ENTANGLEMENT-AWARE FLOW MATCHING FOR POLYMER DYNAMICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Learning long-timescale dynamics in dense polymer systems is important for understanding material behavior, but remains difficult because chain motion is governed by entanglement, a collective topological constraint that existing learned dynamics models often fail to preserve during generation. We formulate windowed Conditional Flow Matching (wCFM) for force-free polymer dynamics, training continuous-time velocity fields on trajectory windows sampled across multiple temporal scales. We further introduce `orthF`, a geometry-aware regularization that penalizes predicted motion across local inter-chain boundaries. We evaluate `orthF` on molecular dynamics trajectories of dense polymer systems across three dynamics models, showing that its benefit is not limited to a single architecture. Across 9 polymer datasets, `orthF` improves aggregate displacement accuracy for all three models, yielding generated trajectories that better match ground-truth distributions and structural and topology-related observables. Once trained, our approach generates a $50\,\mathrm{ns}$ backbone trajectory in $13\,\mathrm{s}$ on a single GPU, compared with $43\,\mathrm{h}$ for all-atom molecular dynamics on 64 CPUs, corresponding to an $\approx 1.2 \times 10^4$ wall-clock sampling-time ratio. These results show that explicit geometric inductive bias is a simple and effective way to improve learned dynamics in entangled molecular systems.

1 INTRODUCTION

Understanding polymer dynamics is central to predicting material behavior because many properties depend on long-timescale relaxation and reorganization, not only on static molecular structure Gedde et al. (2019); Doi & Edwards (1988). These processes are costly to simulate, especially for dynamical and topological observables such as diffusion, viscosity, and entanglement; Table 1 highlights the particularly high cost of entanglement analysis. Molecular dynamics (MD) provides high-fidelity trajectories, but long-time relaxation requires many small integration steps. Machine-learned force fields improve atomistic force prediction Batatia et al. (2022); Musaelian et al. (2023), yet still require repeated force evaluation and numerical integration. A complementary direction learns molecular configurations or dynamics directly from data Noé et al. (2020); Olsson (2026). Dense polymer systems introduce a distinct challenge: local coordinate accuracy alone does not prevent crossing-like motion or degradation of multi-chain structure during rollout.

Table 1: Approximate atomistic MD cost for polymer observables. Entanglement requires long trajectories, large systems, and topology-aware analysis.

Observable	Difficulty	Cost (CPU h)
Density / cohesive energy	Low	$\sim 10^2$
Local structure	Low–Moderate	10^2–10^3
Diffusion / viscosity	High	10^3–10^5
Entanglement	Very high	10^2–10^6

A central challenge in dense polymer systems is *entanglement*. In dense environments, neighboring chains are not chemically bonded but still restrict each other's motion because their backbone paths cannot freely pass through one another Doi & Edwards (1988); Edwards (1977). As a result, polymer relaxation is slow, collective, and shaped by the surrounding multi-chain geometry. For learned dynamics models, locally plausible displacement predictions can still permit crossing-like inter-chain motion, distort topology-related structure, and degrade long-time rollouts.

In this work, we address this failure mode with two complementary components. First, we formulate windowed Conditional Flow Matching (wCFM) for force-free polymer dynamics, using multi-scale

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

trajectory windows to supervise both local segmental motion and longer-time conformational change within a single continuous-time velocity field. Second, we introduce `orthF`, a geometry-aware regularization that penalizes predicted velocity components across local inter-chain boundaries, discouraging crossing-like motion, as visualised in Fig. 1. We evaluate `orthF` on 9 dense polymer datasets constructed from all-atom MD simulations across three dynamics models: our wCFM, built on flow matching Lipman et al. (2023), TFM Zhang et al. (2024), and MLCGMD Fu et al. (2023).

We also introduce a chemistry-aware backbone representation for polymer trajectories. Because entanglement depends on the geometry of each chain's backbone path, we retain the backbone atoms that carry this geometry while preserving their original MD coordinates.

Our contributions are as follows:

- We introduce `orthF`, a soft geometric regularizer that penalizes predicted motion across local inter-chain boundaries and discourages crossing-like dynamics.
- We formulate windowed Conditional Flow Matching (wCFM) for force-free polymer dynamics, using multi-scale trajectory windows to learn a continuous-time velocity field that captures both local motion and longer-time conformational change.
- We propose polymer backbone extraction as a chemistry-aware representation that preserves the atoms defining each chain's backbone path while retaining their original MD coordinates.
- We demonstrate, across three dynamics models, that `orthF` reduces Wasserstein distance (WSD), improves structural and Z1+-based topology-related observables, and enables trajectory generation up to 1.2×10^4 times faster than 50 ns MD.

2 Related Work

Polymer simulation and entanglement analysis. Classical MD and coarse-grained MD are standard tools for studying polymer dynamics Thompson et al. (2022); Kremer & Grest (1990). In dense polymer melts, covalently connected chain contours cannot freely pass through one another, producing entanglement-driven relaxation described by polymer dynamics and tube-model theories Doi & Edwards (1988); Edwards (1977). Entanglement is typically analyzed post hoc using primitive-path methods such as Z1 and Z1+ Kröger (2005); Kröger et al. (2023). Our goal is instead to incorporate local inter-chain geometry directly into learned dynamics.

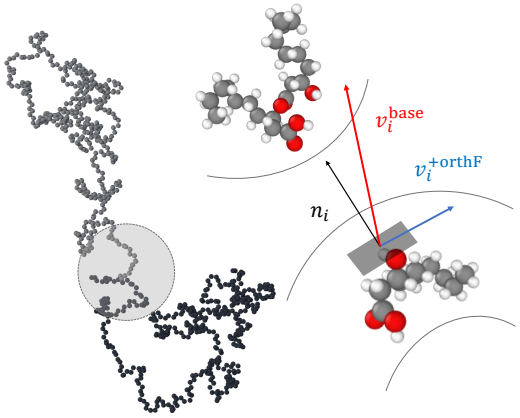


Figure 1: Geometric inductive bias for polymer dynamics. With `orthF`, we regularize the predicted motion toward directions consistent with local inter-chain geometry.

Learned molecular simulation. Machine learning accelerates molecular simulation through force-based and force-free approaches. Machine-learned force fields improve atomistic force prediction Batatia et al. (2022); Musaelian et al. (2023); Wood et al. (2026), including recent polymer-focused efforts Simm et al. (2025); Srivastava et al. (2026), but still require repeated force evaluation and numerical integration. Force-free and generative methods instead learn molecular configurations or dynamics directly from data, including equilibrium generative models Noé et al. (2019); Lewis et al. (2025); Akhound-Sadegh et al. (2026), coarse-grained dynamics Kohler et al. (2023); Arts et al. (2023); Fu et al. (2023), and learned trajectory models Klein et al. (2023); Schreiner et al. (2023); Diez et al. (2026); Kapusniak et al. (2026); Nam et al. (2025). Polymer-specific methods such as PolyConf Wang et al. (2025) and PolyGen Jain & Ramprasad (2025) focus on static conformation generation, whereas our setting starts from a supplied packed multi-chain configuration and models its subsequent evolution.

Continuous-time trajectory learning. Flow matching learns continuous-time generative models by regressing vector fields along probability paths Lipman et al. (2023), and conditional flow matching

provides simulation-free conditional objectives Tong et al. (2023). Recent trajectory generators such as MDGen Jing et al. (2024), ConfRover Shen et al. (2026), and STAR-MD Shoghi et al. (2026) target molecular trajectory generation primarily in single-protein settings. In contrast, we model dense multi-chain polymer systems initialized from supplied packed configurations, where inter-chain contacts and entanglement constrain the admissible motion. We build on continuous-time trajectory learning through wCFM and compare `orthF` across wCFM, TFM Zhang et al. (2024), and MLCGMD Fu et al. (2023). `orthF` introduces an explicit geometric bias by penalizing velocity components across local inter-chain directions.

3 METHOD

We learn force-free polymer dynamics from MD trajectories while biasing the learned velocity field to respect local inter-chain geometry. Our method extracts polymer backbones using chemistry-aware rules, trains a continuous-time velocity field with windowed Conditional Flow Matching (wCFM), and introduces `orthF`, a soft regularization that penalizes velocity components across local inter-chain boundaries.

3.1 POLYMER BACKBONE EXTRACTION

A polymer chain consists of monomers connected into a long molecular path. For entanglement, the relevant geometric object is the chain backbone: the sequence of atoms that carries the large-scale chain geometry through space. We construct a chemistry-aware representation by extracting this backbone from the bonded molecular graph. For each chain m, let $G_m = (V_m, E_m)$ be the bonded graph, with atoms as vertices and covalent bonds as edges. The LAMMPS Thompson et al. (2022) parameter files specify bond types, which we use to identify the monomer-connecting bonds that define the main chain. Let $b : E_m \to \mathcal{B}$ map each bond to its type, and let $\mathcal{B}_{\mathrm{bb}} \subseteq \mathcal{B}$ denote the backbone-relevant bond types, including terminal bonds used to identify chain endpoints. System-specific bond labels are provided in Appendix B.

For chain m, the backbone edge set is

$$E_{\mathrm{bb}}^{(m)} = \{(u, v) \in E_m \mid b(u, v) \in \mathcal{B}_{\mathrm{bb}}\}.$$

Let $\mathcal{P}_m$ denote the set of simple paths in $(V_m, E_{\mathrm{bb}}^{(m)})$ that start from a terminal atom $v_{\mathrm{end}}^{(m)}$. We extract the molecular backbone as

$$P_m^\star = \arg\max_{P \in \mathcal{P}_m} |V(P)|.$$

For a system with M polymer chains, the extracted backbones are

$$\mathcal{P}^\star = \{P_m^\star\}_{m=1}^M, \qquad V_{\mathrm{bb}} = \bigcup_{m=1}^M V(P_m^\star),$$

where $V(P_m^\star)$ denotes the atoms on path $P_m^\star$. The modeled system is then restricted to V_{bb}.

This representation preserves the backbone geometry relevant to entanglement while retaining the original MD coordinates of selected atoms. Unlike clustering-based coarse graining, it does not introduce pseudo-atoms, centroid coordinates, or reconstruction error. Appendix B provides a detailed comparison with clustering-based representations and summarizes the resulting system-size reduction.

3.2 WINDOWED CONDITIONAL FLOW MATCHING

After backbone extraction and coordinate preprocessing, each sample consists of N backbone atoms observed over $T + 1$ frames, indexed by $t = 0, \ldots, T$. We use rescaled unwrapped Cartesian coordinates to avoid discontinuities from periodic boundary crossings and to remove global barostat scaling; details are provided in Appendix B.2. We denote the configuration at frame t by $\mathbf{x}_t \in \mathbb{R}^{N\times 3}$ and associate it with normalized global time $\tau_t = t/T$, so that $\tau_0 = 0$ and $\tau_T = 1$.

We learn a continuous-time velocity field

$$v_\theta(\mathbf{x}_\tau, \tau) \in \mathbb{R}^{N\times 3}, \qquad \tau \in [0, 1],$$

which defines the dynamics

$$\frac{d\mathbf{x}_\tau}{d\tau} = v_\theta(\mathbf{x}_\tau, \tau). \tag{1}$$

Here, τ denotes continuous normalized trajectory time, while t indexes discrete MD frames.

Because MD trajectories are observed only at discrete frames, we use Conditional Flow Matching (CFM) to construct continuous-time velocity targets Lipman et al. (2023); Tong et al. (2023). CFM provides a simulation-free velocity regression objective along paths between endpoint states. We adapt this idea to polymer trajectories by choosing endpoint states from contiguous windows of the same MD trajectory and sampling these windows at multiple temporal scales. We refer to this objective as windowed Conditional Flow Matching (wCFM).

Given a trajectory, we sample a window length $W \in \mathcal{W}$, with $W \geq 2$, and a start index $a \in \{0, \ldots, T - W + 1\}$. The endpoint is $b = a + W - 1$, with normalized global times $\tau_a = \frac{a}{T}$, $\tau_b = \frac{b}{T}$. Within this window, we sample a local interpolation variable $s \sim \mathcal{U}[0, 1]$, where $s = 0$ corresponds to $\mathbf{x}_a$ and $s = 1$ corresponds to $\mathbf{x}_b$. We define the interpolated state and its corresponding global time as

$$\mathbf{x}_s = (1 - s)\mathbf{x}_a + s\mathbf{x}_b, \qquad \tau_s = (1 - s)\tau_a + s\tau_b. \tag{2}$$

The associated target velocity with respect to global normalized time is

$$\mathbf{u}_{\tau_s} = \frac{d\mathbf{x}}{d\tau} = \frac{\mathbf{x}_b - \mathbf{x}_a}{\tau_b - \tau_a}. \tag{3}$$

We train the velocity field with the windowed CFM objective

$$\mathcal{L}_{\mathrm{wCFM}}(\theta) = \mathbb{E}_{(\mathbf{x}_a, \mathbf{x}_b),\, s \sim \mathcal{U}[0,1]} \left[\| v_\theta(\mathbf{x}_s, \tau_s) - \mathbf{u}_{\tau_s} \|_2^2 \right], \tag{4}$$

where the expectation is over sampled trajectory windows and interpolation times.

Multi-scale window sampling exposes the model to both short-time local motion and longer-time conformational change: small windows provide fine temporal supervision, while larger windows provide coarse directional supervision over longer relaxation intervals. This yields a single continuous-time velocity field trained jointly across multiple temporal scales. In contrast to TFM's trajectory-level flow objective and MLCGMD's autoregressive time-coarsened prediction, wCFM uses the same velocity field to learn transitions over multiple window lengths. However, wCFM remains topology-agnostic and does not encode the local inter-chain geometry that constrains entangled polymer motion; we address this next with `orthF`.

3.3 Geometric Conditioning and Soft Flow Constraints

Windowed Conditional Flow Matching provides temporal supervision, but it does not encode the local inter-chain geometry that constrains entangled polymer motion. We build `orthF` by augmenting each atom with two geometry features computed from the current configuration $\mathbf{x}$: an inter-chain direction pointing toward nearby atoms on other chains, and a local tangent direction along the same chain.

Let $m(i)$ denote the polymer chain containing atom i. For a cutoff radius r_{cut}, we define same-chain and other-chain neighborhoods

$$\begin{aligned} \mathcal{N}_i^{\mathrm{in}}(\mathbf{x}) &= \{j \neq i : \|\mathbf{x}_j - \mathbf{x}_i\|_2 \leq r_{\mathrm{cut}},\ m(j) = m(i)\}, \\ \mathcal{N}_i^{\mathrm{out}}(\mathbf{x}) &= \{j \neq i : \|\mathbf{x}_j - \mathbf{x}_i\|_2 \leq r_{\mathrm{cut}},\ m(j) \neq m(i)\}. \end{aligned} \tag{5}$$

In implementation, neighborhoods are constructed from rescaled unwrapped coordinates using radius-graph neighbor lists; preprocessing and neighbor construction details are provided in Appendix B.2.

The other-chain neighborhood defines a local inter-chain direction

$$\bar{\mathbf{r}}_i = \frac{1}{|\mathcal{N}_i^{\mathrm{out}}|} \sum_{j \in \mathcal{N}_i^{\mathrm{out}}} (\mathbf{x}_j - \mathbf{x}_i), \qquad \mathbf{n}_i = \frac{\bar{\mathbf{r}}_i}{\|\bar{\mathbf{r}}_i\|_2 + \epsilon}, \tag{6}$$

with $\mathbf{n}_i = \mathbf{0}$ when $|\mathcal{N}_i^{\mathrm{out}}| = 0$. Thus, $\mathbf{n}_i$ points from atom i toward nearby atoms on neighboring chains. The ϵ term prevents numerical instability when $\bar{\mathbf{r}}_i$ has near-zero norm.

We also estimate the local backbone tangent from the same-chain neighborhood. Let

$$\boldsymbol{\mu}_i = \frac{1}{|\mathcal{N}_i^{\text{in}}|} \sum_{j \in \mathcal{N}_i^{\text{in}}} \mathbf{x}_j, \qquad C_i = \frac{1}{|\mathcal{N}_i^{\text{in}}|} \sum_{j \in \mathcal{N}_i^{\text{in}}} (\mathbf{x}_j - \boldsymbol{\mu}_i)(\mathbf{x}_j - \boldsymbol{\mu}_i)^\top.$$

When $|\mathcal{N}_i^{\text{in}}| \geq 2$, we set

$$\mathbf{t}_i = \text{eig}_{\max}(C_i), \tag{7}$$

where $\text{eig}_{\max}(C_i)$ is the eigenvector with largest eigenvalue; otherwise, $\mathbf{t}_i = \mathbf{0}$. This principal eigenvector estimates the dominant local direction of the same-chain backbone around atom i. Physically, it approximates the local chain contour direction, along which sliding and relaxation are more compatible with the backbone geometry.

The geometry feature for atom i is

$$\mathbf{f}_i = [\mathbf{n}_i, \mathbf{t}_i] \in \mathbb{R}^6, \qquad \mathbf{g}(\mathbf{x}) = \{\mathbf{f}_i\}_{i=1}^N,$$

which is provided to the velocity field together with the atomic coordinates.

Geometric conditioning gives the model access to local chain geometry, but it does not by itself constrain the predicted motion. We therefore add a soft velocity-space constraint, which forms the `orthF` regularization. At a training state $(\mathbf{x}_s, \tau_s)$, let

$$\mathbf{v}_i = \big[v_\theta\big(\mathbf{x}_s, \mathbf{g}(\mathbf{x}_s), \tau_s\big)\big]_i$$

be the predicted velocity of atom i, with $\mathbf{n}_i$ computed from $\mathbf{x}_s$. We penalize alignment between $\mathbf{v}_i$ and the local inter-chain direction:

$$\mathcal{L}_{\text{orthF}} = \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\left(\frac{\mathbf{v}_i \cdot \mathbf{n}_i}{\|\mathbf{v}_i\|_2 \, \|\mathbf{n}_i\|_2 + \epsilon}\right)^2\right]. \tag{8}$$

Here, ϵ prevents numerical instability when $\mathbf{v}_i$ or $\mathbf{n}_i$ has near-zero norm. The loss is a squared cosine-alignment penalty that discourages predicted velocity from aligning with nearby-chain directions while allowing motion in other directions. Thus, `orthF` acts as a soft local geometric regularizer rather than a topological guarantee: it reduces crossing-like motion without enforcing exact non-crossing or global topological invariants. Unlike hard projection, which removes velocity components exactly, `orthF` preserves flexibility for relaxation and local rearrangement. Since $\mathbf{n}_i$ and $\mathbf{t}_i$ are recomputed from the current configuration, the bias adapts during both training and trajectory generation.

3.4 Training and Trajectory Generation

We combine the wCFM objective from Section 3.2 with the local geometric descriptors and `orthF` regularization from Section 3.3. This yields a geometry-conditioned ODE on backbone configurations,

$$\frac{d\mathbf{x}_\tau}{d\tau} = v_\theta\big(\mathbf{x}_\tau, \mathbf{g}(\mathbf{x}_\tau), \tau\big), \tag{9}$$

where $\mathbf{g}(\mathbf{x}_\tau)$ is recomputed from the current configuration and encodes the local inter-chain geometry.

During training, we supervise the velocity field at interpolated states from MD trajectory windows rather than integrating Eq. equation 9. For each sampled window, wCFM provides $(\mathbf{x}_s, \tau_s, \mathbf{u}_{\tau_s})$, and the model is trained with

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{wCFM}}(\theta) + \lambda_{\text{orthF}} \mathcal{L}_{\text{orthF}}(\theta), \tag{10}$$

where λ_{orthF} controls the strength of the soft geometric regularization. Algorithm 1 summarizes this stochastic training procedure.

At generation time, we integrate Eq. equation 9 from an initial configuration $\mathbf{x}_0$. Since $\mathbf{g}$ is recomputed after each ODE step, the inter-chain neighborhoods, other-chain directions, and tangent directions evolve with the generated trajectory. Algorithm 2 summarizes the resulting rollout procedure.

Algorithm 1 Training with `orthF`

Require: Trajectory $\{\mathbf{x}_t\}_{t=0}^{T}$, windows $\mathcal{W}$, weight λ_{orthF}
Ensure: Velocity field v_θ
repeat
  Sample window endpoints (a, b) and interpolation s
  Construct $(\mathbf{x}_s, \tau_s, \mathbf{u}_{\tau_s})$ using Eqs. equation 2–equation 3
  Compute geometry $\mathbf{g}(\mathbf{x}_s)$ and directions $\{\mathbf{n}_i\}_{i=1}^{N}$
  Predict $\mathbf{v} \leftarrow v_\theta(\mathbf{x}_s, \mathbf{g}(\mathbf{x}_s), \tau_s)$
  Compute $\mathcal{L}_{\mathrm{wCFM}}$ and $\mathcal{L}_{\mathrm{orthF}}$ using Eqs. equation 4 and equation 8
  Update θ using $\mathcal{L}_{\mathrm{wCFM}} + \lambda_{\mathrm{orthF}}\mathcal{L}_{\mathrm{orthF}}$
until converged

Algorithm 2 Sampling with `orthF`

Require: Initial configuration $\mathbf{x}_0$, time grid $\{\tau_m\}_{m=0}^{M}$, trained velocity field v_θ
Ensure: Generated trajectory $\{\mathbf{x}_{\tau_m}\}_{m=0}^{M}$
$\mathbf{x}_{\tau_0} \leftarrow \mathbf{x}_0$
for $m = 0, \ldots, M-1$ **do**
  Compute geometry $\mathbf{g}(\mathbf{x}_{\tau_m})$
  $\mathbf{v} \leftarrow v_\theta(\mathbf{x}_{\tau_m}, \mathbf{g}(\mathbf{x}_{\tau_m}), \tau_m)$
  $\mathbf{x}_{\tau_{m+1}} \leftarrow \mathrm{ODEStep}(\mathbf{x}_{\tau_m}, \mathbf{v}, \tau_m, \tau_{m+1})$
end for
return $\{\mathbf{x}_{\tau_m}\}_{m=0}^{M}$

4 Experiments

We evaluate dense polymer trajectory generation along four axes: displacement-distribution accuracy, long-horizon rollout stability, structural and topology-related fidelity, and computational efficiency relative to molecular dynamics (MD). Design studies are reported last.

4.1 Experimental Setup

Data. Our benchmark comprises nine dense biopolymer systems generated by all-atom LAMMPS simulations Thompson et al. (2022) at 700 K. Each system contains multiple chains with degree of polymerization $N = 100$ and is sampled every 10 ps for 5 ns. The trajectories provide coordinates, velocities, molecular identities, bond connectivity, and periodic-cell information. We train on backbone trajectories extracted from the bonded molecular graphs. Following the long-horizon evaluation protocol of MLCGMD Fu et al. (2023), we additionally use a 50 ns 3hb trajectory, train on its first 5 ns, and evaluate beyond the training window. Simulation, preprocessing, periodic-boundary, and neighbor-construction details are given in App. B.

Comparisons. Our main model is wCFM with a Graph Attention Network (GAT) velocity field, selected against MLP Rumelhart et al. (1986), EGNN Satorras et al. (2021), MLP-Mixer Tolstikhin et al. (2021), and Transformer Vaswani et al. (2017) alternatives (App. C). We compare Base and `orthF` versions of wCFM, TFM Zhang et al. (2024), and MLCGMD Fu et al. (2023) under a common data and evaluation protocol. We also compare the soft `orthF` regularizer with the hard velocity projection `orthV`. All quantitative comparisons use MD as the reference. Unless stated otherwise, uncertainty denotes the standard deviation across independent training seeds.

Evaluation metrics. Our primary trajectory metric is the Wasserstein distance (WSD) between generated and MD displacement distributions. For structural fidelity, we compare bond-length, backbone-dihedral, and radius-of-gyration distributions using KL divergence, Jensen–Shannon divergence (JSD), and WSD Jain & Ramprasad (2025); Iyengar et al. (2026); Valentini et al. (2026). We assess topology-related fidelity using self- and cross-average crossing number (ACN), absolute writhe, the absolute Gauss linking integral, and mean-squared displacement (MSD), with complementary Z1+ diagnostics Kröger et al. (2023); Kröger (2005). We additionally report generation time and wall-clock cost relative to LAMMPS.

4.2 Displacement Distribution Accuracy

Tab. 2 shows that `orthF` lowers mean displacement WSD across all three model families, from 0.053 to 0.047 for wCFM, 0.149 to 0.035 for TFM, and 0.628 to 0.618 for MLCGMD. The improvement occurs on six of nine systems for wCFM and on all nine systems for TFM and MLCGMD, showing that the effect is not specific to a single trajectory model. The gains also persist with an SE(3)-

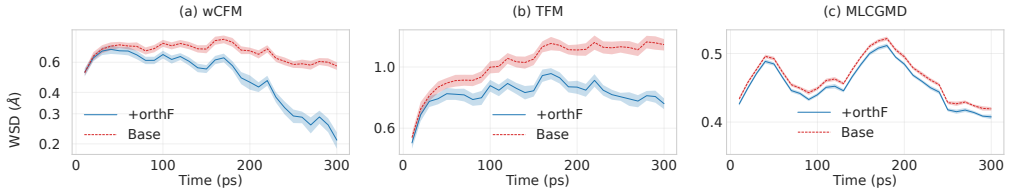


Figure 2: Temporal evolution of displacement Wasserstein distance for wCFM, TFM, and MLCGMD. The vertical axis is logarithmic. Across model families, `orthF` slows error accumulation and improves long-horizon rollout stability.

Table 2: Wasserstein distance between generated and MD displacement distributions. Entries are mean±standard deviation across seeds. Lower is better; bold marks the better result within each Base/`orthF` pair.

	wCFM		TFM		MLCGMD	
Dataset	Base	`orthF`	Base	`orthF`	Base	`orthF`
3hb_3hv	0.055±0.003	**0.044**±0.002	0.228±0.391	**0.028**±0.004	0.463±0.010	**0.453**±0.020
3hb	0.054±0.001	**0.038**±0.012	0.207±0.322	**0.015**±0.001	0.710±0.020	**0.702**±0.040
3hd	0.085±0.009	**0.068**±0.091	0.191±0.347	**0.023**±0.002	0.558±0.009	**0.547**±0.010
3hhx	0.065±0.003	**0.057**±0.030	0.164±0.258	**0.097**±0.120	0.741±0.010	**0.730**±0.100
3hp	**0.029**±0.010	0.034±0.003	0.173±0.292	**0.030**±0.044	0.703±0.090	**0.695**±0.050
3hv	**0.037**±0.002	0.048±0.004	0.195±0.325	**0.034**±0.050	0.728±0.120	**0.720**±0.120
4hb	0.046±0.008	**0.040**±0.001	0.040±0.004	**0.029**±0.000	0.522±0.020	**0.510**±0.030
la	0.032±0.007	**0.020**±0.001	0.106±0.123	**0.029**±0.003	0.397±0.020	**0.386**±0.030
phla	**0.072**±0.005	0.073±0.004	0.034±0.005	**0.030**±0.002	0.831±0.030	**0.820**±0.020
Mean	0.053±0.001	**0.047**±0.001	0.149±0.069	**0.035**±0.023	0.628±0.010	**0.618**±0.020

equivariant EGNN backbone (App. 10), suggesting that `orthF` is complementary to architectural equivariance.

4.3 Rollout Stability

We next evaluate whether improved displacement matching translates into more stable trajectory rollouts. Fig. 2 shows that `orthF` slows displacement-error accumulation across model families; for wCFM, late-rollout WSD decreases from approximately 0.70 to 0.25.

The representative 3hb rollouts in Fig. 3 show the corresponding qualitative effect: the Base model becomes increasingly diffuse over time, whereas wCFM-`orthF` retains more coherent chain structure.

Averaged over all nine systems, `orthF` reduces displacement WSD by 98.1%, 92.7%, and 88.7% over the early, middle, and late rollout intervals, respectively. Error still increases with rollout horizon, indicating that `orthF` improves rather than eliminates long-horizon drift. Full per-system interval results are reported in App. 11.

On the 50 ns 3hb trajectory, `orthF` also improves extrapolation beyond the 5 ns training window; detailed results are reported in App. D.2.

4.4 Structural Fidelity

`orthF` improves every backbone-dihedral and bond-length discrepancy on both 3hb and 3hv (Tab. 3). Radius-of-gyration results are mixed, indicating that improved local structural fidelity does not uniformly translate to global chain-size statistics. Time-resolved R_g behavior is reported in App. E.2.

4.5 Topology-Related Fidelity

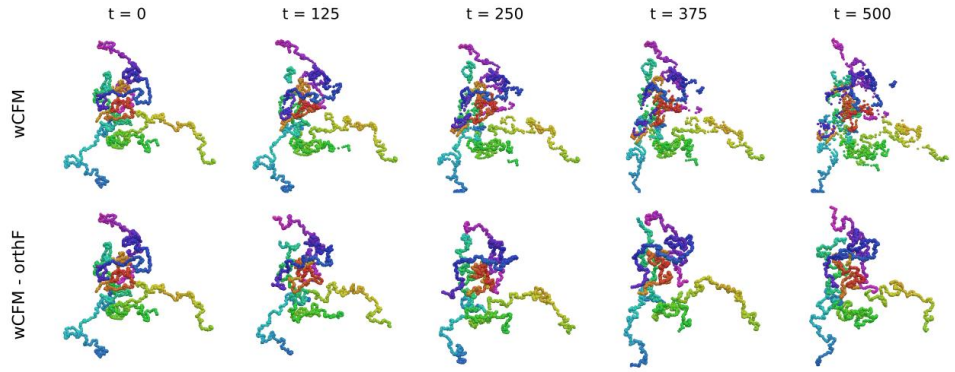


Figure 3: Representative 3hb rollouts. Base becomes increasingly diffuse during rollout, whereas `orthF` retains more coherent chain structure.

Table 3: Agreement between generated and MD structural distributions. KL denotes $D_{\mathrm{KL}}(p_{\mathrm{MD}} \| p_{\mathrm{model}})$, WSD denotes the one-dimensional Wasserstein distance, and JS denotes Jensen–Shannon divergence. Lower is better; bold identifies the better model within each system.

		3hb		3hv	
Observable	Metric	Base	+orthF	Base	+orthF
Backbone dihedral	KL	0.0253	**0.0157**	0.0178	**0.0085**
	WSD	0.1293	**0.0857**	0.1205	**0.0686**
	JS	0.0062	**0.0039**	0.0045	**0.0021**
Bond length	KL	0.6630	**0.4391**	0.5811	**0.4604**
	WSD	0.9534	**0.4889**	0.7474	**0.5287**
	JSD	0.2045	**0.1403**	0.1785	**0.1431**
Radius of gyration (R_g)	KL	**0.1635**	0.4899	0.4288	**0.2251**
	WSD	0.3944	**0.2045**	**0.9017**	1.2379
	JSD	0.0324	**0.0317**	**0.0345**	0.0427

We evaluate topology-related fidelity using ACN, writhe, and the Gauss linking integral over 500 rollout steps (Tab. 4). These Gauss-integral diagnostics can be evaluated on polygonal chains Klenin & Langowski (2000) and provide continuous topology-related descriptors for open polymer configurations Panagiotou et al. (2010; 2013). `orthF` reduces Self-ACN, Cross-ACN, and writhe drift: final Self-ACN decreases from 284.94 to 141.03, and final Cross-ACN from 49.48 to 19.07, compared with MD values of 54.62 and 5.91. The linking diagnostic is mixed, with Base closer in the rollout mean and `orthF` closer at the final frame.

Table 4: Topology-related diagnostics on 3hb. Topology entries show rollout mean±standard deviation (top) and final frame (bottom); MSD is reported at frame 500. Bold marks the learned result closer to MD for each topology statistic.

Metric	Base	+orthF	MD
Self-ACN	137.43±80.75 284.94	**90.04±32.93** **141.03**	54.27±6.16 54.62
Cross-ACN	19.42±22.66 49.48	**10.63±9.92** **19.07**	5.89±4.30 5.91
⟨|Wr|⟩	2.01±1.61 2.95	**1.94±1.25** **2.27**	1.54±1.21 1.62
⟨|Lk|⟩	**0.118±0.270** 0.213	0.136±0.310 **0.188**	0.093±0.190 0.087
Atom MSD (Å^2)	273.5	234.3	322.1
Chain MSD (Å^2)	39.7	36.4	45.8

The complementary Z1+ analysis Kröger et al. (2023); Kröger (2005) in Fig. 4 shows smaller bond-length and end-to-end drift with `orthF`, whereas N_e^{CK} does not show a consistent improvement; detailed values are reported in App. F.1.

The topology-related improvements are not accompanied by complete motion suppression: both learned models exhibit substantial displacement, although both remain under-diffusive relative to MD and `orthF` has somewhat lower MSD than Base. We therefore cannot fully rule out a contribution from reduced motion; a motion-matched control would be required to isolate this effect.

4.6 Design Studies

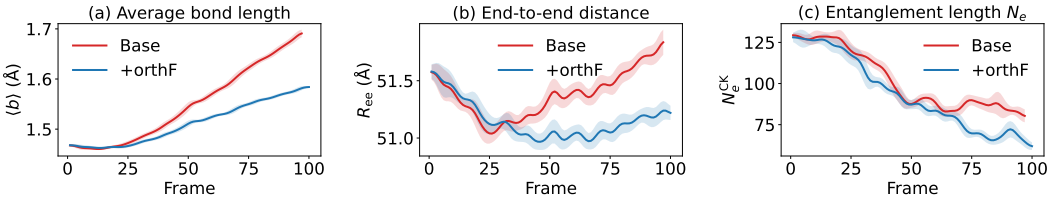


Figure 4: Z1+-based topology-sensitive rollout diagnostics on 3hb: (a) average bond length, (b) root-mean-square end-to-end distance, and (c) classical-kink entanglement length N_e^{CK}. `orthF` exhibits less drift in bond length and end-to-end distance, while N_e^{CK} remains mixed. Detailed Z1+ values are reported in App. F.1.

Across all nine systems, mixed-window training achieves the best average performance with `orthF`, while the Base model becomes unstable under the same setting; the full window-size and regularization-weight ablations are reported in App. E.1.

Tab. 5 compares the soft `orthF` regularizer with the hard `orthV` velocity projection. `orthF` achieves the lowest trajectory MSE for wCFM, TFM, and MLCGMD.

Table 5: Trajectory MSE for the Base model, the soft `orthF` regularizer, and the hard `orthV` projection. Lower is better; bold marks the best result within each model family.

Model	Base	`orthF`	`orthV`
wCFM	0.0076	**0.0071**	0.0075
TFM	0.0064	**0.0062**	0.0075
MLCGMD	0.0560	**0.0530**	0.0750

4.7 Computational Efficiency

A 50 ns LAMMPS trajectory requires approximately 43.0 h wall-clock time on 64 CPUs, whereas wCFM-`orthF` generates a 50 ns trajectory in 13 s on a single GPU after training. This corresponds to an amortized sampling-time wall-clock speedup of 1.19×10^4. Training takes 3.1 h on a single GPU. Sampling uses 40 midpoint ODE steps with $\mathbf{g}(\mathbf{x}_\tau)$ recomputed at each step. The comparison excludes the one-time costs of training and generating the MD training data.

5 Conclusion

We presented a force-free approach for learned polymer dynamics that combines windowed Conditional Flow Matching (wCFM) with `orthF`, a geometry-aware regularization for entangled systems. wCFM learns continuous-time dynamics from multi-scale trajectory windows, while `orthF` biases predicted motion away from local inter-chain crossing directions. Across three dynamics models and 9 polymer datasets, `orthF` consistently improves unconstrained baselines, better matching ground-truth distributions and preserving structural and Z1+-based topology-related observables. Together with a chemistry-aware backbone representation, our approach generates trajectories up to 1.2×10^4 times faster than 50 ns MD. These results show that combining multi-scale temporal supervision with lightweight geometric bias can improve learned polymer rollouts, while exact topological conservation remains an important direction for future work.

Table 6: Wall-clock cost relative to a 50 ns LAMMPS trajectory. Relative sampling speed is computed against the 43.0 h MD reference.

Workflow	Time	Relative speed
MD, 50 ns	43.0 h	1.0×
MD, 5 ns	4.3 h	10.0×
Ours, training	3.1 h	13.9×
Ours, sampling	13 s	**11,900**×

AI use statement

In this work, we used generative AI tools to assist with implementation tasks related to the experiments, debugging, and feedback on the analysis and presentation of results. Generative AI tools were not used to generate datasets or to formulate or prove mathematical claims.

Additionally, generative AI tools were used to assist with code development and editing, manuscript editing, improving clarity and organization, and refining the presentation of the methodology, experimental results, and related work. All AI-assisted code was reviewed, tested, and validated by the authors. The authors independently verified the reported results, scientific claims, mathematical arguments, citations, and interpretations, and made the final decisions regarding the methodology, experimental design, analysis, and conclusions. All AI-assisted text was reviewed and revised by the authors. The authors take full responsibility for the final content of this work, including code, experiments, text, claims, and other artifacts produced with the assistance of generative AI.

Ethics statement

This work concerns computational modeling of polymer dynamics and does not involve human subjects, personal data, or sensitive information. We are not aware of significant ethical concerns arising from the methodology or experiments presented in this work.

Reproducibility statement

We provide the code used for model training and evaluation, together with one simulation dataset, as supplementary material for the review process. The paper and appendix describe the model, training procedure, experimental setup, data generation and preprocessing, baselines, and evaluation metrics used in this work. Following the review process, we plan to publicly release the full code repository and complete dataset.

References

Tara Akhound-Sadegh, Jungyoon Lee, Joey Bose, Valentin De Bortoli, Arnaud Doucet, Michael Bronstein, Dominique Beaini, Siamak Ravanbakhsh, Kirill Neklyudov, and Alexander Tong. Progressive inference-time annealing of diffusion models for sampling from boltzmann densities. *Advances in Neural Information Processing Systems*, 38:63218–63251, 2026.

Marloes Arts, Victor Garcia Satorras, Chin-Wei Huang, Daniel Zugner, Marco Federici, Cecilia Clementi, Frank Noé, Robert Pinsler, and Rianne van den Berg. Two for one: Diffusion models and force fields for coarse-grained molecular dynamics. *Journal of Chemical Theory and Computation*, 19(18):6151–6159, 2023.

Ilyes Batatia, David P Kovacs, Gregor Simm, Christoph Ortner, and Gábor Csányi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. *Advances in neural information processing systems*, 35:11423–11436, 2022.

Y Diao, A Dobay, and A Stasiak. The average inter-crossing number of equilateral random walks and polygons. *Journal of Physics A: Mathematical and General*, 38(35):7601–7616, 2005.

Juan Viguera Diez, Mathias Schreiner, and Simon Olsson. Transferable generative models bridge femtosecond to nanosecond time-step molecular dynamics. *Science advances*, 12(15):eaed2333, 2026.

M. Doi and S. F. Edwards. *The Theory of Polymer Dynamics*. Oxford Univ. Press, 1988.

S. F. Edwards. The theory of rubber elasticity. *British Polymer Journal*, 9:140, 1977.

Xiang Fu, Tian Xie, Nathan J. Rebello, Bradley Olsen, and Tommi S. Jaakkola. Simulate time-integrated coarse-grained molecular dynamics with multi-scale graph networks. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=y8RZoPjEUl.

Ulf W Gedde, Mikael S Hedenqvist, Mats Johansson, Lars Berglund, and Jakob Wohlert. *Fundamental polymer science*, volume 2. Springer, 2019.

William G. Hoover. Canonical dynamics: Equilibrium phase-space distributions. *Phys. Rev. A*, 31: 1695–1697, Mar 1985. doi: 10.1103/PhysRevA.31.1695. URL https://link.aps.org/doi/10.1103/PhysRevA.31.1695.

Pieter J in't Veld and Gregory C Rutledge. Temperature-dependent elasticity of a semicrystalline interphase composed of freely rotating chains. *Macromolecules*, 36(19):7358–7365, 2003.

Aniketh Iyengar, Jiaqi Han, Pengwei Sun, Mingjian Jiang, Jianwen Xie, and Stefano Ermon. Align your structures: Generating trajectories with structure pretraining for molecular dynamics. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=OKQYMeWlGa.

Ayush Jain and Rampi Ramprasad. polygen: A learning framework for atomic-level polymer structure generation. *arXiv preprint arXiv:2504.17656*, 2025.

Bowen Jing, Hannes Stärk, Tommi Jaakkola, and Bonnie Berger. Generative modeling of molecular dynamics trajectories. *Advances in Neural Information Processing Systems*, 37:40534–40564, 2024.

Kacper Kapusniak, Cristian Gabellini, Michael Bronstein, Prudencio Tossou, and Francesco Di Giovanni. Mars-fm: Generative modeling of molecular dynamics via markov state models. In *International Conference on Learning Representations*, volume 2026, pp. 19073–19098, 2026.

Vsevolod Katritch, Jan Bednar, Didier Michoud, Robert G Scharein, Jacques Dubochet, and Andrzej Stasiak. Geometry and physics of knots. *Nature*, 384(6605):142–145, 1996.

Leon Klein, Andrew Foong, Tor Fjelde, Bruno Mlodozeniec, Marc Brockschmidt, Sebastian Nowozin, Frank Noé, and Ryota Tomioka. Timewarp: Transferable acceleration of molecular dynamics by learning time-coarsened dynamics. *Advances in Neural Information Processing Systems*, 36: 52863–52883, 2023.

Konstantin Klenin and Jörg Langowski. Computation of writhe in modeling of supercoiled dna. *Biopolymers: Original Research on Biomolecules*, 54(5):307–317, 2000.

Jonas Kohler, Yaoyi Chen, Andreas Kramer, Cecilia Clementi, and Frank Noé. Flow-matching: Efficient coarse-graining of molecular dynamics without forces. *Journal of Chemical Theory and Computation*, 19(3):942–952, 2023.

Kurt Kremer and Gary S Grest. Dynamics of entangled linear polymer melts: A molecular-dynamics simulation. *The Journal of Chemical Physics*, 92(8):5057–5086, 1990.

Martin Kröger. Shortest multiple disconnected path for the analysis of entanglements in two- and three-dimensional polymeric systems. *Computer Physics Communications*, 168:209–232, 2005.

Martin Kröger, Joseph D Dietz, Robert S Hoy, and Clarisse Luap. The z1+ package: Shortest multiple disconnected path for the analysis of entanglements in macromolecular systems. *Computer Physics Communications*, 283:108567, 2023.

Benedict Leimkuhler and Charles Matthews. Rational construction of stochastic numerical methods for molecular sampling. *Applied Mathematics Research eXpress*, 2013(1):34–56, 2013.

Sarah Lewis, Tim Hempel, José Jiménez-Luna, Michael Gastegger, Yu Xie, Andrew YK Foong, Victor García Satorras, Osama Abdin, Bastiaan S Veeling, Iryna Zaporozhets, et al. Scalable emulation of protein equilibrium ensembles with generative deep learning. *Science*, 389(6761): eadv9817, 2025.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Albert Musaelian, Simon Batzner, Anders Johansson, Lixin Sun, Cameron J Owen, Mordechai Kornbluth, and Boris Kozinsky. Learning local equivariant representations for large-scale atomistic dynamics. *Nature Communications*, 14(1):579, 2023.

Juno Nam, Sulin Liu, Gavin Winter, KyuJung Jun, Soojung Yang, and Rafael Gómez-Bombarelli. Flow matching for accelerated simulation of atomic transport in crystalline materials. *Nature Machine Intelligence*, 7(10):1625–1635, 2025.

Frank Noé, Simon Olsson, Jonas Köhler, and Hao Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147, 2019.

Frank Noé, Alexandre Tkatchenko, Klaus-Robert Müller, and Cecilia Clementi. Machine learning for molecular simulation. *Annual review of physical chemistry*, 71(1):361–390, 2020.

Shuichi Nosé. A unified formulation of the constant temperature molecular dynamics methods. *The Journal of chemical physics*, 81(1):511–519, 1984.

Simon Olsson. Generative molecular dynamics. *Current Opinion in Structural Biology*, 96:103213, 2026.

E Panagiotou, Martin Kröger, and KC Millett. Writhe and mutual entanglement combine to give the entanglement length. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 88(6): 062604, 2013.

Eleni Panagiotou. The linking number in systems with periodic boundary conditions. *Journal of Computational Physics*, 300:533–573, 2015.

Eleni Panagiotou, Kenneth C Millett, and Sofia Lambropoulou. The linking number and the writhe of uniform random walks and polygons in confined spaces. *Journal of Physics A: Mathematical and Theoretical*, 43(4):045208, 2010.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International Conference on Machine Learning*, pp. 9323–9332. PMLR, 2021.

Mathias Schreiner, Ole Winther, and Simon Olsson. Implicit transfer operator learning: Multiple time-resolution models for molecular dynamics. *Advances in Neural Information Processing Systems*, 36:36449–36462, 2023.

Yuning Shen, Lihao Wang, Huizhuo Yuan, Yan Wang, Bangji Yang, and Quanquan Gu. Simultaneous modeling of protein conformation and dynamics via autoregression. *Advances in Neural Information Processing Systems*, 38:109782–109822, 2026.

Nima Shoghi, Yuxuan Liu, Yuning Shen, Rob Brekelmans, Pan Li, and Quanquan Gu. Scalable spatio-temporal SE(3) diffusion for long-horizon protein dynamics. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Q1JpRZkR3S.

Gregor NC Simm, Jean Hélie, Hannes Schulz, Yicheng Chen, Guillem Simeon, Anna Kuzina, Ernesto Martinez-Baez, Piero Gasparotto, Gabriele Tocci, Chi Chen, et al. Simpoly: Simulation of polymers with machine learning force fields derived from first principles. *arXiv preprint arXiv:2510.13696*, 2025.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Ashutosh Srivastava, Sakshi Agarwal, Shivank Shukla, Harikrishna Sahu, and Rampi Ramprasad. On-the-fly machine-learned force fields for high-fidelity polymer glass transition simulations. *Journal of Chemical Information and Modeling*, 66(17):11129–11139, 2026. doi: 10.1021/acs.jcim.6c02116. URL https://doi.org/10.1021/acs.jcim.6c02116.

Huai Sun, Stephen J. Mumby, Jon R. Maple, and Arnold T. Hagler. An ab initio cff93 all-atom force field for polycarbonates. *Journal of the American Chemical Society*, 116(7):2978–2987, 1994. doi: 10.1021/ja00086a030.

A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. J. Stevens, J. Tranchida, C. Trott, and S. J. Plimpton. LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Comp. Phys. Comm.*, 271:108171, 2022. doi: 10.1016/j.cpc.2021.108171.

Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in Neural Information Processing Systems*, 34: 24261–24272, 2021.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *arXiv preprint arXiv:2302.00482*, 2023.

Alessio Valentini, David Pekker, Chungwen Liang, Todd Martinez, and Swagatam Mukhopadhyay. Polyformer: a generative framework for thermodynamic modeling of polymeric molecules. *arXiv preprint arXiv:2604.14241*, 2026.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

Fanmeng Wang, Wentao Guo, Qi Ou, Hongshuai Wang, Haitao Lin, Hongteng Xu, and Zhifeng Gao. Polyconf: Unlocking polymer conformation generation through hierarchical generative models. In *Forty-second International Conference on Machine Learning*, 2025. URL `https://openreview.net/forum?id=BsTLUx38qV`.

Brandon Wood, Misko Dzamba, Xiang Fu, Meng Gao, Muhammed Shuaibi, Luis Barroso-Luque, Kareem Abdelmaqsoud, Vahe Gharakhanyan, John Kitchin, Daniel Levine, et al. Uma: A family of universal models for atoms. *Advances in Neural Information Processing Systems*, 38:129391–129427, 2026.

Xi Zhang, Yuan Pu, Yuki Kawamura, Andrew Loza, Yoshua Bengio, Dennis L Shung, and Alexander Tong. Trajectory flow matching with applications to clinical time series modelling. *Advances in Neural Information Processing Systems*, 37:107198–107224, 2024.

A Relation to Probability-Flow and Langevin Dynamics

This section provides additional intuition for the proposed model by relating it to Langevin dynamics and probability-flow ODEs.

A useful reference point is overdamped Langevin dynamics, which samples from an equilibrium distribution $p_{\mathrm{eq}}(\mathbf{x}) \propto e^{-\beta U(\mathbf{x})}$ Leimkuhler & Matthews (2013); Song et al. (2021):

$$d\mathbf{x}_t = \nabla_{\mathbf{x}} \log p_{\mathrm{eq}}(\mathbf{x}_t)\, dt + \sqrt{2}\, d\mathbf{W}_t = -\beta \nabla U(\mathbf{x}_t)\, dt + \sqrt{2}\, d\mathbf{W}_t.$$

The same marginal density evolution can be represented by the deterministic probability-flow ODE Song et al. (2021)

$$\frac{d\mathbf{x}_t}{dt} = \nabla_{\mathbf{x}} \log p_{\mathrm{eq}}(\mathbf{x}_t) - \nabla_{\mathbf{x}} \log q_t(\mathbf{x}_t),$$

where q_t denotes the transient sample density. Thus, Langevin dynamics and its probability-flow ODE share the same marginals, but differ at the trajectory level: the former is stochastic, while the latter is deterministic.

From this perspective, our model is best viewed as a geometry-aware deterministic transport model rather than a direct simulator of physical MD trajectories. The learned field $v_\theta(\mathbf{x}, \mathbf{g}(\mathbf{x}), \tau)$ is analogous to a probability-flow transport field, while the geometry-dependent feature map $\mathbf{g}(\mathbf{x})$ injects local intermolecular structure into the dynamics.

Conditional flow matching is well-suited to this setting because it learns a continuous transport field without requiring explicit evaluation of the transient score $\nabla_{\mathbf{x}} \log q_t(\mathbf{x})$ Song et al. (2021).

Instead, CFM regresses conditional bridge velocities between sampled endpoint configurations Tong et al. (2023); Lipman et al. (2023). Accordingly, the proposed method should be interpreted as a geometry-aware deterministic sampler for molecular configurations: it is appropriate for equilibrium generation and long-horizon rollout, but not as an exact model of Langevin kinetics.

A.1 Relationship to Trajectory Baselines

Our comparison models differ primarily in how temporal evolution is learned. TFM Zhang et al. (2024) uses a trajectory-level flow-matching objective, whereas MLCGMD Fu et al. (2023) predicts time-coarsened molecular evolution autoregressively. In contrast, wCFM trains a single continuous-time velocity field jointly over trajectory windows sampled at multiple temporal scales.

At a high level, wCFM parameterizes configuration-space dynamics as

$$\frac{d\mathbf{x}_\tau}{d\tau} = v_\theta(\mathbf{x}_\tau, \mathbf{g}(\mathbf{x}_\tau), \tau)\,,$$

while MLCGMD advances the system through learned time-coarsened updates,

$$\mathbf{x}_{t+\Delta t} = \mathcal{F}_\phi(\mathbf{s}_t),$$

where $\mathbf{s}_t$ denotes the model state used to predict the next time-coarsened configuration. Thus, wCFM learns a continuous-time velocity field, whereas MLCGMD uses autoregressive time-coarsened state updates.

The proposed `orthF` regularizer is orthogonal to this temporal parameterization. It acts on predicted motion using geometry recomputed from the current configuration, penalizing velocity components aligned with local inter-chain directions. Evaluating `orthF` across wCFM, TFM, and MLCGMD therefore tests whether the geometric bias transfers across distinct learned-dynamics formulations rather than being specific to wCFM.

B Dataset and Backbone Representation Details

B.1 Polymer Melt Simulations

The polymer dataset consists of nine biopolymer systems simulated at $700\,\mathrm{K}$, including polylactic acid, seven PHA homopolymers, and one PHA copolymer, listed in Table 7. All MD simulations were performed using LAMMPS Thompson et al. (2022) with the Polymer Consistent Force Field (PCFF) Sun et al. (1994). Initial polymer configurations were generated with Enhanced Monte Carlo (EMC v9.4.4) in't Veld & Rutledge (2003), with degree of polymerization $N = 100$. Each system contained 7–22 polymer chains, with roughly $20{,}000$ atoms in total, packed into periodic simulation boxes at melt densities corresponding to $700\,\mathrm{K}$.

After energy minimization and initial equilibration, we performed $5\,\mathrm{ns}$ production runs for all systems in the NPT ensemble at $700\,\mathrm{K}$ and $1\,\mathrm{atm}$. For temporal-extrapolation experiments, we additionally generated a $50\,\mathrm{ns}$ trajectory for the 3hb system. Integration used Velocity-Verlet with a $1\,\mathrm{fs}$ timestep, and temperature and pressure were controlled using the Nosé–Hoover thermostat and barostat Nosé (1984); Hoover (1985). Trajectories were sampled every $10\,\mathrm{ps}$ and include atom coordinates, velocities, molecular identities, bond connectivity, and periodic-cell information.

Z1+-based entanglement descriptors were extracted using Z1+ Kröger (2005); Kröger et al. (2023), including primitive-path networks, entanglement numbers Z, and entanglement kink positions. The entanglement network evolves over the MD trajectory as entanglements form and disappear.

B.2 Coordinate and Neighbor Preprocessing

We train and evaluate models using rescaled unwrapped Cartesian coordinates. The LAMMPS trajectories provide wrapped coordinates, orthorhombic box lengths $L_t = (L_{t,x}, L_{t,y}, L_{t,z})$, and image flags $\mathbf{i}_{t,i} \in \mathbb{Z}^3$. For atom i at frame t, we reconstruct continuous coordinates as

$$\mathbf{x}_{t,i}^{\mathrm{unwrap}} = \mathbf{x}_{t,i}^{\mathrm{wrap}} + \mathbf{i}_{t,i} \odot L_t,$$

Table 7: Monomer-unit SMILES and system-specific backbone bond types used for backbone extraction. Backbone bonds are monomer-linking bonds involving o_2; the terminal marker ho_2-o_2 identifies chain endpoints.

Monomer-unit SMILES

System	SMILES
3hb	`OC(C)CC(=O)O`
3hb_3hv	`OC(C)CC(=O)O` / `OC(CC)CC(=O)O`
3hd	`OC(CCCCCC)CC(=O)O`
3hhx	`OC(CCC)CC(=O)O`
3hp	`OCCC(=O)O`
3hv	`OC(CC)CC(=O)O`
4hb	`OCCCC(=O)O`
la	`OC(C)C(=O)O`
phla	`OC(Cc1ccccc1)C(=O)O`

System-specific backbone bonds

System	Backbone bonds	Terminal marker
3hb	$5, 8, 10$	10
3hb_3hv	$6, 9, 11$	11
3hd	$6, 9, 11$	11
3hhx	$6, 9, 11$	11
3hp	$4, 7, 9$	9
3hv	$6, 9, 11$	11
4hb	$4, 7, 9$	9
la	$5, 8, 10$	10
phla	$6, 9, 13$	13

where $\odot$ denotes componentwise multiplication. This removes artificial jumps caused by atoms crossing periodic boundaries.

Because the simulations are performed in the NPT ensemble, the box size changes over time under the barostat, which adjusts the simulation cell to maintain the target pressure. To remove global barostat-induced scaling, each frame is mapped to a fixed reference box:

$$\tilde{\mathbf{x}}_{t,i} = \mathbf{x}_{t,i}^{\mathrm{unwrap}} \odot \frac{L_{\mathrm{ref}}}{L_t}.$$

In our experiments, L_{ref} is the time-averaged box length,

$$L_{\mathrm{ref}} = \frac{1}{T+1} \sum_{t=0}^{T} L_t,$$

corresponding to `barostat_mode=mean`. All backbone coordinates used by the models are these rescaled unwrapped coordinates.

For geometric features and `orthF`, we construct cutoff-based neighbor lists on the processed backbone coordinates using a radius graph with $r_{\mathrm{cut}} = 5.0$. Neighbor relations are computed in Euclidean space on the rescaled unwrapped coordinates. Bonded pairs are excluded from the nonbonded neighbor list, and the remaining edges are split into same-chain and other-chain pairs using molecular identities. The same-chain neighbors define the local backbone direction, while the other-chain neighbors define the local inter-chain direction used by `orthF`. For storage, per-frame neighbor lists are padded with -1 sentinel values; these invalid entries are ignored when constructing the geometric features.

B.3 Backbone Bond Selection

For each polymer system, chains are constructed as multiple polymer strings with the same degree of polymerization by repeating the corresponding monomer unit. Table 7 lists the monomer-unit SMILES used for each system together with the system-specific bond types used for backbone extraction. The 3hb_3hv system uses a combined repeat unit containing both 3hb and 3hv components, so both SMILES are listed for that system. Bond-type labels are local to each LAMMPS parameter file: the same numerical label in different systems does not necessarily denote the same chemical bond. We select monomer-linking bonds involving the bridging oxygen o_2, together with the terminal ho_2-o_2 bond used to locate chain endpoints.

B.4 Backbone Selection versus Clustering-Based Coarse Graining

The proposed backbone representation differs from clustering-based coarse graining. Rather than defining new coarse variables from geometry-dependent atom groups, we retain a chemically defined subset of atoms on the polymer backbone. This is the structure most directly relevant to entanglement, which is governed by non-crossing constraints on the chain contour.

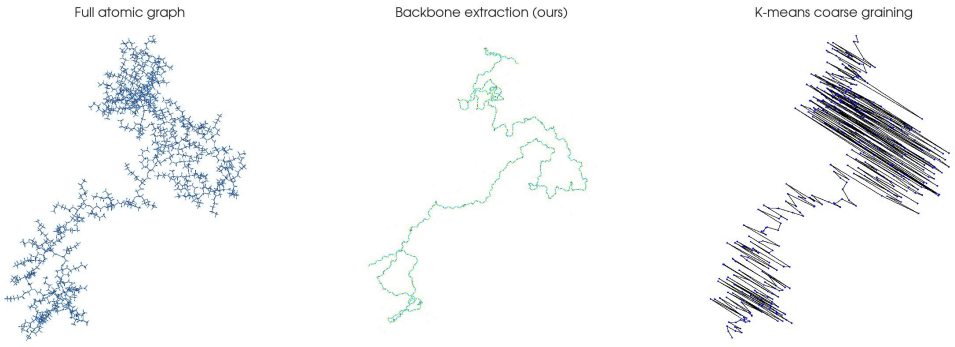


Figure 5: Comparison of representations for a 3hd chain: full atomic graph (3003 atoms), backbone extraction (409 atoms), and k-means coarse graining (409 beads). Backbone extraction preserves the chemical chain contour, whereas clustering groups atoms by spatial proximity.

Table 8: Dataset statistics for each polymer system before and after backbone extraction. Atom and bond counts are reported per chain; covalent bond connectivity is fixed throughout each trajectory.

		3hb_3hv	3hb	3hd	3hhx	3hp	3hv	4hb	la	phla
Chains		16	17	7	11	22	13	17	22	11
Full representation	Atoms	1248	1203	3003	1803	903	1503	1203	903	1903
	Bonds	1247	1202	3002	1802	902	1502	1202	902	2002
Backbone representation	Atoms	403	403	409	405	403	404	503	303	307
	Bonds	402	402	408	404	402	403	502	302	306

A clustering-based coarse-graining method partitions atoms into coordinate-dependent sets $\Pi(\mathbf{x}) = \{C_1, \ldots, C_K\}$ and defines coarse variables such as

$$\mathbf{z}_k = \frac{1}{|C_k|} \sum_{i \in C_k} \mathbf{x}_i.$$

Because the partition depends on geometry, it can vary across frames or across chemically identical chains. The resulting coarse variables may therefore change even when the underlying bonded graph is fixed.

In contrast, our reduction is defined by the fixed bonded graph and bond types. Chains with the same chemistry map to the same reduced backbone graph up to relabeling, and the same chain retains the same reduced structure across frames because the bonded graph does not change during the trajectory. The reduced state is simply the restriction of atomistic coordinates to the retained backbone atoms,

$$\mathbf{x}_{\mathrm{bb}} = \mathbf{x} \restriction_{V_{\mathrm{bb}}},$$

so no pseudo-atoms, centroid coordinates, or decoder are required.

This distinction matters for entanglement-sensitive dynamics: centroid-based coarse graining can distort the chain contour that determines intermolecular topological constraints, whereas backbone extraction preserves the physically relevant contour while reducing dimensionality. Table 8 summarizes the resulting reduction across all systems.

C Velocity Network Ablation

We first compare MLP, GAT, EGNN, MLP-Mixer, and Transformer architectures for parameterizing the velocity field v_θ. Graph-based models use the molecular backbone graph, with backbone atoms as nodes and bonded backbone connectivity as edges. Non-graph models receive the same coordinates and time conditioning but no molecular-edge information. All models use the same wCFM supervision and are evaluated with rollout MSE.

Table 9: Velocity-network architecture ablation. Rollout MSE is reported as mean ± standard deviation. The best result for each task is shown in bold.

Task	MLP	GAT	EGNN	MLP-Mixer	Transformer
3hb_3hv	0.100 ± 0.150	$\mathbf{0.007 \pm 0.003}$	0.010 ± 0.005	0.050 ± 0.020	0.070 ± 0.030
3hb	0.015 ± 0.010	$\mathbf{0.008 \pm 0.003}$	0.011 ± 0.005	0.090 ± 0.040	0.060 ± 0.020
3hd	0.040 ± 0.026	$\mathbf{0.006 \pm 0.002}$	0.042 ± 0.018	0.100 ± 0.030	0.060 ± 0.010
3hhx	0.020 ± 0.013	$\mathbf{0.007 \pm 0.002}$	0.013 ± 0.007	0.080 ± 0.030	0.060 ± 0.020
3hp	1.100 ± 6.140	$\mathbf{0.012 \pm 0.004}$	0.017 ± 0.008	0.080 ± 0.020	0.090 ± 0.040
3hv	0.041 ± 0.023	$\mathbf{0.007 \pm 0.003}$	0.012 ± 0.006	0.070 ± 0.030	0.080 ± 0.030
4hb	$\mathbf{0.009 \pm 0.004}$	0.010 ± 0.004	0.017 ± 0.007	0.120 ± 0.040	0.080 ± 0.030
la	0.021 ± 0.013	$\mathbf{0.006 \pm 0.002}$	0.007 ± 0.003	0.070 ± 0.030	0.050 ± 0.020
phla	0.012 ± 0.006	$\mathbf{0.003 \pm 0.001}$	0.005 ± 0.002	0.050 ± 0.020	0.040 ± 0.010
Total	0.151 ± 2.080	$\mathbf{0.007 \pm 0.004}$	0.015 ± 0.014	0.079 ± 0.030	0.066 ± 0.026

Table 9 shows that GAT achieves the lowest overall MSE and the best result on eight of nine systems. We therefore use the GAT-parameterized model as the default wCFM backbone in the main experiments.

C.1 Compatibility with an SE(3)-equivariant backbone

Although GAT performs best in the architecture ablation, it is not strictly SE(3)-equivariant. To test whether the benefit of `orthF` depends on this choice, we additionally evaluate `orthF` with an EGNN backbone under the same training and evaluation protocol. As shown in Table 10, adding `orthF` reduces displacement WSD across all nine systems, showing that its benefit persists with an SE(3)-equivariant velocity network.

Table 10: Displacement Wasserstein distance (WSD) for an EGNN velocity-field backbone with and without `orthF` across polymer systems. Results are reported as mean ± standard deviation across seeds; lower is better.

Polymer	EGNN	EGNN+`orthF`
3hb	7.719 ± 1.673	$\mathbf{0.025 \pm 0.010}$
3hb_3hv	1149.536 ± 215.707	$\mathbf{0.029 \pm 0.016}$
3hd	1.565 ± 0.695	$\mathbf{0.113 \pm 0.125}$
3hhx	7.480 ± 2.209	$\mathbf{0.040 \pm 0.019}$
3hp	1.544 ± 0.456	$\mathbf{0.020 \pm 0.010}$
3hv	1.910 ± 0.708	$\mathbf{0.032 \pm 0.017}$
4hb	0.791 ± 0.194	$\mathbf{0.107 \pm 0.046}$
la	5.258 ± 0.940	$\mathbf{0.014 \pm 0.006}$
phla	1.576 ± 0.578	$\mathbf{0.009 \pm 0.004}$

D Long-Horizon Rollout Evaluation

D.1 Per-system rollout stability

Across all nine systems, `orthF` yields lower displacement WSD in the early, middle, and late rollout intervals. The gap persists at longer horizons, although WSD increases over time for `orthF`, consistent with gradual rollout drift rather than complete error elimination.

D.2 Temporal Extrapolation

To evaluate temporal extrapolation, we train each model on the first $5\,\mathrm{ns}$ of a $50\,\mathrm{ns}$ MD trajectory for the 3hb system and roll it out beyond the training window. Table 12 reports sampling MSE and WSD over early, middle, and late intervals of the remaining trajectory. Across intervals, `orthF` generally

Table 11: Displacement WSD (Å) across rollout intervals for each polymer system. Values are mean ± standard deviation over three seeds; lower is better.

Dataset	Variant	Early	Middle	Late
3hb_3hv	Base	0.3793 ± 0.2413	0.4919 ± 0.3045	0.5124 ± 0.3052
	orthF	**0.0077 ± 0.0023**	**0.0548 ± 0.0342**	**0.1058 ± 0.0766**
3hb	Base	0.4809 ± 0.1653	0.5740 ± 0.1778	0.5695 ± 0.1759
	orthF	**0.0166 ± 0.0028**	**0.0630 ± 0.0030**	**0.0827 ± 0.0085**
3hd	Base	0.5341 ± 0.7084	0.6953 ± 0.8264	0.7609 ± 0.8556
	orthF	**0.0119 ± 0.0009**	**0.0619 ± 0.0029**	**0.0994 ± 0.0106**
3hhx	Base	0.8608 ± 0.1660	1.0159 ± 0.1681	1.0947 ± 0.1733
	orthF	**0.0107 ± 0.0027**	**0.0486 ± 0.0145**	**0.0976 ± 0.0223**
3hp	Base	0.5465 ± 0.3932	0.6582 ± 0.4898	0.7327 ± 0.4438
	orthF	**0.0154 ± 0.0039**	**0.0436 ± 0.0162**	**0.0731 ± 0.0242**
3hv	Base	1.3067 ± 0.8007	1.5894 ± 1.0415	1.6589 ± 1.0873
	orthF	**0.0165 ± 0.0124**	**0.1044 ± 0.0811**	**0.1468 ± 0.0837**
4hb	Base	0.3340 ± 0.1391	0.4205 ± 0.1620	0.4734 ± 0.1747
	orthF	**0.0077 ± 0.0009**	**0.0752 ± 0.0612**	**0.1173 ± 0.0698**
la	Base	1.0561 ± 0.6865	1.2774 ± 0.8021	1.2186 ± 0.7571
	orthF	**0.0072 ± 0.0038**	**0.0258 ± 0.0141**	**0.0472 ± 0.0308**
phla	Base	0.0844 ± 0.0546	0.1225 ± 0.0861	0.1423 ± 0.0849
	orthF	**0.0103 ± 0.0017**	**0.0255 ± 0.0063**	**0.0429 ± 0.0210**

Table 12: Temporal extrapolation on the 3hb 50 ns trajectory. Models are trained on the first 5 ns and evaluated over later intervals. Values are mean ± standard deviation; lower is better.

Model	Variant	Early interval (0–20 ns)		Middle interval (20–35 ns)		Late interval (35–50 ns)	
		Sampling (MSE↓)	WSD↓	Sampling (MSE↓)	WSD↓	Sampling (MSE↓)	WSD↓
TFM	Base	0.006 ± 0.000	0.020 ± 0.001	0.006 ± 0.001	0.021 ± 0.004	0.006 ± 0.000	0.017 ± 0.006
	orthF	**0.006 ± 0.000**	**0.018 ± 0.002**	**0.006 ± 0.000**	**0.021 ± 0.007**	**0.005 ± 0.000**	**0.017 ± 0.002**
wCFM	Base	0.007 ± 0.000	0.035 ± 0.005	0.007 ± 0.001	0.040 ± 0.014	0.007 ± 0.000	0.033 ± 0.004
	orthF	**0.007 ± 0.000**	**0.029 ± 0.004**	**0.006 ± 0.000**	**0.032 ± 0.011**	**0.006 ± 0.000**	**0.027 ± 0.004**
MLCGMD	Base	0.056 ± 0.006	0.602 ± 0.031	0.058 ± 0.007	0.634 ± 0.028	0.061 ± 0.009	0.688 ± 0.035
	orthF	**0.053 ± 0.006**	**0.587 ± 0.029**	**0.055 ± 0.008**	**0.620 ± 0.032**	**0.060 ± 0.010**	**0.673 ± 0.041**

improves each model, with the clearest gains for wCFM and MLCGMD. TFM performs best overall, while TFM-orthF remains competitive with or slightly improves over TFM-Base. These results suggest that orthF helps preserve rollout quality beyond the 5 ns training window, although the gains are more modest than in displacement-distribution and topology-related evaluations.

E Additional Experimental Results

E.1 Window-size ablation across polymer systems

Across the full benchmark, fixed-window performance is non-monotonic for both variants. With orthF, mixed-window training achieves the lowest mean MSE, improving from 0.0244 for the best fixed window to 0.0156. In contrast, the Base model is unstable under mixed-window training, with mean MSE increasing from 0.593 to 8.826. This suggests that orthF stabilizes learning across temporal scales, rather than multi-scale sampling being uniformly beneficial on its own.

E.2 Time-resolved radius-of-gyration behavior

To complement the distribution-level structural metrics reported in the main text, we examine the time evolution of the radius-of-gyration error during rollout. Fig. 6 shows that the effect of orthF on global chain size is mixed across model families, consistent with the R_g distribution metrics in Tab. 3. We therefore treat R_g as a complementary global-structure diagnostic rather than a uniformly improved observable.

Table 13: Trajectory MSE for different temporal-window settings across the nine polymer systems. Values are mean ± standard deviation; lower is better. Mixed samples from $\mathcal{W} = \{32, 64, 128, 256, 501\}$.

Orth	w	3hb_3hv	3hb	3hd	3hhx	3hp	3hv	4hb	la	phla
Base	32	27.10±6.79	1.45±0.27	0.05±0.03	0.88±0.31	0.92±0.19	0.37±0.09	0.60±0.12	0.21±0.05	0.05±0.01
	64	4.94±1.05	1.03±0.20	0.36±0.08	0.72±0.16	2.44±0.60	0.36±0.08	0.81±0.20	0.70±0.14	0.22±0.06
	128	16.41±2.96	36.29±9.91	0.35±0.11	1.04±0.34	12.28±2.27	0.61±0.15	0.09±0.03	9.59±1.87	0.27±0.05
	256	0.17±0.04	9.44±1.66	0.10±0.04	19.81±8.69	1.42±0.25	0.25±0.10	0.44±0.08	3.36±0.58	0.04±0.01
	501	0.02±0.01	0.89±0.22	0.14±0.07	1.55±0.68	0.08±0.03	0.28±0.08	0.12±0.05	2.09±0.38	0.17±0.10
	Mixed	3.81±1.32	0.67±0.14	0.28±0.09	0.29±0.08	8.56±1.76	0.28±0.05	1.60±0.28	63.89±12.37	0.05±0.01
orthF	32	0.02±0.01	0.02±0.01	0.07±0.04	0.06±0.03	0.02±0.01	0.09±0.05	0.09±0.03	0.02±0.01	0.03±0.02
	64	0.05±0.01	0.02±0.01	0.02±0.01	0.04±0.03	0.03±0.01	0.03±0.02	0.04±0.02	0.02±0.01	0.01±0.00
	128	0.01±0.01	0.04±0.02	0.03±0.02	0.07±0.04	0.02±0.01	0.06±0.05	0.04±0.02	0.01±0.01	0.01±0.00
	256	0.02±0.01	0.02±0.01	**0.01±0.01**	0.04±0.03	0.02±0.01	0.06±0.04	**0.03±0.01**	0.01±0.00	0.01±0.00
	501	0.05±0.01	0.03±0.02	0.03±0.02	0.05±0.04	0.03±0.02	**0.01±0.01**	0.06±0.03	0.20±0.15	0.02±0.01
	Mixed	**0.01±0.00**	**0.01±0.00**	0.03±0.02	**0.02±0.02**	**0.01±0.01**	0.02±0.01	0.03±0.01	**0.01±0.00**	**0.00±0.00**

Table 14: Mean MSE averaged over all 9 polymers. Mixed uses $\mathcal{W} = \{32, 64, 128, 256, 501\}$.

Model	Setting	Mean MSE
Base	Best single window ($w = 501$)	0.593
Base	Mixed	8.826 (unstable)
orthF	Best single window ($w = 256$)	0.0244
orthF	Mixed	**0.0156**

E.3 Additional structural evaluation with PolyConf metrics

We additionally evaluate generated conformations using the PolyConf S-MAT-R and S-MAT-P metrics Wang et al. (2025), for which lower values indicate closer agreement with the reference conformations. The results are mixed across systems: orthF improves all four scores on 3hb, whereas the Base model performs better on 3hv. We therefore treat these metrics as complementary to the bond-length, dihedral, and radius-of-gyration evaluations reported in the main text.

Table 15: PolyConf structural matching metrics on 3hb and 3hv. Lower is better; bold marks the better result within each polymer and metric.

	3hb		3hv	
Metric	Base	+orthF	Base	+orthF
S-MAT-R mean	10.857	**10.458**	**9.950**	12.317
S-MAT-R median	10.938	**10.515**	**9.797**	12.667
S-MAT-P mean	10.673	**10.165**	**9.939**	11.683
S-MAT-P median	10.628	**9.983**	**9.812**	12.316

F Additional Topology Evaluation

F.1 Complementary Z1+ analysis

We additionally evaluate selected chain-level quantities using Z1+ Kröger et al. (2023); Kröger (2005). Compared with Base, orthF remains closer to MD in the root-mean-square end-to-end distance $R_{\mathrm{ee,rms}}$ and average bond length $\langle b \rangle$ at the evaluated rollout frames. In contrast, the classical-kink entanglement length N_e^{CK} does not show a consistent improvement: the two learned models are identical at the intermediate frame, while Base is closer to MD at the final evaluated frame.

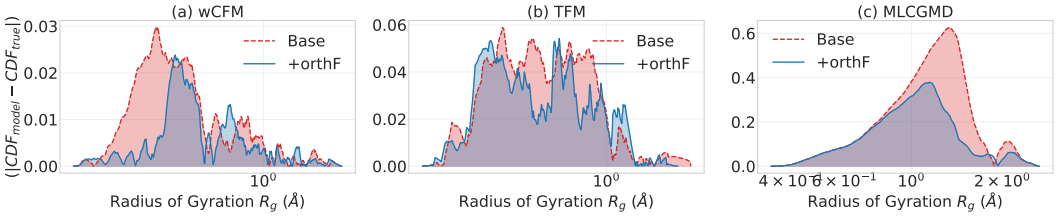


Figure 6: Radius-of-gyration error relative to MD during rollout. Lower values indicate closer preservation of the reference chain-size statistic; Tab. 3 gives the complementary distribution-level comparison.

Table 16: Complementary Z1+ analysis at selected rollout frames. Bold marks the learned result closer to MD.

Metric	Frame	MD	+orthF	Base
$R_{\mathrm{ee,rms}}$ (Å)	1	51.928	51.540	51.540
	50	49.047	**51.103**	51.439
	97	50.728	**51.375**	51.965
N_e^{CK}	1	131.643	129.161	129.161
	50	120.104	86.675	86.675
	97	131.643	67.803	**77.815**
$\langle b \rangle$ (Å)	1	1.468	1.467	1.467
	50	1.468	**1.513**	1.547
	97	1.468	**1.586**	1.699

F.2 Metrics and implementation

We evaluate topology-related fidelity using average crossing number (ACN), writhe, and the Gauss linking integral. These provide continuous geometric descriptors of self-entanglement and inter-chain coupling for open polymer chains Panagiotou et al. (2010; 2013).

Self-ACN. For each chain, the self-average crossing number measures the unsigned geometric complexity of the chain. For a polygonal chain, we compute

$$\mathrm{ACN}_{\mathrm{self}} = \frac{1}{2\pi} \sum_{i<j} |\Omega_{ij}|,$$

where Ω_{ij} is the signed solid-angle contribution associated with segment pair (i, j), evaluated using the polygonal-chain construction of Klenin and Langowski Klenin & Langowski (2000). ACN and related crossing-number measures have been used to characterize knot and polymer-chain geometry Katritch et al. (1996); Diao et al. (2005). We report the mean self-ACN over chains.

Cross-ACN. For two distinct chains A and B, we compute the unsigned crossing contribution

$$\mathrm{ACN}_{\mathrm{cross}} = \frac{1}{4\pi} \sum_{i \in A} \sum_{j \in B} |\Omega_{ij}|,$$

using the same segment-pair construction Klenin & Langowski (2000). This quantity measures geometric interpenetration between different chains while discarding crossing sign. We report the mean cross-ACN over distinct chain pairs. For open chains under periodic boundary conditions, linking and crossing measures require consistent treatment of periodic images Panagiotou (2015).

Writhe. Writhe is the signed Gauss self-interaction of a curve and characterizes its global coiling geometry. For polygonal chains, we evaluate it from the corresponding segment-pair contributions Klenin & Langowski (2000). Because positive and negative values can cancel across an ensemble without a preferred chirality, we report the mean absolute writhe,

$$\langle |\mathrm{Wr}| \rangle,$$

rather than the signed ensemble mean. Writhe is a standard global descriptor of polymer-chain self-entanglement Panagiotou et al. (2010; 2013).

Inter-chain Gauss linking integral. For two distinct chains A and B, we compute

$$\mathrm{Lk}(A,B) = \frac{1}{4\pi}\int_A\int_B \frac{(\mathrm{d}\mathbf{r}_A \times \mathrm{d}\mathbf{r}_B)\cdot(\mathbf{r}_A - \mathbf{r}_B)}{\|\mathbf{r}_A - \mathbf{r}_B\|^3}.$$

For closed curves, this yields the classical linking number. For open chains, it is generally non-integer and is interpreted as a continuous measure of inter-chain geometric coupling Panagiotou et al. (2010; 2013). We report

$$\langle|\mathrm{Lk}|\rangle$$

over distinct chain pairs to avoid cancellation between positive and negative contributions.

Temporal aggregation. For each rollout frame, self-ACN and absolute writhe are averaged over chains, while cross-ACN and absolute linking are averaged over distinct chain pairs. We report statistics aggregated over the rollout together with the final-frame values. When reporting "first $\rightarrow$ last", the first value corresponds to frame 0 and the second to frame 500, the final sampled rollout frame. Because frame 0 is identical across MD, Base, and `orthF`, deviations at later frames quantify topology-related drift from the shared initial configuration.

Signed writhe and linking averages are omitted because sign cancellation makes them less informative for an ensemble without a preferred chirality. We therefore report unsigned ACN together with absolute writhe and linking throughout.

G Training Details and Hyperparameter Selection

We use a single set of training hyperparameters for each model class across all polymer systems unless otherwise stated. Hyperparameters were selected using preliminary validation runs on held-out validation trajectory windows, using distributional accuracy and rollout stability as the main criteria. The test split was not used for hyperparameter selection.

For the GNN-based models, we train with AdamW using learning rate 10^{-4}, cosine decay with a 1000-step warmup, gradient clipping at norm 1.0, hidden dimension 256, 5 message-passing layers, and dropout 0.1. For the MLP baseline, we use Adam with learning rate 10^{-3}, hidden dimension 512, 5 layers, dropout 0.1, and gradient clipping at norm 1.0. Unless otherwise stated, GNN models are trained for 500 epochs and MLP models for 1000 epochs. For `orthF`, we use the same supervised and geometric-loss weights across datasets after validation selection. Sampling uses the midpoint ODE solver with step size 0.05.

All experiments are executed on A100 GPUs.

H Broader Impact

This work aims to accelerate polymer-dynamics simulation by replacing expensive molecular-dynamics rollouts with learned surrogate dynamics. Potential positive impacts include reducing computational cost and energy use in polymer simulation, and enabling faster screening of materials relevant to soft matter, biomaterials, and industrial polymer design. The work does not involve human subjects, personal data, surveillance, or decision-making about individuals. Potential negative impacts are indirect: inaccurate learned rollouts could mislead downstream materials-design decisions if used without validation, and improved simulation throughput could accelerate both beneficial and undesirable materials discovery. We therefore view the method as a scientific surrogate that should be used with domain validation against molecular dynamics or experimental measurements, especially before deployment in high-stakes materials applications.