LEARNING BEAM DYNAMICS IN THE LATENT SPACE OF BEAM DISTRIBUTIONS

N. Wang*†, I. Cao, G. Hoffstaetter, Cornell University, Ithaca, NY, USA

Abstract

We propose a framework for surrogate beam dynamics that generalizes across lattice configurations. A VAE encodes the full 6D phase-space distribution into a compact latent vector; a causal transformer then propagates this state autoregressively through a sequence of tokenized lattice elements, enabling prediction through arbitrary element sequences without retraining. Demonstrated on FODO-style lattices, the model achieves accurate beam distribution prediction over 32-element sequences with end-to-end inference in approximately 67 ms per trajectory on a single GPU.

INTRODUCTION

Predicting the evolution of a beam's six-dimensional (6D) phase-space distribution through an accelerator lattice is the central computational task of beam dynamics. Simulation codes are accurate but computationally demanding, limiting their use for real-time control and iterative optimization [1].

Generative machine learning models have demonstrated accurate reconstruction of 6D beam phase-space distributions from non-invasive measurements [2–4]. Fixed-machine neural surrogate models have achieved large speedups over tracking codes for specific lattice configurations [5–7]. However, these approaches operate over a fixed set of machine parameters and cannot transfer to new lattice configurations.

Learning dynamics in the latent space of an autoencoder has been explored in other physical sciences; in fluid dynamics, for instance, Wiewel et al. [8] couple a convolutional autoencoder with an LSTM, and Solera-Rico et al. [9] pair a β-VAE with a transformer, both forecasting the temporal evolution of flow fields. Within accelerator physics, the closest precedents are the CLARM model [10] and its transformer-based successor, the Latent Evolution Model (LEM) [11], which pair a conditional VAE with an LSTM and transformer, respectively, to propagate 6D phase-space projections module-by-module along a fixed linac. The present work departs from this fixed-machine paradigm: explicit element tokenization treats each lattice element as a discrete input to the dynamics model, allowing the transformer to predict trajectories through arbitrary element sequences without retraining.

We propose a framework that learns beam dynamics as a sequence-to-sequence problem (Fig. 1). The beam's 6D distribution is encoded into a compact latent vector, and a causal transformer propagates this vector autoregressively through a sequence of tokenized lattice elements. Element parameters are embedded through a mapping that accommodates arbitrary element types, sequences, and parameter values. The framework makes no assumptions about the underlying physics — it learns whatever dynamics are present in the training data, whether from linear optics, nonlinear elements, or (in future work) collective effects.

* nw285@cornell.edu

† Ningdong Wang is supported by the U.S. Department of Energy, Office of Science, High Energy Physics, under Award Number DE-SC0024907, the Tigner Traineeships in Accelerator Science program.

METHOD

Beam Distribution Encoding

The full 6D phase space of a beam is spanned by the coordinates $(x, x', y, y', z, \delta)$. We represent a beam snapshot by the $\binom{6}{2} = 15$ unique 2D projections of the 6D distribution, each histogrammed into a 64×64 frequency map and normalized by the total particle count. The resulting input tensor has shape $15 \times 64 \times 64$, carrying all pairwise correlations in the distribution.

A VAE [12,13] encodes this tensor into a 256-dimensional latent vector. The encoder consists of strided convolutional blocks that double the channel width and halve the spatial resolution at each stage. The resulting feature map is flattened and concatenated with a 12-dimensional auxiliary vector of physical beam sizes σ and centroids $\langle \mathbf{x} \rangle$ in all six coordinates, then passed through a fully connected layer to two parallel heads producing the approximate posterior mean $\mu \in \mathbb{R}^{256}$ and log-variance $\log \sigma^2 \in \mathbb{R}^{256}$. Latent samples are drawn as $z = \mu + \sigma\varepsilon$, $\varepsilon \sim \mathcal{N}(0, I)$. The decoder mirrors the encoder with bilinear upsampling and a sigmoid output; auxiliary linear heads branching from z directly predict beam sizes and centroids. The training objective is

$$\begin{aligned}\mathscr{L} = \mathbb{E}\left[\|x - \hat{x}\|^2\right] + \beta\, D_{\mathrm{KL}}[q(z|x)\|p(z)] \\ + \gamma\, \|\hat{\sigma} - \sigma\|^2 + \delta\, \|\langle\hat{\mathbf{x}}\rangle - \langle\mathbf{x}\rangle\|^2,\end{aligned} \tag{1}$$

with $\beta = 1 \times 10^{-5}$, $\gamma = \delta = 1 \times 10^{-4}$. The auxiliary terms anchor the latent space to physically meaningful beam parameters.

Element Tokenization

Each lattice element is described by a 7-dimensional parameter vector: length L, quadrupole gradient K_1, sextupole gradient K_2, dipole bend angle ϕ, RF cavity voltage V_{rf}, RF frequency f_{rf}, and RF phase φ_{rf}. Non-RF elements have $V_{\mathrm{rf}} = f_{\mathrm{rf}} = 0$. Per-parameter normalization is applied before embedding: V_{rf} and f_{rf} are compressed with a $\log(1 + x)$ transform, which reduces their large dynamic range and preserves the zero value for non-RF elements; lengths, magnet strengths, angles, and phases are divided by representative scales (1.0 m, 10.0 m^{-2}, 10.0 m^{-3}, 2π, 2π, respectively).

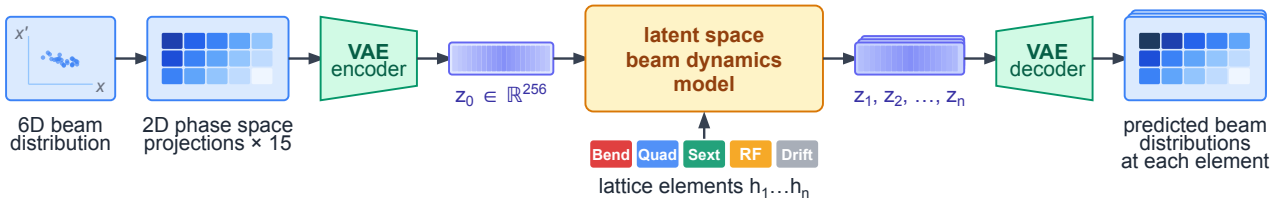

Figure 1: End-to-end pipeline. A beam snapshot is encoded into a latent vector by the VAE. A causal transformer propagates the latent state through a sequence of element tokens, producing a predicted latent trajectory. The VAE decoder reconstructs the beam distribution at any element exit.

The normalized vector is projected to the model dimension d_{model} by a 3-layer MLP with GELU activations.

Positional information is encoded through the cumulative beamline position $s_i = \sum_{j<i} L_j$ (the s-coordinate at the entrance of element i). A Fourier positional encoding uses 32 frequency pairs geometrically spaced in wavelength from 1 cm to 1 km, resolving structure from individual elements to full machine length. The encoding is projected to d_{model} via a learned linear layer and added to the MLP output to yield the element token h_i.

This tokenization accommodates arbitrary element parameters. New element types can be incorporated by extending the 7-dimensional parameter vector and retraining.

Latent Dynamics Model

Given the initial latent state $z_0 \in \mathbb{R}^{256}$ and N element tokens $h_1, \ldots, h_N$, the dynamics model predicts the latent beam state z_t at the exit of each element $t = 1, \ldots, N$ (Fig. 2).

We use a TrackingTransformer with $d_{\text{model}} = 512$, $N_{\text{layers}} = 6$, and $N_{\text{heads}} = 8$. At position t, the input token is formed by projecting z_{t-1} to d_{model}, concatenating with the element token h_t, and passing through a learned fusion projection. These tokens are processed by a GPT-style pre-LayerNorm causal transformer. A causal attention mask ensures that position t attends only to positions $0, \ldots, t$. An output linear head produces a residual update $\Delta z_t \in \mathbb{R}^{256}$, and the latent state is updated as $z_t = z_{t-1} + \Delta z_t$.

The model supports two forward modes: teacher-forcing (TF, ground-truth z_{t-1} provided at each step, used during training), and fully autoregressive inference (AR, z_0 only, all subsequent states predicted sequentially). The training loss is MSE on the full latent trajectory $z_{1:N}$.

Training data consists of 10 000 samples generated by tracking particles through randomly parameterized FODO-style sectioned lattices using Bmad [14, 15], with each sample comprising $N = 32$ elements (drifts, quadrupoles, dipoles, sextupoles, and RF cavities). Physical constraints are imposed to ensure the stability of beam propagation and prevent unbounded beam size growth.

RESULTS

VAE Encoding Quality

The VAE was trained with latent dimension 256 and $\beta = 1 \times 10^{-5}$. Beam sizes (σ_x, $\sigma_{x'}$, σ_y, $\sigma_{y'}$, σ_z, σ_δ) are reconstructed with $R^2 \geq 0.9995$ across all six phase-space

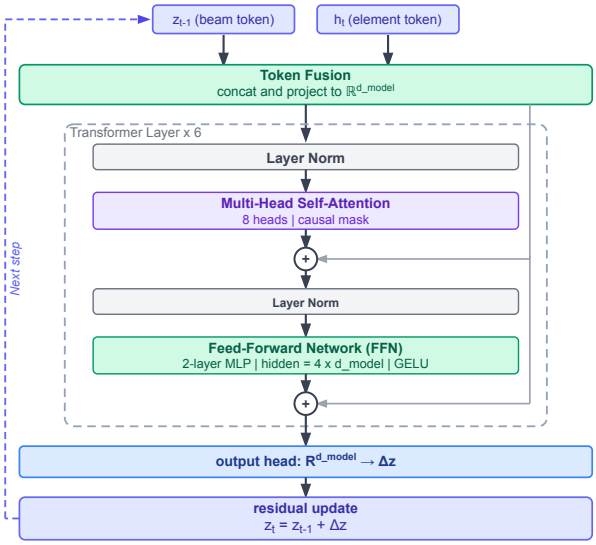

Figure 2: TrackingTransformer. The previous latent z_{t-1} is fused with element token h_t, and processed by a 6-layer pre-LN causal transformer. A linear head outputs Δz_t; the state is updated as $z_t = z_{t-1} + \Delta z_t$.

dimensions; centroid positions achieve $R^2 \approx 1.0000$. The overall reconstruction MSE is 2.99×10^{-9}, and quality is consistent across projections: the best-performing channel (y–δ) achieves 9.51×10^{-10}, while the most demanding channel (x–x') reaches 6.98×10^{-9}, reflecting the richer correlation structure of the transverse phase space. The latent representation is compact: 34, 37, and 41 dimensions suffice to capture 90%, 95%, and 99% of the posterior variance, respectively, indicating that the beam dynamics of these FODO-style lattices are well described by a low-dimensional manifold embedded in $\mathbb{R}^{256}$.

Latent Trajectory Prediction

We evaluated the performance of the best checkpoint on the held-out validation set (Fig. 3). In teacher-forcing mode, the model tracks the ground-truth latent trajectory with MSE $= 6.46 \times 10^{-4}$. In fully autoregressive mode, the mean MSE over all 32 elements is 4.53×10^{-3} and the final-element MSE is 1.53×10^{-2}. Fiure 4 shows the per-step MSE across the lattice. The TF MSE increases with element index because the training distribution grows more diverse along the lattice, which is an artifact of our data generation scheme.

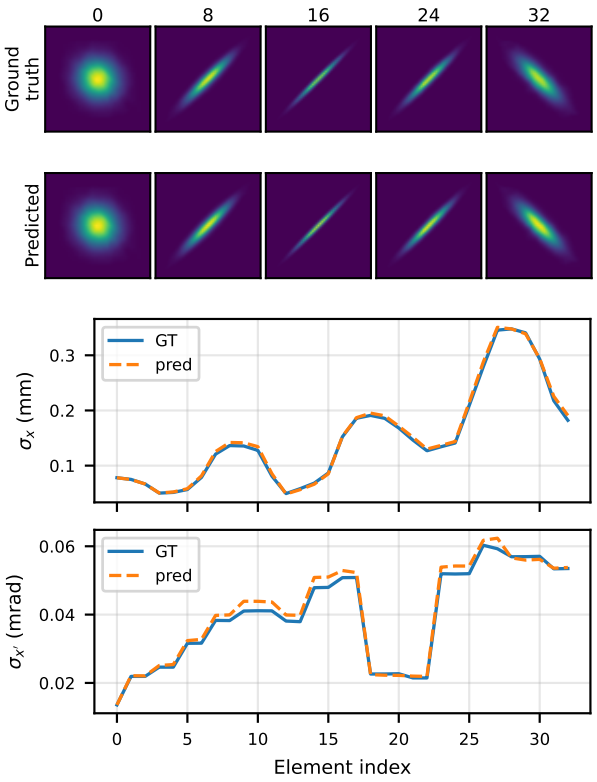

Figure 3: Phase-space reconstruction along a 32-element sextupole lattice (validation sample). Top: x–x' phase space at several elements. Ground truth (top row) vs. model prediction (bottom row). Bottom: predicted transverse beam sizes σ_x and $\sigma_{x'}$ (dashed) compared to ground truth (solid) across all 32 elements.

In autoregressive mode, the mean MSE is an order of magnitude above the median, pulled up by a small tail of outlier trajectories. Error compounds with sequence depth on these outlier trajectories, as seen in the growing gap between TF MSE and AR mean MSE. For typical in-distribution lattices, the AR median MSE tracks the TF error closely across all 32 elements.

Inference Performance

Table 1 reports wall-clock performance measured on a single A100-SXM4-40 GB GPU in float32 precision.

The TrackingTransformer in fully autoregressive (AR) mode over 32-element sequences runs at 66.3 ms per sample at batch size 1. Profiling across d_{model} variants confirms that the bottleneck is the 32 sequential kernel launches of the autoregressive loop, not the per-step computation. A hardware- or compiler-level reduction of this loop overhead (e.g. CUDA graph capture, KV-cache reuse) would directly translate to lower AR latency without architectural changes.

End-to-end inference — VAE encode, 32-step AR rollout, and VAE decode — completes in approximately 67 ms per trajectory on a single GPU. This places the framework in the real-time-capable regime for online control applications.

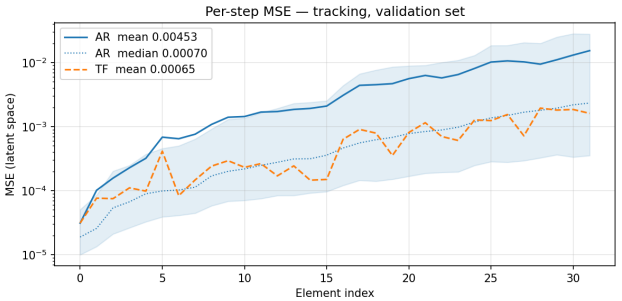

Figure 4: Per-element latent MSE on the validation set. Teacher-forced (TF, dashed) and autoregressive (AR, solid) MSE across the 32-element sequence; the shaded region spans the 10th–90th percentile of AR MSE.

Table 1: Inference Latency per Sample on a Single A100-SXM4-40 GB (float32, batch size 1)

Component	Latency
VAE encode	0.89 ms
VAE decode	0.65 ms
Transformer, TF	2.55 ms
Transformer, AR (32 steps)	66.3 ms
End-to-end	67.8 ms

CONCLUSION

We have presented a framework for learning beam dynamics in the latent space of beam distributions. A VAE encodes the full 6D phase-space distribution into a compact 256-dimensional latent vector, and a causal transformer propagates this state autoregressively through a sequence of element tokens. Element tokenization makes the model lattice-agnostic by construction, enabling prediction through arbitrary element sequences without retraining. Proof-of-concept results on FODO-style lattices demonstrate near-perfect VAE reconstruction of beam sizes, centroids, and shapes. End-to-end inference in approximately 67 ms per trajectory on a single GPU places the framework in the real-time-capable regime for online applications.

The primary limitation of the current work is training data scale and diversity: all results are obtained on randomly parameterized FODO-style lattices with ~10 000 samples, and generalization to qualitatively different lattice topologies and element types remains to be demonstrated. Future work will focus on scaling training data diversity, incorporating collective effects such as space charge and coherent synchrotron radiation, and validating on a specific real machine.

REFERENCES

[1] A. Edelen and X. Huang, "Machine learning for design and control of particle accelerators: A look backward and forward", *Annu. Rev. Nucl. Part. Sci.*, vol. 74, pp. 557–581, 2024. doi:10.1146/annurev-nucl-121423-100719

[2] R. Roussel *et al.*, "Phase space reconstruction from accelerator beam measurements using neural networks and differentiable simulations", *Phys. Rev. Lett.*, vol. 130, p. 145001, 2023. doi:10.1103/PhysRevLett.130.145001

[3] R. Roussel *et al.*, "Efficient six-dimensional phase space reconstructions from experimental measurements using generative machine learning", *Phys. Rev. Accel. Beams*, vol. 27, p. 094601, 2024. doi:10.1103/PhysRevAccelBeams.27.094601

[4] A. Scheinker, "cDVAE: multimodal generative conditional diffusion guided by variational autoencoder latent embedding for virtual 6D phase space diagnostics", *Sci. Rep.*, vol. 14, p. 29158, 2024. doi:10.1038/s41598-024-80751-1

[5] C. Emma, A. Edelen, M. J. Hogan, B. O'Shea, G. White, and V. Yakimenko, "Machine learning-based longitudinal phase space prediction of particle accelerators", *Phys. Rev. Accel. Beams*, vol. 21, p. 112802, 2018. doi:10.1103/PhysRevAccelBeams.21.112802

[6] A. Edelen, N. Neveu, M. Frey, Y. Huber, C. Mayes, and A. Adelmann, "Machine learning for orders of magnitude speedup in multiobjective optimization of particle accelerator systems", *Phys. Rev. Accel. Beams*, vol. 23, p. 044601, 2020. doi:10.1103/PhysRevAccelBeams.23.044601

[7] X. Pang, A. Williams, and L. Rybarcyk, "Machine learning surrogate for charged particle beam dynamics with space charge based on a recurrent neural network with aleatoric uncertainty", *Phys. Rev. Accel. Beams*, vol. 27, p. 024601, 2024. doi:10.1103/PhysRevAccelBeams.27.024601

[8] S. Wiewel, M. Becher, and N. Thuerey, "Latent-space physics: towards learning the temporal evolution of fluid flow", *Comput. Graph. Forum*, vol. 38, no. 2, pp. 71–82, 2019. doi:10.1111/cgf.13620

[9] A. Solera-Rico *et al.*, "β-variational autoencoders and transformers for reduced-order modelling of fluid flows", *Nat. Commun.*, vol. 15, p. 1361, 2024. doi:10.1038/s41467-024-45578-4

[10] M. Rautela, A. Williams, and A. Scheinker, "A conditional latent autoregressive recurrent model for generation and forecasting of beam dynamics in particle accelerators", *Sci. Rep.*, vol. 14, p. 18157, 2024. doi:10.1038/s41598-024-68944-0

[11] M. Rautela and A. Scheinker, "Advancing accelerator virtual beam diagnostics through latent evolution modeling: an integrated solution to forward, inverse, tuning, and UQ problems", in *Proc. NAPAC'25*, paper MOP002, 2025.

[12] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes", 2014, arXiv: 1312.6114,

[13] I. Higgins *et al.*, "β-VAE: learning basic visual concepts with a constrained variational framework", in *Proc. ICLR 2017*, 2017.

[14] D. Sagan, "Bmad: a relativistic charged particle simulation library", *Nucl. Instrum. Methods Phys. Res. A*, vol. 558, pp. 356–359, 2006. doi:10.1016/j.nima.2005.11.001

[15] D. Sagan and J. C. Smith, "The Tao accelerator simulation program", in *Proc. PAC 2005*, pp. 4159–4161, 2005. doi:10.1109/PAC.2005.1591750