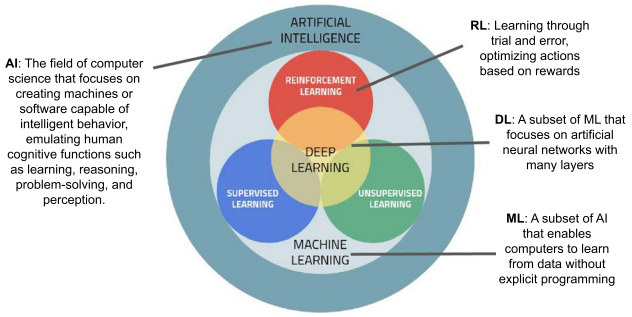

Fig. 1 Taxonomy: a diagrammatic representation of artificial intelligence, machine learning, and deep learning is provided to familiarize readers with the corresponding acronyms utilized in the text

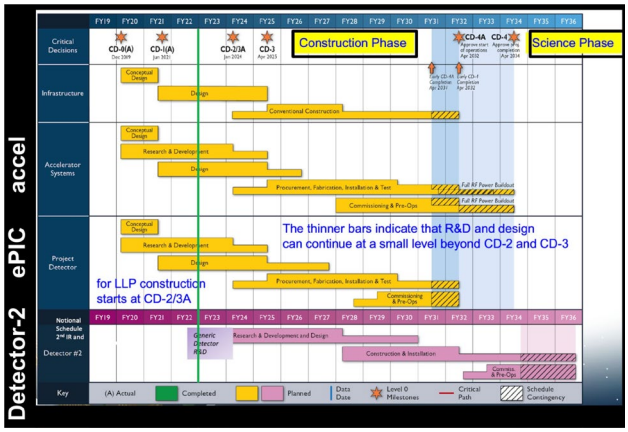

Fig. 2 EIC schedule: the Gantt chart represents different phases (design, construction, science) for accelerator, the ePIC experiment, and a potential detector-2 at EIC. Image taken from [2] and presented in October 2022

Introduction

In October 2022, the second workshop on Artificial Intelligence for the Electron-Ion-Collider (AI4EIC) was held in William & Mary. The workshop delved into a range of active and potential application areas of AI/ML[1] for the EIC, and it was also an opportunity to showcase some of the ongoing research activities in these areas for the recently formed ePIC Collaboration.

The event also had a strong outreach and educational component with different tutorials given by experts in AI and ML from national labs, universities, and industry as well as a hackathon satellite event during the last day of the workshop.

In Abbreviations, we list many of the methods encountered in this work, with their respective acronyms.

As discussed in the EIC Yellow Report [1] and as further deepened during the AI4EIC workshops, AI/ML will permeate all phases of the EIC schedule (shown in Fig. 2), and will involve accelerator and detector activities.

The second *AI4EIC* workshop broadened the scope of its predecessor. Specifically, while the initial workshop was centered on experimental applications for accelerators and detectors, the second workshop pivoted towards more specific applications for the EIC detector program and fostered linkages between theoretical and experimental aspects.

The workshop was structured with the following sessions: AI/ML for Design, Experiment/Theory Connections, Reconstruction and Particle Identification (PID), AI/ML

[1] In this document, we follow a hierarchical taxonomy for artificial intelligence (AI), subdivided into Machine Learning (ML) and Deep Learning (DL). ML, a subset of AI, pertains to a machine's ability to deduce input–output relationships without explicit mathematical instructions. DL, a further refinement within ML, employs intricate neural networks to mimic human brain interactions, facilitating learning from unstructured inputs. See Fig. 1

Infrastructure and Frontiers, and AI/ML in Streaming Readout (SRO). Interwoven throughout the workshop were comprehensive tutorials delivered by seasoned experts from academia, industry, and national labs.

This document is organized as follows:

- In Sect. 3, we delve into discussions from the design session.
- In Sect. 4, we underscore the interplay between theory and experiment through AI/ML applications.
- In Sect. 5, we discuss recent advances in reconstruction and particle identification, emphasizing their applications to the EIC case.
- In Sect. 6, we detail the infrastructure solutions required for transitioning from prototype to production environments. We also address the stimulating panel discussion on AI/ML frontiers, which could shape EIC science in the coming years.
- Section 7 focuses on the potential of integrating AI/ML within a streaming readout data processing environment, prompting a convergence between offline and online analyses.
- Section 8 highlights community efforts, including tutorials and a hackathon, that were conducted during the AI4EIC workshop week.

Concluding our report, Sect. 9 encapsulates our findings and conclusions.

Design of EIC

The development of innovative experimental equipment at the EIC is skillfully leveraging cutting-edge algorithmic advancements within the dynamic landscape of AI-inspired methodologies. Throughout the instrumentation design process, decisions are made with the primary objective of optimizing performance, while thoroughly considering all project limitations and constraints.

Fundamentally, the design evolves into a meticulous optimization process of a multiparameter system, characterized either through Monte Carlo (MC) simulation or by analytical models, which are corroborated by existing experimental data and specific test results. At the EIC, accelerators and spectrometers represent complex systems, and their respective performances are optimized individually, while acknowledging their interconnected requisites. Ideally, these systems should be optimized concurrently, but current practices haven't reached this stage.

In the design session, the presenters provided a comprehensive summary of recent advancements in the application of AI-based methods to the definition and design of both spectrometer components and accelerators,

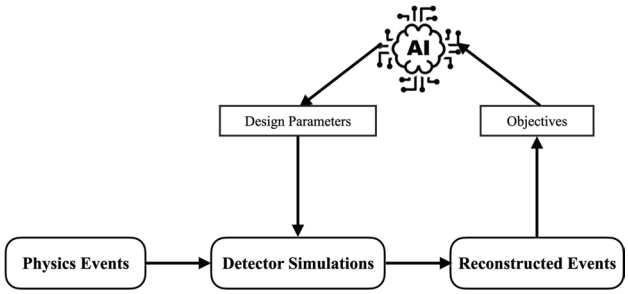

Fig. 3 AI-assisted detector design: flowchart of the main steps characterizing detector design optimization. Image taken from [3]

encapsulating a brief overview of AI-assisted operations. The following points were emphasized: (i) the various sub-detectors within the spectrometer should no longer be approached individually, as was the norm previously, a practice largely due to the diversity in specialized expertise and established work routines. Instead, a holistic perspective that considers all sub-detectors as a unified whole should be adopted [3–7]. (ii) Design of detectors is fundamentally a multi-objective optimization (MOO) process, characterized by numerous parameters that define the system under design and several potentially conflicting objectives that need to be optimized concurrently, subject to constraints. The balancing act between optimizing objectives and adhering to constraints typically necessitates considerable computational effort and time.

The EIC could spearhead the application of AI/ML to assist the design of large scale experiments, starting with the first detector, ePIC, and potentially extending to a second detector planned for the coming years. Considering the ongoing AI revolution, the discussion surrounding the use of AI/ML to aid the design of these experiments is particularly relevant and timely, as their design phase is currently underway.

A typical workflow for detector design is displayed in Fig. 3. An emerging and efficacious strategy to alleviate the computational demands of design optimization is the utilization of Parallel Bayesian Optimization. This method, which focuses on vector-based black-box functions with Expected Hypervolume Improvement [8], a metric encapsulating the range of desirable outcomes, promises superior sample-efficiency. It accomplishes this by identifying the Pareto frontier (optimal solutions) as the most effective trade-offs. These Pareto optimal solutions represent the best possible outcomes where no single objective can be improved without compromising another, offering a clear landscape of optimal choices in complex multi-objective optimization problems. The implementation of this approach is made simpler through the use of existing open-source libraries. These include BoTorch [9], a Bayesian Optimization library built on PyTorch, and Ax [10], an

Adaptive Experimentation Platform, which provides higher-level APIs as well as scheduling, storage, and orchestration capabilities. Both BoTorch and Ax are actively developed by the vibrant data science community with Meta Open Source currently maintaining and leading the development of the code base.

The primary constraints of MOO and the measures to address them are mainly centered around four aspects: firstly, the issue of scalability: the model fitting process, typically utilizing a Gaussian process for the probabilistic surrogate model, escalates at a rate of $O(n^3)$, where n represents the number of data points. The quality of the model and its statistical efficiency degrade with an increase in parameters. Additionally, the hypervolume of the configuration space is super-polynomial in relation to the number of objectives. However, there are promising approaches, such as one based on a sparsity-inducing prior and Markov Chain Monte Carlo inference, designed to address high-dimensional problems where a few parameters exert a significant influence [11]. Secondly, the region of interest: The efficiency of the model can be improved by defining appropriate parameters like objective thresholds, in the regions of interest in the objective functions. To this end, a system is currently being developed known as MORBO (Multi-Objective Trust-Region Bayesian Optimization) [12], implemented using BoTorch. The aim of MORBO is to increase efficiency scaling for many evaluation points by optimizing various parts of the global Pareto frontier simultaneously using a coordinated set of local trust regions. Thirdly, the issue of noise: The model needs to be designed in a way that it can handle noisy data, including intrinsic tolerances and environmental fluctuations. Incorporating this flexibility would likely lead to more realistic and robust optimization outcomes. To optimally utilize noisy data, a MARS (Modified Multivariate value-at-risk Approximation based on Random Scalarizations) approach is currently being developed [13], using BoTorch. Lastly, the matter of data representation: To mitigate ill-conditioned linear systems, a minimum of double precision is recommended. The handling of discrete parameters can be accomplished through probabilistic continuous reparametrization.

The latest implementation of detector design optimization at EIC [7] draws inspiration from the successful pilot attempt on the dual-radiator RICH (dRICH) [5]. It harnesses the power of the Multi-Objective Evolutionary Algorithm (MOEA) and Bayesian Optimization (MOBO) libraries, integrating them with the computationally intensive Geant4-based full simulations to facilitate the ECCE tracker design [3]. This framework incorporates approximately ten design free parameters and three key objectives, subject to a variety of hard and soft constraints. It also deals with the complex requirement of preventing Geant4 volume overlaps. Key facets of the objective functions include momentum and angular resolutions, as well as the efficiency of tracking reconstruction via Kalman filtering. The results of the optimization have been verified by comparing them with the expected baseline performance and post hoc reconstructed physics observables, such as $D^0 \to \pi^+ K^-$ invariant mass reconstruction. The optimization is currently in the process of transitioning from the original ECCE software framework to the more advanced ePIC software framework. This transition aims to expand the AI-assisted design to accommodate a larger parameter space and include multiple sub-detectors (e.g., tracker, PID detectors such as the dRICH, and calorimetry) in the optimization process, along with a broader set of objectives. A significant advantage of this approach is that of utilizing accurate full simulations while limiting the number of design points necessary to approximate the Pareto front in a multi-objective space.

In EIC, an alternative Machine Learning-driven approach has been introduced [14] for calorimetry design application. This approach substitutes the computationally demanding Monte Carlo simulation with an efficient surrogate model. The surrogate model utilizes generative frameworks such as Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and Normalizing Flows (NFs) [15].[2] This results in a differentiable simulation, in which minor perturbations are approximated using a first-order Taylor expansion. In the final step, the optimal detector parameters are identified through a Gradient Descent optimization process, assuming the use of suitable metrics.

A significant portion of the discourse focused on the incorporation of cutting-edge data science tools that enable an interactive visualization of solutions within a multifaceted Pareto front. This front, which exists in a multidimensional objective space, consists of a spectrum of optimal solutions with various trade-offs between competing objectives. This ability to visualize solutions allows for a more intuitive understanding of these trade-offs and assists decision-makers in selecting the most suitable solution based on their specific preferences or constraints. An illustration of these applications, as they allow for an interactive exploration of this space, can be found in Fig. 4. These tools, therefore, represent a critical step forward in managing the complexity of optimization problems in detector and accelerator design.

Besides detector design, the AI/ML techniques and optimization methodologies we discussed hold significant relevance in accelerator science, particularly in advancing the optimization of accelerators. This applicability was a key part of our discussions during the workshop. Particle accelerator optimization poses numerous challenges, primarily stemming from the necessity to navigate non-linear, multi-objective functions that depend on thousands

[2] For additional details on surrogate models, please refer to Sect. 4

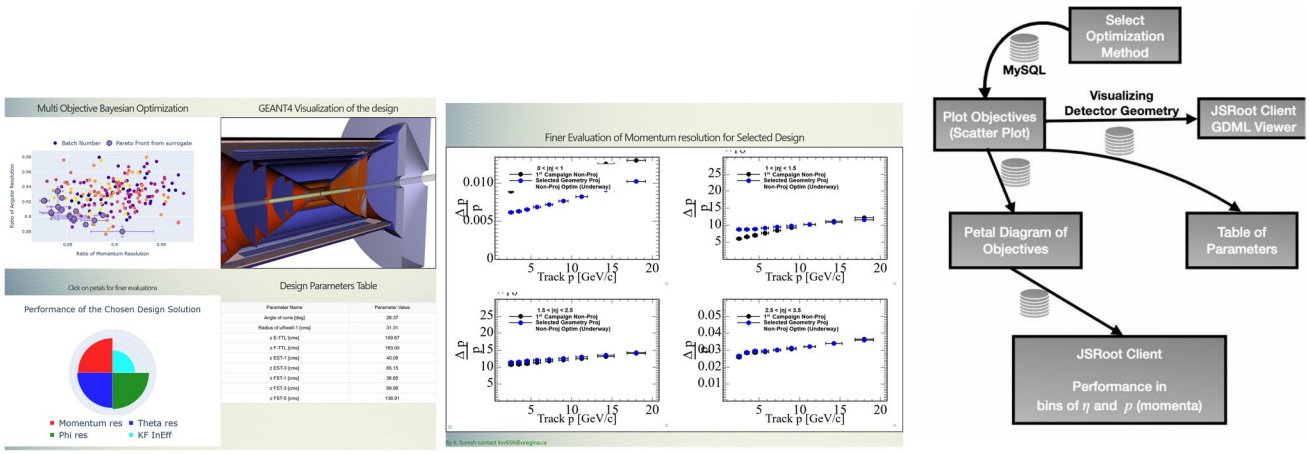

Fig. 4 Interactive Pareto front from AI-assisted design. Left panel: interactive visualization taken from the website [16] showing the performance of a chosen design solution. Middle panel: a fine grained view of the momentum resolution of the design solution in various phasespace bins. Right panel: a schematic of Python and JavaScript libraries facilitating result visualization utilizing advanced data science tools. The app was developed as part of the tracker optimization work [3]

of dynamic machine components and settings. These factors collectively impact the design, operation, and control of particle beams, and often exceed the capacity of traditional optimization methods. However, recent advancements have yielded promising results.

Multi-Objective Genetic Algorithm (MOGA) is a prevalent method used for the optimization of components such as Superconducting Radio Frequency (SRF) guns. Despite its regular use, MOGA still necessitates human intervention in scenarios involving parametric singularities, and the lack of harmony between the myriad approaches in use could potentially limit its overall efficiency and the fluidity of the optimization process. Moreover, the concept of virtual or digital twins has been gaining significant traction due to its ability to generate datasets with minimal effort for the testing and training of AI/ML models, operator training, and as a natural expansion of control room online modeling. This extended exploration capacity could pave the way for the design of innovative solutions for particle acceleration in the near future.

Recent advances hold potential for future accelerator design. These include algorithmic improvements in linear algebra [17] and non-linear/chaotic system forecasting [18, 19], which could significantly influence accelerator surrogate models for non-linear design. However, the impact of these emergent technologies is perhaps not yet robust enough for application to the ongoing EIC accelerator design within the project's timeline.

More standard techniques such as decision tree-based methods have been successfully implemented to enhance Large Hadron Collider (LHC) operations, resulting in improved luminosity through more efficient beam optics control [20]. Techniques such as Isolation and Random Forests have proven effective for instrument fault detection, as well as identification and correction of magnet errors. These applications not only uncover previously undetected hardware and electronics issues, but they also conserve operational time through early detection. Further, autoencoder Neural Networks (NNs) have been employed to de-noise beam measurements on simulated data, leading to an anticipated improvement in measurement quality. Additionally, the use of supervised learning with linear regression models for virtual diagnostics enables the reconstruction of optics observables without direct measurements, potentially accelerating machine commissioning and mitigating the need for time-consuming measurements.

These successful applications have spurred ongoing research for the design and optics corrections in the LHC upgrade, which could potentially be adapted for EIC or inspire new advanced methodologies for collider operations.

In conclusion from the design session of the workshop, it was agreed that EIC is poised to greatly benefit from the application of AI in the control, commissioning, monitoring, and operation of accelerators and spectrometers. It was stressed that recognizing and integrating these opportunities early in the design phase is crucial.

Intersection Between Theory and Experiment

ML techniques have long been successfully employed as data analysis tools within the realm of experimental nuclear and particle physics. However, when it comes to

theoretical or phenomenological perspectives, we have yet to fully explore the potential these techniques offer.

In the context of QCD, ML seems particularly adept at handling non-perturbative phenomena. This includes initial state parton densities (in a broader sense), as well as the final state hadronization process. These complex aspects of QCD could greatly benefit from a comprehensive application of ML. Using QCD factorization theorems and evolution, parton distribution functions (PDFs) and other quantum correlation functions are deduced by conducting global fits on available data. This conventional method involves probing various regions in the parameter space to find the best location, given a specific objective. The ideal parameters are pinpointed by minimizing (or maximizing) a particular cost function, typically through the gradient descent algorithm. To mitigate the influence of parametrization bias, the NNPDF collaboration introduced the application of NNs in extracting collinear proton PDFs (as referenced in [21]). Later, they extended this approach to fragmentation functions (FFs), as detailed in [22]. This innovative use of NNs serves to refine our understanding and analysis of pQCD.

Inspired by their success, several groups have attempted to exploit the flexibility of the NNs to determine more complex, higher dimensional distributions [23, 24]. As a concrete example, we discuss the benefits and challenges of using NNs to extract generalized parton distributions (GPDs) as perused by the FemtoNet Collaboration [24].

The current knowledge of GPDs lags far behind that of collinear PDFs due to their dependence on additional kinematic variables, sparse kinematic coverage, and the overall amount of data being limited. Moreover, in the deeply virtual exclusive scattering processes of interest for GPDs [25], the cross-sections are written in terms of convolutions of the GPDs over one of the kinematic variables. Effective hadron tomography requires incorporating multiple exclusive processes, such as deeply virtual Compton scattering and deeply virtual meson production, which are essential for accessing GPDs, and presents challenges in modeling and fitting procedures due to the complex internal structure of GPD functions and their correlation with observables [26].

The workflow of Fig. 5 depicting the FemtoNet global analysis framework is a response to this challenge [24]. The main goal is to establish an unprecedented precision analysis framework to characterize the quark–gluon structure of matter. Throughout the analysis pipeline, specialized deep learning architectures informed by physics are applied to guarantee compliance with crucial physics constraints. This process intentionally integrates physics insights from multiple sources like theory, lattice QCD, and potential higher twist or beyond-standard-model

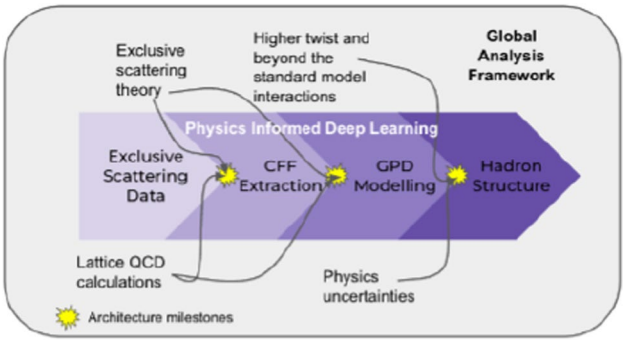

Fig. 5 FemtoNet global analysis workflow: a physics-informed deep learning framework that translates exclusive scattering data into insightful information

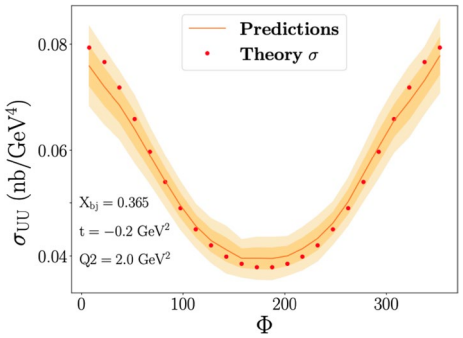

Fig. 6 FemtoNet results on DVCS cross-section modeling: DVCS extrapolation on kinematics outside the range covered in experiment at the kinematic point $x_{Bj} = 0.365$, $t = -0.2GeV^2$, $Q^2 = 2GeV^2$, and $E_b = 5.75GeV$. ML model with Angle Symmetric Constraints. Figure and caption taken from [24]

interactions. Ultimately, this pipeline will be extracting vital information regarding hadronic structure.

As a first step in this analysis to determine the GPDs, the FemtoNet collaboration applies supervised learning utilizing a multi-layer perceptron (MLP) complemented with regularization techniques, namely “dropout", to prevent over-fitting [24, 27]. These studies are then enhanced by the integration of physical information into neural architectures, which guarantees effective generalization. In practice, this involves augmenting the loss functions with additional terms to penalize deviations from theoretical knowledge, reinforcing the model’s adherence to expected behaviors. The performance of the physics informed NN model cross-section prediction is shown in Fig. 6 and its superior performance against its non-physics-based counterparts has been presented in [24].

A pivotal subject of discussion was Monte Carlo event generators (MCEGs) [28], indispensable tools for numerical simulations and subsequent data analysis. MCEGs play

a crucial role in high-energy and nuclear physics, being essential for model validation, facilitating discoveries, experimental planning, and further advancement of theories such as QCD. Their enhancement is geared towards the improvement of prediction accuracy and computational efficiency. Traditionally, non-perturbative aspects rely on phenomenological models. These models, in turn, depend on numerous parameters that are derived from experimental data. An integral facet of the MCEGs is the modeling of hadronization, a transformative process wherein high-energy, color-charged quarks and gluons morph into color-neutral hadrons. Gaining insight into the 'hadronization' process, or the mechanism by which these particles reconfigure into their final state, is crucial for establishing significant comparisons between data and theoretical predictions and improving our understanding of the hadronization process. Due to the curse of dimensionality [29], high-dimensional data often necessitate the use of more complex models to effectively capture the relationships between features. Therefore, it is unsurprising that current models cannot completely encapsulate data across the entire energy spectrum explored. This raises the question: could Machine Learning present a more effective approach?

Presently, the community is exploring three main machine learning strategies for hadronization studies: VAEs, GANs, and NFs [30–32].

VAEs have been deployed to emulate a simplified version of the Lund string model in Pythia, with the assumption of flavor and kinematics of hadron emission being independent [33]. In particular, the conditional sliced-Wasserstein Autoencoder (sWAE) has been presented; it was trained using kinematic distributions for variables such as p_z' (a rescaling of p_z), p_T extracted from Pythia's first emission events (refer to Eq. (2) of [33]), and specific values of string energy. The network was tested using a unique set of string energies not included in the training set. This strategy facilitated a more accurate assessment of the network's ability to generalize across the entire phase space. To simplify the process, only pions in the final state were considered, and the performance was compared with the average Pythia output. This methodology conveniently allows the inclusion of an energy dependence in the hadronization process, if required by the data. The first hadron emissions, which form the basis of the training, are successfully reproduced (refer to Fig.10 in [33]). Comparison with the full hadronization chain (see Fig.11 in [33]) shows a deviation of no more than 10%; such differences originate from the different treatment of the first and subsequent emissions in Pythia which is not considered in the ML approach. While the architecture was applied to a simplified version of the Lund string model, the results are promising and the use of ML is foreseen to be more relevant once training is done on real data, for which the hadronization is not physically accessible.

GANs, instead, were used to learn the cluster decay of the cluster hadronization model using Herwig data [34]. Differing from VAEs, which learn mappings for both encoding and decoding, GANs learn only the decoding from a base distribution utilizing a discriminative loss function, comparing generations with ground truth. This was done for single $e^+ + e^-$ annihilation into two π^0. Despite the simplifications introduced for faster training, it was found that the method generalizes to other hadron species and, even more importantly, that the level of discrepancy with real data is similar to the one achieved with the original cluster decay model.

NFs have been used to further improve the generation scheme, utilizing a Conditional Masked Autoregressive Flow (CMAF) [35] as the generation mechanism for the kinematic [36]. The network is conditioned on a set of hadron masses with differing initial energies. Contrary to earlier methods that restricted consideration to pions alone, the introduction of functional dependence via the hadron mass condition enables the generation of a range of masses in the final state [36]. Additionally, the conditional flow adeptly captures the correlation between the p_T and p_z kinematic distributions.

On the experimental side of connecting theory to experiment, we have identified four major challenges. The first of these challenges pertains to *fast simulation*, a suite of tools designed for the swift transition from particle-level predictions to detector-level observations. Significant progress has been made in ML-based fast simulations, particularly with the advent of 'surrogate models'. These models leverage various deep generative modeling approaches, including VAEs [30], GANs [31], NFs [32], and Diffusion Models (DMs) [37]. Much of the community's attention has been devoted to the simulation of calorimeters, which typically form the slowest segment of the simulation stack [38–41]. Calorimeters, featuring both longitudinal and transverse segmentation, offer a high-dimensional emulation space. Despite the complexity, the latest neural network models have managed to mimic GEANT4 simulations [42] with impressive accuracy [43].

The second significant challenge lies in *reconstruction*. Traditional shallow learning has long been employed for tasks such as momentum reconstruction and particle identification. However, the advent of DL has ushered in innovative methods that process low-level inputs in a more comprehensive manner. Furthermore, ML continues to influence even the most foundational tasks in data analysis. The reconstruction of the kinematic variables in DIS such as Bjorken x and four-momentum transfer squared Q^2 is being reevaluated in light of the advancements made in ML. It has been shown for inclusive DIS measurements that the reconstruction methods benefit from the application of ML-based models [44, 45]. This is now studied further for semi-inclusive DIS where it is important to precisely

determine the inelasticity y of the process and the azimuthal angles of the final state.

The third critical challenge is tied to *parameter inference*. This aspect distinctly differs from reconstruction, which focuses on deducing properties of an individual event or a single object within that event. In contrast, parameter inference is concerned with extracting physical parameters from entire datasets, thereby providing a holistic understanding of the reaction or system under consideration. Such a nuanced analysis is invaluable in scenarios with complex or high-dimensional datasets, where conventional statistical methods may struggle. For example, recently, DL have been used in the search for exotic hadrons where production and decay parameters may be determined from models built on the underlying quantum mechanical amplitudes [46, 47]. By leveraging ML, we can uncover intricate correlations and patterns that might otherwise remain hidden. This makes ML an indispensable tool for parameter inference in the modern data-centric world.

Finally, the fourth challenge pertains to *cross-section inference*. For a vast array of measurements, experimental teams provide corrected differential cross-section results in a readily usable format for subsequent inferential tasks outside the scope of the originating collaboration. ML is precipitating a paradigm shift in how we execute these corrections, commonly known as deconvolution or unfolding. Cutting-edge methods have facilitated the unfolding of high-dimensional and unbinned data [48–50]. This development is paramount to effectively harness EIC data, given that intricate correlations across numerous dimensions are necessary to comprehensively analyze the three-dimensional structure of the proton.

Furthermore, we highlight two frameworks that amalgamate theoretical and experimental aspects, and include uncertainty quantification: the A(i)DAPT group [51, 52], has introduced an innovative employment of ML-based MCEG for data analysis and preservation. Its objectives encompass data compression, providing powerful interpolation tools, and the ability to unfold detector effects, enabling the acquisition of accurate vertex-level data. Additionally, the framework incorporates a GAN-based surrogate model for rapid detector folding, as demonstrated in [53]. Successful testing and validation of this framework, along with its potential to mitigate theory bias during the inference of event distributions, represent a significant advancement towards the reconstruction of physical observables. The QuantOm collaboration [54], has presented another pioneering approach that adopts a holistic strategy for global analysis, seamlessly integrating theoretical and experimental components. By employing an event-based analysis methodology, this approach capitalizes on generative models such as GANs to establish an event-level Quantum Correlation Function (QCF) inference framework. This framework provides a comprehensive and advanced perspective on 3D hadron tomography and nuclear imaging.

In conclusion, the increasing implementation of machine learning in key areas that connect theory and experiment highlights its capability to address complex challenges in nuclear and particle physics. The effectiveness of these ML applications largely depends on the development of robust and flexible techniques, capable of adapting to the demands of a rapidly evolving scientific landscape.

Reconstruction and Particle Identification

Particle identification and reconstruction are crucial components of physics analyses at the EIC, and the integration of AI and ML into these areas is rapidly advancing. This integration offers significant potential for enhancing performance and fully leveraging detector information, moving beyond traditional methods. The discussion during the second workshop encompassed four key-topics and highlighted aspects directly relevant to the current endeavors at EIC:

(i) *Reconstruction and PID.* Calorimetry is a pivotal activity of the ePIC detector at EIC; in [55] we discussed about muon identification with deep learning, showing that modern DL-based architectures can efficiently combine the information coming from the tracking and calorimetry sub-systems and learn how to distinguish charged μs from charged πs, the latter representing the main background source. Another contribution coupled the reconstruction of shower profiles within the hybrid barrel calorimeter in ePIC, integrating sandwiched layers of monolithic silicon sensors AstroPix and Pb/ScFi fibers, with a CNN for PID assessment of these profiles [56], as illustrated in Fig. 7. The key finding of this work is that the utilization of the AstroPix technology with Pb/ScFi and the integration of deep learning algorithms resulted in the most effective reconstruction performance, surpassing other setups that employed various detector technologies and more conventional reconstruction algorithms. Specifically, this configuration enabled superior e/π separation, precise γ and π^0 differentiation, radiative γ tagging, and low-energy μ identification, impacting multiple areas of the EIC physics, such as DIS, DVCS, QED internal corrections, J/ψ and TCS (cf. Abbreviations). Another study delved into the application of ML for pixelated calorimetry, specifically for cluster separation in the electron endcap [58] of ePIC. Different AI/ML techniques have been evaluated, with a particular focus on using VAEs, which have been leveraged on a full-scale simulated calorimeter to condense clusters into single points representing their total energies. Furthermore, another aspect that has been highlighted in our discussion is the use of interpretable networks to have

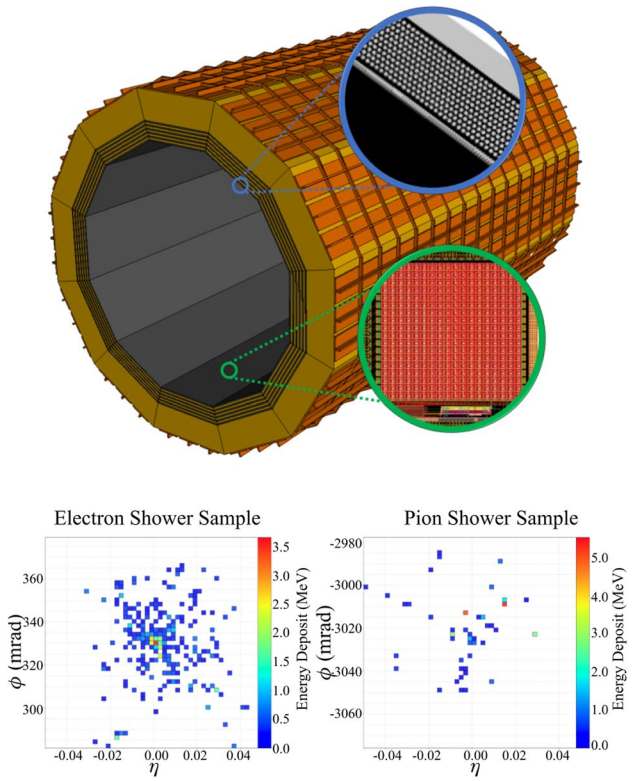

Fig. 7 (top) ECal hybrid concept: the barrel hybrid electromagnetic calorimeter concept for EIC. More details can be found in [57]; (bottom) projection of showers in the ECal: shower projections of electrons (left) and pions (right) as a function of ψ and η. Energy deposition in the pixelated array is represented via color, commonly occupying the channel axis in vision-based neural networks

a more solid interpretation of the results obtained. In particular, we discussed the example of lepton identification amidst jet-induced backgrounds [59] using electromagnetic and hadronic calorimeter information. These studies underlined how deep learning, like CNN, can uncover overlooked low-level image data and isolate novel high-level features, outperforming traditional high-level feature physics models.

In addition to calorimeters, Cherenkov detectors are integral to the ePIC detector's PID system, acting as the backbone of PID for EIC experiments [60, 61]. The ePIC PID system is equipped with a Ring Imaging Cherenkov (RICH) detector in the electron endcap, a dual-radiator Ring Imaging Cherenkov (dRICH) in the hadronic endcap, and a Detection of Internally Reflected Cherenkov light (DIRC) in the central barrel. This setup ensures superior PID capabilities across a broad range of the particle phase space [1]. When it comes to Cherenkov detectors, there are two challenging areas. The first pertains to the simulation of these detectors, which typically demands significant computational resources. This is because the process involves tracking a substantial number of photons across complex surfaces, as illustrated in the contribution [60]. The second area of challenge lies in the reconstruction process, specifically in recognizing patterns of sparse ring images amid noisy conditions. This complexity is further exacerbated in the context of DIRC detectors due to the intricate ring topologies. In addressing these challenges, advancements in ML and DL present considerable promise for enriching the state of the art in both reconstruction and PID in relation to Cherenkov detectors [62]. As discussed in [61, 63], algorithms such as DeepRICH [64] leverage generative models to offer rapid, accurate simulations. Additionally, as demonstrated in the context of the DIRC detector, these algorithms are capable of reconstructing intricate hit patterns, with performance on par with traditional reconstruction methods, but at a significant speed—roughly four orders of magnitude faster during inference time on a graphics processing unit (GPU). Furthermore, we highlighted the substantial opportunities presented by ML/DL applications, which enable learning at the event level as opposed to the particle level. This approach not only leverages the additional information characterizing each event, but also effectively manages the simultaneous detection of multiple particles within the detectors. This shift in focus coupled with the possibility to train these models on high-purity real data, can lead to deeper understanding of the detector response.

(ii) *Tracking*. Regarding AI/ML for tracking at the EIC, we extended the discussion initiated in the first workshop [65, 66], taking cues from the forthcoming upgrade of the LHC to the HL-LHC. Despite the proficiency of existing track reconstruction algorithms based on Kalman filters, they encounter scaling issues with increased data volumes. This necessitates active research into new or enhanced algorithms, involving accelerated hardware application of existing Kalman filters, the integration of ML techniques, and the creation of complete ML-pipelines for tracking like those proposed by the Exa.TrkX project [67]. Also, A Common Tracking Software (ACTS), a new algorithm test bed for track reconstruction research, was highlighted [68], and in the second AI4EIC workshop, we decided to delve deeper into this topic. ACTS is an agnostic, open-source tracking toolkit [69]. Written in C++, ACTS streamlines the entire track fitting process and provides an example framework with Python bindings. Its utilization spans various experiments, like ATLAS, ALICE, sPHENIX, and EIC studies. ACTS serves as a comprehensive tool for developing and testing new ML-based tracking algorithms, making it crucial for current EIC advancements. It also offers an open data detector (ODD) for algorithm benchmarking and ML tracking tests. Noteworthy tools available in ACTS include hashing for hits selection, parameter auto-tuning, and Graph Neural Network (GNN) for track finding. Regarding GNN for tracking, we had discussed the deployment of GNN in a streamlined pipeline for trigger-background event classification in both sPHENIX and

EIC and its implementation on Field Programmable Gate Arrays (FPGAs) [70, 71]. This subject is expanded further in Sect. 7.

(iii) *Jet reconstruction/tagging.* In the realm of jet classification (for a comprehensive review on this topic, the reader may refer to, e.g., [6, 72]), a novel approach, JetVLAD [73, 74], was presented as an application for tagging heavy-flavor jets at RHIC. JetVLAD employs vectors of locally aggregated descriptors (VLAD) to tag heavy-flavor jets, proving instrumental for examining jet interactions with the quark–gluon plasma (QGP) created in high-energy heavy ion collisions. Such interactions are crucial for understanding partonic energy loss within the QGP medium. The JetVLAD architecture, mirroring the ResNet model family [75], uses residual blocks with batch normalization to simplify learning. The model's width was designed to match the output of the NetVLAD layer [76], a CNN architecture created for weakly supervised place recognition. This innovative approach efficiently identifies jet flavor, enabling the analysis of mass dependence in jet-QGP interactions, and sets the stage for high-purity heavy-flavor measurements in contemporary and forthcoming collider experiments like EIC. In the first workshop, we explored the potential of AI/ML in enhancing heavy-flavor and jet tagging in EIC experiments [77]. This insightful presentation emphasized the critical role of merging low-level and high-level track/calorimeter data for the efficient identification of jets or heavy-flavor states, and showcased several effective examples of such implementations drawn from LHC studies, like the possibility of simultaneous estimation of b jet energy and resolution [78]. In the discussion concerning AI/ML applications for jets, it was mentioned a manuscript published concurrently with the workshop in October 2022 [79]. Notably, this work delves in the usage of out-of-jet radiation information, incorporates infrared jet flavor definition for handling non-perturbative QCD effects, and underscores the potential for training such deep learning models with real data. The integration of deep learning for jet analysis could profoundly impact EIC research by reinforcing constraints on transverse momentum-dependent PDFs, augmenting sensitivity to transverse single spin asymmetry, and elucidating cold nuclear matter effects. More comprehensive information can be gleaned from the manuscript [79].

(iv) *Data-driven techniques.* When designing ML models, it is often convenient to train and/or test models utilizing simulated data. Simulated data provide high-purity samples in which it is possible to correctly tag each detector candidate given ground truth information. However, when the model is deployed on actual detector response variables, it is assumed that the two data schemes are exact matches, and thus a bias can be introduced. Figure 8 shows an example in which the target domain (data) does not match the source domain (MC) for the invariant Λ^0 mass spectrum.

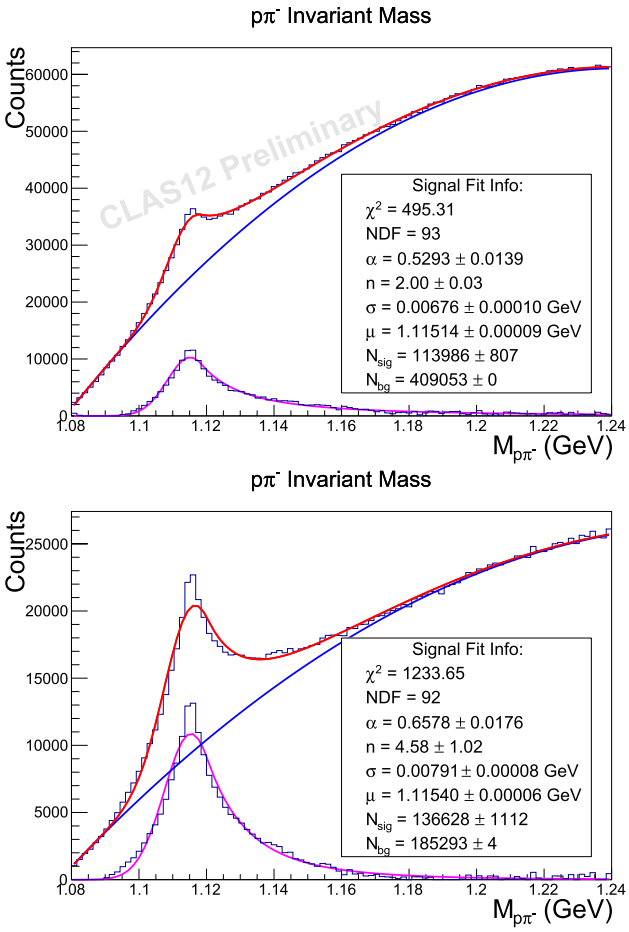

Fig. 8 Comparison between data and simulation at CLAS12: Invariant mass spectrums for the Λ^0 for data (top) and MC (bottom). Notice the distinct differences in the shapes of background distributions. Domain adaptation attempts to overcome this via training the GNN with an adversarial loss between the two data formats. Figure taken from Ref. [80]. Original figure available under https://creativecommons.org/licenses/by/4.0/legalcode

To bridge the gap between the target and source domains, domain adaptation is employed [81]. This machine learning technique modifies a model tailored for a specific task to function efficiently in a related but different domain. In [80], the authors explored the use of Graph Isomorphism Networks (GIN), a form of GNNs known for maintaining injective functions, for Λ-event tagging at CLAS12. By applying adversarial adaptation, they effectively mitigated the discrepancies between simulation-based training and real-data deployment, as discussed in [80, 82]. Information learned from the MC samples should not be disregarded but rather adjusted given the transition to data. After training with an adversarial network with the goal to distinguish between data and simulation, the output distribution of the network becomes significantly more similar as shown in Fig. 9. More details can be found in Ref. [80]. In addition to the domain adaptation previously discussed, recent

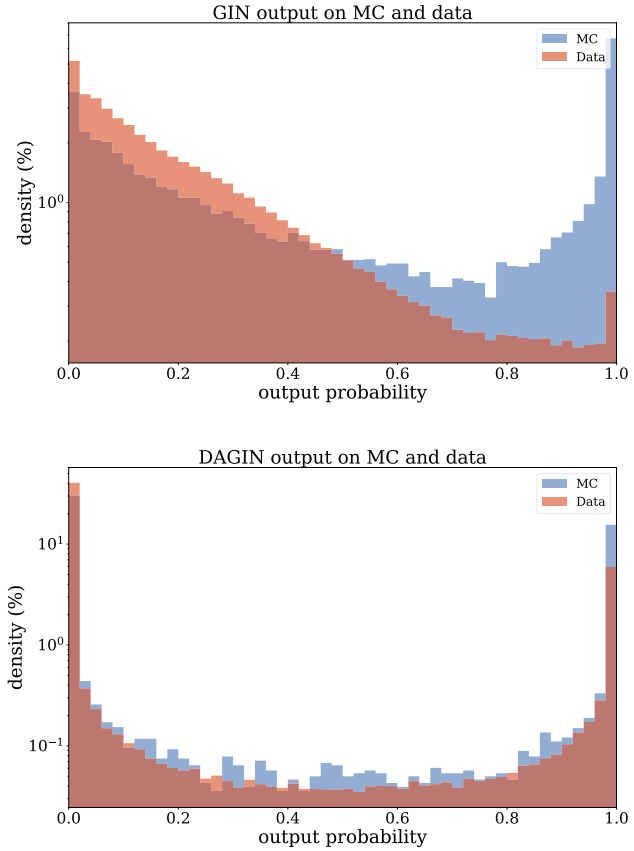

Fig. 9 Comparison between regular GIN and GIN with domain adaptation (DAGIN) for data and simulation at CLAS12: The output of the regular GIN (top) shows significant differences between data (blue) and MC (orange). In comparison, the distribution of the outputs for the DAGIN (bottom) come similar for data and simulation with a Kolmogorov–Smirnov distance for the GIN. Figure taken from Ref. [80]. Original figure available under https://creativecommons.org/licenses/by/4.0/legalcode

developments in ML/DL are enhancing the progress of data-driven techniques for applications such as anomaly detection [83]; these methods often rely on training with high-purity real data samples, provided they are available. In [84, 85], the authors presented a strategy called 'Flux+Mutability', which is based on a combination of a conditional autoencoder (cAE), a cMAF, and Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) for one-class classification and anomaly detection. This method has been employed for both γ/n shower classification in the GlueX barrel calorimeter—which presents similarity with the ePIC barrel calorimeter—and detection of potential beyond Standard Model (BSM) di-jet signatures at the LHC. The F+M algorithm, trained using a single reference class, leverages cAE to filter anomalous events, providing reconstructed features and residuals. The cMAF, fed with these features, generates data for forming a reference cluster, facilitating object-by-object fitting relative to the reference cluster via HDBSCAN. Objects are then labeled using a quantile cut, ensuring class-agnosticity.

Infrastructure and Frontiers

One of the biggest challenges currently facing the EIC is the design and development of future-proof infrastructure, viable both currently and in the next decade when data collection commences. Furthermore, any infrastructure developed should also be modular enough to change during the lifetime of the experiments at the EIC, which are expected to be several decades. Designing and deploying a modular computing infrastructure is therefore essential, as well as defining interfaces between data processing stages such that when new technologies become available, pieces of the overall infrastructure can be updated without disrupting the entire workflow. Lessons can be learned from the LHC, whose computing infrastructure was designed nearly two decades before the accelerator facility began operation. Frameworks did not necessarily just stop working at the LHC as the facility moved farther into the operations phase but rather became inefficient at using the available resources as those resources changed over time which then necessitated changes in the overall infrastructure [86]. With a sufficiently modular design, pieces that become inefficient could be replaced by new efforts that, for example, take advantage of GPU architectures that were not envisioned to play a large role at the time of design. It is also important to consider the role of technologies that are in the early stages of application towards High Energy Physics (HEP) and Nuclear Physics (NP) workflows, such as quantum computing. Continuously checking in and scheduling reviews of the state of technologies, for example when some milestone has been reached, will help assess how applicable they are for workflows at the EIC. Scheduling these "check ins" regularly, and starting them early, will help prepare for their possible integration. As an example, a few decades ago GPUs were not expected to be as computationally valuable as they are currently; therefore, it is essential to remain proactive in evaluating emerging technologies given the timescale of the EIC.

Often times, when designing computing infrastructure, only the hardware and associated software are considered during framework development. However, it is also important to consider the workforce, specifically, how to develop and retain the people necessary to successfully design and implement a computing infrastructure that will serve the EIC science program for its entirety. Building a diverse and interdisciplinary team will help bring technical expertise from computing and physics domains necessary for hardware, algorithm, and physics development. The EIC is a facility that is poised to develop such collaborations due

to the size of the project and the necessary cross-cutting challenges that must be overcome for its success. Large collaborations, such as those at the EIC, can provide a platform for approaching difficult computational problems; as an example, the Worldwide LHC Computing Grid was created to address the challenges of data collection and processing at the LHC [87]. To develop an interdisciplinary team, connections need to be forged, commonly generated through conferences and workshops. At forums such as these, scientists from a variety of domains are able to discuss approaches to the same problem from the different perspectives their expertise offers. Developing a computing infrastructure that can serve the EIC must include hardware, software, and an interdisciplinary team that is capable of designing, implementing, and maintaining the infrastructure needed to serve the lifetime of the EIC project.

Artificial intelligence use cases are indeed one of the primary drivers for developing or utilizing new computing infrastructure for the EIC. For example, many scientific domains have developed the foundation for including high performance computing and next generation architectures into their workflows. Similarly, efforts are being made to push ML models closer to the edge of experiments, such as with FPGAs [88–90]. The utilization of such hardware requires networks with low computational overhead, in terms of both memory and required floating point operations. Research and development is ongoing to integrate these, and other, new infrastructures into EIC workflows. Therefore, in assessing the infrastructure essential for the EIC's software and computing requirements, it is crucial to factor in the rapidly evolving domains of AI/ML and data science.

Considering this, the workshop prominently featured discussions on foundation models (see, e.g., [91–93]), an ascending and increasingly influential field in AI. The emergence of Generative Pretrained Transformers (GPT) [94–96] has offered new potential within the realm of AI for the EIC. With its capability to understand and generate human-like text based on the context provided, GPT models can be pivotal in data interpretation, document generation, and even hypothesis formulation for EIC science and NP at large. This deep learning-based model can sift through large amounts of data, detect patterns, and identify key insights faster and more efficiently than traditional methods, driving further advancements in the field. At the AI4EIC workshop in October 2022, a month prior to the release of chatGPT, the potential of foundation models in nuclear and particle physics was underlined. The advent of chatGPT further emphasizes this potential, illuminating a promising future where AI tools like GPT can accelerate scientific discovery by automating and enhancing various facets of research in the EIC community. Noticeably, the 2023 AI4EIC hackathon's theme was influenced by the advent of GPT [97], and proposed a physics event classification problem using large language models.

Streaming Readout

Streaming Readout (SRO) is rapidly becoming the go-to paradigm for readout processes in contemporary nuclear and high energy physics experiments. Unlike traditional or pipelined methods that rely on hardware signals for initiating data conversion into the digital realm or marking time regions of interest within close-memory buffers, an SRO data acquisition system seamlessly converts and streams detector data to potentially heterogeneous computing systems. The retention of data is determined by software, with possible acceleration by FPGAs or Application-Specific Integrated Circuits (ASICs). Figure 10 provides a conceptual overview of a potential system configuration.

The SRO scheme promptly handles all detector information in digital form, paving the way for AI-powered tagging and filtering algorithms to be employed early in the data collection stage. By making raw data accessible to high-level reconstruction frameworks-typically written in languages such as C++, Python, or Java-SRO allows for the utilization of standard AI tools without necessitating bespoke adjustments for dedicated hardware. A wide array of system scales and implementations exist, ranging from systems that record all data to disk, to those that conduct high-level analyses while data are in transit, only preserving high-level physics objects.

SRO has already been implemented in numerous experiments at the LHC (see, e.g., [98–101]) and has been officially designated as the chosen paradigm for the EIC, as evidenced in the EIC Yellow Report [1]. In the case of the sPHENIX experiment, the conventional triggered readout system is augmented with a streaming system for principal detectors, a strategy that permits the exploration of physics phenomena that would otherwise be missed by a triggered system. Similarly, various experiments at Jefferson Lab are experimenting with partial SRO solutions, thereby paving the way towards a comprehensive transition to a full SRO design [102–104].

The flexible data routing in a streaming readout system enables new or eases the implementation of various quality control and time-to-paper improvements. For example, the INDRA-ASTRA lab at Jefferson Lab is developing techniques to move analysis tasks into the readout [106]. ML has a role here, especially in automatic anomaly detection, for example by using the Adaptive Windowing (ADWIN) technique [107]. In general, streaming readout blurs the lines between online and offline analysis, with the goal to fuse these together as much as possible.

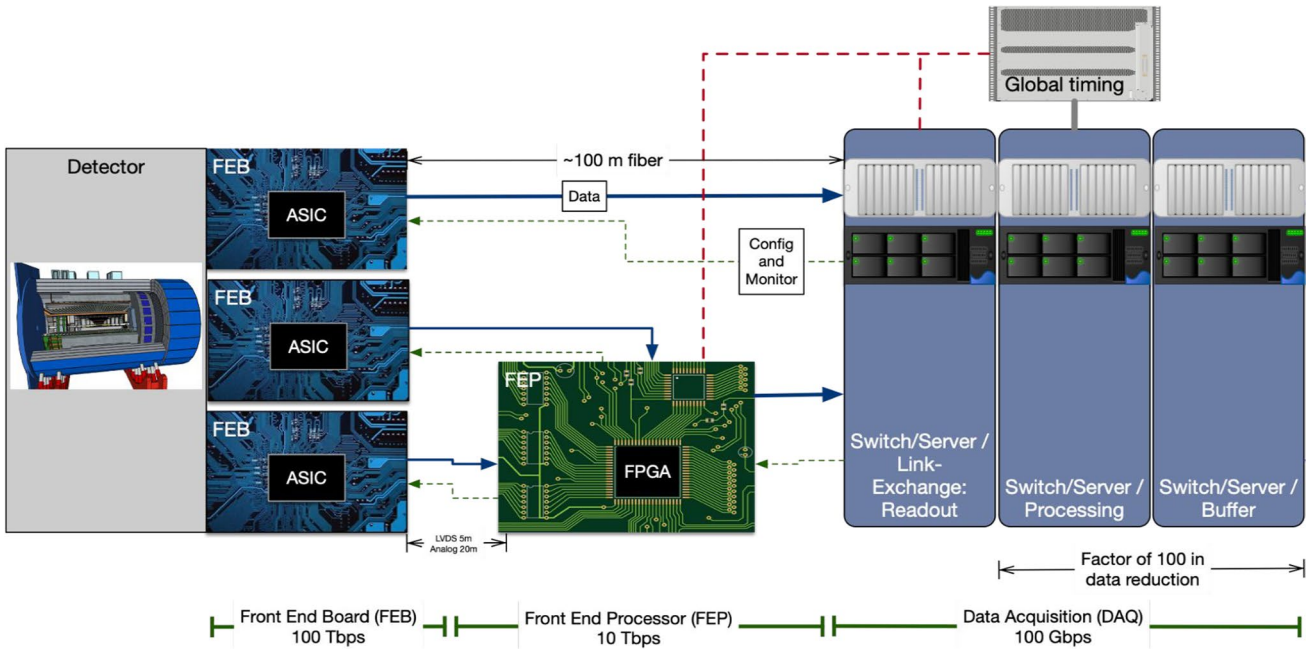

Fig. 10 Conceptual SRO DAQ System: the deployment of DAQ electronics is generally segmented by location, comprising the Front End Electronics (FEE) modules adjacent to the detector, the Front End Processor (FEP) boards for digitizing or reformatting detector data, and Stream Aggregator Boards (SAB) located in the hall for bundling streams, with online filtering and monitoring carried out in the counting room. For additional details, refer to [105]

Similarly, ML has been used for online calibration of the GlueX Central Drift Chamber (CDC) monitoring gain and time-to-distance conversion factors [108, 109]. Implementation of real-time (or quasi-) detector calibration is an essential component of SRO supremacy with respect to conventional triggered Data Acquisition systems (DAQs). HDBSCAN, a form of unsupervised hierarchical clustering detailed in Sect. 5, has been employed for clustering non-calibrated data from the CLAS12 forward tagger calorimeter in SRO mode [110] and reconstruct the electro-production of $\pi^0(\gamma\gamma)$. To take full advantage of the full off-line data reconstruction framework during data acquisition, raw data need to be calibrated and continuously monitored in order to provide reliable information to tagging/filtering algorithms. This request represents a great opportunity for AI-supported calibration and monitoring algorithms like those discussed in [108], where the AI system prototype deployed to control and calibrate the GlueX CDC provided good results, paving the way towards a self-calibrating detector.

Machine Learning, particularly GNNs as outlined in Sect. 5, is adept at managing hit and track identification, as showcased in [111]. This study also examines the use of Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs) for track fitting tasks. ML proves highly effective for noise and background suppression, decreasing the data volume that needs to be transmitted via the readout network and stored on disk. Such implementation yields the most significant impact when deployed early in the readout chain. Considering throughput requirements, there is a clear incentive to incorporate NNs on FPGAs. While packages like hls4ml [112] can aid in the implementation, it is noteworthy that not all network topologies are currently supported. Interesting design advancements have been demonstrated through the development of 'bicephalous autoencoders', which offer a lossy compression scheme that retains critical information while suppressing noise, and that can be even deployed on Intelligence Processing Units (IPU) as illustrated in [113, 114].

Current studies are also exploring the use of GNNs for heavy-flavor tagging and their implementation on FPGAs within the context of the sPHENIX project [70, 115]. The development of a real-time ML FPGA filter for particle identification and tracking in SRO is outlined in [104]. Generally, it is expected that future FPGA devices will include more Intellectual Property (IP) cores aimed at acceleration of NNs, for example by integration of matrix multiplication capabilities or higher number of DSP slices. However, the field will have to watch the developments closely. Our needs are not the driver for these developments, and it is unclear if the addition of these abilities will go hand in hand with a reduction in uncommitted resources required for data acquisition IP. This potentially presents a challenge, as these newly developed NN accelerator cores may not be compatible

with the specific data types required for our unique implementations.

ML also drives developments of new compute models like in-memory computing, with low latencies and very good energy consumption. It is clear that the field needs to further approaches, techniques and packages to ease the implementation of NN on multiple FPGA architectures over multiple generations and capabilities, and also to ease the transition from a Central Processing Unit (CPU)/GPU implementation onto an FPGA. This must include also verification tools. For a streaming readout system, orchestrating a considerable number of nodes is typically required. This circumstance introduces intricate challenges pertaining to system bring-up and configuration, thus necessitating the standardization of communication protocols. One such framework addressing these challenges is APEIRON [116, 117].

As already mentioned, beside the world effort driven by CERN experiments, a significant effort is undergoing at Brookhaven National Laboratory (BNL) and Jefferson Lab (JLab) to test components and concepts of a suitable SRO DAQ for EIC. Prototypes of a full DAQ SRO chain have been deployed and tested in both controlled (lab) and realistic (on-beam) conditions (see, e.g., [110]). Results are generally positive and even if current SRO schemes are not expected to be final, the experience gained by the EIC community is valuable for understanding limitations, requirements and opportunities of SRO at EIC.

Community Efforts

In the next decade, as the EIC reaches its operational phase, the impact of AI will be more pronounced than ever. Recognizing the transformative potential of AI, we have initiated a range of educational activities aimed at enhancing its understanding within the EIC community. These activities are intended to not only increase awareness, but also foster a culture of innovation and exploration centered around AI. One of our core community initiatives includes organizing hackathons designed around specific challenges pertinent to EIC. These hackathons serve as creative platforms for identifying and discussing promising strategies, architectures, and algorithms. By doing so, they present a unique opportunity to unearth solutions that could significantly bolster the EIC physics program.

In the following, we delve into the nuances of these community efforts and elucidate how they are instrumental in shaping the role of AI within the EIC.

Tutorials

The workshop incorporated a robust outreach and educational aspect, featuring a series of tutorials presented by esteemed AI and machine learning experts drawn from national laboratories, universities, and industry. Furthermore, a hackathon satellite event was organized, adding a practical element to the last day of the workshop. Four comprehensive tutorials were offered, each designed to impart knowledge on key topics in AI and machine learning. The subjects of these tutorials included multi-objective optimization with BoTorch/Ax, a technique of unfolding, the concept and applications of Graph Neural Networks, and the Machine Learning lifecycle. This educational component, by bridging the gap between theory and practice, played an essential role in enhancing the attendees' understanding and proficiency in these complex domains.

Multi-objective Optimization with BoTorch/Ax

Optimizing multi-objective problems is vital in particle detector and accelerator design, a critical progress area. Strategies that are both effective and resource-efficient are crucial. The implications extend beyond design, influencing various research areas within EIC that are dependent on optimization for enhanced performance and innovation. BoTorch [9] is a modular and highly customizable library for Bayesian Optimization with state-of-the-art algorithmic capabilities. Ax [10] exposes BoTorch's algorithms through a user-friendly interface and provides additional high-level management, storage, and orchestration capabilities.

In this tutorial, we go over some basic hands-on examples of how to use Ax to perform multi-objective Bayesian Optimization via Ax's Service API (an ask/tell interface) on a synthetic problem. This setup is straightforwardly adapted to any actual multi-objective black-box optimization problem with costly evaluations. The full tutorial is available here: [118] (slides), [119] (colab notebook).

Unfolding

Unfolding aims to correct measured observables for detector distortions and provide easy access to theoretical quantities for the broader nuclear and high energy physics community. Existing unfolding methods require the usage of histograms and are limited to low-dimensional inputs and outputs. Machine learning can naturally incorporate high-dimensional data to estimate the detector response, providing a more accurate estimation of the measured observable. In this tutorial, we introduce OmniFold [48], a machine learning-based method that simultaneously determines the unfolded response of multiple distributions. We present recent results of the application of OmniFold to particle collisions collected by the H1 Collaboration and provide hands-on tutorials on a toy example using normal

distributions as well as an example motivated by the EIC, unfolding the kinematics of leptons and hadrons in DIS. The Colab notebook is available here [120].

Machine Learning Lifecycle

The phases of the machine learning life cycle may be thought of as (1) data analysis, (2) experimentation, (3) model reproducibility, (4) deployment, and (5) production monitoring. This tutorial introduces the Machine Learning Operations (MLOps) open-source platform MLFlow [121] and describes MLFlow's four components to support the ML lifecycle with a specific focus on the MLFlow Tracking component, which is used to record and compare machine learning trials. The python library HyperOpt is also introduced, a library for hyperparameter optimization. The tutorial is contained within a Colab notebook and uses publicly available data from the 2021 Jefferson Lab hackathon when an imaginary calorimeter, with a single shower and no noise, was simulated. The problem is easily solvable with a simple neural network and is used only to illustrate the ease of implementing hyperparameter optimization and MLFlow tracking. First, users are guided through a grid search for a "best model" by creating different types of neural networks with different hyperparameters and tracking the results. Then comparing the results to determine the best-performing model. After users understand the few lines of code needed to implement the grid search implementation of MLFlow Tracking they are introduced to the HyperOpt python library, the concept of the "search space", the "minimization function", and how to combine these concepts into the model's training function and how to track the best performing hyperparameters using MLFlow. The Colab notebook is available [122].

Graph Neural Network

Many real-world data, such as social networks, molecules, roadway maps, cellular biological pathways and so on, are sparse. It is more effective to represent such data as a graph representing relationship among entities. Graph Neural Networks feature permutation invariance on handling graph data. Similar to translation invariance in convolutional neural networks, where the kernel remains the same at different locations of an image, graph neural network is invariant to how the nodes are ordered. This tutorial is a self-contained Colab notebook that goes through a basic graph neural network on solving a regression problem: determine the solubility given a molecule structure. In particular, the tutorial dives deep in practical GNN techniques such as how to generate node features, how to construct a graph convolution layer, how to batch multiple graphs in a mini-batch and so on. At the end, users can pick different hyperparameters to train and evaluate the GNN model. The Colab note is available here [123].

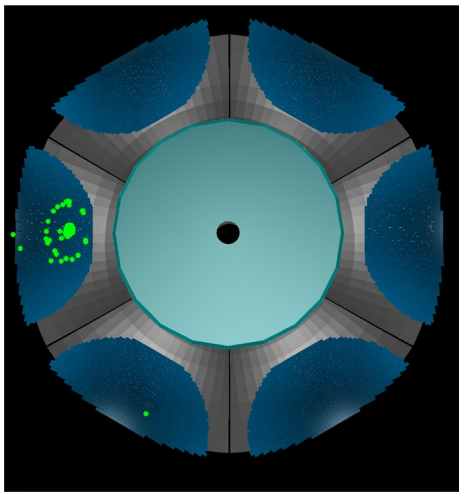

Fig. 11 A sample dRICH π^+ event visualized using ePIC framework

Hackathon

The format of the hackathon was hybrid and international (both local and remote participation), with more than 30 participants connected from around the world (America, Asia and Europe, mainly) grouped in 10 different teams competing to solve the assigned problems. Access to cloud computing resources has been provided during the event, and each team was endowed with an Amazon Web Services (AWS) g5.12xlarge instance, 4 Nvidia a10g GPUs, 48 vCPUs, 192GB Ram, 3.9 TB of disk. For this hackathon we proposed problems with increased level of difficulty and that are deemed to be solvable in a one-day event, starting from a problem that is accessible to everyone. We focused on the dRICH detector under development as part of the particle-identification system at the future ePIC detector at EIC. Data have been produced using the ePIC software stack.

The hackathon was structured around three problems, each escalating in complexity. Initially, we selected a momentum range around 15 GeV as our foundation problem. This range is significant as it corresponds to a momentum zone where both aerogel and gas radiators can potentially contribute to the π/K separation. In order to raise the level of challenge, we embedded realistic photon yields. An exemplar π^+ event as detected in dRICH is depicted in Fig. 11. Moving to the second problem, we expanded the scope by varying the momentum range and altering the positions of the pions and kaons within the dRICH. For the ultimate challenge, the final problem introduced a layer of complexity with a set of random noise hits, making it the most demanding among all three problems. Documentation and data sets have been made available on Zenodo [124]. Despite the inherent 'simplicity' of the problems, given the approximations made as explained in this document, this

event can potentially become a first step towards machine learning/deep learning application for PID with the dRICH. At the end of the hackathon event, the best solutions provided were all machine learning/deep learning-based, they were quite original, and they outperformed other solutions based on more 'classical' approaches (like cut-based analyses). Though an initial foray into leveraging machine learning and deep learning for PID with the dual-RICH, these studies unequivocally point towards the potential these novel approaches hold for reconstruction and PID within the ePIC dual-RICH framework. This endeavor proved to be a valuable learning opportunity, especially for students, and intriguingly, it showcased the potential edge that contemporary AI/ML methods hold over traditional strategies for PID in imaging Cherenkov detectors, as discussed in [62] and references therein.

Conclusions

The AI4EIC workshop has successfully highlighted the critical role that AI/ML play in the design and execution of the Electron Ion Collider. The event was organized into multiple sections, each focusing on various aspects of the EIC science and the connections with AI/ML applications, providing participants with a comprehensive understanding of these complex topics. The community is benefiting from recent funding opportunities for AI/ML in relation to the EIC, which is promising to yield significant results in the coming years. This financial commitment will undoubtedly contribute to the acceleration of research and innovation within the field. The adoption of AI/ML in EIC not only demands new multidisciplinary expertise but also necessitates overcoming cultural barriers to integrate these technologies effectively alongside traditional methods. In this regard, AI4EIC is not only a platform for showcasing the remarkable advancements and progress in AI, but also plays a vital role in increasing AI literacy. By disseminating the knowledge and understanding of AI across the community, we hope to inspire more individuals and institutions to engage with this technology. We are also pleased to collaborate with the ePIC experiment at the EIC. This partnership is set to bring new perspectives and opportunities for progress, strengthening the role of AI in our initiatives. The success of our educational activities, such as hackathons and schools, affirms the effectiveness of these strategies. We are committed to continuing such events, facilitating an environment that encourages learning, innovation, and collaboration.

As plans for future activities, we anticipate formats such as conference, schools, and data challenges, with new upcoming events outlined in the the official website [125].

Acknowledgements The authors would like to acknowledge AWS for providing the cloud computing resources for the hackathon event. We also, thank the College of William and Mary for their support to the hackathon and for sponsoring the prizes.

Author contributions Conveners, hackathon organizers, invited speakers, and editors are reported in the author list. All authors reviewed the manuscript.

Data availability The datasets and lecture materials used in our tutorials are publicly available on the EIC Community website at https://eic.ai/community, which includes all relevant educational resources.

Declarations

Conflict of interest The authors declare no competing interests.

References

1. Abdul Khalek R, et al (2021) Science requirements and detector concepts for the electron-ion collider: EIC yellow report. arXiv preprint arXiv:2103.05419
2. Ent R, Aschenauer E C (2022) EIC schedule and overview . https://indico.bnl.gov/event/16586/contributions/68854/
3. Fanelli C, Papandreou Z, Suresh K et al (2023) AI-assisted optimization of the ECCE tracking system at the Electron Ion Collider. Nucl Instrum Methods Phys Res, Sect A 1047:167748. https://doi.org/10.1016/j.nima.2022.167748
4. Dorigo T, et al (2022) Toward the End-to-End Optimization of Particle Physics Instruments with Differentiable Programming: a White Paper. arXiv preprint arXiv:2203.13818 [physics.ins-det]
5. Cisbani E, Dotto AD, Fanelli C, Williams M et al (2020) Ai-optimized detector design for the future electron-ion collider: the dual-radiator rich case. J Instrum 15(05):P05009. https://doi.org/10.1088/1748-0221/15/05/P05009
6. Boehnlein A et al (2022) Colloquium: machine learning in nuclear physics. Rev Mod Phys 94(3):031003
7. Fanelli C (2022) Design of detectors at the electron ion collider with artificial intelligence. J Instrum 17(04):C04038. https://doi.org/10.1088/1748-0221/17/04/c04038
8. Daulton S, Balandat M, Bakshy E (2020) Differentiable expected hypervolume improvement for parallel multi-objective bayesian optimization . arXiv:2006.05078
9. Balandat M, et al (2020) BoTorch: a framework for efficient Monte-Carlo Bayesian Optimization. https://proceedings.neuri

ps.cc/paper/2020/hash/f5b1b89d98b7286673128a5fb112cb9a-Abstract.html
10. Adaptive experimentation platform. https://ax.dev/. Accessed 2022-12-18
11. Eriksson D, Jankowiak M (2021) High-dimensional bayesian optimization with sparse axis-aligned subspaces. arXiv:2103.00349
12. Daulton S, Eriksson D, Balandat M, Bakshy E (2021) Multi-objective Bayesian optimization over high-dimensional search spaces. arXiv:2109.10964
13. Daulton S, et al (2022) Robust multi-objective bayesian optimization under input noise. arXiv:2202.07549
14. Fast calorimeter simulation challenge 2022. https://calochallenge.github.io/homepage/. Accessed 2022-12-18
15. Aad G et al (2022) Atlfast3: the next generation of fast simulation in atlas. Comput Softw Big Sci 6(1):7. https://doi.org/10.1007/s41781-021-00079-7
16. https://ai4eicdetopt.pythonanywhere.com/
17. Fawzi A et al (2022) Discovering faster matrix multiplication algorithms with reinforcement learning. Nature 610(7930):47–53. https://doi.org/10.1038/s41586-022-05172-4
18. Barbosa WAS, Gauthier DJ (2022) Learning spatiotemporal chaos using next-generation reservoir computing. Chaos Interdiscip J Nonlinear Sci 32(9):093137. https://doi.org/10.1063/5.0098707
19. Li Z et al (2022) Learning chaotic dynamics in dissipative systems. Adv Neural Inf Process Syst 35:16768–16781 arXiv:2106.06898 [cs.LG]
20. Arpaia P et al (2021) Machine learning for beam dynamics studies at the CERN Large Hadron Collider. Nucl Instrum Methods Phys Res, Sect A 985:164652. https://doi.org/10.1016/j.nima.2020.164652
21. Ball RD et al (2017) Parton distributions from high-precision collider data. Eur Phys J C. https://doi.org/10.1140/epjc/s10052-017-5199-5
22. Bertone V, Carrazza S, Hartland NP, Nocera ER, Rojo J (2017) A determination of the fragmentation functions of pions, kaons, and protons with faithful uncertainties. Eur Phys J C. https://doi.org/10.1140/epjc/s10052-017-5088-y
23. Cuic M, Kumericki K, Schafer A (2020) Separation of Quark Flavors using DVCS Data . arXiv:2007.00029
24. Almaeen M, et al (2022) Benchmarks for a global extraction of information from deeply virtual exclusive scattering. arXiv:2207.10766
25. Hyde CE, Guidal M, Radyushkin AV (2011) Deeply virtual exclusive processes and generalized parton distributions. J Phys Conf Ser 299(1):012006. https://doi.org/10.1088/1742-6596/299/1/012006
26. Kumerički K, Liuti S, Moutarde H (2016) GPD phenomenology and DVCS fitting: entering the high-precision era. Eur Phys J A 52:1–31. https://doi.org/10.1140/epja/i2016-16157-3
27. Grigsby J et al (2021) Deep learning analysis of deeply virtual exclusive photoproduction. Phys Rev D 104(1):016001. https://doi.org/10.1103/PhysRevD.104.016001. arXiv:2012.04801 [hep-ph]
28. Campbell J, et al (2022) Event generators for high-energy physics experiments. arXiv. arXiv:2203.11110 [hep-ph]
29. Bellman R (1966) Dynamic programming. Science 153(3731):34–37
30. Kingma D P, Welling M (2022) Auto-Encoding Variational Bayes . arXiv:1312.6114
31. Goodfellow I J, et al (2014) Generative adversarial networks . arXiv:1406.2661
32. Rezende D J, Mohamed S (2016) Variational inference with normalizing flows. arXiv:1505.05770
33. Ilten P, Menzo T, Youssef A, Zupan J (2023) Modeling hadronization using machine learning. SciPost Phys 14 (3): 027. https://doi.org/10.21468/SciPostPhys.14.3.027, arXiv:2203.04983 [hep-ph]
34. Ghosh A, Ju X, Nachman B, Siodmok A (2022) Towards a deep learning model for hadronization. Phys Rev D 106:096020. https://doi.org/10.1103/PhysRevD.106.096020
35. Papamakarios G, Pavlakou T, Murray I (2018) Masked autoregressive flow for density estimation. arXiv:1705.07057
36. Youssef A, et al (2022) Normalizing flows for fragmentation and hadronization
37. Sohl-Dickstein J, Weiss E, Maheswaranathan N, Ganguli S Bach F, Blei D (eds) Deep Unsupervised Learning using Nonequilibrium Thermodynamics. (eds Bach, F. & Blei, D.) Proceedings of the 32nd International Conference on Machine Learning, Vol. 37 of Proceedings of Machine Learning Research, 2256–2265 (PMLR, Lille, France, 2015). https://proceedings.mlr.press/v37/sohl-dickstein15.html
38. Viktoria C et al (2019) Generative models for fast calorimeter simulation: the LHCb case. EPJ Web Conf 214:02034. https://doi.org/10.1051/epjconf/201921402034
39. Rogachev A, Ratnikov F (2023) GAN with an auxiliary regressor for the fast simulation of the electromagnetic calorimeter response. J Phys: Conf Ser 2438(1):012086. https://doi.org/10.1088/1742-6596/2438/1/012086
40. Ratnikov F et al (2023) A full detector description using neural network driven simulation. Nucl Instrum Methods Phys Res A Accel Spectrom Detect Assoc Equip 1046:167591. https://doi.org/10.1016/j.nima.2022.167591
41. Mikuni V, Nachman B (2022) Score-based generative models for calorimeter shower simulation. Phys Rev D 106:092009. https://doi.org/10.1103/PhysRevD.106.092009
42. Agostinelli S et al (2003) Geant4—a simulation toolkit. Nucl Instrum Methods Phys Res Sect A 506(3):250–303. https://doi.org/10.1016/S0168-9002(03)01368-8
43. Krause C, Shih D (2021) CaloFlow: fast and accurate generation of calorimeter showers with normalizing flows. arXiv preprint arXiv:2106.05285 [physics.ins-det]
44. Diefenthaler M, Farhat A, Verbytskyi A, Xu Y (2022) Deeply learning deep inelastic scattering kinematics. Eur Phys J C 82(11):1064. https://doi.org/10.1140/epjc/s10052-022-10964-z. arXiv:2108.11638 [hep-ph]
45. Arratia M, Britzger D, Long O, Nachman B (2022) Reconstructing the kinematics of deep inelastic scattering with deep learning. Nucl Instrum Methods Phys Res A Accel Spectrom Detect Assoc Equip 1025:166164. https://doi.org/10.1016/j.nima.2021.166164
46. Ng L et al (2022) Deep learning exotic hadrons. Phys Rev D 105(9):L091501. https://doi.org/10.1103/PhysRevD.105.L091501. arXiv:2110.13742 [hep-ph]
47. Liu J, Zhang Z, Hu J, Wang Q (2022) Study of exotic hadrons with machine learning. Phys Rev D 105(7):076013. https://doi.org/10.1103/PhysRevD.105.076013. arXiv:2202.04929 [hep-ph]
48. Andreassen A, Komiske PT, Metodiev EM, Nachman B, Thaler J (2020) OmniFold: a method to simultaneously unfold all observables. Phys Rev Lett 124(18):182001. https://doi.org/10.1103/PhysRevLett.124.182001
49. Chan J, Nachman B (2023) Unbinned and Profiled Unfolding. Bull Am Phys Soc
50. Andreev V et al (2022) Measurement of lepton-jet correlation in deep-inelastic scattering with the H1 detector using machine learning for unfolding. Phys Rev Lett 128:132002. https://doi.org/10.1103/PhysRevLett.128.132002
51. Hiller Blin A (2022) A(I)DAPT: AI for Data Analysis and PreservaTion . https://indico.bnl.gov/event/16586/contributions/68737. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
52. Alanazi Y, et al (2021) A survey of machine learning-based physics event generation. arXiv:2106.00643 [hep-ph]

53. Alanazi Y et al (2022) Machine learning-based event generator for electron-proton scattering. Phys Rev D 106(9):096002. https://doi.org/10.1103/PhysRevD.106.096002
54. Sato N (2022) Femtoscale Imaging of Nuclei using ML and Exascale Platforms . https://indico.bnl.gov/event/16586/contributions/68738/. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
55. Phelps W (2022) Muon Identification with Deep Learning at EIC . https://indico.bnl.gov/event/16586/contributions/68784/. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
56. Peng C (2022) ML particle identification with measured shower profiles from calorimetry . https://indico.bnl.gov/event/16586/contributions/68785/. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
57. Apadula N et al (2022) Monolithic active pixel sensors on cmos technologies. arXiv preprint arXiv:2203.07626
58. Branson N (2022) ML for calorimetry. https://indico.bnl.gov/event/16586/contributions/68843/. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
59. Whiteson D (2022) Interpretable Networks for Identifying Leptons. https://indico.bnl.gov/event/16586/contributions/68782/. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
60. Joosten S (2021) Bottlenecks and limitations in classical simulations: where can AI help?. https://indico.bnl.gov/event/10699/contributions/53786/. 1st workshop on Artificial Intelligence for the Electron Ion Collider
61. Fanelli C (2021) AI for Cherenkov detectors. https://indico.bnl.gov/event/10699/contributions/53784. 1st workshop on Artificial Intelligence for the Electron Ion Collider
62. Fanelli C (2020) Machine learning for imaging Cherenkov detectors. J Instrum 15(02):C02012
63. Fanelli C, Mahmood A (2022) Artificial intelligence for imaging Cherenkov detectors at the EIC. J Instrum 17(07):C07011. https://doi.org/10.1088/1748-0221/17/07/C07011
64. Fanelli C, Pomponi J (2020) DeepRICH: learning deeply Cherenkov detectors. Mach Learn Sci Technol 1(1):015010. https://doi.org/10.1088/2632-2153/ab845a
65. Gagnon L-G (2021) ML for tracking in HEP. https://indico.bnl.gov/event/10699/contributions/51456/. 1st workshop on Artificial Intelligence for the Electron Ion Collider
66. Gagnon L-G (2022) Machine learning for track reconstruction at the LHC. J Instrum 17(02):C02026. https://doi.org/10.1088/1748-0221/17/02/C02026
67. Tsaris A, et al (2018) The HEP.TrkX project: deep learning for particle tracking 1085: 042023 . https://exatrkx.github.io/
68. Ai X, Allaire C, Calace N et al (2022) A common tracking software project. Comput Softw Big Sci. https://doi.org/10.1007/s41781-021-00078-8
69. Allaire C (2022) Machine Learning in ACTS. https://indico.bnl.gov/event/16586/contributions/68783/. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
70. Yu D (2021) Real-time AI tracking and tagging. https://indico.bnl.gov/event/10699/contributions/53930/. 1st workshop on Artificial Intelligence for the Electron Ion Collider
71. Xuan T, Durao F, Sun Y (2022) High performance FPGA embedded system for machine learning based tracking and trigger in sPhenix and EIC. J Instrum 17(07):C07003
72. Feickert M, Nachman B (2021) A living review of machine learning for particle physics. arXiv preprint arXiv:2102.02770 . https://iml-wg.github.io/HEPML-LivingReview/
73. Kunnawalkam Elayavalli R (2022) Tagging heavy flavor jets @ RHIC. https://indico.bnl.gov/event/16586/contributions/68787/. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
74. Bielčíková J, Elayavalli RK, Ponimatkin G, Putschke JH, Sivic J (2021) Identifying heavy-flavor jets using vectors of locally aggregated descriptors. J Instrum 16(03):P03017
75. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition 770–778 . https://openaccess.thecvf.com/content_cvpr_2016/papers/He_Deep_Residual_Learning_CVPR_2016_paper.pdf
76. Arandjelovic R, Gronat P, Torii A, Pajdla T, Sivic J (2016) NetVLAD: CNN architecture for weakly supervised place recognition 5297–5307 . https://openaccess.thecvf.com/content_cvpr_2016/papers/Arandjelovic_NetVLAD_CNN_Architecture_CVPR_2016_paper.pdf
77. Sekula S (2022) AI for heavy-flavor and jet tagging at EIC. https://indico.bnl.gov/event/10699/contributions/53924/. 1st workshop on Artificial Intelligence for the Electron Ion Collider
78. Sirunyan AM et al (2020) A deep neural network for simultaneous estimation of b jet energy and resolution. Comput Softw Big Sci 4:1–20
79. Lee K, Mulligan J, Płoskoń M, Ringer F, Yuan F (2023) Machine learning-based jet and event classification at the Electron-Ion Collider with applications to hadron structure and spin physics. J High Energy Phys 2023(3): 1–35 . https://doi.org/10.1007/JHEP03(2023)085, arXiv:2210.06450
80. McEneaney M, Vossen A (2023) Domain-adversarial graph neural networks for Λ hyperon identification with CLAS12. JINST 18(06):P06002. https://doi.org/10.1088/1748-0221/18/06/P06002
81. Farahani A, Voghoei S, Rasheed K, Arabnia H R (2021) A brief review of domain adaptation. Advances in data science and information engineering: proceedings from ICDATA 2020 and IKE 2020 877–894. https://doi.org/10.1007/978-3-030-71704-9_65
82. McEneaney M (2022) Lambda event tagging at CLAS12. https://indico.bnl.gov/event/16586/contributions/68786/. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
83. Pang G, Shen C, Cao L, Hengel AVD (2021) Deep learning for anomaly detection: a review. ACM Comput Surveys (CSUR) 54(2):1–38. https://doi.org/10.1145/3439950
84. Giroux J (2022) Data-driven learning: Flux+Mutability. https://indico.bnl.gov/event/16586/contributions/68844/. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
85. Fanelli C, Giroux J, Papandreou Z (2022) 'Flux+Mutability': a conditional generative approach to one-class classification and anomaly detection. Mach Learn Sci Technol 3(4):045012. https://doi.org/10.1088/2632-2153/ac9bcb
86. Rohr D (2022) The ALICE Run 3 online/offline processing. Nucl Instrum Meth A 1038: 166954. https://doi.org/10.1016/j.nima.2022.166954, arXiv:2208.07412 [physics.ins-det]
87. Shiers J (2007) The worldwide LHC computing grid (worldwide LCG). Comput Phys Commun 177:219–223. https://doi.org/10.1016/j.cpc.2007.02.021
88. Carini Gabriella et al (2022) Smart sensors using artificial intelligence for on-detector electronics and ASICs. arXiv:2204.13223
89. Duarte J, et al (2019) FPGA-accelerated machine learning inference as a service for particle physics computing. Comput Softw Big Sci 3(1): 13. https://doi.org/10.1007/s41781-019-0027-2, arXiv:1904.08986 [physics.data-an]
90. Miniskar N R, et al (2022) Ultra low latency machine learning for scientific edge applications 01–07. https://doi.org/10.1109/FPL57034.2022.00068
91. Bommasani R, et al (2021) On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258 [cs.LG]
92. Taylor R, et al (2022) Galactica: a large language model for science. arXiv preprint arXiv:2211.09085 [cs.CL]
93. Yuan Y (2023) On the power of foundation models 40519–40530 . https://proceedings.mlr.press/v202/yuan23b.html

94. Brown T et al (2020) Language models are few-shot learners. Adv Neural Inf Proc Syst 33: 1877–1901. https://papers.nips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf. arXiv:2005.14165 [cs.CL]
95. OpenAI. Gpt-4 technical report. arxiv 2303.08774. View in Article 2: 13 (2023). arXiv:2303.08774 [cs.CL]
96. Yenduri G, et al (2023) Generative pre-trained transformer: a comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions. arXiv preprint arXiv:2305.10435 [cs.CL]
97. Physics Event Classification Using Large Language Models (2023). https://indico.bnl.gov/event/19560/contributions/83337/attachments/51332/87782/2023%20Hackathon%20Tutorial.pdf. AI4EIC Hackathon
98. Aaij R et al (2020) Allen: a high-level trigger on GPUs for LHCb. Comput Softw Big Sci 4:1–11. https://doi.org/10.1007/s41781-020-00039-7
99. Perez DC et al (2016) The 40 MHz trigger-less DAQ for the LHCb Upgrade. Nucl Instrum Methods Phys Res, Sect A 824:280–283. https://doi.org/10.1016/j.nima.2015.10.047
100. Mitra J et al (2019) Trigger and timing distributions using the TTC-PON and GBT bridge connection in ALICE for the LHC run 3 upgrade. Nucl Instrum Methods Phys Res, Sect A 922:119–133. https://doi.org/10.1016/j.nima.2018.12.076
101. Migliorini M et al (2023) Trigger-less readout and unbiased data quality monitoring of the CMS drift tubes muon detector. J Instrum 18(01):C01003. https://doi.org/10.1088/1748-0221/18/01/C01003
102. Ameli F et al (2022) Streaming readout for next generation electron scattering experiments. Eur Phys J Plus 137(8):958. https://doi.org/10.1140/epjp/s13360-022-03146-z
103. Furletov S et al (2022) Machine learning on FPGA for event selection. J Instrum 17(06):C06009. https://doi.org/10.1088/1748-0221/17/06/C06009
104. Barbosa F et al (2023) Development of ML FPGA filter for particle identification and tracking in real time. IEEE Trans Nucl Sci. https://doi.org/10.1109/TNS.2023.3259436
105. Bernauer J et al (2023) Scientific computing plan for the ECCE detector at the Electron Ion Collider. Nucl Instrum Methods Phys Res, Sect A 1047:167859. https://doi.org/10.1016/j.nima.2022.167859
106. Diefenthaler M (2022) INDRA-ASTRA. https://indico.bnl.gov/event/16586/contributions/68794/. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
107. Bifet A, Gavalda R (2007) Learning from time-changing data with adaptive windowing 443–448. https://doi.org/10.1137/1.9781611972771.42
108. Britton T (2022) AI Experimental Control. https://indico.bnl.gov/event/16586/contributions/68800. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
109. Jeske T et al (2022) AI for experimental controls at Jefferson lab. J Instrum 17(03):C03043. https://doi.org/10.1088/1748-0221/17/03/C03043
110. Bondì M (2022) Streaming Readout for Next Generation e-Scattering Experiments. https://indico.bnl.gov/event/16586/contributions/68798. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
111. Furletov S (2022) Fast ML for FPGA. https://indico.bnl.gov/event/16586/contributions/68795. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
112. FastML Team. fastmachinelearning/hls4ml (2023). https://github.com/fastmachinelearning/hls4ml
113. Huang J (2022) AI-based data reduction for streaming DAQ. https://indico.bnl.gov/event/16586/contributions/68797. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
114. Huang Y, Ren Y, Yoo S, Huang J (2021) Efficient data compression for 3D sparse TPC via bicephalous convolutional autoencoder 1094–1099. https://doi.org/10.1109/ICMLA52953.2021.00179
115. Dean C (2022) Machine learning for heavy flavor identification. https://indico.bnl.gov/event/16586/contributions/68799. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
116. Ammendola R (2022) AI for streaming readout: an architectural perspective. https://indico.bnl.gov/event/16586/contributions/68796. 2nd workshop on Artificial Intelligence for the Electron Ion Collider
117. Ammendola R et al (2023) APEIRON: composing smart TDAQ systems for high energy physics experiments. arXiv:2307.01009 [cs.DC]
118. Balandat M (2022) Multi-Objective Bayesian Optimization with BoTorch and Ax (slides). https://indico.bnl.gov/event/16586/contributions/68649/. [Online; accessed 07-Jun-2023]
119. Balandat M (2022) Multi-objective Bayesian optimization with BoTorch and Ax (colab notebook). https://colab.research.google.com/drive/1c6JY4tcwGzlQuGbFFv6ZIcPRrLI6_AD7#scrollTo=XcOhF2r0p2df. [Online; accessed 17-Jun-2023]
120. Torales Acosta F, Mikuni V (2022) Unfolding with Omnifold (colab notebook). https://colab.research.google.com/drive/1zuU9MezTIQGPhXlPG1Y9QilyDcQk6L0K?usp=sharing. [Online; accessed 1-Dec-2022]
121. Zaharia M, et al (2018) Accelerating the machine learning lifecycle with MLflow. IEEE Data Eng Bull 41(4): 39–45. https://cs.stanford.edu/~matei/papers/2018/ieee_mlflow.pdf
122. McSpadden D, Rajput K (2022) MLFlow and Hyperparameter Optimization. https://colab.research.google.com/drive/1qPIyfefaqofX1wNQ3TYPT_ABy749Ohd2?usp=sharing. [Online; accessed 22-Dec-2022]
123. Ren Y (2022) Graph Neural Network Tutorial. https://colab.research.google.com/drive/16fF6q1CSnxnEqRSl7LDAb0evscfqMOrf?usp=sharing. [Online; accessed 12-Dec-2022]
124. Fanelli C, Giroux J, McSpadden D, Rajput K, Suresh K (2022) AI4EIC Hackathon. https://doi.org/10.5281/zenodo.7197023
125. AI4EIC website. https://eic.ai

Authors and Affiliations

C. Allaire[60] · R. Ammendola[22] · E.-C. Aschenauer[3] · M. Balandat[33] · M. Battaglieri[36] · J. Bernauer[6,46] · M. Bondì[35] · N. Branson[14,32] · T. Britton[27] · A. Butter[28] · I. Chahrour[55] · P. Chatagnon[27] · E. Cisbani[37] · E. W. Cline[46] · S. Dash[23] · C. Dean[31] · W. Deconinck[54] · A. Deshpande[3,6] · M. Diefenthaler[27] · R. Ent[27] · C. Fanelli[27,64] · M. Finger[10] · M. Finger Jr.[10] · E. Fol[5] · S. Furletov[27] · Y. Gao[3] · J. Giroux[56,64] · N. C. Gunawardhana Waduge[58] · O. Hassan[54,57] · P. L. Hegde[9] · R. J. Hernández-Pinto[16] · A. Hiller Blin[25] · T. Horn[47] · J. Huang[3] · A. Jalotra[53] · D. Jayakodige[21,27] · B. Joo[39] · M. Junaid[56] · N. Kalantarians[62] · P. Karande[30] · B. Kriesten[8] · R. Kunnawalkam Elayavalli[61] · Y. Li[41] · M. Lin[3] · F. Liu[39] · S. Liuti[58] · G. Matousek[15] · M. McEneaney[15] · D. McSpadden[27] · T. Menzo[51] · T. Miceli[17] · V. Mikuni[65] · R. Montgomery[52] · B. Nachman[29,48] · R. R. Nair[34] · J. Niestroy[64] · S. A. Ochoa Oregon[16] · J. Oleniacz[63] · J. D. Osborn[3] · C. Paudel[18] · C. Pecar[15] · C. Peng[1] · G. N. Perdue[17] · W. Phelps[11,27] · M. L. Purschke[3] · H. Rajendran[9] · K. Rajput[27] · Y. Ren[29] · D. F. Renteria-Estrada[16] · D. Richford[2] · B. J. Roy[38] · D. Roy[45] · A. Saini[17] · N. Sato[27] · T. Satogata[27,40] · G. Sborlini[12,20] · M. Schram[27] · D. Shih[44] · J. Singh[43] · R. Singh[4,7] · A. Siodmok[26] · J. Stevens[64] · P. Stone[64] · L. Suarez[64] · K. Suresh[56,64] · A.-N. Tawfik[19] · F. Torales Acosta[29] · N. Tran[17] · R. Trotta[47] · F. J. Twagirayezu[50] · R. Tyson[52] · S. Volkova[42] · A. Vossen[15,27] · E. Walter[64] · D. Whiteson[49] · M. Williams[31] · S. Wu[54] · N. Zachariou[59] · P. Zurita[13,24]

✉ C. Fanelli
cfanelli@wm.edu

[1] Argonne National Laboratory, Lemont, IL 60439, USA

[2] Baruch College, New York, NY 10010, USA

[3] Brookhaven National Lab, Upton, NY 11973, USA

[4] Department of Physics, Brookhaven National Laboratory, Upton, NY 11973-5000, USA

[5] Accelerators and Beam Physics Group, CERN, Geneva 1211, Switzerland

[6] Center for Frontiers in Nuclear Science, Stony Brook University, Stony Brook, NY 11790-3800, USA

[7] Department of Physics and Astronomy, Center for Frontiers in Nuclear Science, Stony Brook University, Stony Brook, NY 11794, USA

[8] Center for Nuclear Femtography, Washington, DC 20005, USA

[9] Central University of Karnataka, Kalaburagi 585367, India

[10] Charles University, Faculty of Mathematics and Physics, Prague 18000, Czech Republic

[11] Christopher Newport University, Newport News, VA 11606, USA

[12] Departamento de Física Fundamental e IUFFyM, Universidad de Salamanca, Salamanca 37008, Spain

[13] Departamento de Física Teórica & IPARCOS, Universidad Complutense de Madrid, Adrid 28040, Spain

[14] Drexel University, Philadelphia, PA 19104, US

[15] Duke University, Durham, NC 27708, USA

[16] Facultad de Ciencias Físico-Matemáticas, Universidad Autónoma de Sinaloa, Ciudad Universitaria, Culiacán, Sinaloa 80000, Mexico

[17] Fermilab, Batavia, IL 60510, USA

[18] Florida International University, Miami, FL 33199, USA

[19] Future University in Egypt, New Cairo 11835, Egypt

[20] GFIMA, Escuela de Ciencias, Ingeniería y Diseño, Universidad Europea de Valencia, Valencia 46010, Spain

[21] Hampton University, Hampton, VA 23668, USA

[22] Sezione di Roma Tor Vergata, INFN, Rome 00133, Italy

[23] Indian Institute of Technology Bombay, Mumbai, Maharashtra 400076, India

[24] Institut für Theoretische Physik, Universität Regensburg, Regensburg 93040, Germany

[25] Institute for Theoretical Physics, TübingenUniversity, Tübingen 72076, Germany

[26] Jagiellonian University, Kraków 31-007, Poland

[27] Jefferson Lab, Newport News, VA 293606, USA

[28] Laboratory Nuclear and High-Energy Physics, CNRS, Paris 75005, France

[29] Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

[30] Lawrence Livermore National Laboratory, Livermore, CA 94550, USA

[31] Massachusetts Institute of Technology, Cambridge 021339, MA, USA

[32] Messiah University, Mechanicsburg 17055, PA, USA

[33] Meta, Menlo Park, CA 935025, USA

[34] National Centre For Nuclear Research, Warsaw 02-093, Poland

[35] National Institute for Nuclear Physics, Catania 95125, Italy

[36] National Institute for Nuclear Physics, Genoa 16146, Italy

[37] National Institute for Nuclear Physics, Roma, Rome 00185, Italy

[38] Nuclear Physics Division, Bhabha Atomic Research Centre, Mumbai 400085, India

[39] Oak Ridge National Laboratory, Oak Ridge, TN 37830, USA

[40] Old Dominion University, Norfolk, VA 23529, USA

[41] Department of Computer Science, Old Dominion University, Norfolk 23529, USA

[42] Pacific Northwest National Laboratory, Richland, WA 99354, USA

[43] Panjab University, Chandigarh 160014, India

[44] Rutgers University, Camden, NJ 08102, USA

[45] Rutgers University, Piscataway, NJ 08854, USA

[46] Stony Brook University, Stony Brook, NY 11794, USA

[47] The Catholic University of America, Washington, DC 20064, USA

[48] Berkeley Institute for Data Science, University of California, Berkeley, CA 94720, USA

[49] University of California Irvine, Irvine, CA 92697, USA

[50] University of California Los Angeles, Los Angeles, CA 90095, USA

[51] University of Cincinnati, Cincinnati, OH 45221, USA

[52] University of Glasgow, Glasgow G12 8QQ, UK

[53] University of Jammu, Jammu, Jammu and Kashmir 180006, India

[54] Department of Physics, University of Manitoba, Winnipeg, MB R3T 2N2, Canada

[55] University of Michigan, Ann Arbor, MI 48104, USA

[56] University of Regina, Regina, SK S4S0A2, Canada

[57] University of Victoria, Victoria V8P 5C2, Canada

[58] University of Virginia, Charlottesville, VA 22904, USA

[59] University of York, York YO422SF, UK

[60] Université Paris-Saclay, CNRS/IN2P3, IJCLAB, Orsay 91400, France

[61] Vanderbilt University, Nashville, TN 37235, USA

[62] Virginia Union University, Richmond, VA 23220, USA

[63] Warsaw University of Technology, Warsaw 00-661, Poland

[64] William & Mary, Williamsburg, VA 231866, USA

[65] National Energy Research Scientific Computing Center, Berkeley Laboratory, Berkeley, CA, 94720, USA