Machine-Learned Particle Flow as a Foundation Model for Collider Physics

Farouk Mokhtar [1,2,*] Joosep Pata [3] Michael Kagan [4] and Javier Duarte [1]
[1]Department of Physics, University of California San Diego, La Jolla, California 92093, USA
[2]European Organization for Nuclear Research (CERN), Geneva 1211, Switzerland
[3]National Institute of Chemical Physics and Biophysics, Tallinn 12618, Estonia
[4]SLAC National Accelerator Laboratory, Menlo Park, California 94025, USA

(Received 15 June 2026; accepted 17 September 2026; published 4 November 2026)

The workflow from particle collision to physics analysis passes through a series of reconstruction steps that are traditionally modular and disconnected, with no shared representation linking low-level detector data to high-level analysis tasks. We show that casting event reconstruction as a machine learning problem naturally produces such a shared representation. We repurpose a machine learning model trained for particle-flow reconstruction (MLPF) to perform three distinct analysis tasks: jet flavor identification, jet energy regression, and missing momentum regression. By appending the per-particle latent representations learned during reconstruction as additional input features, we substantially improve over baselines that use conventional particle-level features alone. We further demonstrate that a single linear layer trained using only the latent representations achieves competitive performance against state-of-the-art baseline architectures, and outperforms the baseline for missing momentum regression with approximately 35 times fewer parameters. These results demonstrate that the latent representations learned during reconstruction encode essential physics information needed for downstream analysis, establishing MLPF as a foundation model and offering a concrete step toward an end-to-end pipeline from detector data to physics analysis.

DOI: 10.1103/v5n3-d3tn

I. INTRODUCTION

In collider experiments, the workflow from particle collision to physics analysis passes through a series of reconstruction steps that are traditionally modular and disconnected, with no shared representation linking low-level detector data to high-level analysis tasks. Translating raw detector signals into physics objects, a task known as event reconstruction, is typically solved using the particle-flow (PF) algorithm [1,2], which combines information from all subdetectors to produce a list of stable particle candidates, or PF candidates. This list then serves as the interface between the detector and all downstream physics analyses. Once it is produced, however, the rich low-level correlations encoded in the detector signals are no longer directly accessible. Recovering task-relevant information beyond the PF candidate four-momenta typically requires hand-engineering additional input features from lower-level quantities. For instance, jet flavor identification relies on track-displacement variables that must be carefully derived from raw track parameters.

Machine learning (ML)-based reconstruction changes this picture. Models such as machine-learned particle flow (MLPF) [3] learn an internal high-dimensional representation of each collision event—referred to hereafter as *latent representations*—as an intermediate step before reconstructing the PF candidates. These representations are learned end-to-end and encode the full complexity of the detector response and particle interactions, including lower-level information that would otherwise require hand-engineering from raw quantities. In this paper, we show that they also encode information useful beyond reconstruction. By appending the per-particle latent representations to the standard PF candidate features, we substantially improve performance across three downstream analysis tasks that probe different levels of event complexity: jet flavor identification, which requires track-level quantities such as impact parameters (IPs) that lie below the PF candidate interface; jet energy regression, which operates at the jet constituent level; and missing momentum ($\vec{p}^{\,\mathrm{miss}}$) regression, which aggregates information across the full event. Figure 1 contrasts the conventional pipeline, which produces only a list of PF candidates, with the ML-based pipeline, which additionally provides the latent representations to analysis tasks.

The prospect of building *foundation models*, large models pretrained on broad data that can be adapted or fine-tuned to a wide range of downstream tasks [4], for collider physics has attracted significant recent interest, with several works demonstrating that models pretrained on large jet datasets can be fine-tuned for downstream tasks with improved physics performance [5–18]. For instance, Ref. [14] demonstrates that jet-level latent representations from a pretrained transformer-based jet tagger can be input to an analysis-level event classifier to improve signal sensitivity significantly compared

*Contact author: fmokhtar@ucsd.edu

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

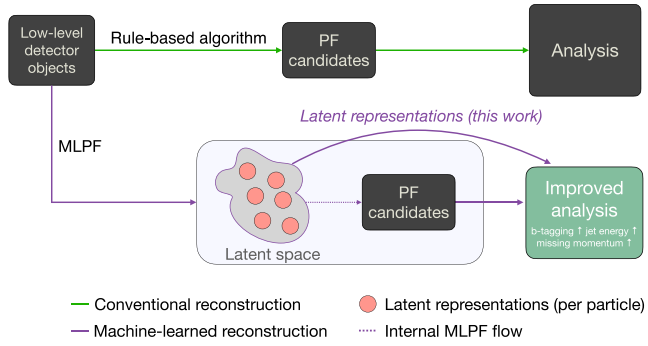


FIG. 1. Schematic comparison of conventional and machine-learned particle-flow reconstruction. In the machine-learned approach, MLPF learns a rich per-particle latent representation that improves performance across different analysis tasks.

to using only conventional high-level features. Reviews of this emerging field are given in Refs. [19,20].

In parallel, MLPF reconstruction has become a viable alternative to rule-based PF, with approaches developed for multiple detector geometries [21–25] and recently commissioned in CMS data [26]. In Ref. [22], we further demonstrated that an MLPF model pretrained on one detector geometry can be rapidly fine-tuned to a new detector with substantially less data than training from scratch, establishing cross-detector transfer as a first step toward a foundation model for collider physics.

The present work extends this picture by demonstrating cross-task transfer. Rather than explicitly designing a foundation model from the outset, we demonstrate that MLPF already produces representations that are generically useful for downstream analysis tasks. The MLPF backbone requires no modification, additional pretraining, or fine-tuning; its latent representations are extracted as a byproduct of standard reconstruction inference. This suggests that ML-based reconstruction and physics analysis need not be treated as separate pipeline stages, and that reconstruction models may serve as natural foundation models for collider physics.

The paper is organized as follows. Section II describes the MLPF architecture and extraction of the latent representations. Section III presents the experimental setup. Sections IV–VI describe the three downstream tasks and report the results for each. Section VII discusses the findings and their broader implications, and Sec. VIII summarizes the main conclusions.

II. MLPF LATENT REPRESENTATIONS

The MLPF algorithm [3] was originally designed as a graph neural network and later evolved into a transformer-based architecture [22,26]. The precise underlying mechanism, whether message passing or attention, is not central to the following discussion. In both cases, the model processes a set of input detector elements (tracks and calorimeter clusters) and learns to map them into a high-dimensional *latent space*. Unlike the input space, which is defined by raw detector measurements, the latent space is learned. Its dimensions have no fixed physical meaning but are optimized during training to capture the correlations and structures in the data most useful for solving the reconstruction task. The model then uses this representation to predict the kinematic properties (four-momentum) and particle identity (PID) of each reconstructed PF candidate. Throughout this paper, we refer to the particles reconstructed by MLPF as PF candidates, using the same terminology as conventional PF algorithms to emphasize that MLPF serves as a drop-in replacement for traditional reconstruction.

In the latest implementation [22] used here, the latent space has 2048 dimensions. This arises from a design choice in the MLPF architecture to maintain two separate 1024-dimensional encoder branches, one specialized for particle classification and the other for momentum regression. This allows the classification and regression objectives to be optimized jointly during training while keeping their internal representations distinct. This separation is not fundamental. A single unified encoder of equivalent capacity would serve our purposes equally well. We extract per-particle latent representations from the layer immediately preceding the output heads. These representations encode structural information about the collision event that conventional reconstruction algorithms discard once the list of PF candidates is produced.

Since the full 2048-dimensional space is impractical as a direct input to downstream models, and to control model capacity, we apply a simple unsupervised compression step to retain the top k dimensions (details in Sec. III B). The specific choice of compression technique is not central to our argument. We deliberately use a well-established, parameter-free method to demonstrate that the improvement is robust and does not depend on task-specific optimization of the compression. We expect further gains from more sophisticated approaches, which we leave to future work.

As a qualitative illustration of the structure captured by the MLPF latent representations, Fig. 2 shows a two-dimensional t-distributed stochastic neighbor embedding (t-SNE) [27] projection of latent representations averaged across the constituents of each jet in a sample of 50 000 jets. Each point is colored by its generator-level flavor label (defined in Sec. IV). Despite the fact that the MLPF backbone never used jet-flavor labels during training, the projection shows visibly separated regions of b jets and light-flavor jets, suggesting that jet-flavor information is encoded in the latent representations.

III. EXPERIMENTAL SETUP

Throughout this work, we use the MLPF model in Ref. [22] as our backbone model. The MLPF backbone is pretrained on simulated datasets from the Compact Linear Collider detector (CLICdet) and subsequently fine-tuned on simulated datasets from the CLIC-like detector (CLD) [28], a general-purpose detector concept proposed for the Future Circular Collider in electron-positron mode (FCC-ee) [29]. The model was trained using a total of 25 million events (more details in Ref. [22]). This is the most recent MLPF checkpoint trained on the CLD geometry and is therefore the natural choice given that all events in this paper are simulated with the same detector geometry.

We adopt two deliberately conservative choices throughout. First, the MLPF backbone weights remain completely frozen. This approach is practically useful for collider analy-

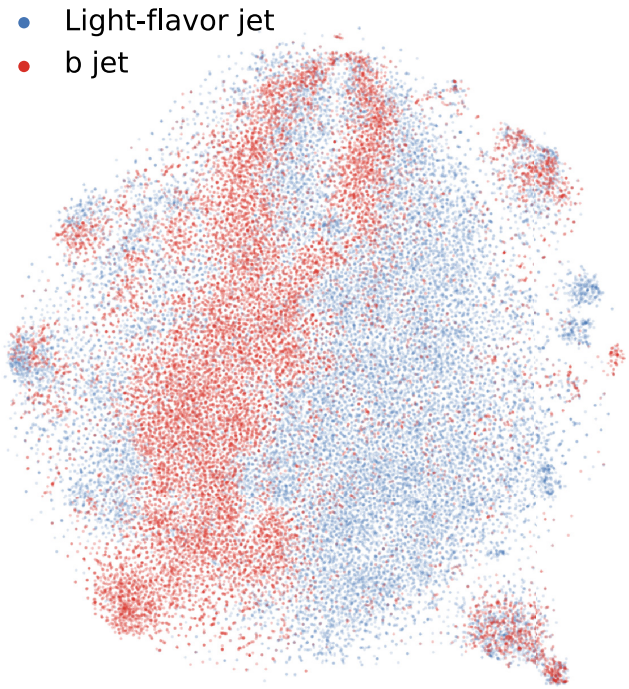


FIG. 2. Two-dimensional t-SNE projection of the MLPF latent representations averaged across the constituents of each jet on a sample of 50 000 jets, colored by generator-level flavor (red: b jets; blue: light-flavor jets). The MLPF backbone is never trained using jet-flavor labels, yet the latent representations exhibit visible flavor separation.

ses because analysts can leverage MLPF latent representations for downstream tasks without modifying the reconstruction pipeline or retraining the backbone. Second, all downstream experiments use events drawn exclusively from the held-out test split of the MLPF fine-tuning procedure, so training-set contamination does not arise by construction and the observed improvements are attributable only to genuine transfer of learned representations. Earlier studies indicated that allowing overlap with the MLPF fine-tuning set yields further performance gains, suggesting additional improvements are achievable when this constraint is relaxed. We nevertheless adopt the stricter setup to remove any contamination concern.

The remainder of this section presents the dataset splits, the compression of the MLPF latent representations, and the evaluation design used in all three downstream studies.

A. Dataset

We use the CLD dataset described in Ref. [22], consisting of top quark pair production ($e^+e^- \to t\bar{t}$) events at a center-of-mass energy of 365 GeV. The CLD detector concept is a general-purpose detector sharing the same broad design philosophy as the ATLAS and CMS detectors at the LHC, comprising a silicon pixel and strip tracking system, a high-granularity electromagnetic calorimeter, a hadronic calorimeter, and a superconducting solenoid. Events are generated with PYTHIA [30] and passed through full GEANT4 [31–33] simulation of the CLD geometry, providing a realistic description of particle interactions and detector response. Each event contains a variable number of PF candidates, characterized by their transverse momentum (p_T), pseudorapidity (η), azimuthal angle (ϕ), energy (E), and PID. The all-inclusive $t\bar{t}$ sample is well suited for this study as it naturally provides all objects relevant to the downstream tasks considered: b jets from top quark decays, c jets from $W \to c\bar{s}$ decays, light-flavor jets from hadronic W boson decays, and genuine $\vec{p}^{\,\text{miss}}$ from neutrinos produced in leptonic W boson decays.

TABLE I. Dataset splits used in this work. All events are drawn from the MLPF held-out test split of Ref. [22].

Split	Events	Purpose
Train	350 000	Train downstream models
Validation	5000	Track validation loss
Test	40 000	Benchmark physics performance

The dataset is divided into three nonoverlapping splits as shown in Table I. An additional 5000 events are reserved exclusively for the latent compression step described in the following subsection.

B. PCA compression of latent representations

As described in Sec. II, the MLPF encoder produces a 2048-dimensional latent vector per PF candidate. To reduce this to a computationally practical size, we apply principal component analysis (PCA) in an entirely unsupervised manner, using only the geometry of the MLPF latent space and introducing no task-specific information during the compression stage. We intentionally use a task-agnostic compression to demonstrate that the MLPF latent representations carry generic information useful across different downstream tasks.

We apply PCA to the 2048-dimensional latent vectors, projecting them onto their k leading principal components. These are the k directions that capture the most variance in the latent space. We fit the projection on the dedicated 5000 events reserved for this purpose, yielding approximately 744 000 PF candidate latent vectors, using the `IncrementalPCA` implementation from scikit-learn [34] to avoid loading all vectors into memory at once. We choose $k = 128$. For this value, the projection retains 55% of the total variance. The cumulative explained variance as a function of k is notably flat, reflecting a latent space where information is broadly distributed across many dimensions rather than concentrated in a few dominant directions. The resulting k-dimensional compressed vectors are appended as additional per-particle features alongside the standard kinematic inputs. During inference, the compression reduces to a single $2048 \times k$ matrix multiply per PF candidate (∼0.3 M FLOPs), adding negligible overhead, orders of magnitude below the MLPF inference and downstream-model costs.

C. Evaluation design

For each downstream task, we train and evaluate three model variants:

(1) *BASELINE*: established task-specific architecture with standard PF candidate kinematic features as input (four-momentum, PID).

(2) *LATENT-AUGMENTED*: same architecture and inputs as BASELINE, augmented with the 128-dimensional MLPF latent vector per PF candidate.

(3) *LINEAR-PROBE*: a single linear layer using only the 128-dimensional MLPF latent vector representations per PF candidate as input.

Particle identity is encoded as a one-hot vector over the five MLPF output classes (charged hadron, neutral hadron, photon, electron, muon). Depending on the downstream task, additional input features beyond PF candidate kinematics and PID may be provided to the BASELINE and LATENT-AUGMENTED models. For jet flavor identification, for instance, track IP variables are included. In all cases, we follow established conventions from prior publications, as detailed in Secs. IV–VI.

Since the LATENT-AUGMENTED model receives additional input features, its parameter count is inherently larger than that of the BASELINE model even with identical hidden layer widths. To verify that the observed gains are not simply a consequence of this increased capacity, we also train a BASELINE-LARGE model with the same inputs as BASELINE but with wider hidden layers matching the parameter count of LATENT-AUGMENTED. We observe that the BASELINE-LARGE model performs comparably to the BASELINE model across all three tasks, confirming that capacity alone does not explain the performance gains. Full results are reported in Appendix A.

The LINEAR-PROBE model serves a distinct diagnostic purpose. In representation learning, a *linear probe* [35] is a single linear layer trained on top of a frozen embedding, quantifying how much task-relevant information is linearly accessible in the latent space, independently of any nonlinear processing by the downstream architecture. A strong linear probe result indicates that the latent representations intrinsically organize the relevant physics signal in a geometrically separable way. In the LINEAR-PROBE model, all learned parameters reside in a single linear layer that takes the MLPF latent representations as input. Any kinematic operations follow the natural output structure of each task and are held fixed. For the jet-level tasks, the per-particle latent representations are averaged across the constituents of each jet before the linear layer produces the output. For $\vec{p}^{\,\mathrm{miss}}$ regression, the linear layer produces a per-particle scalar weight, and the predicted $\vec{p}^{\,\mathrm{miss}}$ is the weighted vector sum of the PF candidate momenta in the event. The LINEAR-PROBE models have 129 parameters for the two regression tasks (128 weights+1 bias for the scalar output). For jet flavor identification, which involves three output classes, the LINEAR-PROBE model has 387 parameters (128×3 weights+3 biases). All LINEAR-PROBE models are trained on the same data and with the same protocol as the other model variants.

While the LINEAR-PROBE model quantifies the information that is linearly accessible in the latent space, we additionally study how much is accessible to a more expressive model by training NONLINEAR-PROBE, in which the full task-specific architecture is applied to the MLPF latent representations alone, with all other input features removed. We observe that the NONLINEAR-PROBE model matches the LATENT-AUGMENTED model for the two regression tasks and approaches the BASELINE model for jet flavor identification, indicating that most of the task-relevant information is accessible from the latent space alone. Full results are reported in Appendix B.

The scaling figures (Figs. 3, 5, and 7) are evaluated on the validation set and show the median (M) and interquartile range (IQR) across five random-seed trainings as a function of the size of the training set. For jet flavor identification, we plot the area under the curve (AUC) rather than the validation loss, as it is more directly interpretable. The final physics performance results (Figs. 4, 6, and 8) use the held-out test set and the best model for each task (corresponding to the seed with the lowest validation loss when trained on the full training set), with shaded bands showing the statistical uncertainty per bin where applicable.

IV. JET FLAVOR IDENTIFICATION

We formulate jet flavor identification as a multiclass classification task where, given the PF candidates clustered into a jet, the model predicts one of three flavor categories: b jets (from beauty quarks), c jets (from charm quarks), or light-flavor jets (from strange, up, down quarks, or gluons). The three-class setup separates the two heavy-flavor categories whose discrimination is the main target of modern flavor taggers. We use the $\mathrm{t\bar{t}}$ sample described in Sec. III A, which naturally provides all three classes: b jets from top quark decays, c jets from $\mathrm{W}\to\mathrm{c\bar{s}}$ decays, and light-flavor jets from hadronic W boson decays and final-state radiation.

A. Jet clustering and labeling

Reconstructed jets are clustered from the PF candidates using the generalized-k_{T} algorithm [36] with $R = 0.4$ and exponent $p = -1$, requiring a minimum $p_{\mathrm{T}} > 5\,\mathrm{GeV}$, consistent with the convention used in Ref. [22]. Jet-flavor labels are assigned directly to the reconstructed jets using the ghost-association technique [37], where generator-level B and D hadrons are added to the input list of the jet-clustering algorithm with infinitesimal momentum, so their presence assigns them to a jet without affecting the jet kinematics. A reconstructed jet is labeled as a b jet if any B hadron is associated with it in this way, a c jet if no B hadron but at least one D hadron is associated, and a light-flavor jet otherwise.

B. Dataset statistics

A kinematic selection of $p_{\mathrm{T}} \in [10, 200]\,\mathrm{GeV}$ and $|\eta| < 2.5$ is applied, retaining 89.3% of jets. Selected jets have a mean constituent multiplicity of 13 PF candidates and a p_{T} distribution with mean 42.2 GeV, median 39.4 GeV, and 90th percentile at 71.7 GeV. The resulting flavor composition of the training set is approximately 51% light-flavor jets, 13% c jets, and 36% b jets. Table II summarizes the jet dataset after the kinematic selection.

C. Input features

Each PF candidate is described by a set of kinematic, PID, and track-displacement variables, listed in Table III. The kinematic features include the angular separations $\Delta\eta$ and $\Delta\phi$ with respect to the jet axis, $\ln(1+p_{\mathrm{T}})$, $\ln(1+E)$, $\ln(p_{\mathrm{T}}/p_{\mathrm{T\,jet}})$, $\ln(E/E_{\mathrm{jet}})$, and ΔR, the angular distance to the

TABLE II. Jet counts for the jet flavor identification downstream task.

Split	Total jets	Light-flavor jets	c jets	b jets
Train	1 819 536	937 080	227 422	655 034
Validation	25 900	13 362	3165	9373
Test	207 891	107 170	25 874	74 847

jet axis. Particle identification is encoded as a one-hot vector over the five MLPF output classes, together with the electric charge q.

B and D hadrons produced in heavy-flavor decays travel a macroscopic distance before decaying, leaving a measurable displacement between the primary vertex (PV) and the hadron decay point. Exploiting this displacement is the cornerstone of heavy-flavor jet identification, and providing the model with track-level displacement information is standard practice for ML-based taggers. In the MLPF reconstruction, each PF candidate is aligned with the input detector element from which it is reconstructed: a track for charged particles and a calorimeter cluster for neutral particles [22]. For charged PF candidates, we therefore derive the full set of displacement variables listed in Table III from the associated track, following Ref. [38] and the numerical implementation of the official CLD flavor-tagging code [39]: the transverse and longitudinal IPs (d_0, z_0), the signed two-dimensional (2D) and three-dimensional (3D) IPs and their significances ($\mathrm{SIP_{2D}}$, $\mathrm{SIP_{2D}}/\sigma_{\mathrm{2D}}$, $\mathrm{SIP_{3D}}$, $\mathrm{SIP_{3D}}/\sigma_{\mathrm{3D}}$), the 3D distance between the track and the jet axis at their point of closest approach (d_{3D}) and its significance ($d_{\mathrm{3D}}/\sigma_{d_{\mathrm{3D}}}$), and the full 15-element track parameter covariance matrix C_{ij}. Neutral PF candidates have no associated track, so all displacement and covariance features are set to zero for them.

TABLE III. Per-particle input features for the jet flavor identification and jet energy regression tasks, following Ref. [38].

Index	Feature	Description
0–1	$\Delta\eta$, $\Delta\phi$	η, ϕ relative to jet axis
2	$\ln(1+p_{\mathrm{T}})$	Log-scaled p_{T}
3	$\ln(1+E)$	Log-scaled energy
4	$\ln(p_{\mathrm{T}}/p_{\mathrm{T\,jet}})$	Log-scaled fraction of jet p_{T}
5	$\ln(E/E_{\mathrm{jet}})$	Log-scaled fraction of jet energy
6	ΔR	Angular distance to jet axis
7	d_0	Transverse IP
8	z_0	Longitudinal IP
9	$\mathrm{SIP_{2D}}$	Signed 2D IP
10	$\mathrm{SIP_{2D}}/\sigma_{2D}$	Signed 2D IP significance
11	$\mathrm{SIP_{3D}}$	Signed 3D IP
12	$\mathrm{SIP_{3D}}/\sigma_{3D}$	Signed 3D IP significance
13	d_{3D}	Jet-track 3D distance
14	$d_{\mathrm{3D}}/\sigma_{d_{3D}}$	Jet-track 3D distance significance
15–29	C_{ij}	Track parameter covariance matrix
30	q	Electric charge
31–35	PID flags	Five-class type encoding
Used by LATENT-AUGMENTED and LINEAR-PROBE only:		
36–163	Latent	128-dim latent vectors

TABLE IV. Model variants for the jet flavor identification task. The LINEAR-PROBE model is a single linear layer with 387 parameters (128×3 weights+3 biases), where the factor of 3 corresponds to the three flavor classes (b, c, light).

Model	Architecture	Inputs	Parameters
BASELINE	ParticleNet	36	360 651
LATENT-AUGMENTED	ParticleNet	164	508 363
LINEAR-PROBE	Linear	128	387

Since the displacement variables require knowledge of the PV position, which is not provided by default in the CLD reconstruction chain, we use the PYTHIA generator-level PV smeared with a Gaussian distribution of width 3 μm in each spatial coordinate. This corresponds to the expected vertex resolution of the FCC-ee collider design [40] and serves as a stand-in for a full PV reconstruction, which we leave to future CLD reconstruction work. The complete feature list is given in Table III.

D. Model architecture and training

We use ParticleNet [41] as a well-established architecture baseline, which allows the contribution of the MLPF latent representations to be isolated from model-specific gains. We follow Ref. [41] for the architecture details, with EdgeConv channels 64/128/256, $\Delta\eta$ and $\Delta\phi$ as graph coordinates for the k-nearest-neighbor construction with $k = 16$, and a dropout rate of 0.1 in the postpooling fully connected head. The parameter counts for the three model variants are summarized in Table IV.

Models are trained on a heterogeneous pool of graphics processing units (GPUs) (NVIDIA A10, L40S, V100, RTX 3090, and RTX 4090) on the National Research Platform (NRP) Kubernetes cluster [42], with wall-clock training time ranging from ∼14 min (smallest dataset, fastest GPU) to ∼9 h (full dataset, on slower GPUs). We use the AdamW optimizer [43] with an initial learning rate of 10^{-3}, cosine annealing to 10^{-5}, batch size 256, and a weight decay of 10^{-4}. We minimize the categorical cross-entropy loss with per-class weights set to the inverse training-set frequency to compensate for the imbalance between the three flavor classes. The corresponding weights are approximately (0.46, 1.88, 0.66) for light, c, and b jets, respectively. Training is capped at 100 epochs with early stopping after 20 epochs without improvement in validation loss.

To characterize the data-scaling behavior, we train on five training-set sizes: 100k, 200k, 500k, 1M, and the full 1.82M jets.

E. Results

To evaluate the classification performance, we use the ROC curve, which reports the jet misidentification rate as a function of b-jet identification efficiency, and the corresponding AUC. Figure 3 shows the AUC at the best validation epoch, evaluated on the held-out validation set, for b-versus-light (top) and b-versus-c (bottom) discrimination, as a function of the number of training jets.

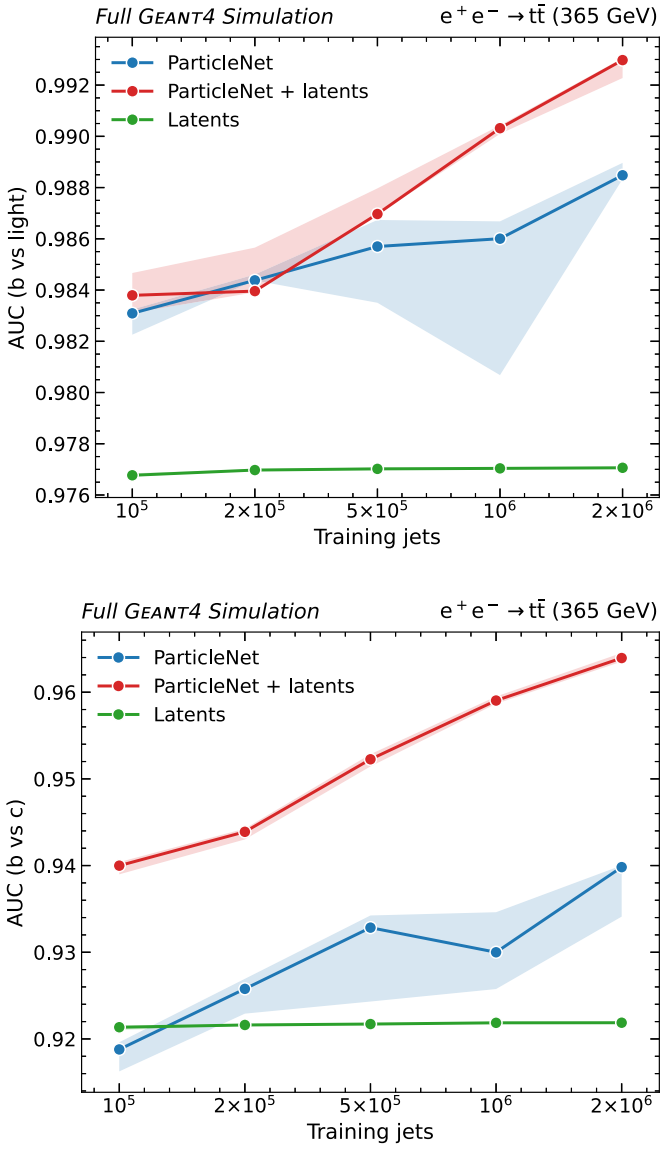


FIG. 3. AUC for b-vs-light (top) and b-vs-c (bottom) discrimination as a function of the number of training jets, comparing the three model variants (blue: BASELINE; red: LATENT-AUGMENTED; green: LINEAR-PROBE), evaluated on the validation set over all reconstructed jets ($p_T > 10$ GeV, $|\eta| < 2.5$). Each curve shows the median across five random seeds, and the shaded bands indicate the interquartile range.

The LATENT-AUGMENTED model dominates both AUC scaling curves at every training-set size, with the advantage more pronounced for the harder b-versus-c discrimination. Trained on only 100k jets, it already reaches an AUC of approximately 0.940 for b-versus-c, almost the same value as the AUC of the BASELINE model trained on the full training set. The BASELINE model is unable to close the AUC gap for both b-versus-light and b-versus-c discrimination at the largest available training size, indicating that the MLPF latent representations supply information that is not present in the baseline input feature set, despite the latter already including track-level discriminant variables.

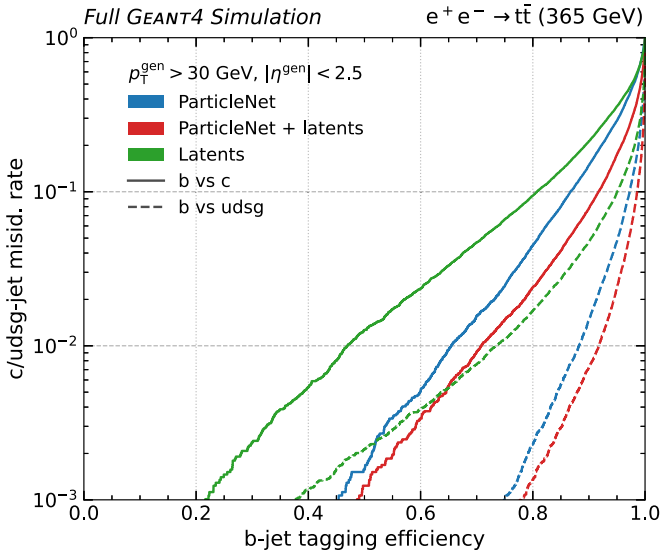


FIG. 4. Receiver operating characteristic (ROC) curves for the three model variants trained on the full training dataset and evaluated on the test set using the best model (the lowest-validation-loss seed) for each variant. Shown are the misidentification rates for light-flavor jets (dashed) and c jets (solid) as a function of b-jet identification efficiency, with a fiducial selection of $p_T^{gen} > 30$ GeV and $|\eta^{gen}| < 2.5$.

The LINEAR-PROBE curve is essentially flat on both plots across the entire range, saturating at AUC ≈0.977 for b-versus-light and ≈0.922 for b-versus-c and showing no improvement with training-set size. This flatness is expected by construction. A linear head has no nonlinear capacity to fit, so its performance reflects only the discrimination power already encoded in the frozen latent representations. However, the LINEAR-PROBE model achieves a nontrivial AUC well above random chance despite the fact that the MLPF backbone never uses jet-flavor labels during training, indicating that flavor-discriminating structure is intrinsically present in the latent representations. At the full training-set size the LINEAR-PROBE model trails the BASELINE model by approximately 0.012 in AUC for b-versus-light and 0.018 for b-versus-c.

In Fig. 4, we present the ROC curves for the three model variants trained on the full training dataset and evaluated on the test set with a fiducial kinematic selection of $p_T^{gen} > 30$ GeV and $|\eta^{gen}| < 2.5$, separately for the b-versus-light (dashed) and b-versus-c (solid) cases.

At a typical working point of 1% misidentification rate, the LATENT-AUGMENTED model improves the b-jet identification efficiency by approximately 3% against light-flavor jets and approximately 6% against c jets, relative to the BASELINE model. At the same working point, the LINEAR-PROBE model shows b-jet identification efficiency losses of approximately 15% against light-flavor jets and 19% against c jets, relative to the BASELINE model. This gap reflects the substantial discriminating power supplied by the explicit track-displacement features in the BASELINE model input set, which a linear projection of the latent space alone cannot fully recover.

V. JET ENERGY REGRESSION

We formulate jet energy regression as a per-jet scalar regression task where, given the PF candidates clustered into a jet, the model predicts a multiplicative correction factor that maps the reconstructed jet energy to the corresponding generator-level jet energy. This is the form adopted in recent ML-based jet energy regression studies at CMS using ParticleNet-based architectures [44].

We use the same jet dataset (Table II) and input features (Table III) as the jet flavor identification task, including the track-displacement variables. Although these variables are not required for jet energy regression, retaining them makes the model flavor aware and follows the convention adopted in Ref. [44]. The task-specific differences from jet flavor identification are the regression target, the loss function, and the output head, described in the following subsections.

A. Regression target and loss

For each reconstructed jet, we form the multiplicative correction factor,

$$y = \frac{E_{\text{jet}}^{\text{gen}}}{E_{\text{jet}}^{\text{reco}}}, \tag{1}$$

where $E_{\text{jet}}^{\text{reco}}$ is the energy of the reconstructed jet and $E_{\text{jet}}^{\text{gen}}$ is the energy of its matched generator-level jet. Generator-level jets are obtained by applying the same generalized-k_t algorithm to stable particles at generator level and are matched to reconstructed jets within $\Delta R < 0.4$. Reconstructed jets without a generator-level match within $\Delta R < 0.4$ are excluded from the sample.

We use the log-cosh loss, which is approximately quadratic for small residuals and approximately linear for large ones. This loss profile is robust to the heavy tail of rare misreconstructed jets without sacrificing sensitivity to small residuals. Predicting the ratio rather than the generator-level energy decouples the target from the absolute p_{T} scale, which prevents the loss from being dominated by the large absolute residuals of high-p_{T} jets and keeps the log-cosh objective in its quadratic regime across the full p_{T} range.

B. Model architecture and training

The architecture and training protocol are identical to those of the jet flavor identification task (Sec. IV), with the only change being that the final head outputs a scalar prediction instead of the three-class output. For the LINEAR-PROBE model, we initialize the linear head with zero weights and unit bias so the untrained model outputs the constant value 1.0 for every jet, close to the empirical target mean of 1.02. Training then measures the additional information that can be extracted by a linear readout of the MLPF latent representations.

The parameter counts are 360 137 for the BASELINE model, 507 849 for the LATENT-AUGMENTED model, and 129 for the LINEAR-PROBE model, slightly smaller than the corresponding jet flavor identification counts (Table IV) because the output head produces a single scalar prediction rather than three flavor outputs.

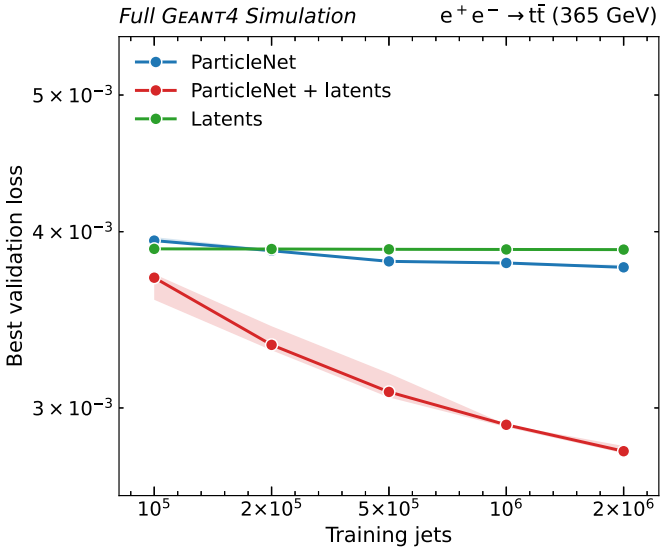


FIG. 5. Best validation loss as a function of the number of training jets, for the jet energy regression task, comparing the three model variants (blue: BASELINE; red: LATENT-AUGMENTED; green: LINEAR-PROBE), evaluated on the validation set. Each curve shows the median across five random seeds, and the shaded bands indicate the interquartile range.

To characterize the data-scaling behavior, we train on the same five training-set sizes as jet flavor identification: 100k, 200k, 500k, 1M, and the full 1.82M jets.

C. Results

We first study how performance scales with the size of the training set. Figure 5 shows the best validation loss as a function of the number of training jets, evaluated on the held-out validation set. The data-scaling behavior is similar to that observed in the jet flavor identification task. The LATENT-AUGMENTED model dominates at every training-set size. As in the jet flavor identification case, the LINEAR-PROBE curve is essentially flat by construction throughout the full range, plateauing slightly above the BASELINE curve at larger training sizes.

We evaluate the physics performance using the jet energy response, defined as $E_{\text{jet}}^{\text{reco}}/E_{\text{jet}}^{\text{gen}}$, where the corrected reconstructed-jet energy is obtained by multiplying the raw reconstructed energy by the model's predicted correction factor. The inclusive response distribution at the full training set is shown in the left panel of Fig. 6. All three distributions are centered close to unity, with the LATENT-AUGMENTED model achieving the tightest distribution with an IQR of 0.052 against 0.057 for the BASELINE model, an improvement of approximately 10%. The LINEAR-PROBE distribution sits at an IQR of 0.059, trailing the BASELINE by approximately 3%.

The right panel of Fig. 6 shows the jet energy resolution, defined as IQR/M, as a function of the generator-level jet p_{T}. The resolution improves rapidly with increasing p_{T} for all three models, as expected. The LATENT-AUGMENTED model achieves the best resolution at every p_{T} bin, with a relative improvement over the BASELINE model of approximately 10%

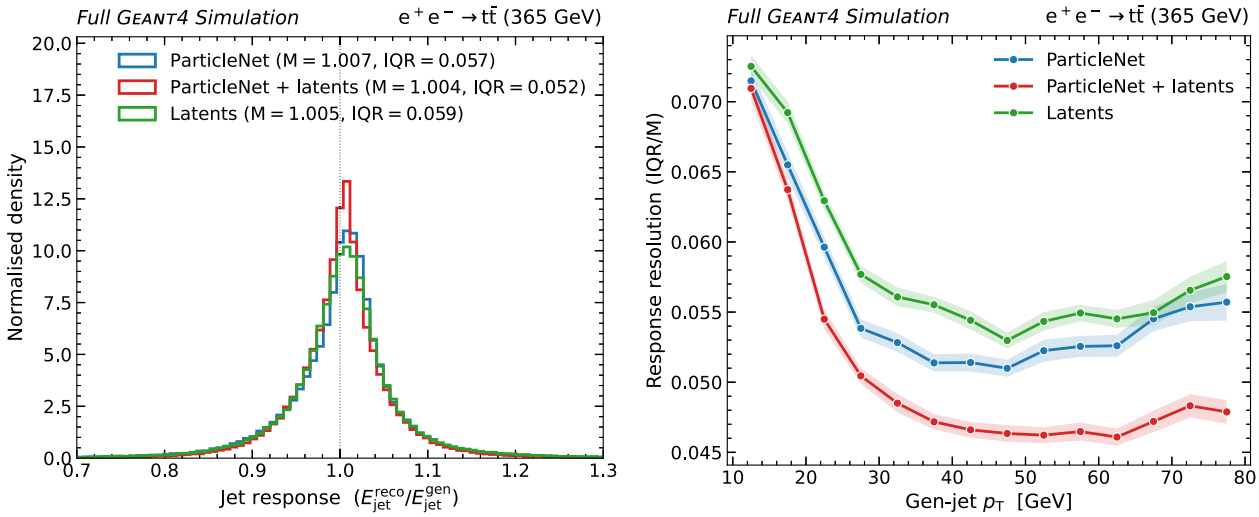


FIG. 6. Jet energy performance at the largest training-set size (blue: BASELINE; red: LATENT-AUGMENTED; green: LINEAR-PROBE), evaluated on the test set using the best model (the lowest-validation-loss seed) for each variant. Left: inclusive distribution of the jet energy response. Right: jet energy resolution as a function of the generator-level jet p_T. Shaded bands indicate the per-bin statistical uncertainty.

at intermediate p_T and approximately 15% at the highest p_T bins. The LINEAR-PROBE model trails the BASELINE model by approximately 5%–8% at intermediate p_T but approaches it at the highest p_T bins.

VI. MISSING MOMENTUM REGRESSION

The third downstream task we consider is the regression of $\vec{p}^{\,\text{miss}}$, which represents the imbalance of the visible transverse and longitudinal momenta and is a key observable for physics analyses involving escaping neutral particles such as neutrinos. At the LHC, ML-based missing transverse momentum (p_T^{miss}) estimators have become standard tools in precision analyses. For instance, the recent CMS W boson mass measurement [45] relied on the DeepMET algorithm [46] to estimate the p_T^{miss}, which improves the reconstruction of the W boson. The FCC-ee environment is pileup free, but the same fundamental challenge persists. The p_T^{miss} must be inferred from a list of reconstructed PF candidates whose individual momentum estimates carry detector-resolution uncertainties that accumulate when summed over the event.

Unlike at hadron colliders, where the longitudinal boost of the partonic initial state is unknown and only the p_T^{miss} is well defined, the e^+e^- initial state has fixed, known kinematics. The missing momentum vector $\vec{p}^{\,\text{miss}}$ is therefore physically meaningful and gives direct access to observables such as the missing mass, which is central to several FCC-ee physics analyses [29]. We accordingly extend the DeepMET regression target from the transverse plane to the full three-dimensional missing momentum. The remainder of this section defines the input features, the DeepMET-based BASELINE architecture, the training procedure, and the resulting physics performance.

A. Dataset statistics

We use the inclusive $t\bar{t}$ sample described in Sec. III A, which naturally provides a mix of events with genuine missing momentum from neutrinos in leptonic W boson decays and events with no true missing momentum from fully hadronic final states. We make no distinction between these classes in the performance plots. The p_T^{miss} distribution over the validation set has a mean of 31.1 GeV, a median of 23.4 GeV, and a 95th percentile at 82.7 GeV. We therefore restrict the range of the physics performance plots to 100 GeV.

B. Input features

Each PF candidate is described by eight continuous features and two categorical features. The continuous features are p_T, E, η, the Cartesian momentum components p_x, p_y, and p_z, and the IPs (d_0 and z_0) relative to the reconstructed PV (Sec. IV). The two kinematic variables p_T and E enter through a $\ln(1+x)$ transformation to soften their heavy tails; p_x, p_y, and p_z retain their physical units since they enter the $\vec{p}^{\,\text{miss}}$ kinematic sum directly. The categorical features are the PID class (one of the five MLPF output classes) and the electric charge, amounting to 14 input features per PF candidate.

C. Model architecture

We implement the DeepMET architecture in PYTORCH [47], closely following the original CMS implementation [46]. The network is a per-particle multilayer perceptron applied independently to each PF candidate, with hidden dimensions [64, 32, 16] and batch normalization and ReLU activations after each fully connected layer. For each PF candidate, the network outputs four quantities: A scalar weight w_i shared across the three spatial axes, and three bias terms $b_{i,x}$, $b_{i,y}$, and $b_{i,z}$. The event-level prediction is then assembled as a masked weighted sum along each axis,

$$\hat{p}_x^{\text{miss}} = -\sum_i (w_i\, p_{i,x} + b_{i,x}), \tag{2}$$

$$\hat{p}_y^{\text{miss}} = -\sum_i (w_i\, p_{i,y} + b_{i,y}), \tag{3}$$

$$\hat{p}_z^{\text{miss}} = -\sum_i (w_i\, p_{i,z} + b_{i,z}), \tag{4}$$

TABLE V. Model variants for the missing momentum regression task. The LINEAR-PROBE model is a single linear layer with 129 parameters (128 weights+1 bias).

Model	Architecture	Inputs	Parameters
BASELINE	DeepMET	14	4564
LATENT-AUGMENTED	DeepMET	142	12 756
LINEAR-PROBE	Linear	128	129

where the sum runs over the PF candidates in the event. We adapt the original CMS DeepMET architecture to the e^+e^- environment in two ways. We omit the pileup-per-particle identification [48] weight feature, since the e^+e^- environment is pileup free. We replace the PF candidate mass input with the PF candidate energy, since the mass of the PF candidate is set by its identified particle species and is therefore redundant with the PID embedding, whereas the energy carries independent kinematic information. The parameter counts for the three model variants are summarized in Table V.

D. Training

We train the model to minimize the mean squared error between the predicted and generator-level $\vec{p}^{\,\mathrm{miss}}$ vectors,

$$\mathcal{L} = \tfrac{1}{2}\left\langle \left(\hat{p}_x^{\mathrm{miss}} - p_x^{\mathrm{miss,gen}}\right)^2 + \left(\hat{p}_y^{\mathrm{miss}} - p_y^{\mathrm{miss,gen}}\right)^2 + \left(\hat{p}_z^{\mathrm{miss}} - p_z^{\mathrm{miss,gen}}\right)^2 \right\rangle. \quad (5)$$

Models are trained on the same heterogeneous GPU pool as the other downstream tasks, with wall-clock training time ranging from ∼30 min (smallest dataset, fastest GPU) to ∼13 h (full dataset, slower GPUs). We use the AdamW optimizer [43] with a constant learning rate of 10^{-3}, no weight decay, and batch size 128. Training is capped at 100 epochs with early stopping after 20 epochs without improvement in validation loss.

To characterize the data-scaling behavior, we train on five training-set sizes: 10k, 50k, 100k, 200k, and the full 350k events.

E. Results

Figure 7 shows the best validation loss, evaluated on the held-out validation set, as a function of the number of training events. The LATENT-AUGMENTED model reaches a validation loss of approximately 47.9 at the full training set, compared with approximately 65.0 for the BASELINE model, a 26% improvement. The LATENT-AUGMENTED model trained on only 50k events already beats the BASELINE model trained on the full sample, indicating that the MLPF latent representations supply information of central importance to the $\vec{p}^{\,\mathrm{miss}}$ regression task. The LINEAR-PROBE curve is essentially flat across the full range of training-set sizes, plateauing near 64.2.For the physics performance, we first inspect the inclusive distribution of the $p_{\mathrm{T}}^{\mathrm{miss}}$ response, defined as the ratio of the reconstructed $p_{\mathrm{T}}^{\mathrm{miss}}$ to its generator-level value $p_{\mathrm{T}}^{\mathrm{miss,reco}}/p_{\mathrm{T}}^{\mathrm{miss,gen}}$. All three distributions are centered slightly below unity, with medians of 0.967, 0.974, and 0.962 for the BASELINE, LATENT-AUGMENTED, and LINEAR-PROBE models, respectively.

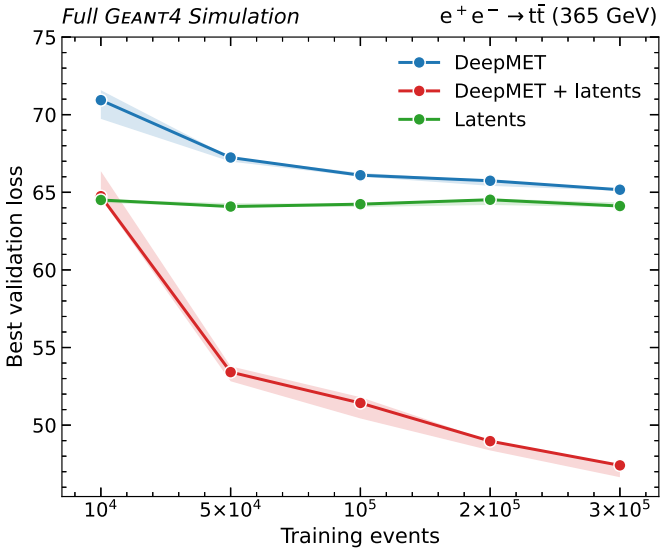


FIG. 7. Best validation loss as a function of the training dataset, for the missing momentum regression task, comparing the three model variants (blue: BASELINE; red: LATENT-AUGMENTED; green: LINEAR-PROBE), evaluated on the validation set. Each curve shows the median across five random seeds, and the shaded bands indicate the interquartile range.

Following the methodology established in Ref. [46], we further evaluate the recoil resolution in two orthogonal projections. The recoil $\vec{u} = -\vec{p}_{\mathrm{T}}^{\,\mathrm{miss,reco}}$ is decomposed into components parallel ($u_\parallel$) and perpendicular ($u_\perp$) to the generator-level $p_{\mathrm{T}}^{\mathrm{miss}}$ direction. The parallel resolution $\sigma(u_\parallel)$ measures how well the recoil tracks the generator-level $p_{\mathrm{T}}^{\mathrm{miss}}$ in magnitude, while the perpendicular resolution $\sigma(u_\perp)$ measures how well the direction is reconstructed. These are reported in bins of the generator-level $p_{\mathrm{T}}^{\mathrm{miss}}$ magnitude in the left and middle panels of Fig. 8. The LATENT-AUGMENTED model improves the recoil resolution by approximately 15%–20% across the full range in both projections, while the LINEAR-PROBE model gives a smaller but consistent improvement of approximately 2%–5% over the BASELINE model. The longitudinal resolution $\sigma(p_z^{\mathrm{miss}})$, defined as the standard deviation of the per-event residual $(p_z^{\mathrm{miss,gen}} - \hat{p}_z^{\mathrm{miss}})$ and shown in the right panel of Fig. 8, follows the same hierarchy with the LATENT-AUGMENTED model improving over the BASELINE model by approximately 10% across the full range.

VII. DISCUSSION

The three downstream studies presented in Secs. IV–VI together support the central claim of this work. The three tasks are qualitatively different: jet flavor identification, a per-jet multiclass classification that exploits track-level displacement variables; jet energy regression, a per-jet scalar regression; and missing momentum regression, a per-event vector regression. In all three cases, augmenting the task-specific baseline with the frozen MLPF latent representations yields a consistent improvement on the physics-performance metric appropriate to each task, despite the three tasks differing in

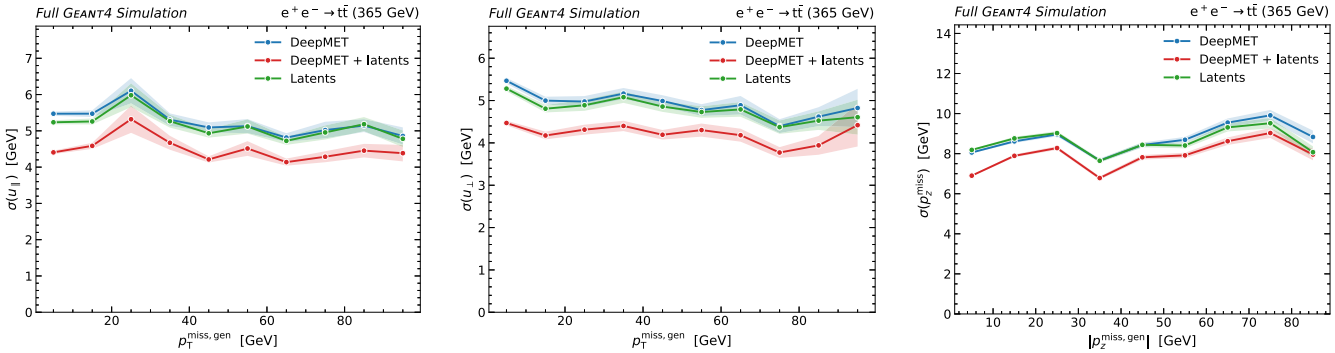


FIG. 8. Missing momentum performance at the largest training-set size (blue: BASELINE; red: LATENT-AUGMENTED; green: LINEAR-PROBE), evaluated on the test set using the best model (the lowest-validation-loss seed) for each variant. Left, middle: recoil resolution parallel ($\sigma(u_\parallel)$) and perpendicular ($\sigma(u_\perp)$) to the generator-level $p_{\mathrm{T}}^{\mathrm{miss}}$ direction, as a function of the $p_{\mathrm{T}}^{\mathrm{miss}}$ magnitude. Right: longitudinal missing-momentum resolution $\sigma(p_z^{\mathrm{miss}})$ as a function of $|p_z^{\mathrm{miss,gen}}|$. Shaded bands indicate the per-bin statistical uncertainty.

structure, the input modalities they emphasize, and their loss objectives. For jet flavor identification, at a 1% jet misidentification rate, the LATENT-AUGMENTED model improves the b-jet identification efficiency by approximately 3% against light-flavor jets and 6% against c jets. For jet energy regression, the LATENT-AUGMENTED model improves the jet energy resolution by approximately 10%–15% across the full jet p_{T} range, with the largest gains at the highest jet p_{T}. The greatest improvement is observed for missing momentum regression, where the recoil resolution improves by approximately 15%–20% in both projections, and the longitudinal resolution improves by approximately 10%, across the full range.

The LINEAR-PROBE model results strengthen the foundation-model interpretation by quantifying the fraction of task-relevant information that is already linearly accessible in the MLPF latent space, even though the MLPF backbone was never trained using jet-flavor labels, generator-level jet energies, or generator-level $\vec{p}^{\,\mathrm{miss}}$ information. The hierarchy varies by task, and the pattern itself is a useful diagnostic. For missing momentum regression, a 129-parameter linear layer outperforms the DeepMET-based BASELINE model at every training-set size while using approximately 35 times fewer parameters. For the two jet-level tasks, the LINEAR-PROBE model trails the BASELINE model by approximately 3% in the inclusive jet-energy resolution and more substantially in jet flavor identification. Two factors contribute to this gap. First, the BASELINE model input set includes explicit track-displacement features—the IP significances and the signed 2D and 3D IPs—that are highly discriminating for jet flavor identification but were not part of the MLPF input. MLPF sees only the raw perigee impact parameters d_0 and z_0, so these engineered features are not encoded in the latent representations, and a linear projection of the latent space cannot recover their discriminating power. Second, the LINEAR-PROBE model uniformly averages the latent representations across jet constituents, diluting constituent-level structure that the ParticleNet-based BASELINE can exploit through its learned aggregation. A more task-specific aggregation of the latent representations could further close the gap.

The scaling of the performance as a function of dataset size provides further evidence that the performance gap reflects information content rather than model capacity or training efficiency. For the jet energy regression and missing momentum estimation tasks, the BASELINE model performance saturates at large training-set sizes and the gap with the LATENT-AUGMENTED model does not close with additional labeled data. The capacity-control study of Appendix A confirms that this saturation is not a consequence of model size, and the relevant information is therefore not recoverable from the task-specific input features regardless of training-set size or model capacity. For jet flavor identification, both models continue to improve with dataset size, yet the gap between them persists. This persistence suggests that the advantage of the MLPF latent representations reflects information content rather than data efficiency. The backbone encodes flavor-relevant physics that remains inaccessible to the downstream model through particle-level features alone.

To understand the physics mechanisms driving the improvements from the latent representations, we perform several checks. For jet flavor identification, we zero the latent representations of neutral PF candidates at inference time for the LATENT-AUGMENTED model and find that both discriminations degrade, with the larger drop observed for b-versus-c. With a fiducial selection of $p_{\mathrm{T}}^{\mathrm{gen}} > 30\,\mathrm{GeV}$ and $|\eta^{\mathrm{gen}}| < 2.5$, the AUC evaluated on the test set decreases from 0.994 to 0.993 for b-versus-light and from 0.970 to 0.961 for b-versus-c, corresponding to a ΔAUC of approximately 0.001 and 0.009, respectively. This is consistent with the known physics of the two tasks. For b-versus-light, large-IP tracks in light-flavor jets arise from fake tracks, resolution effects, and some long-lived hadrons, so the explicit IP features in the BASELINE model already provide strong discrimination. For b-versus-c, both jet classes carry genuine secondary vertices (from particles with similar masses and lifetimes) and large-IP tracks, so in addition to IP-based discrimination the differences in quark fragmentation are also important. The b $\rightarrow$ c $\rightarrow$ daughters cascade produces neutral activity across multiple decay steps, and we suspect that the global attention

mechanism in MLPF encodes the collective neutral deposits from the full cascade jointly, a signal that is absent from the per-particle PF candidate interface and that the downstream model cannot reconstruct from kinematic features alone.

For missing momentum regression, we inspect the mean scalar weight that the LINEAR-PROBE model assigns to each PID class over the test set. The hierarchy follows the expected ordering of per-particle momentum reconstruction reliability, where charged hadrons receive the highest mean weight ($\bar{w} = 0.995$), neutral hadrons the lowest ($\bar{w} = 0.902$), and electrons sit notably below muons ($\bar{w} = 0.926$ versus 0.973), consistent with bremsstrahlung-induced reconstruction losses degrading per-particle momentum accuracy. The weight standard deviation is highest for neutral hadrons ($\sigma_w = 0.117$) and lowest for muons ($\sigma_w = 0.042$), indicating that the LINEAR-PROBE model varies weights at the per-particle level rather than applying a fixed per-class correction. This hierarchy suggests that MLPF encodes per-particle reconstruction reliability in the latent representations.

For jet energy regression, we find that the improvement is uniform across jet flavors (approximately 9%–11% for all three classes) and shows no significant trend with constituent multiplicity, consistent with the latent representations supplying generic per-particle reconstruction-quality information rather than a topology-specific signal.

VIII. CONCLUSION

Combined with the cross-detector transfer-learning study of Ref. [22], this work establishes a two-axis foundation-model picture for ML-based event reconstruction. The MLPF algorithm transfers across detector geometries with substantially less data than training from scratch, and the representations it learns transfer across qualitatively different downstream analysis tasks without retraining. These two dimensions together motivate treating reconstruction models as natural foundation models for collider physics, useful both for detector R&D and for downstream analysis.

The practical implication for analysis pipelines is direct. Modern collider analyses devote significant effort to feature engineering and to training task-specific ML architectures, often on dataset slices that are themselves expensive to generate, simulate, or label. In jet flavor identification, a 387-parameter linear classifier using only the MLPF latent representations achieves nontrivial discrimination without any hand-crafted track impact parameter features, suggesting that the reconstruction model already encodes flavor-relevant structure that would otherwise require careful feature engineering to exploit.

A similar implication holds for missing momentum regression. For instance, the recent CMS W boson mass measurement [45] relies on a DeepMET estimator, which was trained on a few tens of thousands of events with performance reported to be only weakly sensitive to larger training samples [46]. An estimator built on MLPF latent representations could improve the performance at fixed data scale with no modification to the upstream reconstruction and would continue to improve at larger training-set sizes where the BASELINE model shows clear signs of saturation. The $p_{\mathrm{T}}^{\mathrm{miss}}$ study presented here is in a pileup-free environment, so the translation to LHC analyses with significant pileup remains to be tested, but the gains observed are large enough to motivate that test.

ACKNOWLEDGMENTS

F.M. and J.D. were supported by the Research Corporation for Science Advancement (RCSA) under Grant No. CS-CSA-2023-109, U.S. Department of Energy (DOE), Office of Science, Office of High Energy Physics under Grant No. DE-SC0009919, and the U.S. National Science Foundation (NSF) Harnessing the Data Revolution (HDR) Institute for Accelerating AI Algorithms for Data Driven Discovery (A3D3) under Cooperative Agreement No. PHY-2117997. J.P. was supported by the Estonian Research Council Grant No. PSG864, TARISTU24-TK10 and by the European Regional Development Fund through the CoE program Grant No. TK202. M.K. was supported by the U.S. Department of Energy (DOE) under Grant No. DE-AC02-76SF00515.

The authors contributed according to the contributor roles taxonomy (CRediT) categories as follows. F.M.: conceptualization, investigation, methodology, software, validation, data curation, writing (original draft), writing (review and editing). J.P.: conceptualization, validation, supervision, writing (review and editing). M.K.: conceptualization, supervision, writing (review and editing). J.D.: conceptualization, resources, funding acquisition, supervision, writing (review and editing). The authors acknowledge the use of Anthropic's Claude (Opus 4.7 and 4.8) for assistance with code development and manuscript editing. The authors verified all AI-assisted output and take full responsibility for the scientific content of this paper.

DATA AVAILABILITY

The code and data that support the findings of this article are openly available [49].

APPENDIX A: CAPACITY CONTROL WITH BASELINE-LARGE

For each downstream task, the LATENT-AUGMENTED model has more parameters than the BASELINE model because it receives an additional 128-dimensional MLPF latent vector per particle as input. To verify that the performance gain attributed to the latent representations is not simply a consequence of the larger parameter count, we train an additional model variant, BASELINE-LARGE, with the same inputs as the BASELINE model but with wider hidden layers chosen to match the parameter count of the LATENT-AUGMENTED model. The BASELINE-LARGE model is trained with a single random seed per training-set size as a capacity check, and the comparison below uses the best model (i.e., the one trained on the full training dataset).

For jet flavor identification, BASELINE-LARGE widens the ParticleNet EdgeConv channels from (64, 128, 256) to (80, 160, 320), raising the parameter count from 360 651 to 510 955, approximately matching the 508 363 parameters of the LATENT-AUGMENTED model. For jet energy regression, BASELINE-LARGE widens the same EdgeConv channels, raising the parameter count from 360 137 to 510 441,

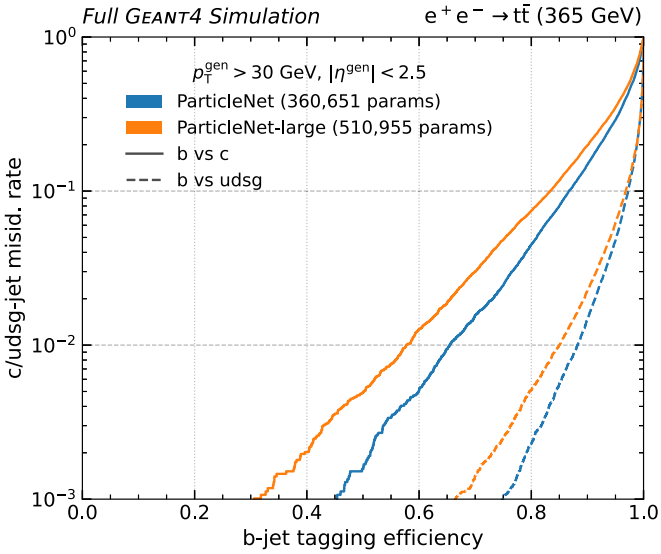


FIG. 9. Capacity-control comparison for jet flavor identification: ROC curves at the full training set, evaluated on the test set, for the BASELINE (blue) and BASELINE-LARGE (orange) models. Dashed lines show the light-flavor jet misidentification rate vs b-jet identification efficiency; solid lines show the c-jet misidentification rate. A fiducial selection of $p_T^{gen} > 30$ GeV and $|\eta^{gen}| < 2.5$ is applied. The BASELINE-LARGE model is trained with a single random seed, and BASELINE shows its best seed (i.e., lowest validation loss).

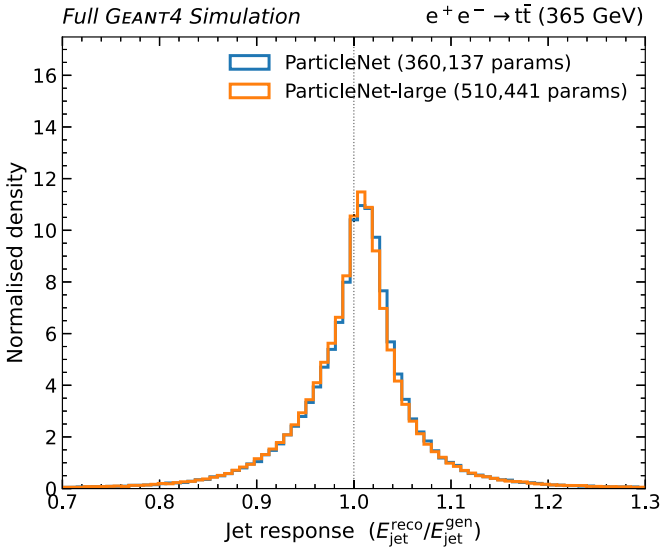


FIG. 10. Capacity-control comparison for jet energy regression: inclusive jet-energy response distribution at the full training set, evaluated on the test set, for the BASELINE (blue) and BASELINE-LARGE (orange) models. The BASELINE-LARGE model is trained with a single random seed, and BASELINE shows its best seed (i.e., lowest validation loss).

approximately matching the 507 849 parameters of the LATENT-AUGMENTED model. For the missing momentum regression task, BASELINE-LARGE widens the first two DeepMET hidden dimensions from (64, 32, 16) to (128, 64, 16), raising the parameter count from 4564 to 13 044, approximately matching the 12 756 parameters of the LATENT-AUGMENTED model. The additional capacity is distributed across the hidden layers rather than concentrated in the output head so that the BASELINE-LARGE model has more room to learn richer features rather than only a wider final projection, making the capacity-control comparison as generous as possible to the BASELINE-LARGE model. All other training hyperparameters are identical to those of the BASELINE model.

Across all three tasks, the BASELINE and BASELINE-LARGE models perform comparably on their physics-performance metrics, with the BASELINE-LARGE slightly trailing on jet flavor identification. For jet flavor identification (Fig. 9), the AUCs at the full training set are 0.990 (BASELINE) versus 0.988 (BASELINE-LARGE) against light-flavor jets, and 0.952 versus 0.938 against c jets. The BASELINE-LARGE model is in fact worse than the BASELINE model on both AUCs, marginally against light-flavor jets and more substantially against c jets, confirming that the larger parameter count does not improve performance.

For jet energy regression (Fig. 10), the inclusive jet-energy response distributions agree to within 0.002 in the median (1.007 for the BASELINE model versus 1.005 for BASELINE-LARGE model), with both IQRs at approximately 0.057.

For missing momentum regression (Fig. 11), the inclusive p_T^{miss} response distributions are essentially indistinguishable, with medians of 0.967 for the BASELINE model and 0.970 for the BASELINE-LARGE model, and IQRs of approximately 0.123 for both. This confirms that the performance gains observed in the paper are not a consequence of the increased parameter count, but reflect genuine additional information supplied by the MLPF latent representations.

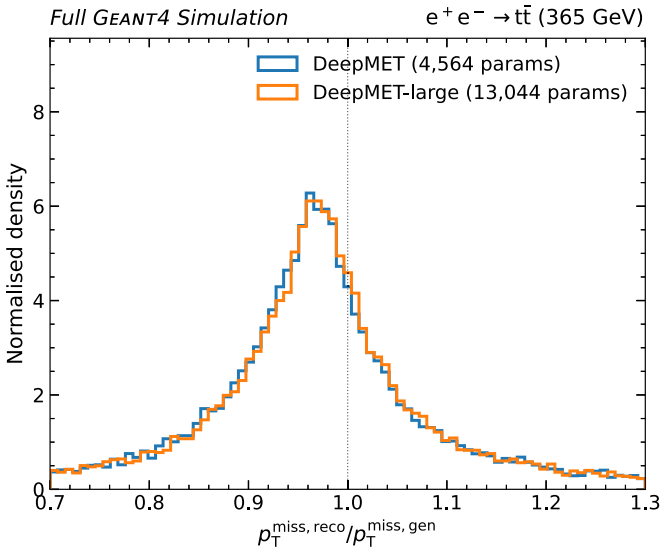


FIG. 11. Capacity-control comparison for missing momentum regression: inclusive p_T^{miss} response distribution at the full training set, evaluated on the test set, for the BASELINE (blue) and BASELINE-LARGE (orange) models. The BASELINE-LARGE model is trained with a single random seed, and BASELINE shows its best seed (i.e., lowest validation loss).

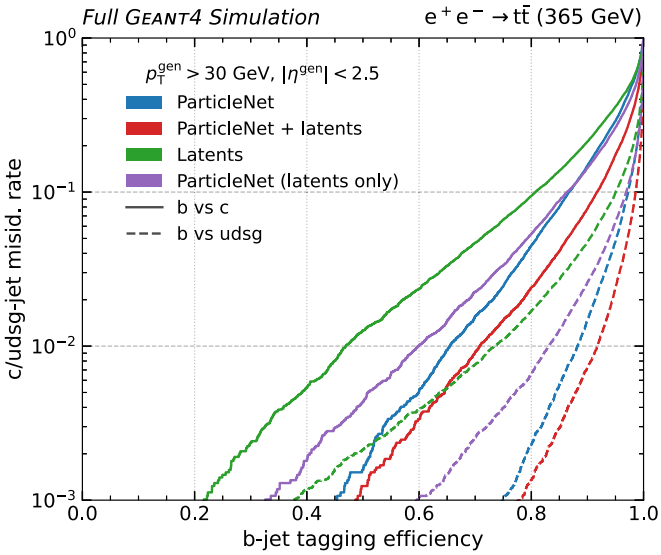


FIG. 12. ROC curves for the NONLINEAR-PROBE (purple) model alongside the BASELINE (blue), LATENT-AUGMENTED (red), and LINEAR-PROBE (green) models, trained on the full training dataset and evaluated on the test set, showing the misidentification rate for light-flavor jets (dashed) and c jets (solid) as a function of b-jet identification efficiency, with a fiducial selection of $p_T^{gen} > 30\,\text{GeV}$ and $|\eta^{gen}| < 2.5$.

APPENDIX B: NONLINEAR PROBE OF THE LATENT REPRESENTATIONS

The LINEAR-PROBE model of Sec. III C captures only the *linearly* accessible information in the latent space, so its gap relative to the BASELINE model could reflect either information absent from the latent representations or information present but not linearly separable. To disentangle these, we train NONLINEAR-PROBE, the full task-specific architecture applied

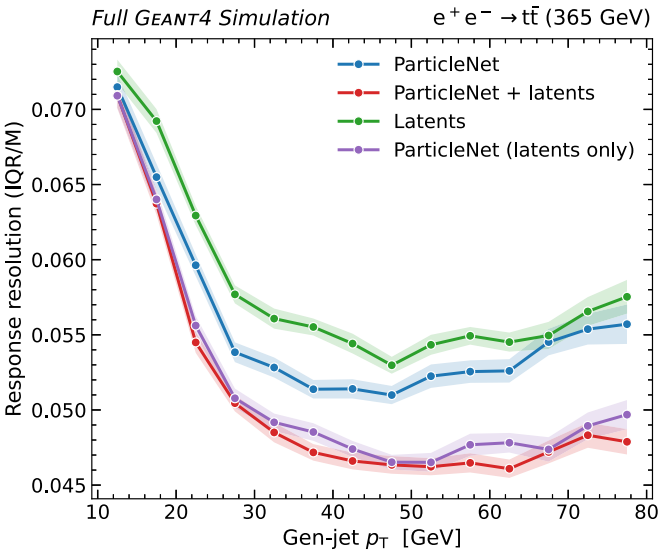


FIG. 13. Jet energy resolution as a function of the generator-level jet p_T of the NONLINEAR-PROBE (purple) model alongside the BASELINE (blue), LATENT-AUGMENTED (red), and LINEAR-PROBE (green) models at the full training set, evaluated on the test set. Shaded bands indicate the per-bin statistical uncertainty.

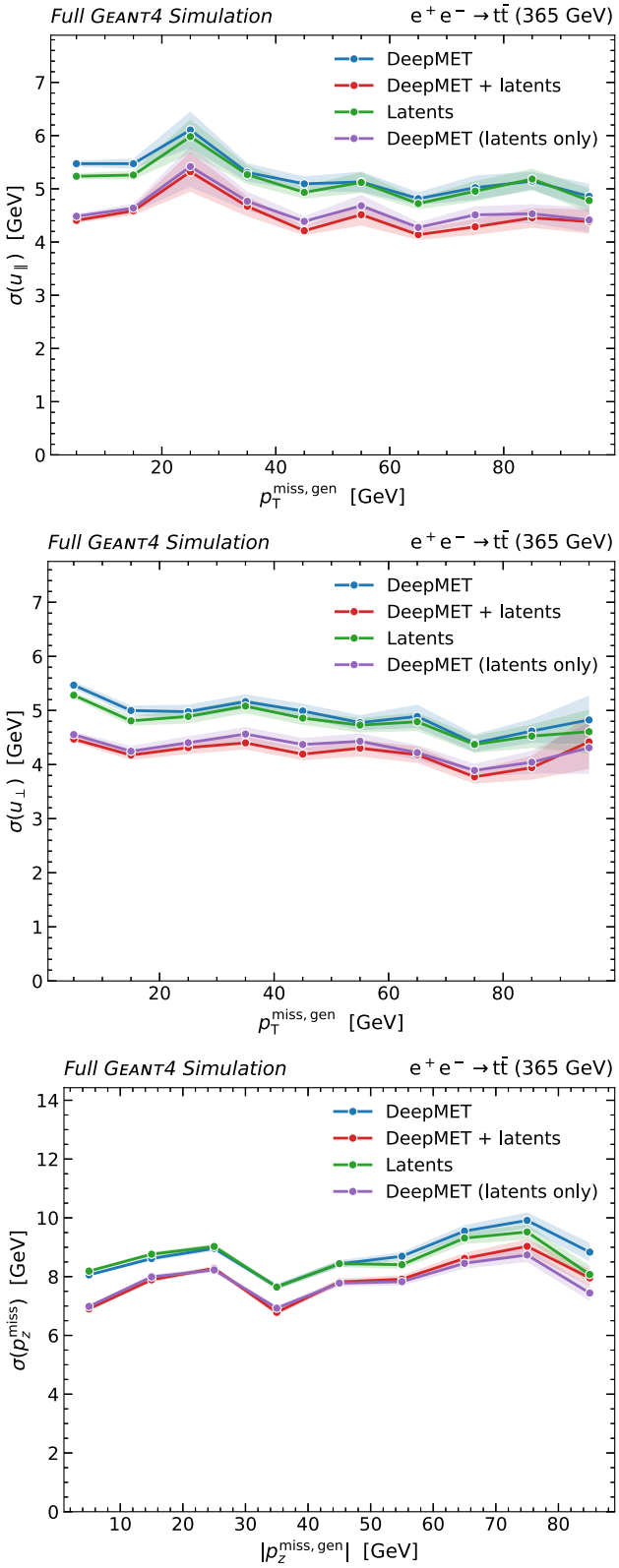


FIG. 14. Missing momentum performance of the NONLINEAR-PROBE (purple) model alongside the BASELINE (blue), LATENT-AUGMENTED (red), and LINEAR-PROBE (green) models at the full training set, evaluated on the test set. Top, middle: recoil resolution parallel ($\sigma(u_\parallel)$) and perpendicular ($\sigma(u_\perp)$) to the generator-level p_T^{miss} direction, as a function of the p_T^{miss} magnitude. Bottom: longitudinal missing-momentum resolution $\sigma(p_z^{miss})$ as a function of $|p_z^{miss,gen}|$. Shaded bands indicate the per-bin statistical uncertainty.

to the 128-dimensional MLPF latent representations alone, with all kinematic, PID, and track-displacement inputs removed.

The architecture is ParticleNet for jet flavor identification and jet energy regression, and DeepMET for missing momentum regression, identical to the BASELINE models except that the input is the latent vector. For the ParticleNet tasks, the graph building in the first layer is built from the constituent angular coordinates (η, ϕ) and the subsequent layers from the dynamically updated latent features, following the standard ParticleNet construction. For DeepMET, the latent vector determines the per-candidate weights entering the $\vec{p}^{\,\text{miss}}$ sum. As for the other variants, we train five models with different random seeds and report the best model on the test set.

For jet flavor identification (Fig. 12), at a 1% misidentification rate the NONLINEAR-PROBE model reaches a b-jet identification efficiency of approximately 83% against light-flavor jets and 60% against c jets, compared with 88% and 66% for the BASELINE model and substantially exceeding the LINEAR-PROBE model (74% and 47%). The NONLINEAR-PROBE model thus recovers most of the BASELINE discrimination using no kinematic or track-displacement features, while the LATENT-AUGMENTED model remains the best-performing variant, indicating that the explicit track-displacement features still add complementary information beyond the latent space.

For jet energy regression, the inclusive jet energy response has an IQR of 0.052, matching the LATENT-AUGMENTED model and improving on the BASELINE model by approximately 10%, and the resolution is comparable to the LATENT-AUGMENTED model across the full p_T range as shown in Fig. 13.

For missing momentum regression (Fig. 14), the recoil resolution improves over the BASELINE model by approximately 12% in both projections and the longitudinal resolution by approximately 11%, comparable to the LATENT-AUGMENTED model.

Across the three tasks, a model trained on the latent representations alone matches the LATENT-AUGMENTED model for the two regression tasks, indicating that the latent space captures essentially all of the task-relevant information there. For jet flavor identification, it approaches but does not reach the BASELINE model, because the BASELINE model additionally exploits explicit track-displacement variables—the IP significances and the signed 2D and 3D IPs—that are constructed downstream from the track covariance and the jet axis and are therefore not encoded in the latent representations. Even without these inputs, the NONLINEAR-PROBE model recovers most of the BASELINE model discrimination, confirming that flavor-relevant information is already accessible to some extent from the latent space.

[1] A. M. Sirunyan *et al.* (CMS Collaboration), Particle-flow reconstruction and global event description with the CMS detector, J. Instrum. **12**, P10003 (2017).

[2] D. Buskulic *et al.* (ALEPH Collaboration), Performance of the ALEPH detector at LEP, Nucl. Instrum. Methods Phys. Res. Sect. A **360**, 481 (1995).

[3] J. Pata, J. Duarte, J.-R. Vlimant, M. Pierini, and M. Spiropulu, MLPF: Efficient machine-learned particle-flow reconstruction using graph neural networks, Eur. Phys. J. C **81**, 381 (2021).

[4] R. Bommasani *et al.*, On the opportunities and risks of foundation models, Stanford CRFM report, arXiv:2108.07258.

[5] T. Golling, L. Heinrich, M. Kagan, S. Klein, M. Leigh, M. Osadchy, and J. A. Raine, Masked particle modeling on sets: Towards self-supervised high energy physics foundation models, Mach. Learn.: Sci. Technol. **5**, 035074 (2024).

[6] J. Birk, A. Hallin, and G. Kasieczka, OmniJet-α: The first cross-task foundation model for particle physics, Mach. Learn.: Sci. Technol. **5**, 035031 (2024).

[7] M. Leigh, S. Klein, F. Charton, T. Golling, L. Heinrich, M. Kagan, I. Ochoa, and M. Osadchy, Is tokenization needed for masked particle modelling? Mach. Learn. Sci. Tech. **6**, 025075 (2025).

[8] P. Harris, J. Krupa, M. Kagan, B. Maier, and N. Woodward, Resimulation-based self-supervised learning for pretraining physics foundation models, Phys. Rev. D **111**, 032010 (2025).

[9] T. Kishimoto, M. Morinaga, M. Saito, and J. Tanaka, Pre-training strategy using real particle collision data for event classification in collider physics, in NeurIPS 2023 Workshop: Machine Learning and the Physical Sciences Workshop, arXiv:2312.06909.

[10] V. Mikuni and B. Nachman, Solving key challenges in collider physics with foundation models, Phys. Rev. D **111**, L051504 (2025).

[11] J. Ho, R. Roberts, S. Han, and H. Wang, Pretrained event classification model for high energy physics analysis, J. Instrum. **21**, P08006 (2026).

[12] T.-H. Hsu, B.-H. Zhou, Q. Liu, Y. Xu, S. Li, G. W.-S. Hou, B. Nachman, S.-C. Hsu, V. Mikuni, Y.-T. Chou, and Y. Zhang, EveNet: A foundation model for particle collision data analysis, arXiv:2601.17126.

[13] O. Amram, L. Anzalone, J. Birk, D. A. Faroughy, A. Hallin, G. Kasieczka, M. Krämer, I. Pang, H. Reyes-Gonzalez, and D. Shih, Aspen open jets: Unlocking LHC data for foundation models in particle physics, Mach. Learn.: Sci. Technol. **6**, 030601 (2025).

[14] M. Vigl, N. Hartman, and L. Heinrich, Finetuning foundation models for joint analysis optimization in high energy physics, Mach. Learn.: Sci. Technol. **5**, 025075 (2024).

[15] C. Li *et al.*, Accelerating resonance searches via signature-oriented pre-training, arXiv:2405.12972.

[16] Z. Zhao, F. Mokhtar, R. Kansal, H. Li, and J. Duarte, Large-scale pretraining and finetuning for efficient jet classification in particle physics, in Proceedings of 22nd International Workshop on Advanced Computing and Analysis Techniques inPhysics Research, arXiv:2408.09343.

[17] V. Mikuni and B. Nachman, Method to simultaneously facilitate all jet physics tasks, Phys. Rev. D **111**, 054015 (2025).

[18] W. Bhimji, C. Harris, V. Mikuni, and B. Nachman, Foundation model framework for all tasks involving jet physics, Phys. Rev. D **113**, 032020 (2026).

[19] A. Hallin, Foundation models for high-energy physics, in Proceedings of 2nd European AI for Fundamental Physics Conference, arXiv:2509.21434.

[20] Particle Data Group Collaboration, Machine learning, in Review of particle physics, Int. J. Mod. Phys. A **41**, 2630011 (2026).

[21] J. Pata, E. Wulff, F. Mokhtar, D. Southwick, M. Zhang, M. Girone, and J. Duarte, Improved particle-flow event reconstruction with scalable neural networks for current and future particle detectors, Commun. Phys. **7**, 124 (2024).

[22] F. Mokhtar, J. Pata, D. Garcia, E. Wulff, M. Zhang, M. Kagan, and J. Duarte, Fine-tuning machine-learned particle-flow reconstruction for new detector geometries in future colliders, Phys. Rev. D **111**, 092015 (2025).

[23] D. Garcia, G. Krzmanc, P. Zehetner, J. Kieseler, and M. Selvaggi, Towards particle flow event reconstruction at the Future Circular Collider with GNNs, in *NeurIPS 2023 Workshop: New Frontiers in Graph Learning* (2023).

[24] P. Wahlen and T. Suehara, Particle-flow reconstruction with Transformer, EPJ Web Conf. **315**, 03010 (2024).

[25] D. Garcia, L. Herrmann, G. Krzmanc, and M. Selvaggi, End-to-end event reconstruction for precision physics at future colliders, arXiv:2603.04084.

[26] A. Hayrapetyan *et al.* (CMS Collaboration), Full event interpretation with machine-learning-based particle-flow reconstruction in the CMS detector, Eur. Phys. J. C **86**, 1025 (2026).

[27] L. van der Maaten and G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. **9**, 2579 (2008).

[28] N. Bacchetta *et al.*, CLD—A detector concept for the FCC-ee, arXiv:1911.12230.

[29] A. Abada *et al.* (FCC Collaboration), FCC-ee: The lepton collider: Future Circular Collider conceptual design report volume 2, Eur. Phys. J. Spec. Top. **228**, 261 (2019).

[30] C. Bierlich, S. Chakraborty, N. Desai, L. Gellersen, I. Helenius, P. Ilten, L. Lönnblad, S. Mrenna, S. Prestel, C. T. Preuss, T. Sjöstrand, P. Skands, M. Utheim, and R. Verheyen, A comprehensive guide to the physics and usage of PYTHIA 8.3, SciPost Phys. Codebases 8 (2022).

[31] S. Agostinelli *et al.* (Geant4 Collaboration), GEANT4—A simulation toolkit, Nucl. Instrum. Methods Phys. Res. Sect. A **506**, 250 (2003).

[32] J. Allison *et al.* GEANT4 developments and applications, IEEE Trans. Nucl. Sci. **53**, 270 (2006).

[33] J. Allison *et al.* (GEANT4 Collaboration), Recent developments in GEANT4, Nucl. Instrum. Methods Phys. Res. Sect. A **835**, 186 (2016).

[34] F. Pedregosa *et al.*, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. **12**, 2825 (2011).

[35] G. Alain and Y. Bengio, Understanding intermediate layers using linear classifier probes, arXiv:1610.01644.

[36] M. Cacciari, G. P. Salam, and G. Soyez, The anti-k_T jet clustering algorithm, J. High Energy Phys. 04 (2008) 063.

[37] M. Cacciari, G. P. Salam, and G. Soyez, The catchment area of jets, J. High Energy Phys. 04 (2008) 005.

[38] F. Bedeschi, L. Gouskos, and M. Selvaggi, Jet flavour tagging for future colliders with fast simulation, Eur. Phys. J. C **82**, 646 (2022).

[39] S. Aumiller, D. Garcia, and M. Selvaggi, "Jet flavor tagging performance at FCC-ee", 2024, https://doi.org/10.17181/e537w-n6886.

[40] N. Bacchetta, P. Collins, and P. Riedler, Tracking and vertex detectors at FCC-ee, Eur. Phys. J. Plus **137**, 231 (2022).

[41] H. Qu and L. Gouskos, ParticleNet: Jet tagging via particle clouds, Phys. Rev. D **101**, 056019 (2020).

[42] L. Smarr *et al.*, The Pacific research platform: Making high-speed networking a reality for the scientist, in *Practice and Experience on Advanced Research Computing (PEARC18)* (ACM Press, New York, 2018).

[43] I. Loshchilov and F. Hutter, Decoupled weight decay regularization, arXiv:1711.05101.

[44] CMS Collaboration, A unified approach for jet tagging in Run 3 at $\sqrt{s} = 13.6$ TeV in CMS, CMS Detector Performance Summary CMS-DP-2024-066, 2024.

[45] V. Chekhovsky *et al.* (CMS Collaboration), High-precision measurement of the W boson mass with the CMS experiment, Nature (London) **652**, 321 (2026).

[46] A. Hayrapetyan *et al.* (CMS Collaboration), Improving missing transverse momentum estimation with a deep neural network, Phys. Rev. D **113**, 072010 (2026).

[47] A. Paszke *et al.*, PyTorch: An imperative style, high-performance deep learning library, in *Advances in Neural Information Processing Systems*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Curran Associates, Red Hook, NY, 2019), Vol. 32, p. 8024.

[48] D. Bertolini, P. Harris, M. Low, and N. Tran, Pileup per particle identification, J. High Energy Phys. 10 (2014) 059.

[49] F. Mokhtar, J. Pata, M. Kagan, and J. Duarte, MLPF as a foundation model—Downstream evaluation dataset (FCC-ee CLD, $e^+e^- \to t\bar{t}$, $\sqrt{s} = 365$ GeV) [Dataset], Zenodo (2026), https://doi.org/10.5281/zenodo.20663506.