ABLE: ALL-ATOM BIOMOLECULAR TRAJECTORY GENERATION WITH ACTIVE-LIGAND ENRICHMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Molecular dynamics simulations provide an atomistic bridge between *in silico* modeling and experiments, but their high computational cost limits applications in high-throughput drug discovery. Recent generative models accelerate molecular dynamics by modeling large-lag transitions, yet models trained primarily on active protein–ligand complexes may inherit inductive biases that weaken active–decoy discrimination, limiting their ability to serve as general substitutes for physics-based simulations. To address this limitation, we introduce ABLE, an all-atom biomolecular trajectory generation framework designed to extend trajectory modeling toward virtual screening. At the post-training stage, ABLE incorporates an on-policy reinforcement learning procedure centered on a newly defined protein–ligand interfacial expansion rate. By quantifying the instantaneous entropy growth of a local interfacial distribution, this quantity provides the optimization signal that drives distinct dynamical behaviors for active ligands and decoys using only static cross-docked complex structures. Experiments demonstrate that ABLE achieves competitive trajectory generation performance on both proteins and protein–ligand complexes and, after reinforcement learning, delivers state-of-the-art early enrichment in virtual screening among the evaluated baselines.

1 INTRODUCTION

Molecular Dynamics (MD) simulations provide an atomistic view of molecular motion and are indispensable for studying biomolecular mechanisms and structure-based drug discovery (Lindorff-Larsen et al., 2011; Dror et al., 2011). Yet, conventional MD relies on numerical integration with femtosecond-scale time steps, making biologically relevant timescales prohibitively expensive to simulate (Leimkuhler & Matthews, 2013; Balusek et al., 2019). Although enhanced-sampling techniques such as umbrella sampling and metadynamics can accelerate transitions across free-energy barriers (Torrie & Valleau, 1977; Laio & Parrinello, 2002; Barducci et al., 2008), they often require carefully designed collective variables and remain costly for large biomolecular systems (Blumer et al., 2024; Ray & Rizzi, 2025; Invernizzi & Parrinello, 2022). Efficiently generating physically meaningful molecular dynamics therefore remains a fundamental challenge.

Generative models offer a promising alternative by learning molecular distributions or dynamical transitions directly from data. One line of work generates equilibrium conformational ensembles, as exemplified by AlphaFlow (Jing et al., 2024a) and BioEmu (Lewis et al., 2025), enabling highly parallel sampling of protein conformations. However, such models do not explicitly describe kinetic pathways or temporal evolution, which are essential for characterizing dynamic processes such as ligand binding and unbinding (Tiwary et al., 2015; Bianciotto et al., 2021). A complementary line of work instead learns a finite-time transition kernel $q_\tau(x, \mathrm{d}y) := \mathcal{P}(X_{t+\tau} \in \mathrm{d}y \mid X_t = x)$ at a lag time τ much larger than the integration time step Δt of conventional MD simulations, and trajectories can then be generated autoregressively by repeatedly propagating the current state according to the learned transition kernel (Klein et al., 2023; Jing et al., 2024b). Recent trajectory generative models have been extended from small molecules (Wu & Li, 2023) to proteins and protein–ligand complexes (Jing et al., 2024b; Liu et al., 2025; Yu et al., 2026), raising the possibility of replacing expensive portions of MD simulation with learned dynamics.

However, existing trajectory generative models remain insufficient for practical MD-driven applications. One important limitation is the highly skewed composition of existing open-source trajectory

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.

datasets, which predominantly comprise experimentally validated protein–ligand complexes and contain relatively few trajectories for negative protein–decoy pairs. Training on such datasets may introduce a spurious inductive bias that protein–ligand complexes are generally associated with favorable interactions. As a result, models trained on such data may have limited ability to distinguish active ligands from decoys in virtual screening. By contrast, physics-based MD simulations can, in principle, capture differences in their dynamical behavior, including variations in binding stability (Guterres & Im, 2020).

To address this limitation, we introduce an **A**ll-atom **B**iomolecular trajectory generation framework for active-**L**igand **E**nrichment (ABLE). ABLE leverages static cross-docked complexes containing both active ligands and decoys to learn ligand-dependent dynamics without requiring protein–decoy MD trajectories, realized through a novel on-policy reinforcement learning procedure. Specifically, given a complex structure, we introduce tracer particles that are initialized around the protein–ligand interface and evolved under a stochastic differential equation driven by an interpolated spatial velocity field over $\mathbb{R}^3$. The instantaneous entropy growth of the tracer distribution measures interfacial expansion and serves as the reward, encouraging active complexes to preserve stable binding behavior while promoting separation of decoys from the protein. In this way, ABLE introduces active–decoy discrimination into trajectory generation using only static complex structures. Beyond reinforcement learning, ABLE builds on the variational bridge framework (Yu et al., 2026), which unifies pretraining on high-quality static structures with finetuning on MD trajectories across diverse molecular domains, enabling strong distributional fidelity while maintaining high physical validity.

Overall, the contributions of this work are summarized as follows:

1. To the best of our knowledge, ABLE is the first trajectory generation framework tailored for active-ligand enrichment and virtual screening, extending generative dynamics modeling to a key application of physics-based MD simulations.
2. We propose a novel on-policy reinforcement learning approach that induces distinct kinetic behaviors for active ligands and decoys using only static cross-docked complex structures, without requiring MD trajectories of protein–decoy complexes.
3. ABLE achieves competitive performance in trajectory generation for both protein monomers and protein–ligand complexes, and attains state-of-the-art early enrichment in virtual screening after reinforcement learning.

2 Related Work

Trajectory Generation Trajectory generation offers a generative alternative for modeling molecular kinetics and approximating physics-based MD simulations. By operating at a large lag time $\tau \gg \Delta t$, these models learn the transition kernel $q_\tau(x, \mathrm{d}y) \coloneqq \mathcal{P}(X_{t+\tau} \in \mathrm{d}y \mid X_t = x)$, focusing on large-scale conformational transitions rather than high-frequency local vibrations. Recent advances in diffusion models and flow matching have enabled trajectory generation across small molecules (Wu & Li, 2023), peptides and proteins (Klein et al., 2023; Schreiner et al., 2023; Jing et al., 2024b; Yu et al., 2025; Shen et al., 2025; Xu et al., 2026), and protein–ligand complexes (Liu et al., 2025; Feng et al., 2026a;b; Shi et al., 2026). Among them, MDGen (Jing et al., 2024b) unifies multiple dynamical prediction tasks through flexible frame conditioning, while PVB (Yu et al., 2026) shows that large-scale pretraining on equilibrium conformations can improve the physical validity of generated trajectories, thereby addressing the overemphasis of prior work on distributional similarity. Despite these advances, existing protein–ligand trajectory models are predominantly trained on active complexes, potentially introducing a spurious bias toward favorable interactions and limiting their ability to distinguish active ligands from decoys through kinetic behavior.

Virtual Screening Virtual screening aims to efficiently prioritize candidate ligands according to their binding propensity toward a target protein. Classical docking methods, such as AutoDock Vina (Trott & Olson, 2010), enable efficient large-scale screening with approximate scoring functions, whereas more rigorous physics-based approaches, such as free energy perturbation (FEP) (Wang et al., 2015), offer higher-fidelity binding free-energy estimates at substantially greater computational cost. To improve scalability, recent learning-based methods either infer protein–ligand compatibility from molecular and protein representations (Öztürk et al., 2018; Nguyen et al., 2021; Gao et al.,

2023; Zhang et al., 2025a), or explicitly model interactions from static three-dimensional complex structures (Stepniewska-Dziubinska et al., 2018; Jiménez et al., 2018; Jin et al., 2023; Feng et al., 2024; Kong et al., 2024). Despite substantial progress, these approaches largely characterize binding from sequences or static structural observations, without explicitly modeling the temporal evolution of protein–ligand interactions. Recent evidence further highlights the importance of ligand-dependent conformational flexibility for binding affinity prediction (Min et al., 2024; Zhang et al., 2025b), suggesting that a single static structure may provide an incomplete description of molecular recognition. These observations motivate trajectory-based virtual screening, where efficiently generated dynamics can provide complementary kinetic information for distinguishing active ligands from decoys without the computational cost of explicit MD simulations.

3 METHOD

3.1 PROBLEM FORMULATION

Notations Each molecular system is represented by $\{z, x, \mathcal{E}\}$, where $z \in \mathbb{N}^N$ denotes the atomic numbers of the N heavy atoms, $x \in \mathbb{R}^{N\times 3}$ denotes their Cartesian coordinates, and $\mathcal{E} \in \mathbb{N}^{2\times E}$ specifies the indices of the E covalent bonds in the system. Accordingly, an MD trajectory of the system is represented as $\{z, (x_0, x_{\Delta t}, x_{2\Delta t}, \ldots), \mathcal{E}\}$, where Δt denotes the integration time step. Given a lag time $\tau \gg \Delta t$, we define the data transition kernel as $q_\tau(x, \mathrm{d}y) \coloneqq \mathcal{P}(X_{t+\tau} \in \mathrm{d}y \mid X_t = x)$. During training, a pair of conformations separated by the lag time τ, *e.g.*, $(x_t, x_{t+\tau})$, is randomly sampled from a trajectory. For notational simplicity, we hereafter denote such a pair by (x_0, x_1). In addition, W and φ (with appropriate subscripts) refer to linear projections and parameterized networks, respectively. We also use $H \in \mathbb{N}_+$ for the hidden dimension of the model, $D \in \mathbb{N}_+$ for the dimension of the latent scalar feature space, $v[i]$ for the i-th element of the vector v, abs for the element-wise absolute value operation, $\langle\cdot,\cdot\rangle$ for the inner product, $\cos(u, v)$ for the cosine similarity between two vectors u and v, and $\|\cdot\|_2$ for the ℓ_2-norm of a vector.

We pursue two primary objectives in this work: (i) Train a conditional generative model $p_\theta(\cdot \mid x)$, parametrized by θ, to approximate the transition kernel q_τ while balancing distributional fidelity and the physical validity of generated trajectories. (ii) Finetune p_θ on cross-docked datasets to distinguish active ligands from decoys based on the kinetic behaviors captured by the generated trajectories.

3.2 MODEL ARCHITECTURE

The overall workflow of ABLE is presented in Figure 1. The architecture of ABLE comprises three main components: encoder, conditioner, and decoder. All three components are constructed from Memory-Efficient Equivariant Transformer (MEET) blocks (Jiao et al., 2026a), with different numbers of blocks stacked in each component. A standard MEET block takes as input the atomic features $h_{\mathrm{in}} \in \mathbb{R}^{N\times H}$, Cartesian coordinates $x \in \mathbb{R}^{N\times 3}$, and covalent bond indices $\mathcal{E}$, and produces

$$h_{\mathrm{out}}, \boldsymbol{v}_{\mathrm{out}} = \mathrm{MEET}(h_{\mathrm{in}}, x, \mathcal{E}), \tag{1}$$

where $h_{\mathrm{out}} \in \mathbb{R}^{N\times H}$ denotes $\mathrm{SE}(3)$-invariant scalar features, while $\boldsymbol{v}_{\mathrm{out}} \in \mathbb{R}^{N\times 3\times H}$ denotes $\mathrm{SO}(3)$-equivariant vector features. As the encoder, conditioner, and decoder are each composed of stacked MEET blocks, they inherit the corresponding invariant and equivariant properties.

Encoder The encoder φ_e maps the atomic features and Cartesian coordinates of an input structure to stochastic latent representations. Specifically, given the encoder outputs

$$h_e, \boldsymbol{v}_e = \varphi_e(\mathrm{embed}(z), x, \mathcal{E}), \tag{2}$$

we parameterize the latent distributions and apply the reparameterization trick (Kingma & Welling, 2013; Mueller et al., 2017) as follows:

$$\mu_h = h_e W_1^\top \in \mathbb{R}^{N\times D}, \quad V_h = -\mathrm{abs}(h_e W_2^\top) \in \mathbb{R}^{N\times D}, \quad V_x = -\mathrm{abs}(h_e W_3^\top) \in \mathbb{R}^{N\times 1}, \tag{3}$$

$$\hat{h} = (\mu_h + \exp(V_h/2)\epsilon_h)\, W_4^\top, \qquad \hat{x} = x + s_x \cdot \exp(V_x/2)\epsilon_x, \qquad \epsilon_h, \epsilon_x \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}). \tag{4}$$

Here, V_h and V_x parameterize the log-variances, $\hat{h} \in \mathbb{R}^{N\times H}$ and $\hat{x} \in \mathbb{R}^{N\times 3}$ denote the latent atomic features and coordinates, respectively. The scalar $s_x \in \mathbb{R}_{\geq 0}$ controls the scale of stochastic perturbations applied to the latent coordinates, and its role will be discussed in detail in Section 3.3.

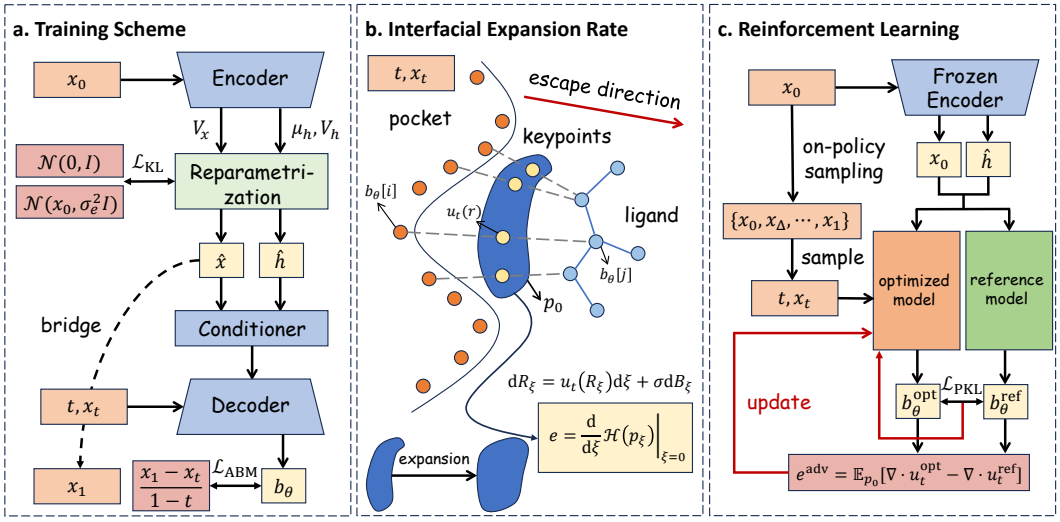


Figure 1: Overview of ABLE. **a.** ABLE encodes x_0 into variational latent variables $(\hat{x}, \hat{h})$ and learns the transition to x_1 through augmented bridge matching. **b.** Given a protein–ligand conformation x_t, we define an initial tracer distribution p_0 concentrated around the interface and represent it using a finite set of sampled interfacial keypoints. The model-predicted vector field $b_\theta(t, x_t)$ induces a local velocity field $u_t(r)$ that transports these tracers. Its instantaneous entropy growth rate e measures interfacial expansion, with larger values indicating a more source-like outward flow and a stronger tendency toward protein–ligand separation. **c.** During reinforcement learning, an on-policy state (t, x_t) is evaluated by the optimized and reference models. The relative expansion rate e^{adv}, derived from the expected difference in local velocity-field divergence, provides the optimization signal together with path-space KL regularization.

We impose a standard Gaussian prior $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$ on the latent scalar features and a Gaussian prior $\mathcal{N}(x, \sigma_e^2 \boldsymbol{I})$ centered at the input coordinates x on the latent coordinates, where σ_e is a hyperparameter controlling the scale of coordinate noise (Yu et al., 2026). Accordingly, the corresponding Kullback–Leibler (KL) divergence terms are given by

$$D_{\text{KL}}^h = -\frac{1}{2}\left(1 + V_h - \|\mu_h\|_2^2 - \exp(V_h)\right), \; D_{\text{KL}}^x = -\frac{1}{2}\left(1 + V_x - 2\log\sigma_e - \frac{\exp(V_x)}{\sigma_e^2}\right). \quad (5)$$

Empirically, we observe that the model tends to underutilize the information encoded in the latent scalar features. To alleviate this issue, we employ the free-bits strategy (Kingma et al., 2016) by introducing a threshold $\varepsilon \in \mathbb{R}_+$ for D_{KL}^h, such that the KL regularization on the latent scalar features is activated only when D_{KL}^h exceeds ε. The resulting KL loss is defined as

$$\mathcal{L}_{\text{KL}} = w_h D_{\text{KL}}^h \mathbb{I}(D_{\text{KL}}^h > \varepsilon) + w_x D_{\text{KL}}^x, \quad (6)$$

where w_h and w_x control the relative contributions of two KL terms, respectively.

Conditioner The conditioner φ_c processes the latent atomic features $\hat{h}$ and coordinates $\hat{x}$ to obtain

$$h_c, \boldsymbol{v}_c = \varphi_c(\hat{h}, \hat{x}, \mathcal{E}). \quad (7)$$

The scalar representation $h_c \in \mathbb{R}^{N \times H}$ encodes the reparameterized latent coordinates and is subsequently provided to the decoder as the conditioning signal for bridge matching.

Decoder The decoder φ_d transports the latent coordinates $\hat{x}$ toward the target coordinates x_1 using augmented bridge matching (De Bortoli et al., 2023). Let $X_0 = \hat{x}$ and $X_1 = x_1$, and denote their joint distribution by $\Pi_{0,1}$. We construct a scaled Brownian bridge with diffusion coefficient σ between the two endpoints. For any $t \in [0, 1]$, its conditional marginal admits the reparameterization

$$X_t = (1-t)X_0 + tX_1 + \sigma\sqrt{t(1-t)}\epsilon, \qquad \epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}), \qquad (X_0, X_1) \sim \Pi_{0,1}. \quad (8)$$

Conditioned on (X_0, X_1), the bridge equivalently satisfies the forward SDE

$$\mathrm{d}X_t = \frac{X_1 - X_t}{1-t}\,\mathrm{d}t + \sigma\,\mathrm{d}B_t. \tag{9}$$

Bridge matching replaces the conditional bridge drift in Equation (9) with a learnable vector field b_θ. In ABLE, the decoder parameterizes this drift at the intermediate state X_t and diffusion time t:

$$b_\theta(t, X_0, X_t, \mathcal{E}) \coloneqq \boldsymbol{v}_d[\ldots, 0] \in \mathbb{R}^{N\times 3}, \quad \text{s.t.} \quad h_d, \boldsymbol{v}_d = \varphi_d(h_c, X_t, t, \mathcal{E}). \tag{10}$$

The decoder is trained to regress the predicted drift toward the conditional Brownian bridge drift:

$$\mathcal{L}_{\mathrm{ABM}} = \mathbb{E}_{t\sim\mathrm{Unif}(0,1),(X_0,X_1)\sim\Pi_{0,1},X_t\sim\mathbb{Q}_{t|0,1}(\cdot|X_0,X_1)}\left[\left\| b_\theta(t, X_0, X_t, \mathcal{E}) - \frac{X_1 - X_t}{1-t}\right\|_2^2\right]. \tag{11}$$

3.3 Learning to Generate Trajectories

Following Yu et al. (2026), the training paradigm of ABLE consists of two stages: (I) pretraining on equilibrium conformations across diverse molecular domains and (II) finetuning on domain-specific MD trajectories. Both stages share the architecture introduced in Section 3.2, with different constructions of the source and target states.

In Stage I, given an equilibrium conformation $\{z, x, \mathcal{E}\}$, we formulate pretraining as a self-reconstruction task by setting both the input and target coordinates to x. To prevent trivial reconstruction, we set $s_x = 1$ in Equation (4), introducing stochastic perturbations into the latent coordinates from which the decoder reconstructs the original conformation.

In Stage II, given a trajectory pair $\{z, (x_0, x_1), \mathcal{E}\}$, the encoder takes x_0 as input and the decoder is trained toward the target endpoint x_1. To gradually remove stochasticity from the latent coordinates, we anneal the coordinate noise scale as $s_x = \max(0, 1 - N_{\text{iter}}/S)$, where $N_{\text{iter}} \in \mathbb{N}$ is the current training iteration and $S \in \mathbb{N}_+$ is the total number of annealing steps. Once $N_{\text{iter}} \geq S$, the coordinate perturbation vanishes and $\hat{x} = x_0$, so the decoder directly learns the conditional transition $q_\tau(x_1 \mid x_0)$.

For both training stages, the overall objective is defined as $\mathcal{L}_{\mathrm{MD}} = \mathcal{L}_{\mathrm{KL}} + \mathcal{L}_{\mathrm{ABM}}$.

3.4 Modulating Kinetic Behaviors via On-Policy RL

After training on MD trajectories, we further modulate the learned dynamics to distinguish active ligands from decoys. Motivated by the observation that active ligands generally maintain more stable binding poses during MD simulations (Guterres & Im, 2020), we perform on-policy Reinforcement Learning (RL) on cross-docked complexes to induce such discriminative kinetic behaviors without requiring MD trajectories of protein–decoy systems.

Formally, with the encoder φ_e frozen, we aim to optimize the vector field b_θ^{opt}, which is initialized from the MD-finetuned reference vector field b_θ^{ref}. As an on-policy training scheme, we consider trajectories $X_{0:1}$ generated by b_θ^{opt} from the initial state $X_0 = x_0$, and sample a diffusion time $t \in [0, 1]$ along each trajectory. For $m \in \{\mathrm{ref}, \mathrm{opt}\}$, let $u_t^m : \mathbb{R}^3 \to \mathbb{R}^3$ denote a continuous spatial velocity field interpolated from the atomic velocities given by b_θ^m at X_t. We introduce an auxiliary tracer process $(R_\xi)_{\xi\geq 0}$ evolving under u_t^m according to

$$\mathrm{d}R_\xi = u_t^m(R_\xi)\,\mathrm{d}\xi + \sigma\,\mathrm{d}B_\xi, \qquad R_0 \sim \gamma_t(\cdot \mid X_t), \tag{12}$$

where $\xi \geq 0$ denotes the auxiliary evolution time, and γ_t denotes a probability distribution concentrated around the protein–ligand interface at X_t. Let p_ξ^m denote the probability density of R_ξ, such that $p_0^m = \gamma_t$. We define the interfacial expansion rate as the instantaneous entropy growth rate of the tracer distribution under the learned dynamics:

$$e^m(t, X_t) \coloneqq \frac{\mathrm{d}}{\mathrm{d}\xi}\mathcal{H}(p_\xi^m)\bigg|_{\xi=0} = -\frac{\mathrm{d}}{\mathrm{d}\xi}\mathbb{E}_{R_\xi\sim p_\xi^m}\left[\log p_\xi^m(R_\xi)\right]\bigg|_{\xi=0}. \tag{13}$$

Interpretation of e^m We next provide a physical interpretation of the interfacial expansion rate e^m. By definition, the spatial velocity field u_t^m characterizes the local motion induced by the molecular

dynamics around the configuration X_t, while the tracer distribution $\gamma_t(\cdot \mid X_t)$ localizes this flow near the protein–ligand interface. Accordingly, e^m measures the local expansion of the tracer distribution: a larger e^m corresponds to a more source-like, positively divergent flow around the interface, causing tracer mass to spread outward. Such expansion provides an approximate measure of the tendency toward protein–ligand separation and, when accumulated over time, can give rise to distinct kinetic behaviors in the generated trajectories.

Training Objective We define the relative interfacial expansion as

$$e^{\mathrm{adv}}(t, X_t) \coloneqq e^{\mathrm{opt}}(t, X_t) - e^{\mathrm{ref}}(t, X_t), \tag{14}$$

which measures the change in interfacial expansion induced by the optimized vector field relative to the MD-trained reference. For decoys, we encourage e^{adv} to exceed a positive margin λ_m, promoting interfacial expansion and facilitating ligand separation from the protein. For active ligands, we instead penalize deviations of e^{adv} from zero to preserve the reference interfacial dynamics. A shared path-space KL regularization further constrains the optimized dynamics from deviating excessively from the reference. Formally, taking the expectation over the input structure x, the diffusion time $t \sim \mathrm{Unif}(0, 1)$, and on-policy trajectories $X_{0:1} \sim \mathbb{P}^{\mathrm{opt}}_{|0}(\cdot \mid X_0 = x)$, the resulting objective is

$$\mathcal{L}_{\mathrm{RL}} = \mathbb{E}\left[\lambda_s \,\mathrm{softplus}\left(\frac{\lambda_m - e^{\mathrm{adv}}(t, X_t)}{\lambda_s}\right)\mathbb{I}_n(x) + \frac{1}{2\lambda_m}\left\|e^{\mathrm{adv}}(t, X_t)\right\|_2^2 \mathbb{I}_p(x)\right] + \beta\mathcal{L}_{\mathrm{PKL}}. \tag{15}$$

where $\mathbb{I}_n(x)$ and $\mathbb{I}_p(x)$ indicate whether x is a decoy or an active protein–ligand complex, respectively, λ_m specifies the target expansion margin for decoys, λ_s controls the smoothness of the expansion regularization, and β weights the path-space KL regularization term $\mathcal{L}_{\mathrm{PKL}}$.

Construction of u_t^m To evaluate e^m defined in Equation (13), we first construct the spatial velocity field u_t^m governing the tracer process in Equation (12). Specifically, we extend the atom-wise molecular vector field $b_\theta^m(t, X_t)$ to the continuous spatial domain through Gaussian kernel interpolation:

$$u_t^m(r) = \frac{1}{\sum_{i=0}^{N-1}\alpha_i(r)}\sum_{i=0}^{N-1}\alpha_i(r) b_\theta^m(t, X_t)[i], \quad \alpha_i(r) = \exp\left(-\frac{\|r - X_t[i]\|_2^2}{d^2}\right). \tag{16}$$

Here, $d > 0$ is a prescribed bandwidth. For any $d > 0$, u_t^m defines a smooth velocity field over $\mathbb{R}^3$ and, as a normalized weighted average of the atom-wise vectors, follows the local motion encoded by nearby atoms without introducing excessive velocity magnitudes.

Subsequently, Proposition 3.1 provides a closed-form expression for the relative interfacial expansion defined in Equation (14).

Proposition 3.1. *Given a diffusion time $t \in [0, 1]$ and molecular configuration X_t, let u_t^m be the spatial velocity field constructed from b_θ^m according to Equation* (16)*, for $m \in \{\mathrm{ref}, \mathrm{opt}\}$. Then, the relative interfacial expansion rate admits the following expression:*

$$e^{\mathrm{adv}}(t, X_t) = \mathbb{E}_{r\sim\gamma_t(\cdot|X_t)}\left[\nabla \cdot u_t^{\mathrm{opt}}(r) - \nabla \cdot u_t^{\mathrm{ref}}(r)\right]. \tag{17}$$

Moreover, the divergence of u_t^m can be evaluated in closed form as

$$\nabla \cdot u_t^m(r) = \frac{2}{d^2\sum_{i=0}^{N-1}\alpha_i(r)}\sum_{i=0}^{N-1}\alpha_i(r)\left\langle b_\theta^m(t, X_t)[i] - u_t^m(r), X_t[i] - r\right\rangle, \tag{18}$$

Sampling from γ_t The remaining step in evaluating e^{adv} in Equation (17) is to sample spatial points from $\gamma_t(\cdot \mid X_t)$, which we refer to as *keypoints*. In practice, we approximate these samples using the midpoints of selected protein–ligand atom pairs as representative locations along the interface. The selection criterion is specified as follows.

We first identify an *escape direction*, corresponding to the spatial direction along which the ligand experiences the least obstruction from the surrounding protein. Specifically, we generate 10^3 approximately uniformly distributed candidate directions l_i $(1 \le i \le 10^3)$ on the unit sphere using the spherical Fibonacci lattice (González, 2010), and determine the escape direction as

$$\hat{l} = \arg\min_{l_i}\sum_{j\in\mathcal{I}_p}\frac{\max\left(\cos(X_t[j] - c_l, l_i), 0\right)}{\|X_t[j] - c_l\|_2^2}, \quad c_l = \frac{1}{|\mathcal{I}_l|}\sum_{j\in\mathcal{I}_l} X_t[j]. \tag{19}$$

where $\mathcal{I}_p$ and $\mathcal{I}_l$ denotes indexes of the protein and ligand atoms, respectively. The objective measures the degree of spatial occlusion along each candidate direction: protein atoms located in front of the ligand contribute positively, with nearby atoms exerting a larger contribution, such that minimizing the objective identifies a relatively unobstructed direction away from the binding pocket.

Next, we construct a candidate set of protein–ligand atom pairs, $\mathbb{V} = \{(X_t[i], X_t[j]) \mid i \in \mathcal{I}_l,\ j \in \mathcal{I}_p,\ \|X_t[i] - X_t[j]\|_2 < 6\text{Å}\}$. Each candidate pair $(u, v) \in \mathbb{V}$ is assigned the score

$$s(u, v) = \frac{1 - \cos\left(\frac{u+v}{2} - c_l, \hat{l}\right)}{\|u - v\|_2^2}. \tag{20}$$

The score favors spatially close atom pairs whose midpoint lies opposite to the escape direction relative to the ligand center. Such pairs characterize interfacial contacts that tend to separate as the ligand moves along the escape direction, making them informative probes of escape-related interfacial expansion. Finally, we select the top-K pairs according to $s(u, v)$ and use their midpoints as representative keypoints from $\gamma_t(\cdot \mid X_t)$.

KL Regularization By Girsanov's theorem (Girsanov, 1960), the path-space KL regularization admits the following closed-form expression:

$$\mathcal{L}_{\text{PKL}} = \frac{1}{2}\mathbb{E}_{x\sim\mathcal{D}, t\sim\text{Unif}(0,1), X_{0:1}\sim\mathbb{P}^{\text{opt}}_{|0}(\cdot|x)}\left[\left\|b^{\text{opt}}_\theta(t, X_0, X_t, \mathcal{E}) - b^{\text{ref}}_\theta(t, X_0, X_t, \mathcal{E})\right\|_2^2\right]. \tag{21}$$

4 Experiment

4.1 Experimental Setup

Dataset The datasets used for training and evaluation are grouped into three categories according to their roles. More details about data preparation are provided in Section C.1.

Pretraining. To pretrain ABLE on equilibrium conformations across diverse molecular domains, we use: (i) Protein Data Bank (PDB) (Berman et al., 2000), a repository of experimentally determined biomolecular structures, and CATH (Orengo et al., 1997), a hierarchical classification of protein domains, with data splits following Jiao et al. (2026b); (ii) PDBbind2020 (Wang et al., 2004; 2005; Liu et al., 2015), which provides experimentally resolved protein–ligand complexes, using the splits adopted by DynamicBind (Lu et al., 2024); and (iii) ANI-1x (Smith et al., 2020) and PCQM4Mv2 (Hu et al., 2021), two large-scale small-molecule datasets providing diverse molecular conformations and quantum-chemical structures, respectively, with data splits following PVB (Yu et al., 2026).

Finetuning. ABLE is further finetuned on MD trajectories from two molecular domains: (i) protein monomers, using ATLAS (Vander Meersche et al., 2024) with the data split from AlphaFlow (Jing et al., 2024a); and (ii) protein–ligand complexes, using MISATO (Siebenmorgen et al., 2024) with its original data split. For the latter, we evaluate on 93 test systems due to computational constraints.

Virtual screening. To learn distinct kinetic behaviors of active ligands and decoys, we use the DUDE-Z dataset (Stein et al., 2021), an optimized version of the DUD-E benchmark (Mysinger et al., 2012) designed for virtual screening evaluation with challenging property-matched decoys across 43 targets. For each target, ligands are partitioned into data splits based on chemical similarity, with the resulting test set containing at least one active ligand and 100 decoys.

Baselines The following deep learning-based methods for biomolecular dynamics are selected as baselines: AlphaFlow (Jing et al., 2024a), BioEmu (Lewis et al., 2025), Boltz-2 (Passaro et al., 2025), MDGen (Jing et al., 2024b), UniSim (Yu et al., 2025), NeuralMD (Liu et al., 2025), ConfRover (Shen et al., 2025), and PVB (Yu et al., 2026). Among them, AlphaFlow, BioEmu, and Boltz-2 generate conformational ensembles, whereas the other methods generate time-ordered trajectories. Further details of the baseline methods are provided in Section C.2.

4.2 Trajectory Generation on Proteins

We finetune the pretrained ABLE checkpoint on ATLAS for protein trajectory generation and evaluate it on 82 test systems. For each test system, all models are required to generate 1,000 conformations,

which are evaluated against the corresponding reference MD ensemble. Detailed definitions of the evaluation metrics are given in Section C, with further qualitative analyses presented in Section D.1.

Evaluation results on the 82 ATLAS test systems are summarized in Table 1. For a controlled comparison, we distinguish between models evaluated with released pretrained checkpoints and those trained on the same ATLAS split. Among the models trained on the same ATLAS split, ABLE yields the lowest torsion JSD and remains competitive in radius of gyration (Rg), although its agreement with the reference distribution is relatively weaker in TICA space. For dynamic behavior, its weak- and transient-contact similarities are close to those of AlphaFlow, accompanied by strong RMSF correlation for residue-level flexibility despite relying on neither MSA features nor an AlphaFold-based backbone. ABLE also maintains perfect structural validity, matching the MD references and substantially exceeding AlphaFlow and MDGen on this criterion. We further report an aggregate score over the major evaluation metrics, for which ABLE ranks first among all compared models, demonstrating strong overall trajectory generation performance on protein monomers.

Table 1: Evaluation results for trajectory generation on 82 protein systems from ATLAS. The best result for each metric is shown in **bold**, and the second best is underlined. JSD: Jenson–Shannon distance; $\mathcal{J}$: Jaccard similarity; r: Pearson correlation.

Models	JSD (↓)				Contact $\mathcal{J}$ (↑)		Per-target RMSF		Validity (↑)	Score (↑)
	Rg	Torsion	TIC	TIC2D	Weak	Transient	r (↑)	RMSE (↓)		
MD (Oracle)	0.209	0.079	0.373	0.557	0.807	0.686	0.955	0.450	1.000	0.779
MD (10 ns)	0.333	0.147	0.682	0.774	0.697	0.533	0.912	0.894	1.000	0.651
Models evaluated with released pretrained checkpoints										
UniSim	0.520	0.483	**0.390**	0.693	0.152	0.023	-0.043	13.92	0.003	0.256
BioEmu	**0.429**	**0.291**	–	–	**0.499**	**0.367**	**0.389**	**1.651**	0.390	–
PVB	0.505	0.340	0.399	**0.654**	0.346	0.102	0.336	3.459	**0.976**	0.483
Models trained on the ATLAS data split										
AlphaFlow	0.496	0.503	0.559	0.739	**0.523**	0.286	**0.844**	1.084	0.013	0.421
MDGen	0.524	0.309	**0.404**	**0.626**	0.458	0.199	0.651	**1.012**	0.129	0.447
ConfRover	0.491	0.436	0.444	0.710	0.267	0.063	0.254	9.267	0.038	0.318
ABLE (*ours*)	**0.485**	**0.249**	0.551	0.737	0.521	**0.299**	0.736	1.182	**1.000**	**0.567**

4.3 Trajectory Generation on Protein–Ligand Complexes

For protein–ligand complex trajectory generation, ABLE is further finetuned on MISATO and evaluated on 93 test systems. All models are required to generate an ensemble of 100 conformations for each test system, matching the length of the reference MD trajectories in MISATO.

Evaluation results on the MISATO test set are reported in Table 2. At the protein–ligand interface, ABLE shows the strongest agreement with the reference atomic fluctuations, achieving the highest RMSF Pearson correlation and the lowest RMSE, together with competitive positional distribution matching as measured by RMWD. In terms of physical validity, ABLE produces no protein–ligand clashes and maintains a low intra-ligand clash rate comparable to the baselines. Notably, ABLE achieves the lowest trajectory RMSD (t-RMSD) error by a substantial margin over all other trajectory generation models, demonstrating more faithful conformational evolution over time. Overall, these results highlight ABLE's strong and balanced performance for protein–ligand trajectory generation.

Table 2: Evaluation results for trajectory generation on 93 protein–ligand systems from MISATO, with each metric reported as the mean and standard deviation across all test systems. The best result for each metric is shown in **bold**, and the second best is underlined. † indicates that the model is evaluated using its released weights. r: Pearson correlation; RMWD: root mean $\mathcal{W}_2$-distance.

Models	Per-target RMSF		Steric Clashes (↓)		RMWD (↓)	t-RMSD error (↓)
	r (↑)	RMSE (↓)	intra	inter		
Boltz-2†	0.410/0.240	1.563/1.636	**0.005**/0.023	**0.000**/0.000	3.765/4.237	–
NeuralMD-SDE	0.175/0.255	0.955/0.639	0.539/1.496	0.001/0.012	**1.538**/1.368	1.360/1.123
PVB	0.574/0.105	0.965/0.542	0.060/0.147	**0.000**/0.000	2.767/1.304	1.570/0.827
ABLE (*ours*)	**0.612**/0.122	**0.593**/0.551	0.110/0.229	**0.000**/0.000	2.004/1.355	**0.836**/0.982

4.4 VIRTUAL SCREENING

Finally, we evaluate ABLE on virtual screening by assessing whether the generated trajectories exhibit distinct kinetic behaviors for active ligands and decoys. Starting from the MISATO-finetuned weights, ABLE is further optimized on DUDE-Z using the on-policy RL procedure in Section 3.4 and evaluated across all 43 targets on test ligands that are chemically dissimilar to those used for training.

For each protein–ligand system, we generate a 100-frame trajectory for evaluation. The mean pocket-aligned ligand RMSD over the final 20% of frames is used as the screening score, with lower values indicating more stable binding poses. Test ligands are then ranked by this score to compute the corresponding virtual screening metrics. Additional qualitative analyses are presented in Section D.2.

The evaluation results are reported in Table 3. Among the trajectory generation models, NeuralMD, PVB and the MISATO-finetuned ABLE checkpoint show near-random AUROC and weak early enrichment, consistent with our expectation that training solely on active-complex MD trajectories does not confer effective virtual screening capability. In contrast, lightweight RL using only static cross-docked structures substantially improves all screening metrics over the reference checkpoint, yielding strong early enrichment and supporting efficient high-throughput virtual screening.

Furthermore, Figure 2 illustrates how reinforcement learning reshapes the dynamical behavior of ABLE. Compared with the reference model, the RL-optimized checkpoint induces clearer kinetic differences between active ligands and decoys while preserving stable binding poses for actives. These observations highlight the effectiveness of the proposed RL strategy in introducing ligand-discriminative kinetic behavior without requiring additional MD trajectories.

Table 3: Evaluation results for virtual screening on 43 targets from DUDE-Z, with the enrichment factor reported as the mean and standard deviation across all test systems. The best result for each metric is shown in **bold**, and the second best is underlined.

Methods	Enrichment Factor (↑)				AUROC (↑)		
	1%	2%	5%	10%	median	mean	std
NeuralMD-SDE	1.214/3.660	1.422/3.229	0.796/1.672	0.831/1.240	0.495	0.478	0.168
PVB	0.430/2.134	0.488/1.911	0.862/1.676	0.980/1.183	0.535	0.529	0.126
ABLE (ref)	0.524/1.960	0.551/1.819	0.965/1.734	1.034/1.207	0.523	0.499	0.158
ABLE (opt)	**2.807**/4.760	**2.009**/3.465	**1.754**/2.179	**1.593**/1.744	**0.553**	**0.542**	0.185

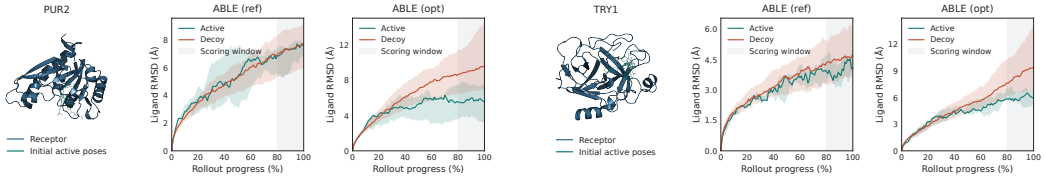


Figure 2: Comparison of dynamical behaviors between the MISATO-finetuned ABLE checkpoint (ref) and the RL-optimized checkpoint (opt) on two DUDE-Z targets, PUR2 and TRY1.

5 CONCLUSION

We introduce ABLE, a unified all-atom framework for biomolecular trajectory generation. To mitigate the positive-pair bias in existing trajectory datasets, ABLE incorporates a novel RL post-training stage using static cross-docked complexes, through which the model learns distinct dynamical behaviors for active ligands and decoys. Experiments show competitive trajectory generation performance and state-of-the-art early enrichment in virtual screening among trajectory generation baselines.

ABLE nevertheless has several limitations. First, its RL stage depends on docked starting structures and may therefore be sensitive to pose quality. Second, the current experiments focus on a limited set of biomolecular systems and fixed lag-time settings. Extending ABLE to broader molecular classes and longer timescales remains an important direction for future work.

AI USE STATEMENT

In this work, we used generative AI tools for assisting with mathematical derivations and the writing of proofs, cleaning and reformatting datasets, and assisting with translation. We have not used generative AI tools for developing theoretical models or conceptual frameworks, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, supporting qualitative and thematic data analysis, or interpreting results, and generating synthetic data sets is not applicable to this work. Additionally, we used generative AI tools for creating and editing auxiliary software code, searching for relevant literature, editing the manuscript for clarity and readability, and suggesting titles and keywords. We have reviewed all AI-assisted work. In particular, we independently verified the correctness of all AI-assisted mathematical derivations and code, and ensured that all manuscript revisions were consistent with the authors' intended claims. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human participants, personal or sensitive data, or experiments requiring institutional ethical approval. The datasets used in this study are publicly available scientific datasets, and the proposed methods are intended for computational modeling and evaluation of biomolecular dynamics. We are not aware of any specific concerns related to privacy, discrimination, security, legal compliance, conflicts of interest, or other ethical issues arising from this work.

REPRODUCIBILITY STATEMENT

We include comprehensive implementation and experimental details to facilitate reproducibility of our results. The model architecture is described in Section 3.2, and pseudocode for both training and inference is given in Section B.2. For the theoretical analysis, a complete proof of Proposition 3.1 is included in Section A. The experimental setup, including dataset construction and preprocessing, is summarized in Section 4.1 and documented in greater detail in Section C.1, while the evaluation protocols and metric definitions are specified in Section C.3. In addition, the source code is available at `https://anonymous.4open.science/r/ABLE-09F0` and contains the complete pipelines required to reproduce model training, inference, and all reported evaluations.

REFERENCES

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.

Curtis Balusek, Hyea Hwang, Chun Hon Lau, Karl Lundquist, Anthony Hazel, Anna Pavlova, Diane L Lynch, Patricia H Reggio, Yi Wang, and James C Gumbart. Accelerating membrane simulations with hydrogen mass repartitioning. *Journal of chemical theory and computation*, 15(8):4673, 2019.

Alessandro Barducci, Giovanni Bussi, and Michele Parrinello. Well-tempered metadynamics: a smoothly converging and tunable free-energy method. *Physical review letters*, 100(2):020603, 2008.

Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic acids research*, 28(1): 235–242, 2000.

Marc Bianciotto, Paraskevi Gkeka, Daria B Kokh, Rebecca C Wade, and Hervé Minoux. Contact map fingerprints of protein–ligand unbinding trajectories reveal mechanisms determining residence times computed from scaled molecular dynamics. *Journal of Chemical Theory and Computation*, 17(10):6522–6535, 2021.

Ofir Blumer, Shlomi Reuveni, and Barak Hirshberg. Combining stochastic resetting with metadynamics to speed-up molecular dynamics simulations. *Nature Communications*, 15(1):240, 2024.

Valentin De Bortoli, Guan-Horng Liu, Tianrong Chen, Evangelos A Theodorou, and Weilie Nie. Augmented bridge matching. *arXiv preprint arXiv:2311.06978*, 2023.

Ron O Dror, Albert C Pan, Daniel H Arlow, David W Borhani, Paul Maragakis, Yibing Shan, Huafeng Xu, and David E Shaw. Pathway and mechanism of drug binding to g-protein-coupled receptors. *Proceedings of the National Academy of Sciences*, 108(32):13118–13123, 2011.

Bin Feng, Jiying Zhang, Xini Zhang, Ming Zhang, Patrick Barth, Zijing Liu, and Yu Li. Physically grounded generative modeling of all-atom biomolecular dynamics. *bioRxiv*, pp. 2026–02, 2026a.

Bin Feng, Jiying Zhang, Xinni Zhang, Zijing Liu, and Yu Li. Biomd: All-atom generative model for biomolecular dynamics simulation. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 73466–73491, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/776f8da4b9e7007f92797ddfdf063131-Paper-Conference.pdf.

Shikun Feng, Minghao Li, Yinjun JIA, Wei-Ying Ma, and Yanyan Lan. Protein-ligand binding representation learning from fine-grained interactions. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 10625–10640, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/2c28efa5a86dca4b603a36c08f49f240-Paper-Conference.pdf.

Bowen Gao, Bo Qiang, Haichuan Tan, Yinjun Jia, Minsi Ren, Minsi Lu, Jingjing Liu, Wei-Ying Ma, and Yanyan Lan. Drugclip: Contrastive protein-molecule representation learning for virtual screening. *Advances in Neural Information Processing Systems*, 36:44595–44614, 2023.

Igor Vladimirovich Girsanov. On transforming a certain class of stochastic processes by absolutely continuous substitution of measures. *Theory of Probability & Its Applications*, 5(3):285–301, 1960.

Álvaro González. Measurement of areas on a sphere using fibonacci and latitude–longitude lattices. *Mathematical geosciences*, 42(1):49–64, 2010.

Hugo Guterres and Wonpil Im. Improving protein-ligand docking results with high-throughput molecular dynamics simulations. *Journal of chemical information and modeling*, 60(4):2189, 2020.

Moritz Hoffmann, Martin Scherer, Tim Hempel, Andreas Mardt, Brian de Silva, Brooke E Husic, Stefan Klus, Hao Wu, Nathan Kutz, Steven L Brunton, et al. Deeptime: a python library for machine learning dynamical models from time series data. *Machine Learning: Science and Technology*, 3(1):015009, 2022.

Weihua Hu, Matthias Fey, Hongyu Ren, Maho Nakata, Yuxiao Dong, and Jure Leskovec. OGB-LSC: A large-scale challenge for machine learning on graphs. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=qkcLxoC52kL.

Michele Invernizzi and Michele Parrinello. Exploration vs convergence speed in adaptive-bias enhanced sampling. *Journal of Chemical Theory and Computation*, 18(6):3988, 2022.

Rui Jiao, Xiangzhe Kong, Yinjun Jia, Yijia Zhang, Ziyi Yang, Yang Liu, and Jianzhu Ma. Scalable peptide design via memory-efficient equivariant transformer. *arXiv preprint arXiv:2606.25006*, 2026a.

Rui Jiao, Xiangzhe Kong, Li Zhang, Ziyang Yu, Fangyuan Ren, Wenjuan Tan, Wenbing Huang, and Yang Liu. An equivariant pretrained transformer for unified 3d molecular representation learning. *Nature Communications*, 17(1):2606, 2026b.

José Jiménez, Miha Skalic, Gerard Martinez-Rosell, and Gianni De Fabritiis. K deep: protein–ligand absolute binding affinity prediction via 3d-convolutional neural networks. *Journal of chemical information and modeling*, 58(2):287–296, 2018.

Wengong Jin, Siranush Sarkizova, Xun Chen, Nir HaCohen, and Caroline Uhler. Unsupervised protein-ligand binding energy prediction via neural euler's rotation equation. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 33514–33528. Curran Associates, Inc., 2023. doi: 10.52202/075280-1457. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/6a45a1b0697ee086bd8bf494cacc6567-Paper-Conference.pdf.

Bowen Jing, Bonnie Berger, and Tommi Jaakkola. Alphafold meets flow matching for generating protein ensembles. In *Forty-first International Conference on Machine Learning*, 2024a. URL https://openreview.net/forum?id=rs8Sh2UASt.

Bowen Jing, Hannes Stärk, Tommi Jaakkola, and Bonnie Berger. Generative modeling of molecular dynamics trajectories. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 40534–40564. Curran Associates, Inc., 2024b. doi: 10.52202/079017-1282. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/478b06f60662d3cdc1d4f15d4587173a-Paper-Conference.pdf.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/ddeebdeefdb7e7e7a697e1c3e3d8ef54-Paper.pdf.

Leon Klein, Andrew Foong, Tor Fjelde, Bruno Mlodozeniec, Marc Brockschmidt, Sebastian Nowozin, Frank Noe, and Ryota Tomioka. Timewarp: Transferable acceleration of molecular dynamics by learning time-coarsened dynamics. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 52863–52883. Curran Associates, Inc., 2023. doi: 10.52202/075280-2300. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a598c367280f9054434fdcc227ce4d38-Paper-Conference.pdf.

Xiangzhe Kong, Wenbing Huang, and Yang Liu. Generalist equivariant transformer towards 3d molecular interaction learning, 2024. URL https://openreview.net/forum?id=z3mPLBLfGY.

Alessandro Laio and Michele Parrinello. Escaping free-energy minima. *Proceedings of the national academy of sciences*, 99(20):12562–12566, 2002.

Benedict Leimkuhler and Charles Matthews. Robust and efficient configurational molecular sampling via langevin dynamics. *The Journal of chemical physics*, 138(17), 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.

Kresten Lindorff-Larsen, Stefano Piana, Ron O Dror, and David E Shaw. How fast-folding proteins fold. *Science*, 334(6055):517–520, 2011.

Shengchao Liu, Weitao Du, Hannan Xu, Yanjing Li, Zhuoxinran Li, Vignesh Bhethanabotla, Divin Yan, Christian Borgs, Anima Anandkumar, Hongyu Guo, et al. A multi-grained symmetric differential equation model for learning protein-ligand binding dynamics. *Nature Communications*, 17(1):1049, 2025.

Zhihai Liu, Yan Li, Li Han, Jie Li, Jie Liu, Zhixiong Zhao, Wei Nie, Yuchen Liu, and Renxiao Wang. Pdb-wide collection of binding data: current status of the pdbbind database. *Bioinformatics*, 31(3): 405–412, 2015.

Wei Lu, Jixian Zhang, Weifeng Huang, Ziqiao Zhang, Xiangyu Jia, Zhenyu Wang, Leilei Shi, Chengtao Li, Peter G Wolynes, and Shuangjia Zheng. Dynamicbind: predicting ligand-specific protein-ligand complex structure with a deep equivariant generative model. *Nature Communications*, 15(1):1071, 2024.

Robert T. McGibbon, Kyle A. Beauchamp, Matthew P. Harrigan, Christoph Klein, Jason M. Swails, Carlos X. Hernández, Christian R. Schwantes, Lee-Ping Wang, Thomas J. Lane, and Vijay S. Pande. Mdtraj: A modern open library for the analysis of molecular dynamics trajectories. *Biophysical Journal*, 109(8):1528 – 1532, 2015. doi: 10.1016/j.bpj.2015.08.015.

Yaosen Min, Ye Wei, Peizhuo Wang, Xiaoting Wang, Han Li, Nian Wu, Stefan Bauer, Shuxin Zheng, Yu Shi, Yingheng Wang, et al. From static to dynamic structures: Improving binding affinity prediction with graph-based deep learning. *Advanced Science*, 11(40):2405404, 2024.

Jonas Mueller, David Gifford, and Tommi Jaakkola. Sequence to better sequence: Continuous revision of combinatorial structures. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2536–2544. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/mueller17a.html.

Michael M Mysinger, Michael Carchia, John J Irwin, and Brian K Shoichet. Directory of useful decoys, enhanced (dud-e): better ligands and decoys for better benchmarking. *Journal of medicinal chemistry*, 55(14):6582, 2012.

Thin Nguyen, Hang Le, Thomas P Quinn, Tri Nguyen, Thuc Duy Le, and Svetha Venkatesh. Graphdta: predicting drug–target binding affinity with graph neural networks. *Bioinformatics*, 37 (8):1140–1147, 2021.

Christine A Orengo, Alex D Michie, Susan Jones, David T Jones, Mark B Swindells, and Janet M Thornton. Cath–a hierarchic classification of protein domain structures. *Structure*, 5(8):1093–1109, 1997.

Hakime Öztürk, Arzucan Özgür, and Elif Ozkirimli. Deepdta: deep drug–target binding affinity prediction. *Bioinformatics*, 34(17):i821–i829, 2018.

Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *BioRxiv*, 2025.

Guillermo Pérez-Hernández, Fabian Paul, Toni Giorgino, Gianni De Fabritiis, and Frank Noé. Identification of slow molecular order parameters for markov model construction. *The Journal of chemical physics*, 139(1), 2013.

Dhiman Ray and Valerio Rizzi. Enhanced sampling with suboptimal collective variables: Reconciling accuracy and convergence speed. *Journal of Chemical Theory and Computation*, 21(1):58–69, 2025.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

RA Sayle. 2d similarity, diversity and clustering in rdkit. *RDKit UGM*, 2019.

Mathias Schreiner, Ole Winther, and Simon Olsson. Implicit transfer operator learning: Multiple time-resolution models for molecular dynamics. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 36449–36462. Curran Associates, Inc., 2023. doi: 10.52202/075280-1582. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/7274ed909a312d4d869cc328ad1c5f04-Paper-Conference.pdf.

Yuning Shen, Lihao Wang, Huizhuo Yuan, Yan Wang, Bangji Yang, and Quanquan Gu. Simultaneous modeling of protein conformation and dynamics via autoregression. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, pp. 109782–109822. Curran Associates, Inc., 2025. doi: 10.52202/085713-3666. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/9e1b570c9f8df9ea884b3307e86a7810-Paper-Conference.pdf.

Liang Shi, Jiarui Lu, Junqi Liu, Chence Shi, Zhi Yang, and Jian Tang. Atomic trajectory modeling with state space models for biomolecular dynamics. *arXiv preprint arXiv:2603.17633*, 2026.

Till Siebenmorgen, Filipe Menezes, Sabrina Benassou, Erinc Merdivan, Kieran Didi, André Santos Dias Mourão, Radosław Kitel, Pietro Liò, Stefan Kesselheim, Marie Piraud, et al. Misato: machine learning dataset of protein–ligand complexes for structure-based drug discovery. *Nature computational science*, 4(5):367–378, 2024.

Justin S Smith, Roman Zubatyuk, Benjamin Nebgen, Nicholas Lubbers, Kipton Barros, Adrian E Roitberg, Olexandr Isayev, and Sergei Tretiak. The ani-1ccx and ani-1x data sets, coupled-cluster and density functional theory properties for molecules. *Scientific data*, 7(1):134, 2020.

Reed M Stein, Ying Yang, Trent E Balius, Matt J O'Meara, Jiankun Lyu, Jennifer Young, Khanh Tang, Brian K Shoichet, and John J Irwin. Property-unmatched decoys in docking benchmarks. *Journal of chemical information and modeling*, 61(2):699, 2021.

Marta M Stepniewska-Dziubinska, Piotr Zielenkiewicz, and Pawel Siedlecki. Development and evaluation of a deep learning model for protein–ligand binding affinity prediction. *Bioinformatics*, 34(21):3666–3674, 2018.

Pratyush Tiwary, Vittorio Limongelli, Matteo Salvalaglio, and Michele Parrinello. Kinetics of protein–ligand unbinding: Predicting pathways, rates, and rate-limiting steps. *Proceedings of the National Academy of Sciences*, 112(5):E386–E391, 2015.

Glenn M Torrie and John P Valleau. Nonphysical sampling distributions in monte carlo free-energy estimation: Umbrella sampling. *Journal of computational physics*, 23(2):187–199, 1977.

Oleg Trott and Arthur J Olson. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of computational chemistry*, 31(2):455–461, 2010.

Jean-François Truchon and Christopher I Bayly. Evaluating virtual screening methods: good and bad metrics for the "early recognition" problem. *Journal of chemical information and modeling*, 47(2): 488–508, 2007.

Yann Vander Meersche, Gabriel Cretin, Aria Gheeraert, Jean-Christophe Gelly, and Tatiana Galochkina. Atlas: protein flexibility description from atomistic molecular dynamics simulations. *Nucleic acids research*, 52(D1):D384–D392, 2024.

Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272, 2020.

Lingle Wang, Yujie Wu, Yuqing Deng, Byungchan Kim, Levi Pierce, Goran Krilov, Dmitry Lupyan, Shaughnessy Robinson, Markus K Dahlgren, Jeremy Greenwood, et al. Accurate and reliable prediction of relative ligand binding potency in prospective drug discovery by way of a modern free-energy calculation protocol and force field. *Journal of the American Chemical Society*, 137 (7):2695–2703, 2015.

Renxiao Wang, Xueliang Fang, Yipin Lu, and Shaomeng Wang. The pdbbind database: collection of binding affinities for protein- ligand complexes with known three-dimensional structures. *Journal of medicinal chemistry*, 47(12):2977–2980, 2004.

Renxiao Wang, Xueliang Fang, Yipin Lu, Chao-Yie Yang, and Shaomeng Wang. The pdbbind database: methodologies and updates. *Journal of medicinal chemistry*, 48(12):4111–4119, 2005.

Fang Wu and Stan Z Li. Diffmd: a geometric diffusion model for molecular dynamics simulations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 5321–5329, 2023.

Yaoyao Xu, Di Wang, Zihan Zhou, Tianshu Yu, and Mingchen Chen. Tempo: Temporal multi-scale autoregressive generation of protein conformational ensembles. *Advances in Neural Information Processing Systems*, 38:104292–104324, 2026.

Ziyang Yu, Wenbing Huang, and Yang Liu. Unisim: A unified simulator for time-coarsened dynamics of biomolecules. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=W598uW37zt.

Ziyang Yu, Wenbing Huang, and Yang Liu. Unified biomolecular trajectory generation via pretrained variational bridge. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=8HH9dBOxwu.

Chenbin Zhang, Zhiqiang Hu, Jiang Chuchu, Wen Chen, JIE XU, and Shaoting Zhang. Rethinking the generalization of drug target affinity prediction algorithms via similarity aware evaluation. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 45891–45917, 2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/7172e147d916eef4cb1eb30016ce725f-Paper-Conference.pdf.

Diya Zhang, Mengwei Sun, Xingdan Wang, Cheng Liang, Qiaozhen Meng, Shiqiang Ma, and Fei Guo. Dynapharm: Adaptive and physics-constrained modeling for target-drug complexes with drug-specific adaptations. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, pp. 1674–1707. Curran Associates, Inc., 2025b. doi: 10.52202/085713-0063. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/027af285dc29d3388002c2d223ab1772-Paper-Conference.pdf.

A Proof of Propositions

A.1 Proof of Proposition 3.1

We start from the tracer process $(R_\xi)_{\xi\geq 0}$ defined in Equation (12). Applying Itô's formula to the log-density $\log p_\xi^m(R_\xi)$ yields

$$\mathrm{d}\log p_\xi^m(R_\xi) = \left[\partial_\xi \log p_\xi^m + u_t^m \cdot \nabla \log p_\xi^m + \frac{\sigma^2}{2}\Delta \log p_\xi^m\right]\mathrm{d}\xi + \sigma(\nabla \log p_\xi^m)^\top \mathrm{d}B_\xi. \tag{22}$$

By the Fokker–Planck equation, the density p_ξ^m evolves according to

$$\partial_\xi p_\xi^m = -\nabla \cdot (u_t^m p_\xi^m) + \frac{\sigma^2}{2}\Delta p_\xi^m. \tag{23}$$

Dividing both sides by p_ξ^m gives

$$\partial_\xi \log p_\xi^m = -\nabla \cdot u_t^m - u_t^m \cdot \nabla \log p_\xi^m + \frac{\sigma^2}{2}\frac{\Delta p_\xi^m}{p_\xi^m}. \tag{24}$$

Moreover,

$$\begin{aligned}
\Delta \log p_\xi^m &= \nabla \cdot \left(\frac{\nabla p_\xi^m}{p_\xi^m}\right) \\
&= \frac{\Delta p_\xi^m}{p_\xi^m} + \nabla p_\xi^m \cdot \nabla\left(\frac{1}{p_\xi^m}\right) \\
&= \frac{\Delta p_\xi^m}{p_\xi^m} - \frac{\|\nabla p_\xi^m\|_2^2}{(p_\xi^m)^2} \\
&= \frac{\Delta p_\xi^m}{p_\xi^m} - \|\nabla \log p_\xi^m\|_2^2.
\end{aligned} \tag{25}$$

Substituting this identity into Equation (24) yields

$$\partial_\xi \log p_\xi^m = -\nabla \cdot u_t^m - u_t^m \cdot \nabla \log p_\xi^m + \frac{\sigma^2}{2}\left(\Delta \log p_\xi^m + \|\nabla \log p_\xi^m\|_2^2\right). \tag{26}$$

Substituting Equation (26) into Equation (22), we obtain

$$\mathrm{d}\log p_\xi^m(R_\xi) = \left[-\nabla\cdot u_t^m + \sigma^2\Delta\log p_\xi^m + \frac{\sigma^2}{2}\|\nabla\log p_\xi^m\|_2^2\right]\mathrm{d}\xi + \sigma(\nabla\log p_\xi^m)^\top\,\mathrm{d}B_\xi. \tag{27}$$

Taking expectations over $R_\xi \sim p_\xi^m$ and using the fact that the Itô stochastic integral has zero expectation under the standard integrability conditions, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}\xi}\mathbb{E}_{R_\xi\sim p_\xi^m}\left[\log p_\xi^m(R_\xi)\right] = \mathbb{E}_{R_\xi\sim p_\xi^m}\left[-\nabla\cdot u_t^m + \sigma^2\Delta\log p_\xi^m + \frac{\sigma^2}{2}\|\nabla\log p_\xi^m\|_2^2\right]. \tag{28}$$

Therefore, the interfacial expansion rate defined in Equation (13) admits the following closed-form expression:

$$\begin{aligned} e^m(t,X_t) &= -\frac{\mathrm{d}}{\mathrm{d}\xi}\mathbb{E}_{R_\xi\sim p_\xi^m}\left[\log p_\xi^m(R_\xi)\right]\Big|_{\xi=0} \\ &= \mathbb{E}_{r\sim\gamma_t(\cdot|X_t)}\left[\nabla\cdot u_t^m - \sigma^2\Delta\log\gamma_t - \frac{\sigma^2}{2}\|\nabla\log\gamma_t\|_2^2\right]. \end{aligned} \tag{29}$$

Since $\gamma_t(\cdot \mid X_t)$ is shared by the optimized and reference dynamics for a given X_t, all terms depending solely on γ_t cancel when taking their difference. Consequently,

$$e^{\mathrm{adv}}(t,X_t) = e^{\mathrm{opt}}(t,X_t) - e^{\mathrm{ref}}(t,X_t) = \mathbb{E}_{r\sim\gamma_t(\cdot|X_t)}\left[\nabla\cdot u_t^{\mathrm{opt}}(r) - \nabla\cdot u_t^{\mathrm{ref}}(r)\right], \tag{30}$$

which completes the proof of the first part.

Next, we derive a closed-form expression for the divergence of the velocity field u_t^m defined in Equation (16). Let

$$\alpha_i(r) = \exp\left(-\frac{\|r - X_t[i]\|_2^2}{d^2}\right), \qquad Z(r) = \sum_{i=0}^{N-1}\alpha_i(r). \tag{31}$$

Then, the spatial velocity field can be written as

$$u_t^m(r) = \frac{1}{Z(r)}\sum_{i=0}^{N-1}\alpha_i(r)b_\theta^m(t,X_t)[i]. \tag{32}$$

Since $b_\theta^m(t,X_t)[i]$ is independent of r, the gradient of the Gaussian weight is given by

$$\nabla\alpha_i(r) = \frac{2}{d^2}\alpha_i(r)\left(X_t[i] - r\right). \tag{33}$$

Applying the quotient rule to Equation (32), we obtain

$$\begin{aligned} \nabla\cdot u_t^m(r) &= \nabla\cdot\left(\frac{\sum_{i=0}^{N-1}\alpha_i(r)b_\theta^m(t,X_t)[i]}{Z(r)}\right) \\ &= \frac{1}{Z(r)}\sum_{i=0}^{N-1} b_\theta^m(t,X_t)[i]^\top\nabla\alpha_i(r) - \frac{\sum_{i=0}^{N-1}\alpha_i(r)b_\theta^m(t,X_t)[i]}{Z(r)^2}^\top\nabla Z(r) \\ &= \frac{1}{Z(r)}\sum_{i=0}^{N-1} b_\theta^m(t,X_t)[i]^\top\nabla\alpha_i(r) - \frac{u_t^m(r)^\top}{Z(r)}\sum_{i=0}^{N-1}\nabla\alpha_i(r) \\ &= \frac{1}{Z(r)}\sum_{i=0}^{N-1}\nabla\alpha_i(r)^\top\left(b_\theta^m(t,X_t)[i] - u_t^m(r)\right). \end{aligned} \tag{34}$$

Substituting Equation (33) into Equation (34), we obtain

$$\nabla\cdot u_t^m(r) = \frac{2}{d^2\sum_{i=0}^{N-1}\alpha_i(r)}\sum_{i=0}^{N-1}\alpha_i(r)\left\langle b_\theta^m(t,X_t)[i] - u_t^m(r), X_t[i] - r\right\rangle. \tag{35}$$

Then we complete the proof. □

B TRAINING DETAILS

B.1 DATA AUGMENTATION

To improve robustness during autoregressive rollout, we employ two data augmentation strategies at different training stages (Yu et al., 2026).

First, during pretraining and MD finetuning, training inputs are sampled from valid molecular conformations, whereas inference recursively feeds generated configurations back into the model. This train–inference mismatch can cause prediction errors to accumulate over successive transitions, gradually shifting the inputs away from the training distribution. To mitigate this effect, we independently add isotropic Gaussian noise with a standard deviation of 0.2 Å to the Cartesian coordinates of x_0 during training. This exposes the model to mildly perturbed configurations and improves its robustness to rollout errors.

Second, during RL finetuning, each protein–ligand system is represented by a single docking pose in the cross-docked dataset, resulting in substantially fewer input configurations than in trajectory-based training. To mitigate overfitting to these fixed poses, we perturb the ligand pose with a random rigid transformation at each training iteration while keeping the protein coordinates unchanged. Specifically, we sample a rotation angle $\theta \sim \mathrm{Unif}(-\pi/3, \pi/3)$ together with a unit rotation axis, and construct the corresponding rotation matrix $\boldsymbol{Q} \in \mathrm{SO}(3)$ using Rodrigues' rotation formula. We further sample a translation vector $\boldsymbol{t} \sim \mathcal{N}(\mathbf{0}, \frac{1}{3}\boldsymbol{I})$, corresponding to a root-mean-square translation magnitude of 1 Å. The resulting transformation $(\boldsymbol{Q}, \boldsymbol{t})$ is applied only to the ligand coordinates in x_0, thereby augmenting the initial binding pose while preserving the protein structure. The bounded rotation angle and moderate translation scale increase pose diversity while reducing the likelihood of introducing severe steric clashes with the protein.

B.2 ALGORITHMS

To facilitate precise implementation, we provide pseudocode for trajectory generation and on-policy RL finetuning in Algorithm 1 and Algorithm 2, respectively.

Algorithm 1 Trajectory Generation with ABLE

Require: Input conformation $\{z, x_0, \mathcal{E}\}$, number of SDE steps T, trajectory length L
Ensure: Generated trajectory $\mathcal{T} = (x_0, x_1, \ldots, x_{L-1})$

function SAMPLE(x, T)
  Sample latent representation $(\hat{h}, \hat{x})$ from $\varphi_e(\mathrm{embed}(z), x, \mathcal{E})$
  $X^{(0)} \leftarrow \hat{x}$, $\Delta t \leftarrow 1/T$
  for $k = 0, \ldots, T-1$ **do**
    $t_k \leftarrow k/T$
    $b_\theta^{(k)} \leftarrow b_\theta(t_k, X^{(0)}, X^{(k)}, \mathcal{E})$
    if $k < T-1$ **then**
      Sample $\epsilon_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
      $X^{(k+1)} \leftarrow X^{(k)} + b_\theta^{(k)} \Delta t + \sigma\sqrt{\Delta t}\,\epsilon_k$
    else
      $X^{(k+1)} \leftarrow X^{(k)} + b_\theta^{(k)} \Delta t$
    end if
  end for
  return $X^{(T)}$
end function
function SAMPLETRAJ(x_0, T, L)
  $\mathcal{T} \leftarrow [x_0]$
  $x \leftarrow x_0$
  for $\ell = 1, \ldots, L-1$ **do**
    $x \leftarrow$ SAMPLE(x, T)
    Append x to $\mathcal{T}$
  end for
  return $\mathcal{T}$
end function
function REALIZATION(x, T)
  Sample latent representation $(\hat{h}, \hat{x})$ from $\varphi_e(\mathrm{embed}(z), x, \mathcal{E})$
  $X^{(0)} \leftarrow \hat{x}$, $\Delta t \leftarrow 1/T$
  $\mathcal{X} \leftarrow [X^{(0)}]$
  for $k = 0, \ldots, T-1$ **do**
    $t_k \leftarrow k/T$
    $b_\theta^{(k)} \leftarrow b_\theta\left(t_k, X^{(0)}, X^{(k)}, \mathcal{E}\right)$
    if $k < T-1$ **then**
      Sample $\epsilon_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
      $X^{(k+1)} \leftarrow X^{(k)} + b_\theta^{(k)} \Delta t + \sigma\sqrt{\Delta t}\,\epsilon_k$
    else
      $X^{(k+1)} \leftarrow X^{(k)} + b_\theta^{(k)} \Delta t$
    end if
    Append $X^{(k+1)}$ to $\mathcal{X}$
  end for
  return $\mathcal{X}, \{k/T\}_{k=0}^{T}$
end function

Algorithm 2 On-Policy RL Finetuning of ABLE for Ligand Enrichment

Require: Mini-batch of protein–ligand complexes $\{z, x_0, \mathcal{E}\}$, MD-finetuned model parameters θ, number of SDE steps T, number of interfacial keypoints K, bandwidth d, relative expansion margin λ_m, expansion scale λ_s, KL weight β
Ensure: RL-finetuned model parameters θ'

Freeze the reference model parameters θ
Initialize $\theta' \leftarrow \theta$
Freeze the encoder parameters in θ'
for each training iteration **do**
  $x_0 \leftarrow \textsc{DataAugmentation}(x_0)$ {Section B.1}
  $\mathcal{X}, \{t_k\}_{k=0}^{T} \leftarrow \textsc{Realization}(x_0, T)$ {On-policy sampling with θ'}
  Sample $j \sim \mathrm{Unif}\{0, \ldots, T\}$
  $x_t \leftarrow X^{(j)}$, $t \leftarrow t_j$
  $\Gamma_t \leftarrow \textsc{InteractionKeypoints}(x_t, K)$
  $b_\theta \leftarrow b_\theta(t, x_0, x_t, \mathcal{E})$
  $b_{\theta'} \leftarrow b_{\theta'}(t, x_0, x_t, \mathcal{E})$
  $e_\theta \leftarrow \textsc{ExpansionRate}(x_t, b_\theta, \Gamma_t, d)$ {Proposition 3.1}
  $e_{\theta'} \leftarrow \textsc{ExpansionRate}(x_t, b_{\theta'}, \Gamma_t, d)$
  $e^{\mathrm{adv}} \leftarrow e_{\theta'} - e_\theta$
  for each complex i in the mini-batch **do**
    $\mathcal{D}_i \leftarrow \dfrac{1}{N_i} \sum_{n=1}^{N_i} \|b_{\theta',i,n} - b_{\theta,i,n}\|_2^2$
    if complex i is a decoy **then**
      $\mathcal{L}_i \leftarrow \beta \mathcal{D}_i + \dfrac{\lambda_s}{K} \sum_{k=1}^{K} \mathrm{softplus}\left(\dfrac{\lambda_m - e_{i,k}^{\mathrm{adv}}}{\lambda_s}\right)$
    else
      $\mathcal{L}_i \leftarrow \beta \mathcal{D}_i + \dfrac{1}{2\lambda_m} \left\|e_i^{\mathrm{adv}}\right\|_2^2$
    end if
  end for
  $\mathcal{L}_{\mathrm{RL}} \leftarrow \dfrac{1}{B} \sum_{i=1}^{B} \mathcal{L}_i$
  $\theta' \leftarrow \textsc{Optimizer}(\theta', \mathcal{L}_{\mathrm{RL}})$
end for
return θ'

B.3 Hyperparameters

The hyperparameter settings used for ABLE in our experiments are summarized in Table 4.

C Experimental Details

C.1 Data Preparation

ATLAS ATLAS (Vander Meersche et al., 2024) is a large-scale dataset comprising 100 ns explicit-solvent MD trajectories for thousands of protein monomers, with three independent replicates provided for each system. Following AlphaFlow (Jing et al., 2024a), we use the split of 1,266/39/82 protein systems for training, validation, and testing, respectively. For each training and validation system, we randomly sample 500 conformation pairs (x_0, x_1) from each of the three MD replicates using a lag time of $\tau = 100$ ps, which are used for model training.

MISATO MISATO (Siebenmorgen et al., 2024) comprises 8 ns explicit-solvent MD trajectories for over 10,000 protein–ligand complexes. We use the original split of 13,765/1,595/1,612 systems for training, validation, and testing, respectively. For each training and validation system, all 100 conformation pairs (x_0, x_1) separated by a lag time of $\tau = 80$ ps are used for model training and validation. Due to computational constraints, we further select 93 protein–ligand systems from the original test set for evaluation. The corresponding PDB identities are listed below.

Table 4: Hyperparameter settings for ABLE.

Category	Hyperparameter	Value
Model Architecture	Hidden dimension H	256
	Latent dimension D	32
	Number of attention heads	8
	Number of encoder layers	6
	Numder of conditioner layers	3
	Number of decoder layers	6
Pretraining	Coordinate noise scale σ_e	$\sqrt{0.3}$ Å
	SDE noise scale σ	0.2 Å
	KL weight for scalar features w_h	0.05
	KL weight for coordinates w_x	0.5
	KL threshold ε	0.3
MD finetuning	Annealing steps S	10,000
RL finetuning	Spatial bandwidth d	3.0 Å
	Expansion scale λ_s	0.01
	Relative expansion margin λ_m	0.02
	Number of interfacial keypoints K	16
	Path-space KL weight β	0.2
Optimization	Learning rate	[1e-4,5e-5]
	Optimizer	Adam
	Warmup steps	1,000
	Warmup scheduler	LambdaLR
	Training Scheduler	ReduceLROnPlateau
Inference	Number of SDE steps	10

5DU4	4UU5	5I38	4I7D	3TZ0	4NCM
5N2F	4JHT	3T70	2VQT	2WW0	2QPQ
3T4V	6NDL	6AJH	6CFD	1AJQ	6AAD
3BXF	1S89	6AX1	6SEN	1YYR	2WZS
5AI1	4R3W	3WDE	4NRQ	3MET	7STD
5NGZ	5KBE	3RV8	5NP8	5YH8	5L8A
6RNT	2VMF	6CWI	6QAT	5V3X	6HLZ
4EUO	4DFU	3TZ2	4JG8	5O9O	4G0P
5LSY	4NRU	5IOK	2VNP	5WL0	1H46
4D1C	2AMT	2FJ0	4DU8	1RSD	6BBX
6M88	5NDF	5O3Q	3TIF	6H1I	4Y24
4BTX	5HGC	4GS9	3FFU	2WJI	2ZA3
6M89	3ZOT	3W54	1JN2	6O5U	4AYR
6HKE	3OQ5	5KAU	6CUS	3CZ1	4NRT
1B74	5E2L	3T3Y	3N2E	1RSI	5C0M
2WPB	6BHH	1E3V			

DUDE-Z DUDE-Z (Stein et al., 2021) is a virtual screening benchmark comprising 43 protein targets, 2,312 active ligands, and 69,994 property-matched decoys. For each target, active ligands and decoys are docked to the corresponding receptor using DOCK3.7.2 (Stein et al., 2021), and the resulting poses serve as the initial states in our experiments. To construct chemically separated data splits, active ligands and decoys are clustered independently according to ECFP4-based Tanimoto similarity (Rogers & Hahn, 2010), using a threshold of 0.65 (Sayle, 2019). The resulting clusters are assigned to the training, validation, and test sets with an 80%/10%/10% ratio, preventing highly similar molecules within the same class from appearing across different splits. All training and validation molecules are retained, while test decoys are randomly subsampled to at most 100 per target. After preprocessing, each test target contains at least one active ligand and 100 decoys.

C.2 BASELINES

We compare ABLE against the following deep learning-based approaches for biomolecular dynamics: (i) AlphaFlow (Jing et al., 2024a), which repurposes AlphaFold as a flow-matching generative model for sampling protein conformational ensembles. Following the official recommendation, we use the `alphaflow_md_distilled_202402.pt` checkpoint to generate 1,000 conformations per system. (ii) BioEmu (Lewis et al., 2025), a generative model built upon pretrained AlphaFold2 representations for efficiently sampling protein equilibrium conformational ensembles. (iii) Boltz-2 (Passaro et al., 2025), a strong biomolecular foundation model for co-folding and binding affinity prediction. (iv) MDGen (Jing et al., 2024b), a flow-based generative model that formulates molecular dynamics as conditional trajectory generation and supports multiple tasks, including forward simulation, interpolation, and temporal upsampling. We generate a 1,000-frame trajectory through five consecutive 200-frame rollouts, with each rollout initialized from the final frame of the preceding one. (v) UniSim (Yu et al., 2025), a unified time-coarsened dynamics simulator across diverse molecular domains with physics-based guidance. (vi) NeuralMD (Liu et al., 2025), a semi-flexible machine learning surrogate for protein–ligand molecular dynamics, where the protein coordinates remain fixed and only the ligand trajectory is propagated. (vii) ConfRover (Shen et al., 2025), an autoregressive framework that unifies protein ensemble sampling and trajectory generation by modeling temporal dependencies across conformational frames. For computational efficiency, we generate 1,000-frame trajectories in consecutive 11-frame segments, restarting each segment from the final frame of the preceding one, rather than retaining the full trajectory history in the KV cache. (viii) PVB (Yu et al., 2026), a variational bridge framework that unifies training on equilibrium structures and paired MD trajectory data, with the first demonstration of MD-level physical validity in trajectory generation.

C.3 METRICS

Binding Pocket Following AlphaFold3 (Abramson et al., 2024), we define the binding pocket based on its spatial proximity to the ligand. Specifically, a protein residue is considered part of the binding pocket if any of its heavy atoms lies within 10 Å of any ligand heavy atom. All atoms belonging to the selected residues are then included as pocket atoms, while their Cα atoms are used for pocket-based structural alignment.

Time-lagged Independent Component Trajectories of proteins in ATLAS are featurized using time-lagged independent component analysis (TICA) (Pérez-Hernández et al., 2013). The collective variables consist of the sine and cosine of the backbone ϕ and ψ torsion angles and all side-chain χ torsion angles. For each protein, the TICA model is constructed from the first MD replicate, denoted by the suffix `R1`, using `deeptime` (Hoffmann et al., 2022) with a lag time of 100 ps. The generated trajectories and reference trajectories are then projected onto the two slowest time-lagged independent components, denoted by TIC0 and TIC1. While the TICA model is fitted using only the first replicate, the reference ensemble used for evaluation is formed by pooling all three MD replicates.

Jensen–Shannon Distance The Jensen–Shannon distance (JSD) is used to quantify the distributional similarity between generated and reference ensembles. In our experiments, we compute the JSD for the radius of gyration (Rg), backbone ϕ and ψ torsion angles (Torsion), the two slowest time-lagged components (TIC), and their joint two-dimensional distribution (TIC2D). For one-dimensional features, we divide the range between the minimum and maximum values of the reference features into 49 equally spaced bins, construct histograms for the generated and reference samples, and compute their JSD using `SciPy` (Virtanen et al., 2020). For multi-dimensional features, the JSD is computed independently along each dimension and then averaged across dimensions. As an exception, for the TIC2D representation, we construct a joint two-dimensional histogram over the slowest two TICs using 49×49 bins and compute the JSD directly between the resulting joint distributions.

Jaccard Similarity Following AlphaFlow (Jing et al., 2024a), we use the weak and transient contact Jaccard similarities (denoted by $\mathcal{J}$) to evaluate the agreement of dynamic contact patterns between generated and reference trajectories. A pair of Cα atoms is considered to be in contact when their distance is below 8 Å. A weak contact is defined as a pair that is in contact in the reference structure but becomes separated in more than 10% of the ensemble structures, whereas a transient contact is a pair that is not in contact in the reference structure but forms a contact in more than 10%

of the ensemble structures. For each contact type, we construct the corresponding contact sets for the generated and reference trajectories and compute their Jaccard similarity.

Per-target RMSF To evaluate whether the generated trajectories reproduce local structural flexibility, we compute the Root Mean Square Fluctuation (RMSF) profiles of selected atoms for both the generated and reference trajectories using `mdtraj` (McGibbon et al., 2015). For ATLAS protein monomers, RMSF is evaluated over Cα atoms, whereas for MISATO protein–ligand complexes, it is evaluated over all heavy atoms in the binding pocket and ligand. For each system, we report the Pearson correlation coefficient r between the generated and reference RMSF profiles, together with their Root Mean Square Error (RMSE) in Å.

Structural Validity To assess the physical validity of generated protein trajectories, we compute the fraction of valid conformations in each trajectory. Following PVB (Yu et al., 2026), a conformation is considered valid if it satisfies both of the following criteria: (i) the Euclidean distance between the Cα atoms of any two distinct residues is at least 3.0 Å, preventing severe steric clashes; and (ii) the Euclidean distance between the Cα atoms of adjacent residues within the same protein chain does not exceed 4.19 Å, preventing unrealistic backbone discontinuities.

Aggregate Score To summarize model performance across complementary evaluation criteria, we report an aggregate score with equal weighting across eight metrics: the JSDs of Rg, Torsion, TIC, and TIC2D, the weak- and transient-contact Jaccard similarities, the RMSF Pearson correlation, and structural validity. For the four JSD metrics, where lower values indicate better performance, we transform each metric by taking one minus its value, while the remaining four metrics are used directly. The eight resulting values are then averaged, with a higher score indicating better overall performance across distributional similarity, structural flexibility, and physical validity.

Steric Clashes To assess the physical validity of generated protein–ligand trajectories, we measure steric clashes between heavy atoms. Following AlphaFold3 (Abramson et al., 2024), a steric clash is identified when the distance between two heavy atoms is less than 1.1 Å. We report the average number of steric clashes per trajectory frame, where *intra* denotes clashes between ligand heavy atoms and *inter* denotes clashes between pocket and ligand heavy atoms.

Root Mean $\mathcal{W}_2$-Distance To quantify the distributional similarity between generated and reference protein–ligand trajectories, we compute the Root Mean $\mathcal{W}_2$-Distance (RMWD) (Shi et al., 2026). Both trajectories are first aligned to the reference structure using the Cα atoms of the binding pocket. For each atom i in the binding pocket and ligand, we approximate its three-dimensional positional distributions in the reference and generated trajectories by Gaussian distributions and compute their 2-Wasserstein distance, denoted by $\mathcal{W}_{2,i}$. The RMWD is then defined as

$$\mathrm{RMWD} = \left(\frac{1}{N_{\mathrm{eval}}} \sum_{i=1}^{N_{\mathrm{eval}}} \mathcal{W}_{2,i}^2 \right)^{1/2}, \tag{36}$$

where N_{eval} denotes the total number of pocket and ligand heavy atoms. A lower RMWD indicates closer agreement between the generated and reference positional distributions around the protein–ligand interface.

t-RMSD Error Following TEMPO (Xu et al., 2026), we report the trajectory RMSD (t-RMSD) error to evaluate the agreement of conformational evolution between the generated and reference trajectories. For each trajectory, we compute the RMSD of every frame relative to its respective initial frame over the pocket and ligand atoms using `mdtraj` (McGibbon et al., 2015), and report the root mean square error between the resulting RMSD sequences.

AUROC Given a ranked list of active ligands and decoys, we use the Area Under the Receiver Operating Characteristic Curve (AUROC) to evaluate the overall ranking performance. For a target containing N_a active ligands and N_d decoys, let r_i denote the rank of active ligand i, with tied candidates assigned their average rank. The AUROC is computed as

$$\mathrm{AUROC} = \frac{N_a N_d + N_a(N_a+1)/2 - \sum_{i=1}^{N_a} r_i}{N_a N_d}. \tag{37}$$

The AUROC can be interpreted as the probability that a randomly selected active ligand is ranked ahead of a randomly selected decoy, with a larger value indicating better overall discrimination.

Enrichment Factor To evaluate the early enrichment of active ligands, we further report the Enrichment Factor (EF) (Truchon & Bayly, 2007). The enrichment factor at the top $x\%$ of the ranked list is defined as

$$\mathrm{EF}_{x\%} = \frac{n_a^{x\%}/K_{x\%}}{N_a/(N_a+N_d)}, \tag{38}$$

where $K_{x\%} = \max(1, \lceil x(N_a+N_d)/100 \rceil)$ denotes the number of candidates considered in the top $x\%$, and $n_a^{x\%}$ denotes the number of active ligands among them. The EF quantifies the fold enrichment of active ligands in the top-ranked subset relative to their overall prevalence in the candidate pool, with $\mathrm{EF}_{x\%} = 1$ corresponding to random selection and larger values indicating stronger early enrichment. We report $\mathrm{EF}_{1\%}$, $\mathrm{EF}_{2\%}$, $\mathrm{EF}_{5\%}$, and $\mathrm{EF}_{10\%}$ in our experiments.

D Additional Experimental Results

D.1 Qualitative Analysis of Trajectory Generation

To provide a more intuitive assessment of ABLE's trajectory generation performance, we visualize representative systems from ATLAS and MISATO in Figure 3 and Figure 4, respectively.

D.2 Virtual Screening Analysis

We select representative DUDE-Z targets for a case study of the virtual screening performance of the RL-finetuned ABLE checkpoint, with the results shown in Figure 5. Across these targets, active ligands and decoys exhibit distinct kinetic behaviors: actives generally maintain more stable binding poses, as reflected by lower ligand RMSD, and preserve a larger fraction of their initial protein–ligand contacts. These differences translate into effective active–decoy discrimination, yielding strong AUROC and early enrichment in virtual screening.

Additionally, for some targets (*e.g.*, ROCK1), the generated trajectories exhibit substantially increased ligand RMSD during rollout. This suggests that RL finetuning can, in certain cases, induce excessive deviation from the dynamics captured by the reference model, even when active–decoy discrimination remains effective. These results indicate that the RL hyperparameters should be tuned carefully to balance screening performance with the physical plausibility of the generated trajectories.

D.3 Ablation Study

To assess the sensitivity of the reinforcement learning hyperparameters, we conduct ablation studies on the path-space KL regularization weight β, bandwidth d, and number of keypoints K for virtual screening on DUDE-Z. The results are summarized in Table 5, from which we draw the following observations.

Regularization weight β Across all tested values of β, RL finetuning consistently improves virtual screening over the reference checkpoint, indicating that the proposed optimization remains effective within a reasonable range of regularization strengths. At $\beta = 0.3$, ABLE attains the highest AUROC but comparatively weaker early enrichment, suggesting better overall ranking of active ligands without concentrating them as strongly at the top of the ranked list. In contrast, $\beta = 0.2$ and $\beta = 0.4$ produce stronger early enrichment while maintaining competitive AUROC. The variation across settings does not exhibit a systematic trend and is likely influenced in part by the stochastic nature of reinforcement learning optimization.

Bandwidth d ABLE maintains competitive screening performance across all tested bandwidths $d \in \{2.0, 3.0, 4.0, 5.0\}$, with consistently favorable early enrichment and only modest variation in AUROC. No systematic trend emerges as d changes, indicating robust performance across the tested range.

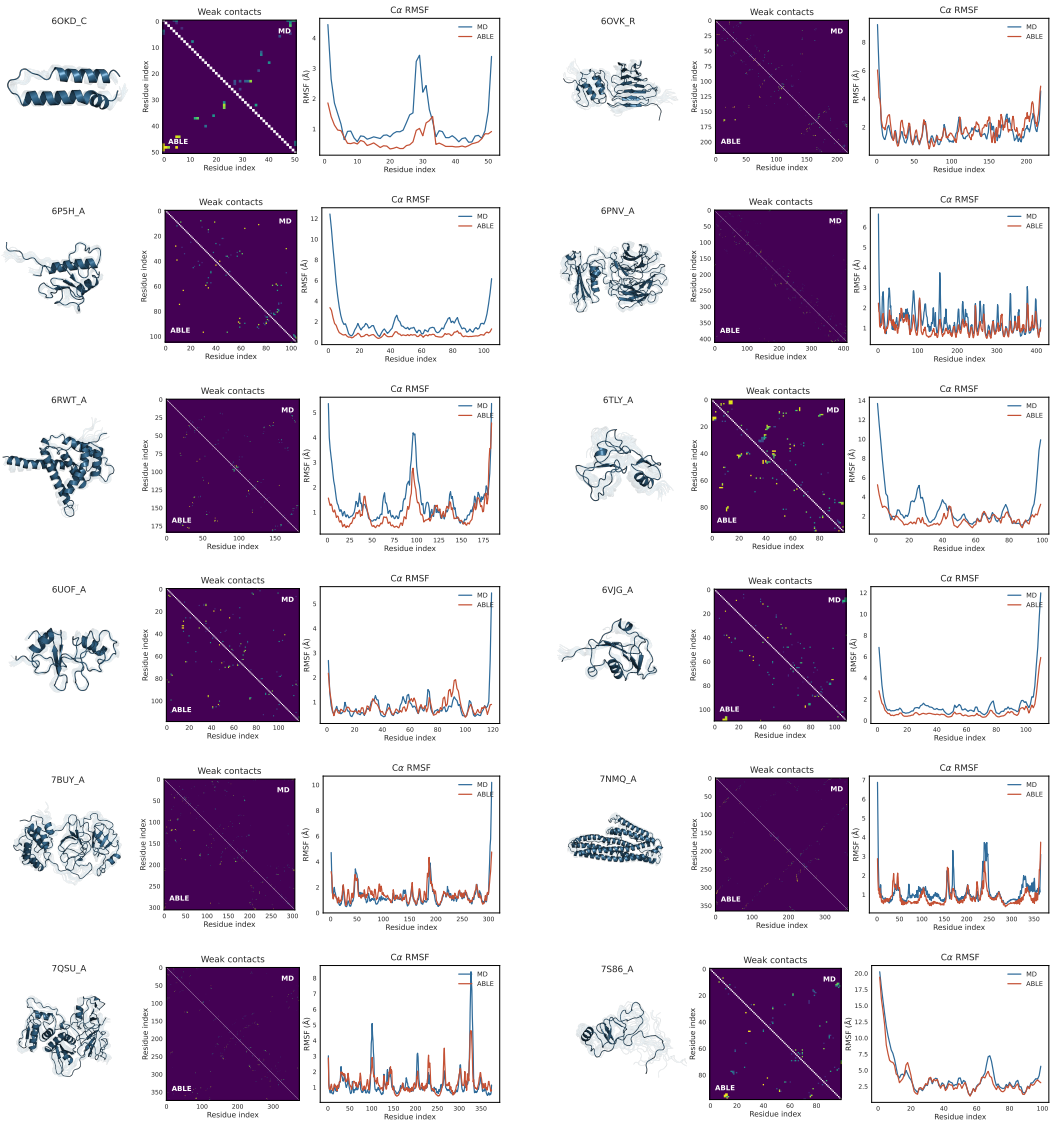


Figure 3: Trajectory generation analysis of 12 protein systems from the ATLAS test set. For each system, we show the generated protein ensemble structures (left), the corresponding weak-contact map (middle), and the Cα RMSF profiles of the reference MD trajectory and ABLE (right).

Number of keypoints K Increasing the number of keypoints from $K = 16$ to $K = 32$ leads to a noticeable degradation in screening performance. This trend may be attributed to the keypoint selection procedure, which ranks candidate protein–ligand atom pairs according to a prescribed scoring criterion that is not perfectly reliable. As K increases, lower-confidence pairs that may not accurately represent the protein–ligand interface are more likely to be included, potentially introducing noisy optimization signals and steering the RL update toward less desirable directions. These results suggest that a moderate number of high-confidence keypoints is preferable to using a larger, noisier set.

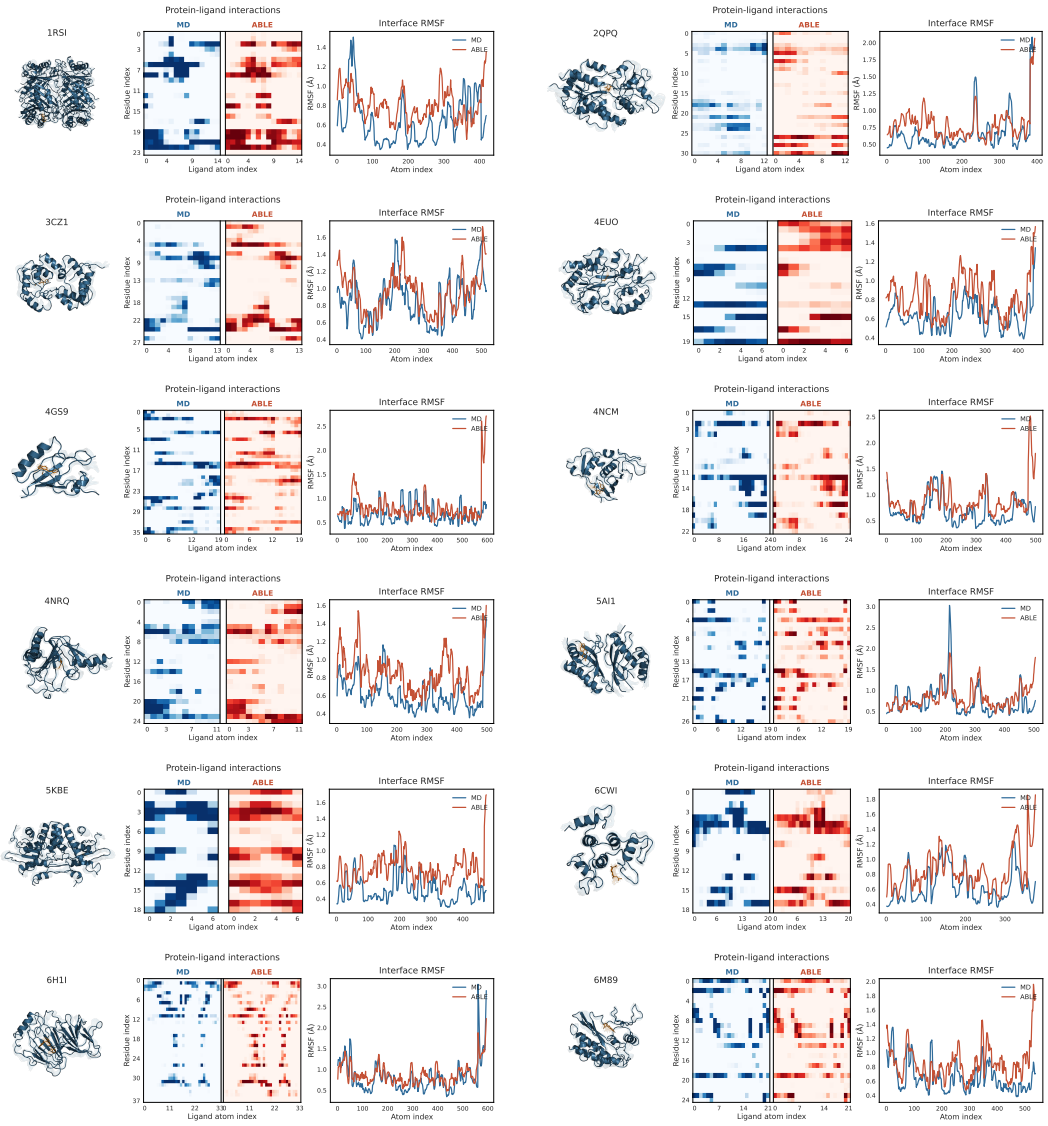


Figure 4: Trajectory generation analysis of 12 protein–ligand systems from the MISATO test set. For each system, we show the generated complex ensemble structures (left), the protein–ligand interaction map (middle), and the RMSF profiles of the protein–ligand interface atoms from the reference MD trajectory and ABLE (right).

Table 5: Ablation studies of the path-space KL regularization weight β, bandwidth d, and number of keypoints K on the 43 DUDE-Z targets. The best result for each metric is shown in **bold**, and the second best is underlined.

Hyperparameters			Enrichment Factor (↑)				AUROC (↑)		
β	d	K	1%	2%	5%	10%	median	mean	std
0.2	3.0	16	**2.807**/4.760	2.009/3.465	1.754/2.179	**1.593**/1.744	0.553	0.542	0.185
0.3	3.0	16	1.393/4.005	1.476/3.269	1.258/1.904	1.149/1.168	**0.580**	**0.547**	0.167
0.4	3.0	16	2.231/4.759	**2.420**/4.094	1.712/2.280	1.491/1.424	0.543	0.536	0.169
0.2	2.0	16	2.713/5.005	2.357/3.518	1.521/2.111	1.188/1.424	0.540	0.528	0.196
0.2	4.0	16	2.182/4.434	2.312/4.320	1.448/2.164	1.299/1.405	0.525	0.538	0.145
0.2	5.0	16	2.136/5.029	2.257/3.855	**1.846**/2.665	1.574/1.971	0.543	0.536	0.196
0.2	3.0	32	1.931/3.971	1.450/2.778	1.337/1.838	1.313/1.553	0.543	0.541	0.162

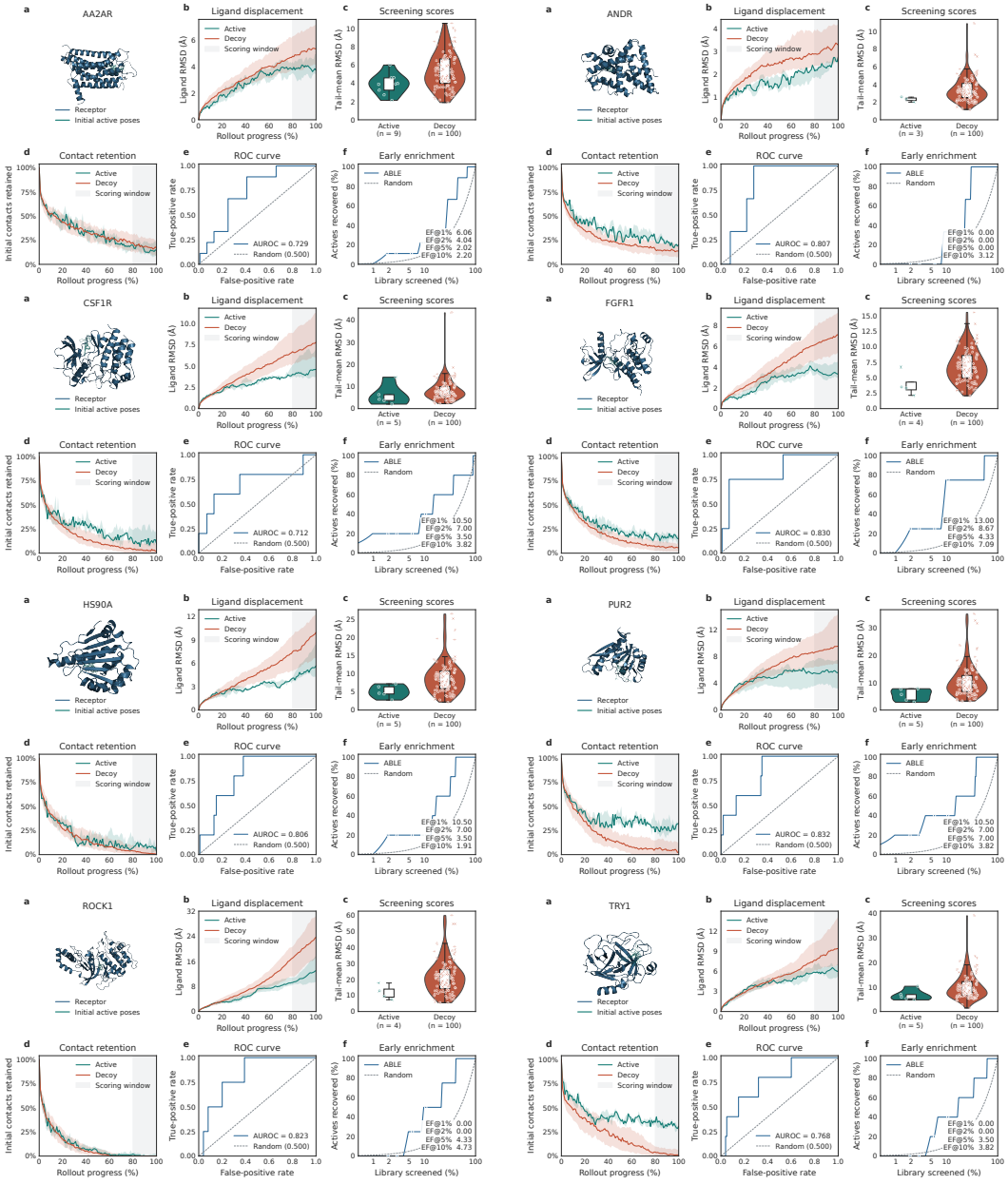
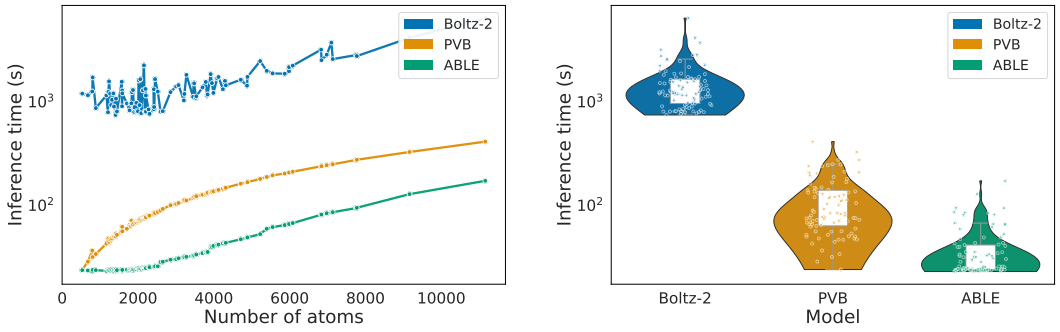


Figure 5: Virtual screening analysis of representative targets from DUDE-Z. For each target, we show **(a)** the receptor structure with the initial poses of active ligands; **(b)** ligand RMSD along the generated trajectories after pocket alignment, with the final 20% of frames highlighted as the screening window; **(c)** the distributions of screening scores for active ligands and decoys, where the score is defined as the mean ligand RMSD within the screening window; **(d)** the retention of initial protein–ligand heavy-atom contacts along the trajectories, where contacts are defined by an interatomic distance below 4.5 Å; **(e)** the ROC curve for active–decoy discrimination; and **(f)** the early enrichment curve showing the fraction of actives recovered as a function of the fraction of the library screened.

D.4 Inference Efficiency

To evaluate inference efficiency, we compare the runtime of different models on the 93 MISATO test systems, with each model generating 100 conformations per system, as shown in Figure 6. As illustrated in Figure 6a, ABLE exhibits approximately linear scaling with system size and requires only on the order of 10^2 seconds even for systems containing more than 10,000 atoms, benefiting

from its memory-efficient attention implementation. Across the full test set, Figure 6b shows that ABLE consistently achieves the lowest inference time with relatively small variation. The median runtime of ABLE is roughly 2–3$\times$ lower than PVB and more than an order of magnitude lower than Boltz-2. These results demonstrate the favorable scalability and efficiency of ABLE, supporting its potential for high-throughput trajectory generation.



(a) Inference time as a function of system size.

(b) Distribution of inference time across test systems.

Figure 6: Inference-time comparison on the 93 MISATO test systems for generating 100 frames per system.

E Computational Infrastructure

The training and inference of ABLE were conducted using up to eight NVIDIA GeForce RTX 3090 GPUs. The complete training pipeline required approximately 5,160 GPU hours, including about 240 GPU hours for pretraining, 2,400 GPU hours each for finetuning on ATLAS and MISATO, and 120 GPU hours for on-policy RL finetuning.