From Conformational Ensembles to Kinetic Trajectories in Biomolecular Modelling

Jiying Zhang[1,2,3], Guanzi Chen[1], Bin Feng[1], Pierre Vandergheynst[3], Zijing Liu[1,+], Patrick Barth[2,+], and Yu Li[1,+]

[1]AI for Science, International Digital Economy Academy (IDEA), Shenzhen, China
[2]Laboratory of Protein and Cell Engineering, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
[3]Signal Processing Laboratory LTS2, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
[+]corresponding liyu@idea.edu.cn, liuzijing@idea.edu.cn, patrick.barth@epfl.ch

ABSTRACT

Many biomolecular processes are governed by conformational dynamics: proteins fold through metastable intermediates, receptors cross activation barriers, and ligands bind and dissociate through transient contacts. Molecular dynamics (MD) simulations resolve atomic motion at femtosecond time steps but remain limited in the timescales they can access. Predictive and generative deep models are reshaping this paradigm by replacing or augmenting the explicit integration of Newton's equations with neural transition kernels and path distributions, thereby recasting trajectory generation as a probabilistic sampling problem. These models can accelerate conformational exploration and trajectory simulation, although thermodynamic accuracy and kinetic consistency remain difficult to guarantee. Central to this Review is a comprehensive and actionable roadmap for biomolecular simulation, providing guidance on how to utilize the state-of-the-art deep models in accelerating molecular simulation and how to evaluate the deep learning methods with comprehensive metrics. Specifically, we organize recent advances into three foundational paradigms: sequential predictive models, trajectory-segment generative models, and latent-space dynamics models. Viewed through this taxonomy, a central trade-off emerges: models optimized for sampling efficiency are often the hardest to interpret kinetically, whereas those most faithful to classical dynamics inevitably inherit the sampling limitations and force-field biases of their training data. Looking ahead, we outline future directions, highlighting the vast potential of deep learning to revolutionize dynamic simulation and expedite drug development, positioning it as a transformative force at the forefront of drug discovery.

Key points

- Predictive and generative trajectory models move beyond static structure and conformational ensembles by learning finite-lag transitions, multi-frame path distributions, or latent dynamics with explicit temporal ordering.
- Biomolecular trajectory models can be organized into three broad paradigms: sequential prediction, trajectory-segment generation, and latent-space dynamics, according to their primary probabilistic objects and representations.
- Structural realism and ensemble agreement do not establish kinetic fidelity; quantitative interpretation requires time-dependent observables, sufficiently sampled reference trajectories, and careful treatment of heterogeneous simulation protocols.
- Trajectory models can broaden conformational and pathway exploration and support dynamic screening in drug discovery, but remain learned surrogates rather than general molecular simulators; quantitative use requires validated kinetics and robust transfer across molecular systems and environmental conditions.

1. Introduction

Molecular dynamics (MD) simulation provides an atomistic description of how chemical and biological systems evolve over time. Molecular function often involves transitions among conformational states, including protein folding[1], allosteric regulation[2], enzymatic catalysis[3], and ligand binding and dissociation[4]. Classical MD resolves these processes by propagating molecular states through Newton's equations of motion under an explicit physical model[5–7]. Long-timescale simulations have revealed transient intermediates and rare transitions that are difficult to resolve from static measurements, particularly in protein folding and conformational dynamics[1,8]. Their reach nevertheless remains limited by the femtosecond integration steps required for atomistic simulation, leaving many biologically and chemically relevant processes beyond routine sampling.

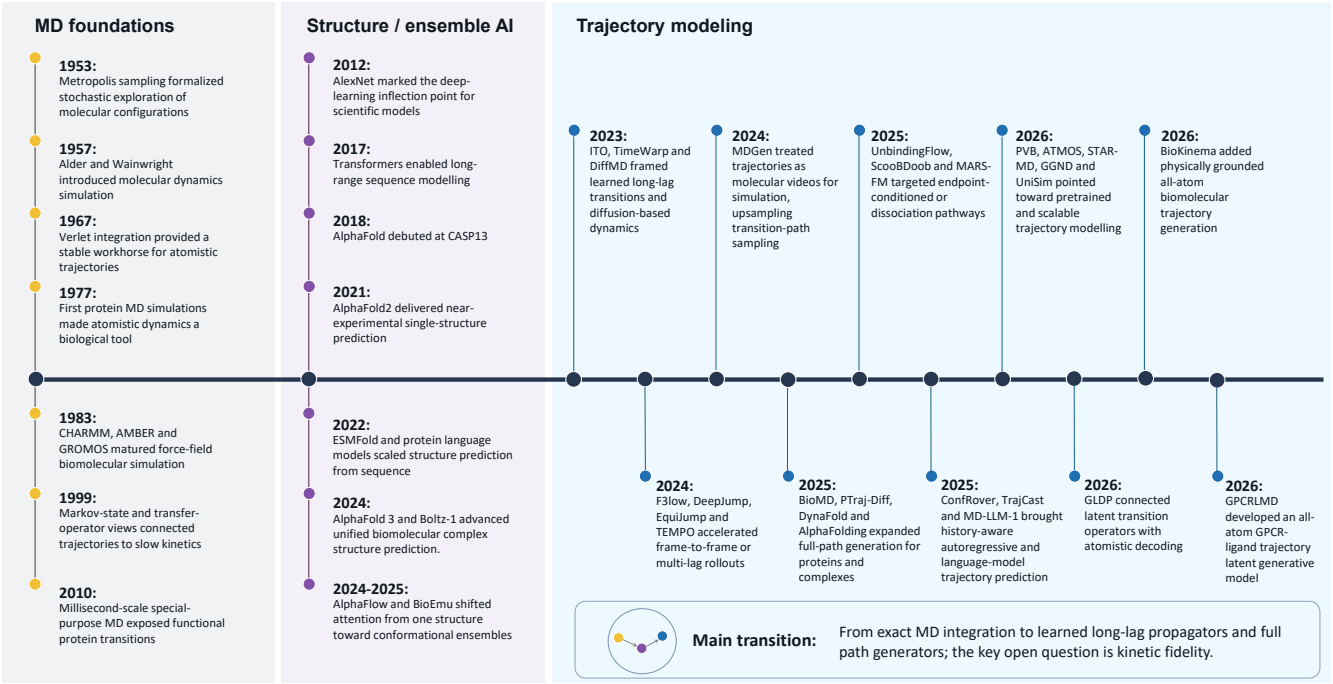


Figure 1. Historical development of deep learning for molecular trajectory prediction. Selected milestones are organized into three stages: molecular-dynamics and kinetic foundations, structure and ensemble AI, and trajectory modeling. Early developments established stochastic sampling, numerical integration, biomolecular force fields, and Markov-state or transfer-operator descriptions of slow molecular kinetics. Structure-prediction and ensemble-generation models subsequently provided accurate structural anchors and conformational priors. More recent trajectory-focused methods learn long-lag transition kernels, autoregressive rollouts, trajectory-segment distributions, endpoint-conditioned bridges, and latent dynamics with atomistic decoding. The timeline highlights the broader transition from exact short-step MD integration and static or unordered structural sampling toward learned time-ordered molecular paths. The central unresolved challenge is whether these generated trajectories preserve kinetic fidelity. Representative methods are selected to illustrate major conceptual developments rather than provide an exhaustive chronology.

Recent advances in Artificial Intelligence (AI) have substantially improved structural modelling but do not remove the need for explicit dynamical descriptions. AlphaFold2[9] and AlphaFold 3[10] provide accurate structural anchors, while AlphaFold2-based sampling strategies have begun to expose conformational heterogeneity[11–13]. Such approaches do not, however, define a calibrated molecular clock, assign transition probabilities, or establish whether an interpolated route is kinetically accessible. These distinctions matter for highly dynamic systems such as G-protein coupled receptor (GPCR) signalling complexes[14–16] and kinase active sites[17,18], where function depends on transitions among conformational states rather than on individual structures alone[19,20].

Deep generative models have extended structural prediction towards conformational ensembles. AlphaFlow[21], AnewSampling[22], BioEmu[23], and energy-feedback alignment[24], for example, sample broad distributions of plausible or equilibrium-like conformations. Ensemble agreement is an important step, but it does not determine how these conformations are connected in time: the same equilibrium distribution can support different transition rates, residence times, and pathways. Trajectory modelling therefore introduces a stronger objective in which temporal ordering and transition statistics become part of the prediction rather than an annotation added afterwards (Fig. 2).

This shift has motivated a rapidly growing class of models that learn molecular trajectories directly. Since 2024, approaches based on transition kernels, multi-frame generation, and learned latent dynamics have expanded alongside increasingly large public MD repositories (Fig. 1). We organize these methods by their primary probabilistic object and representation: (i) sequential predictive models learn state-level or finite-lag transition distributions and construct trajectories through causal rollout, including TimeWarp[25], ConfRover[26], STAR-MD[27], and ATMOS[28]; (ii) trajectory-segment generative models learn joint or endpoint-conditioned distributions over finite-horizon paths, including MDGen[29], BioMD[30], and ScooBDoob[31]; and (iii) latent-space dynamics models evolve learned representations before reconstructing atomistic trajectories, including GLDP[32], DynaFold[33], and GPCRLMD[34]. These categories describe the dominant modeling abstraction rather than mutually exclusive

architectures: diffusion and flow objectives can parameterize either state transitions or multi-frame segments, while latent representations can support both sequential and segment-level generation.

Several reviews have covered related areas, including classical MD workflows[35,36], AI methods for protein dynamics[37,38], and conformational ensemble generation[39]. Here, we focus specifically on predictive and generative trajectory models and on the physical interpretation of their outputs: when a generated path provides a useful mechanistic hypothesis, what evidence is needed for quantitative dynamical interpretation, and how such models can complement biomolecular simulation and downstream applications. Accordingly, we organize recent methods by their modelling paradigm, examine current approaches to kinetic evaluation, and discuss their practical integration with molecular simulation, experiment, and AI-assisted drug discovery.

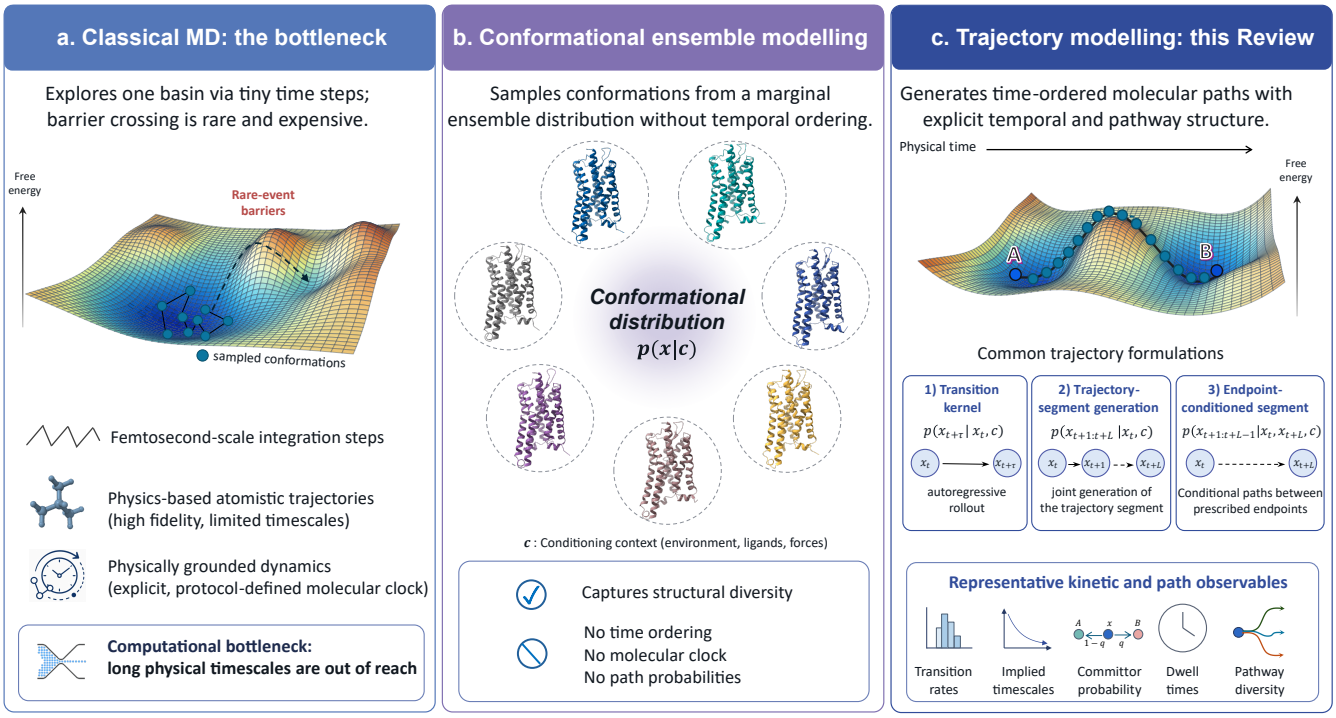


Figure 2. From classical molecular dynamics and conformational ensembles to learned molecular trajectories. a, Classical molecular dynamics (MD) propagates atomistic configurations through femtosecond-scale integration under a specified physical model, providing physics-based trajectories and an explicit molecular clock, but leaving long-timescale barrier crossing computationally demanding. **b,** Conformational ensemble models learn a marginal distribution $p(x \mid c)$ over molecular structures, where c denotes conditioning information such as the environment, ligand identity or external forces. These models capture structural diversity without specifying temporal ordering, a molecular clock or path probabilities. **c,** Trajectory models instead represent time-ordered molecular motion through finite-lag transition kernels $p_\theta(x_{t+\tau} \mid x_t, c)$, jointly generated trajectory segments $p_\theta(x_{t+1:t+L} \mid x_t, c)$ or endpoint-conditioned segments $p_\theta(x_{t+1:t+L-1} \mid x_t, x_{t+L}, c)$. Representative kinetic and path observables include transition rates, implied timescales, committor probabilities, dwell times and pathway diversity. Here, x_t denotes the molecular configuration at physical time t, τ the lag time, L the trajectory-segment horizon and θ the model parameters. Free-energy landscapes are schematic; A and B denote metastable conformational basins.

The kinetic gap in molecular simulation

Molecular dynamics is determined not only by which conformations are accessible, but also by how they are connected in time. A useful way to see this distinction is through a free-energy landscape: an ensemble model may correctly reproduce the Boltzmann-weighted populations of low-free-energy basins, yet this distribution alone does not specify how the system moves between them. The same equilibrium distribution can be consistent with very different residence times, transition frequencies and pathways. A trajectory model makes the stronger claim of describing these temporal connections. At an appropriate lag time, transitions among metastable states can often be summarized by an approximately Markovian process, making state-to-state probabilities, relaxation times and pathway fluxes natural kinetic quantities[40,41].

This distinction is increasingly important as predictive and generative models move from conformational ensembles to time-ordered molecular trajectories[25,28–31]. Many current evaluations still emphasize structural diversity or agreement with

Table 1. Representative deep generative and predictive models for molecular trajectory prediction and transition-path generation, organized according to the three-part taxonomy used in the main text. Target systems indicate the intended application scope, whereas training data list the principal structural or trajectory sources used for model development, and the details of these data can be found in Table 2.

Category	Model	Representation	Method	Target systems	Training data
Sequential predictive models	TimeWarp[25]	All-atom	Normalizing flow + MH correction	Peptides	Peptide MD
	ConfRover[26]	All-atom	AR + diffusion	Proteins	ATLAS
	STAR-MD[27]	All-atom	Spatiotemporal AR + diffusion	Proteins	ATLAS
	MD-LLM-1[42]	Latent / tokens	Latent LM	Proteins	Mad2 and T4 lysozyme MD
	EquiJump[43]	All-atom	Stochastic interpolant	Proteins	Fast-folding protein MD
	TEMPO[44]	Backbone	Hierarchical two-timescale stochastic autoregression	Proteins	ATLAS; mdCATH
	MARS-FM[45]	All-atom	MSM-guided flow matching	Peptides; proteins	Peptide MD; mdCATH
	ATMOS[28]	All-atom	SSM + diffusion	Proteins; P–L complexes	PDB; mdCATH; MISATO
	TrajCast[46]	All-atom	Autoregressive equivariant GNN	System-specific small molecules, solids	System-specific MD trajectories
	DiffMD[47]	All-atom / geometric	GNN + diffusion	System-specific small molecules; C7O2H10	MD17; C7O2H10 isomer MD
	GGND[48]	All-atom / geometric graph	Geometric graph diffusion	System-specific molecules	3BPA[49] and SAMD23 SiN and HfO[50](separate system-specific)
	NeuralMD[51]	All-atom	Neural ODE + GNN	P–L complexes	MISATO
	ITO[52]	All-atom / Cα	Conditional diffusion	Peptides; fast-folding proteins	Peptide and fast-folding MD
	PLaTITO[53]	Cα	Conditional flow matching + pLM conditioning	Proteins	mdCATH
	F3low[54]	Backbone	Guided flow matching	Proteins	Fast-folding protein MD
	DeepJump[55]	All-atom	Conditional flow matching	Proteins; fast-folding MD	mdCATH
	UniSim[56]	All-atom / latent	Stochastic interpolants + force guidance	small molecules, peptides, and proteins	Multi-domain pretraining; PepMD[57]; ATLAS; MD17
	PVB[58]	All-atom / latent	Variational encoder + augmented bridge matching	Small molecules; proteins; P–L complexes	PCQM4Mv2[59]; ANI-1x[60]; PDB; PDBBind (pretrain); ATLAS; mdCATH; MISATO (MD finetune)
	UnbindingFlow[61]	All-atom / coarse-grained	Flow matching	P–L unbinding	DD-13M
Trajectory-segment generative models	DynamicsDiffusion[62]	All-atom	Trajectory-level DDPM	Toy systems; peptides	Double-well trajectories; alanine-dipeptide MD
	PTraj-Diff[63]	Backbone	SE(3)-parameterized diffusion	P–P complexes	1HPV and 1BRS MD
	AlphaFolding[64]	All-atom	4D diffusion	Proteins	ATLAS; DynamicPDB MD
	GeoTDM[65]	All-atom / geometric	GNN + diffusion	System-specific small molecules	MD17 (separate training for each molecule)
	EGInterpolator[66]	All-atom / geometric	Structure-pretrained diffusion + temporal interpolation	Small molecules; peptides; proteins	GEOM-QM9/Drugs; QM9/Drugs MD; TimeWarp; ATLAS
	STFlow[67]	All-atom	Flow matching + GNN	System-specific small molecules	MD17 (separate training for each molecule)
	DynaMode[68]	Cα / spectral	Spectral diffusion + inverse DCT	Proteins	mdCATH
	MDGen[29]	All-atom	Flow matching	Proteins; peptides	ATLAS; peptide MD
	GP-EquiFlow[69]	All-atom	SE(3)-equivariant flow matching + GP prior	N-body; small molecules	MD17; synthetic N-body
	GSE-Flow[70]	All-atom	SE(3)-equivariant flow matching	Small molecules; macromolecules; human motion	MD17; MD22; CMU MoCap
	BioMD[30]	All-atom	DiT + flow matching + hierarchical modeling	P–L complexes	DD-13M; MISATO
	BioKinema[71]	All-atom	Spatiotemporal diffusion + hierarchical modeling	Proteins; peptides; P–L complexes	DD-13M; MISATO; ATLAS; MDposit; MSR (CATH/MegaSim/octapeptide MD)
	PIPS[72]	All-atom	Stochastic optimal control + neural bias force	Peptides; proteins	On-the-fly biased OpenMM MD
	Doob's Lagrangian[73]	All-atom	Variational Doob h-transform	Peptides; proteins	Simulation-free; AMBER14/OpenMM energy model
	TPS-DPS[74]	All-atom	Off-policy diffusion path sampling + equivariant bias force	Peptides; fast-folding proteins	On-the-fly biased OpenMM MD
	ScooBDoob[31]	CV / MSM states	Schrödinger bridge + Doob transform	Peptides	Peptide MD
	TPS-Flow[75]	Backbone	Flow matching + spatiotemporal gated attention	Proteins; P–P complexes	System-specific protein MD
	DeepPath[76]	Reduced coordinates / all-atom	Progressive GAN + active learning	Proteins; protein complexes	Endpoint structures; MM-guided active-learning structures
Latent-space dynamics	DeepGenMSM[77]	All-atom	latent MSM + decoder	System-specific small molecules	Alanine-dipeptide MD
	LSS[78]	Cα	MDN latent propagator + decoder	System-specific proteins	System-specific MD
	GLDP[32]	All-atom	Latent operator learning + decoder	System-specific peptides/proteins	System-specific MD
	DynaFold[33]	All-atom	Latent diffusion + backmapping	Proteins	PDB; AFDB; ATLAS; fast-folding MD
	GPCRLMD[34]	All-atom	Latent flow matching	GPCR–ligand complexes	GPCRMD

Note: Categories reflect primary generative objects rather than mutually exclusive architectures. Hybrid methods are listed once according to their dominant role: state-level prediction is classified as sequential, whereas joint generation of temporally coupled frames is classified as trajectory-segment generation. Methods are ordered approximately by modeling family. AR, autoregressive; CV, collective variable; GNN, graph neural network; LM, language model; MH, Metropolis–Hastings; MSM, Markov state model; P–P, protein–protein; P–L, protein–ligand; SSM, state-space model.

reference ensembles, which establish where a model samples but not whether it reproduces how molecular states interconvert. This review therefore places kinetic fidelity alongside structural and ensemble accuracy, asking which generated trajectories can support mechanistic hypotheses, which preserve quantitative dynamics, and how these claims should be tested.

2. Theoretical Foundations: Formulating Dynamics as Predictive and Generative Tasks

This review focuses on models whose primary objective is to learn or generate biomolecular time evolution, rather than methods designed primarily to learn interatomic potential-energy surfaces or force fields[79–81]. Although learned force fields can substantially reduce the cost of force evaluation, they retain the underlying integration-based formulation and do not by themselves remove the long-timescale sampling bottleneck. We therefore treat force-field learning as a complementary route to accelerated simulation and concentrate on models that directly parameterize finite-lag transitions, trajectory-segment distributions, or latent dynamics.

2.1 Trajectories as Stochastic Processes and Path-Space Distributions

For a molecular system with coordinates $x_t \in \mathbb{R}^{N\times 3}$, a trajectory is a time-ordered random object $X_{0:T} = (x_0, x_1, \ldots, x_T)$ rather than a bag of structures. Classical simulation defines this object through an integrator, force field, thermostat, barostat, solvent model, and boundary conditions. A trajectory model replaces part of this machinery with a learned distribution. We denote a finite-horizon trajectory segment by $X_{t:t+L} = (x_t, x_{t+1}, \ldots, x_{t+L})$, where L is the segment horizon and need not span the complete physical trajectory. Three recurring formulations are useful. First, sequential prediction learns a time-coarsened conditional distribution $p_\theta(x_{t+\tau} \mid x_t, c)$ or $p_\theta(x_{t+\tau} \mid x_{\le t}, c)$ and constructs longer trajectories through causal rollout, where c denotes optional conditioning information such as environment, ligands, and forces. Second, trajectory-segment generation learns a coupled multi-frame distribution such as $p_\theta(X_{t+1:t+L} \mid x_t, c)$ or $p_\theta(X_{t+1:t+L} \mid x_{\le t}, c)$. Third, endpoint-conditioned generation models $p_\theta(X_{t+1:t+L-1} \mid x_t, x_{t+L}, c)$, which is natural for transition-path sampling between prescribed basins (Fig. 2c).

These formulations differ primarily in their temporal factorization and scientific claim, rather than in the neural architecture alone. A reusable transition kernel can be tested as an accelerated simulator, but repeated rollout accumulates its own errors. A trajectory-segment generator can couple fluctuations and transitions across several frames within one finite horizon, yet computational limits may require long trajectories to be extended autoregressively from one segment to the next, reintroducing drift or boundary inconsistencies. An endpoint-conditioned bridge can efficiently propose rare routes, but the resulting path distribution is conditional on reaching the target and does not by itself determine the unconditioned transition rate. Throughout this Review we therefore separate structural validity from kinetic fidelity. Structural validity asks whether frames are chemically plausible. Kinetic fidelity asks whether the generated process reproduces transition probabilities, implied timescales, committor structure and time-correlation functions. Much of the field currently has evidence for the former and only partial evidence for the latter.

2.2 Physical Symmetries and Equivariance in Time Evolution

Translations and rotations of the input coordinates should produce corresponding transformations of the generated trajectory. Permutations of identical atoms should likewise preserve its probability distribution. These requirements motivate $E(3)$- or $SE(3)$-equivariant neural networks, tensor-product attention, equivariant generative kernels and geometric message passing.

Equivariance constrains molecular geometry but does not ensure thermodynamic or kinetic fidelity. Faithful dynamics also depend on equilibrium or controlled irreversibility, conservation behaviour and coupling between slow collective modes and fast local fluctuations. Time-coarsened predictors are particularly sensitive to this coupling. Mean-only updates can suppress fluctuations, while incompatible stochastic drift can alter energies or move trajectories away from thermodynamically relevant regions. Additional constraints may enter through equilibrium corrections, energy or force guidance, conservation objectives, and dynamical-consistency losses. Their value should be judged through the physical observables they preserve, rather than through the presence of a nominal physical prior. These constraints remain conditional on the selected physical model and may inherit force-field, protocol and sampling biases from the training trajectories.

2.3 An Operational Taxonomy

These distinctions determine how the following sections are organized. Sections 3 and 4 classify models by their primary temporal object: a finite-lag transition or a jointly generated trajectory segment. Endpoint-conditioned paths are treated as a specialized form of segment-level generation because conditioning changes the sampled path distribution but not the underlying temporal object.

Section 5 considers latent-space models separately because encoding and atomistic backmapping introduce an additional validation problem. Latent models may nevertheless use either sequential propagation or segment-level generation, so temporal

factorization and molecular representation are not mutually exclusive classification axes. The same diffusion, flow or transformer architecture may also parameterize different generative objects.

Hybrid methods make these overlaps explicit. BioKinema, for example, generates frames jointly within each segment and extends successive segments autoregressively[71]. Other methods, including BioMD, PVB and MD-LLM-1, combine choices across forecasting, endpoint conditioning, autoregressive extension or molecular representation[30,42,58]. We assign each method to the category that best reflects its primary scientific claim and validation burden. These burdens concern rollout stability for sequential models, segment-level path fidelity for joint generators, and kinetic preservation plus atomistic backmapping for latent models. The taxonomy of representative methods is shown in Fig. 3, Table 1 and Sup. Fig. 1. For the validation coverage, such as Structural fidelity, Thermodynamic fidelity, and Kinetic fidelity, please refer to the Sup. Table 2

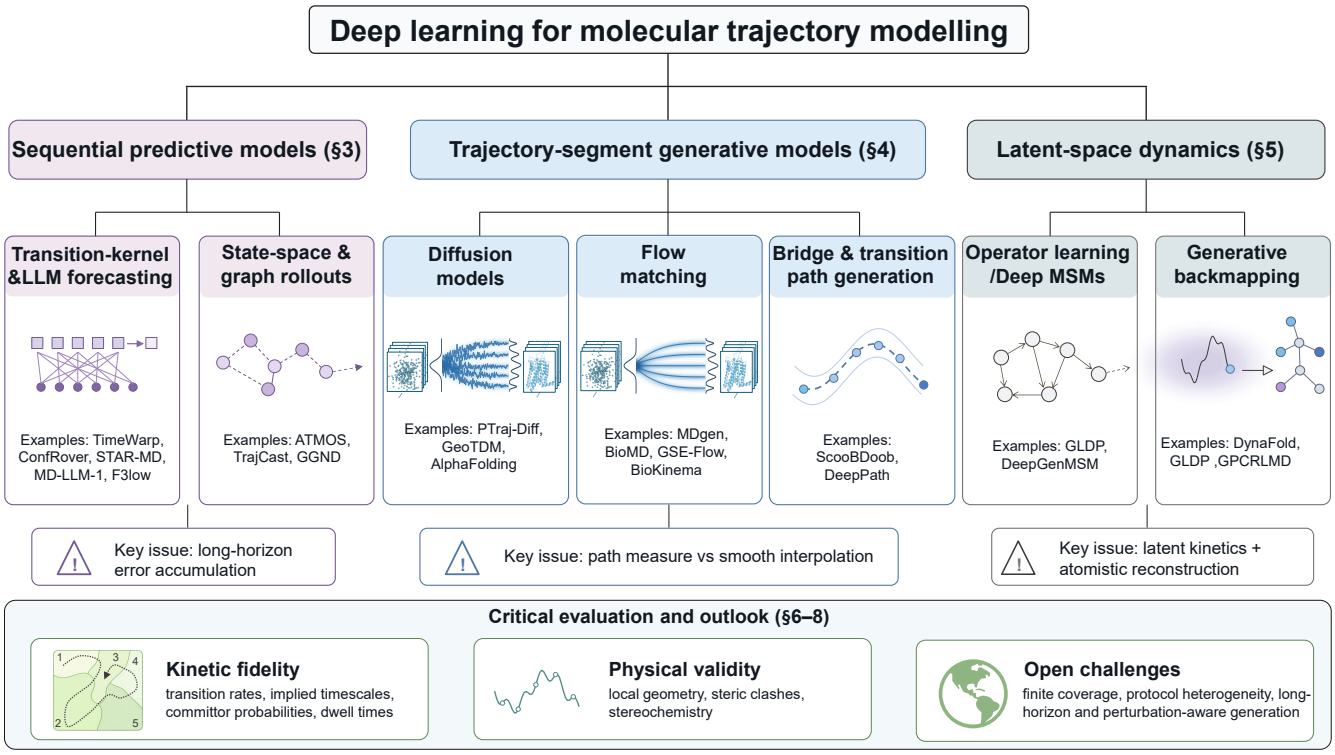


Figure 3. Taxonomy of deep learning methods for molecular trajectory modelling. Methods are organized by their primary probabilistic object and representation of dynamics. Sequential predictive models learn finite-lag transitions and construct trajectories through causal rollout, with long-horizon error accumulation as a central limitation. Trajectory-segment generative models use diffusion, flow matching or bridge formulations to model temporally coupled finite-horizon paths, raising the question of whether generated paths reproduce physical dynamics rather than smooth structural interpolation. Latent-space approaches learn dynamical representations or operators before reconstructing atomistic trajectories, coupling latent kinetic fidelity to the quality of backmapping. The categories denote the dominant modelling abstraction rather than mutually exclusive architectures, as diffusion, flow, and autoregressive mechanisms can occur across different formulations. The bottom panel summarizes cross-cutting considerations discussed in Sections 6–8, including kinetic fidelity, physical validity, and challenges arising from finite sampling, heterogeneous simulation protocols and the development of calibrated trajectory generators. Examples are representative rather than exhaustive.

3. Sequential Predictive Models: Transition Kernels and Autoregressive Rollouts

Sequential predictive models make the most direct bargain with MD: exchange many small integration steps for a larger learned update. If this bargain holds, a model can act as a time-coarsened simulator. If it fails, the error is not a local imperfection but a dynamical bias that compounds with every generated frame. The most important question for this family is therefore not whether a single predicted step is close to a reference frame, but whether repeated steps preserve the slow processes that define the system.

3.1 Time-Coarsened Transition Kernels and Sequence Forecasting

Time-coarsened models learn a stochastic propagator $p_\theta(x_{t+\tau} \mid x_t, c)$ at a lag time τ substantially larger than the integration step of classical MD, allowing trajectories to be constructed from a sequence of learned coarse updates. Existing approaches mainly differ in how this transition distribution is parameterized. TimeWarp combines normalizing-flow proposals with Metropolis–Hastings correction[25], while TEMPO separates slow and fast molecular modes within the learned dynamics[44]. Diffusion-based formulations include ITO and its protein-language-model extension PLaTITO, which model transition densities over multiple lag times[52,53]. EquiJump and UniSim instead use stochastic interpolants, with UniSim further incorporating force guidance[43,56]. PVB takes a related bridge-based approach, using augmented bridge matching to unify structural pretraining with finite-lag transition learning from MD trajectories[58]. Flow-based objectives are adopted by F3low, DeepJump and UnbindingFlow for conformational transitions or ligand dissociation[54,55,61].

The conditioning need not be restricted to the current molecular state. ConfRover incorporates latent trajectory history with diffusion decoding[26], STAR-MD uses spatiotemporal diffusion for history-conditioned forecasting[27], and MD-LLM-1 represents molecular trajectories as token sequences for language-model-style prediction[42]. Longer temporal context can capture correlated motions and help resolve transitions whose direction is ambiguous from a single conformation.

The main advantage of these models is that a learned transition kernel can be repeatedly applied beyond the training horizon. Long-time behavior, however, depends on repeated composition of this approximate propagator, so even small finite-lag errors can alter its stationary distribution, rare-event fluxes and slow dynamical modes. Good one-step prediction is therefore not sufficient to establish that the learned process remains faithful after many transitions. Evaluation should examine how errors evolve with rollout length and whether stationary populations, multi-step transition probabilities, Chapman–Kolmogorov consistency and implied timescales remain consistent with the reference dynamics[40].

3.2 State-Space and Graph-Based Rollouts

Other sequential models introduce more explicit structure into how molecular states are represented and propagated. ATMOS uses state-space modelling to retain temporal information over extended trajectories[28], while TrajCast directly forecasts molecular positions and velocities without repeated force evaluation[46]. MARS-FM performs generative modelling over MSM-defined metastable states[45]. NeuralMD combines graph-based modelling of molecular interactions with neural differential-equation integration to propagate protein–ligand dynamics under Newtonian constraints[51]. Graph representations are also used in UnbindingFlow[61], DiffMD[47] and GGND[48], where geometric interactions and diffusion-based refinement are incorporated into the rollout. These structured representations can provide useful inductive biases for local interactions and temporal evolution, although accurate geometry or short-term forecasting does not ensure that the resulting process preserves equilibrium populations or long-timescale kinetics.

3.3 Mitigating Error Accumulation in Long-Horizon Rollouts

Long-horizon rollout amplifies small prediction errors because models increasingly condition on states generated by themselves rather than reference trajectories. This distribution shift affects history-conditioned models such as ConfRover[26], STAR-MD[27] and ATMOS[28], while repeated application of time-coarsened kernels can similarly distort populations and slow dynamics. Stochastic updates, refinement mechanisms, physical constraints and rollout-aware training can reduce such drift, with broader strategies for long-horizon generation discussed in Section 8.2. Their effectiveness should ultimately be assessed over increasing rollout horizons through structural stability and kinetic observables such as transition probabilities, Chapman–Kolmogorov consistency and implied timescales.

4. Trajectory-Segment Generative Models: Diffusion, Flow and Conditional Path Synthesis

Unlike transition-kernel models, which propagate trajectories one state at a time through repeated rollout, trajectory-segment models jointly generate a finite block of future conformations conditioned on an anchor state. Given $x_t \in \mathbb{R}^{N\times 3}$, a model samples the subsequent segment $X_{t+1:t+L} \sim p_\theta(X_{t+1:t+L} \mid x_t, c)$ or $p_\theta(X_{t+1:t+L} \mid x_{\le t}, c)$ through a coupled denoising, transport or bridge process. This batchwise formulation captures correlations across frames and amortizes computation over the segment, improving parallelism and sampling throughput. Sup. Table 1 summarizes the Sampling Efficiency of these methods. Longer trajectories can still be assembled autoregressively from successive segments. The approach is particularly useful for rare events, where alternative paths are more informative than a single future state. Validation must address frame-level physical validity, segment-level distributional fidelity and calibration to molecular time.

4.1 Diffusion Models for Trajectory Segments

Trajectory diffusion models treat multiple temporally ordered conformations as a coupled generative object rather than predicting each future state separately. DynamicsDiffusion provided an early formulation by applying denoising diffusion directly to molecular trajectory segments for unconditional generation and rare-event sampling[62]. Subsequent work has largely focused on

introducing molecular geometry and temporal structure into this formulation. PTraj-Diff uses $SE(3)$-parameterized diffusion for protein and protein–protein complex trajectories[63], GeoTDM combines equivariant diffusion with temporal attention[65], and AlphaFolding incorporates reference and motion guidance into four-dimensional diffusion of all-atom trajectories[64]. BioKinema extends spatiotemporal diffusion to continuous-time all-atom biomolecular trajectories and combines coarse forecasting with interpolation for long-horizon generation[71]. Other approaches modify the representation or supervision used for trajectory generation: EGInterpolator couples diffusion-based structural pretraining with an MD-trained temporal interpolation module[66], while DynaMode performs diffusion over discrete-cosine-transform representations of finite trajectories, providing a spectral decomposition of slow motions and faster fluctuations[68]. Across these formulations, the diffusion variable spans multiple physical frames, which distinguishes them from the finite-lag transition models in Section 3. Latent trajectory diffusion, such as DynaFold, is discussed in Section 5 because reconstruction from the latent representation introduces an additional modelling challenge. Further details of diffusion formulations are provided in Sup. S.5.1.

Joint generation is particularly useful when similar initial conformations can lead to different dynamical outcomes, such as local fluctuations, productive transitions or unsuccessful transition attempts. Modelling these alternatives within a multi-frame distribution can retain temporal correlations and multimodality that are difficult to capture with a single-state prediction. The generated segment, however, is not necessarily sampled with the probability or timescale of the underlying molecular process: accurate denoising and structurally realistic trajectories can coexist with distorted pathway frequencies or transition durations. Kinetic assessment therefore requires comparison of basin populations, transition probabilities, time-correlation functions and committor structure, together with tests of continuity and dynamical stability when successive segments are connected into longer trajectories.

4.2 Flow Matching over Trajectory Segments

Flow matching learns a time-dependent vector field that transports a simple prior toward the distribution of molecular trajectory segments[82]. Because the transport path can be trained without explicitly simulating its intermediate distributions, generation proceeds by integrating the learned field with an ODE solver and can potentially require fewer network evaluations than iterative denoising. MDGen introduced this formulation for molecular dynamics by treating a trajectory segment as a spatiotemporal generative object[29]. The subsequent model, BioMD, has adapted the same principle to more complex molecular settings, combining hierarchical forecasting and interpolation for all-atom protein–ligand trajectories[30]. A second line of development has focused on making the transport itself better matched to trajectory geometry and temporal correlations. STFlow introduces conditioned random-walk priors and spatiotemporal graph structure[67]; GP-EquiFlow uses vector-valued Gaussian processes to construct $SE(3)$-equivariant, temporally correlated priors[69]; and GSE-Flow combines equivariant flow matching with history-aware spatiotemporal representations[70]. Although these formulations differ in their priors and architectures, they generate multiple physical frames as a coupled segment, with longer trajectories obtained when needed by connecting successive segments. Further details of flow-matching formulations are provided in Sup. S.5.2.

Like diffusion models, flow-matching generators can reproduce a joint distribution over several frames, but this does not guarantee recovery of the physical dynamics that produced it. An additional distinction arises from the vector field itself: the ODE coordinate parameterizes generative transport rather than molecular time, so the learned field should not be interpreted as a molecular force or velocity. Consequently, a model may reproduce structural diversity and state populations while still distorting temporal correlations, residence times or transition probabilities. Kinetic assessment should therefore include implied timescales, transition probabilities, time-correlation functions and dwell-time statistics, together with tests of their stability when finite segments are extended into longer trajectories[40,41].

4.3 Bridge-Based and Transition-Path Generation

Bridge-based and transition-path models focus on the endpoint-conditioned trajectory-segment distribution $p(X_{t+1:t+L-1} \mid x_t, x_{t+L}, c)$, describing pathways between prescribed molecular states such as bound and unbound ligand configurations, active and inactive receptors, or folded and unfolded basins. One line of work seeks to approximate the conditioned path measure through controlled stochastic dynamics. PIPS combines transition-path sampling with Schrödinger bridges and stochastic optimal control to learn collective-variable-free bias policies[72]; Doob's Lagrangian develops a variational approximation to the Doob h-transform[73]; and TPS-DPS amortizes sampling from a target transition-path distribution through off-policy diffusion-path training[74]. Related ideas can also be applied at coarser resolution, as in ScooBDoob, which combines Schrödinger bridges and the Doob transform over MSM-defined metastable states[31]. A different route generates the intermediate conformations more directly: TPS-Flow uses physics-guided endpoint-conditioned flow matching for multi-frame protein pathways[75], while DeepPath learns structural progressions between known endpoints through progressive adversarial training[76]. These formulations differ in resolution and sampling procedure, but all place the transition path, rather than an unrestricted future trajectory, at the centre of the generative task.

Endpoint conditioning narrows the path ensemble and can make rare transitions substantially easier to sample, but it also changes the interpretation of the resulting trajectories. A conditional path distribution does not determine the equilibrium

populations of the endpoint states or the rate at which the unconditioned system moves between them, and the sampled routes may depend strongly on endpoint definitions, state representations, reference dynamics and imposed physical constraints. Narrow conditioning can also exclude alternative channels, while progression along a generated path need not provide a calibrated molecular clock. Where kinetic interpretation is intended, structural plausibility should therefore be complemented by committor probabilities, transition-path fluxes, pathway diversity and free-energy profiles, together with comparison against independent molecular simulations.

5. Latent-Space Dynamics and Spatiotemporal Coarse-Graining

Latent-space models seek representations in which molecular dynamics can be described more compactly or with a simpler transition structure. Although these representations are often lower-dimensional, compression is not essential; latent variables may instead provide stochastic, geometry-aware or otherwise structured reparameterizations of Cartesian coordinates. This perspective cuts across the temporal formulations introduced above, because latent states can be propagated sequentially or generated jointly over finite segments. Their interpretation nevertheless introduces an additional layer of uncertainty: the learned representation must preserve the dynamical information of interest, and decoding must recover chemically and temporally consistent atomistic structures. Slow collective coordinates may dominate the latent dynamics, while local packing, ligand contacts, solvent exposure and side-chain rearrangements remain important after reconstruction.

5.1 Operator Learning: Koopman Theory and Deep MSMs

A natural use of latent representations is to describe dynamics through the evolution of probabilities or observables rather than direct coordinate prediction. Markov state models (MSMs)[41], transfer operators and Koopman formulations[83] all follow this principle. Given an encoder E_ϕ, a molecular state is mapped to $z_t = E_\phi(x_t, c)$, after which the dynamics can be represented by a latent transition model $p_\theta(z_{t+\tau} \mid z_{\leq t}, c)$ or, in the Koopman setting, by an operator $\mathscr{K}_\tau$ satisfying $(\mathscr{K}_\tau f)(z_t) = \mathbb{E}[f(z_{t+\tau}) \mid z_t]$ for observables f. Latent dynamics can then be formulated at different levels of resolution. DeepGenMSM models transitions among latent metastable states and generatively maps them back to molecular conformations[77]. LSSs instead propagate continuous slow coordinates learned with reversible VAMPnets and reconstruct the corresponding molecular conformations[78]. GLDP further illustrates that the choice of latent representation can be separated from the propagator itself, comparing autoregressive, Koopman and score-guided Langevin dynamics within a shared representation[32]. These formulations connect learned trajectories to transition probabilities, slow modes and implied timescales without requiring dynamics to be propagated directly in Cartesian coordinates.

The quality of these quantities, however, depends on the representation in which the dynamics are modelled. An encoder that merges kinetically distinct states or removes variables needed for approximately Markovian evolution can yield accurate reconstruction or short-lag prediction while distorting slow relaxation and transition pathways. Operator agreement should therefore be tested in a common state space using kinetic observables, and conclusions drawn in latent space should, where possible, be checked again after reconstruction into atomistic coordinates.

5.2 Generative Backmapping from Latent Kinetics

For models that generate trajectories in latent space, kinetic fidelity and atomistic reconstruction are closely coupled. A latent process may preserve the desired slow dynamics, yet decoding can alter local geometry, molecular interactions or temporal continuity; conversely, accurate reconstruction cannot compensate for incorrect latent kinetics. GPCRLMD generates latent trajectory segments with flow matching before decoding them into all-atom GPCR–ligand conformations[34], while DynaFold applies diffusion to latent trajectories followed by atomistic reconstruction[33]. GLDP couples learned latent dynamics to an all-atom decoder[32]. Related backmapping problems also arise in methods classified elsewhere by their primary temporal formulation: PVB bridges latent and coordinate representations[58], and MD-LLM-1 reconstructs molecular dynamics from tokenized states[42]. Further details of latent generative model formulations are provided in Sup. S.5.3.

Compression or reparameterization can make slow dynamics easier to model, but decoding may reintroduce degrees of freedom that were only weakly constrained in latent space. Stereochemistry, side-chain packing, ligand contacts, solvent-mediated interactions and temporal continuity can therefore deteriorate even when the latent trajectory reproduces the intended slow modes. Evaluation should consequently examine both kinetic observables before and after decoding and the physical quality of the reconstructed trajectory, including local geometry, thermodynamic plausibility and continuity across successive frames or segment boundaries.

6. Benchmarking and Critical Evaluation of Kinetic Fidelity

Evaluation of molecular trajectory models should proceed through progressively stronger levels of evidence. Frame-level metrics test whether individual conformations are reconstructed accurately, ensemble metrics test whether the distribution of

sampled structures is reproduced, and kinetic metrics test whether those structures are connected with the correct temporal statistics. These levels are complementary but not interchangeable: agreement at a lower level does not by itself establish agreement at a higher one.

6.1 From Ensemble Agreement to Trajectory Fidelity

A first level of evaluation asks whether generated conformations reproduce the structural statistics of a reference ensemble. Existing ensemble benchmarks, such as those introduced for AlphaFlow[21], combine pairwise RMSD and RMSF correlations with Wasserstein distances in Cartesian or principal-component spaces and higher-order observables such as inter-residue contacts, solvent exposure, and correlations between structural features. Together, these metrics assess conformational breadth, flexibility, collective variation, and correlated structural fluctuations, and are substantially more informative than frame-wise reconstruction error alone.

However, ensemble agreement does not establish trajectory fidelity because these observables generally do not depend on temporal ordering. Randomly permuting the frames of a trajectory would leave its RMSF, conformational distribution, contact frequencies, and most other ensemble statistics unchanged, while altering or destroying its autocorrelation decay, dwell times, transition rates, and reactive pathways. In probabilistic terms, ensemble metrics primarily characterize the marginal distribution $p(\{x_t\} \mid c)$, while a trajectory model makes the stronger claim of learning a transition kernel $p(x_{t+\tau} \mid x_t, c)$ or a path distribution $p(X_{0:T} \mid c)$. This imbalance is reflected in current validation practice. Of the 41 representative trajectory models summarized in Sup. Table 2, 39 (95%) include explicit structural evaluation, but only 14 (34%) report explicit quantitative kinetic validation; the remainder rely on partial or indirect tests or report no kinetic evaluation.

Agreement with ensemble benchmarks should therefore be treated as necessary evidence of conformational coverage, but not as evidence of kinetic fidelity. The appropriate evaluation also depends on the intended use of the model. Conformational sampling requires accurate state populations and structural diversity; pathway proposal requires plausible intermediates and alternative routes; and quantitative kinetic inference additionally requires a calibrated time mapping and transition statistics. The following section introduces time-dependent observables for testing this stronger claim. Regardless of the intended use, generated frames must also satisfy basic geometric and stereochemical checks before ensemble or kinetic agreement can be considered meaningful.

6.2 Kinetic and Temporal Metrics

Once conformational coverage has been established, the stronger test is whether generated structures are connected with the correct temporal statistics. The appropriate observables depend partly on the molecular system. Velocity autocorrelation functions, mean-squared displacement and diffusion coefficients are informative for small molecules and materials, while protein conformational dynamics are often analysed through metastable-state kinetics and slow collective coordinates. Markov-state analysis therefore provides a useful common framework, complemented by representations such as TICA or VAMP[40,84]. Relevant kinetic tests include[40]:

- **implied timescales**, quantifying relaxation of slow dynamical processes;
- **continuous-time transition rates**, quantifying state-to-state kinetics;
- **dwell-time distributions**, measuring persistence within metastable states;
- **mean first-passage times**, measuring the average time required to first reach a target state;
- **TICA autocorrelation functions**, measuring relaxation along shared slow coordinates;
- **Chapman–Kolmogorov consistency**, testing whether short-lag dynamics predict longer-time evolution; and
- **committor probabilities and reactive fluxes**, characterizing transition progress and dominant pathways.

For these comparisons to reflect differences in dynamics rather than differences in representation, reference and generated trajectories should be analysed in the same dynamical state space. Let $\mathbf{x}_n$ denote the conformation at frame n, separated by Δt, and assign each frame to a metastable state $s_n = g(\mathbf{x}_n) \in \{1, \ldots, M\}$. The mapping g, including input features, TICA or VAMP projection, clustering and state definitions, should be constructed from the reference data and then applied unchanged to generated trajectories, as adopted in recent kinetic evaluations of trajectory generators[71]. A model may reproduce the occupancy of these states while still assigning incorrect transition probabilities, dwell times or relaxation times, making temporal ordering essential to the comparison. Quantitative kinetic interpretation additionally requires a calibrated physical interval Δt; coordinate-only paths without such calibration can support pathway analysis, but not direct estimates of rates or timescales. Detailed definitions of these metrics are provided in Sup. S.6.

6.3 Current Datasets and the Need for Dynamic Benchmarks

Available trajectory resources cover three broad dynamical regimes. Small-molecule datasets such as MD17 and MD22 provide high-quality trajectories for force learning and geometric prediction, but offer limited access to rare biomolecular events[85,86]. Protein datasets span conformational fluctuations, folding and slow collective motions, from large-scale resources such as ATLAS and mdCATH to the longer DESRES and Anton trajectories that remain particularly valuable for kinetic analysis[1,8,87,88]. A third group targets interaction-driven dynamics, including protein–ligand and membrane systems represented by MISATO, DD-13M, MemProtMD, and GPCRmd[61,89–91]. Broader repositories such as MDRepo and DynaRepo complement these curated datasets by collecting trajectories across systems and simulation protocols[92,93]. Their scope and simulation settings are summarized in Table 2.

Table 2. Selected public molecular-dynamics datasets, trajectory repositories, and MD-derived conformational resources relevant to molecular trajectory modelling.

Dataset	Molecular scope	Scale / duration	Protocol snapshot	Use case
MD17 / MD22 [85,86]	Molecular and supramolecular systems	Original MD17: 8 systems, $\sim 10^5$–10^6 conformations per molecule; MD22: 7 systems, ∼5k–85k systems	DFT-based ab initio MD with energy/force labels; MD17: PBE+vdW-TS, 500 K, 0.5 fs; MD22: PBE+MBD, 400–500 K, 1 fs	Force learning; trajectory benchmarks
DESRES fast-folding [8]	Fast-folding proteins	12 proteins; 100 μs–1 ms/system; ∼8.2 ms aggregate	Anton all-atom explicit-solvent MD; CHARMM22* with modified TIP3P; protein-specific temperatures	Folding kinetics
Shaw2010 [1]	Small proteins	FiP35: 2×100 μs; BPTI: 1 ms	Anton long-timescale all-atom explicit-solvent MD; system-specific Amber force fields, water models, and temperatures	ms dynamics
ATLAS [87]	Protein monomers	Main set: 1,390 chains; 4,170 trajectories; 3×100 ns/chain; 417 μs	GROMACS; CHARMM36m/TIP3P; 0.15 M NaCl; 300 K; three independent 100 ns replicas	Monomer dynamics
DynamicPDB [94]	Protein monomers	12,643 protein chains; 1 μs/chain; 1 ps output interval	OpenMM; Amber ff14SB/TIP3P; 0.15 M NaCl; 300 K; 1 fs; coordinates and physical properties every 1 ps	Dynamics pretraining
mdCATH [88]	CATH domains	5,398 domains; 134,950 trajectories; > 62 ms aggregate; mean 464 ns/trajectory	CHARMM22*/TIP3P; 0.15 M NaCl; 5 temperatures (320–450 K) × 5 replicas; coordinates and forces every 1 ns	Domain pretraining
ProteinConformers [95]	Protein conformers	87 CASP14/15 targets; 40,387 seed decoys; 381,546 MD-refined conformations	Short all-atom MD refinement of structural decoys; 125–375 ps unrestrained runs; snapshots every 25 ps	Landscape diversity
DynaRepo [93]	Proteins and macromolecular complexes	∼450 complexes + ∼270 single chains; mostly 3×500 ns/system; > 1.1 ms aggregate	Standard protein protocol: GROMACS 2024.2; CHARMM36m/TIP3P; 0.15 M NaCl; 310 K NPT; 2 fs; coordinates every 100 ps; nucleosome subset uses a distinct protocol	Curated MD reuse
MISATO [89]	Protein–ligand complexes	16,972 complexes; 10 ns simulated/complex; 8 ns retained (100 snapshots); ∼170 μs simulated	Amber20; ff14SB protein, GAFF2 ligand, TIP3P; 300 K NVT; first 2 ns discarded and final 8 ns retained	Pocket dynamics
Octapeptides [23]	8-residue peptides	1,100 peptides; 5 new 1 μs runs/system; 8.0 ms aggregate including original data	Amber ff99SB-ILDN; explicit TIP3P; 0.1 M NaCl; 300 K, 1 bar; 4 fs production timestep	Peptide ensembles
CATH1 [23]	Selected CATH domains	50 domains; 100 μs/domain in new simulations; 5.2 ms total including original data	Amber ff99SB-ILDN; explicit TIP3P; 300 K; adaptive sampling with 1–5 μs trajectories	Long domain dynamics
CATH2 [23]	Broad CATH domains	1,040 domains; ∼39 μs/domain; 41.0 ms aggregate	Amber ff99SB-ILDN; explicit TIP3P; 300 K; adaptive sampling with 1 μs trajectory segments	Broad pretraining
ManyPeptidesMD [96]	2–8 AA peptides	21,700 training peptides; 200 ns/system (4.34 ms aggregate); evaluation: 5 μs/system	OpenMM; Amber14/OBC1 implicit solvent; 310 K; 1 fs Langevin dynamics; positions/velocities every 1 ps (train) and 10 ps (validation/test)	Peptide samplers
GPCRmd[91,97]	GPCR monomers and complex	190 experimental structures; 371 systems (181 holo + 181 matched apo + 9 apo-only); 3×500 ns/system; 556.5 μs total	POPC/TIP3P; 0.15 M NaCl; CHARMM36 (July 2020), CGenFF/ParamChem for ligands; ACEMD3; 310 K NVT; 4 fs; three independent 500 ns replicas	GPCR signalling
MemProtMD [90,98]	Membrane proteins	2015 release: 2,294 structures identified, 2,219 completed 1 μs CGMD simulations; current database >8,000 structures	MARTINI coarse-grained membrane self-assembly; 100 ns initial insertion + 900 ns extension; final CG frame backmapped to atomistic resolution	Membrane dynamics
MDRepo [92]	Public protein MD	Initial imports: 2,798 ATLAS + 1,457 GPCRmd simulations; community repository continues to grow	No uniform simulation protocol; force fields, solvents, temperatures, and trajectory lengths are source-dependent; metadata retained per entry	Public MD reuse
DD-13M [61]	Ligand dissociation	565 complexes; 26,612 dissociation trajectories; 12.79M frames; median 21.8 ps/trajectory	Regular metadynamics; 50 parallel runs/complex with different seeds; 3D ligand COM coordinates (x, y, z) used as CVs; coordinates recorded every 0.1 ps	Unbinding paths

Note: Aggregate simulation times are rounded. Counts for evolving repositories such as ATLAS, DynaRepo, GPCRmd, MemProtMD, and MDRepo refer to publication or release snapshots and may increase over time. ProteinConformers is an MD-refined conformational benchmark rather than a conventional long-timescale equilibrium trajectory dataset. MD17/MD22 were originally developed primarily as energy/force-learning benchmarks but are also widely used for geometric trajectory prediction.

This diversity is useful for training, but complicates benchmarking. Force fields, solvent models, temperatures, protonation states, sampling schemes and saved frame intervals define different dynamical processes, so trajectories from different sources cannot always be treated as samples from the same underlying distribution. Large pooled datasets may consequently improve

structural coverage while blurring state populations or transition statistics. Dynamic benchmarks should therefore preserve simulation metadata, separate train and test sets by both molecular identity and dynamical event, and distinguish interpolation within sampled basins from transitions into previously unseen regions of conformational space. For kinetic evaluation in particular, reference trajectories must also be sufficiently long and statistically rich to support the time-dependent observables discussed in Section 6.2.

7. Discussion and Practical Guidance

The rapid development of molecular trajectory models has expanded the range of paths and conformational changes that can be proposed computationally. However, the interpretation of these outputs depends on what the model has learned, how its training data were generated, and which observables have been validated. Three distinctions are particularly important: an ordered sequence of plausible structures is not necessarily a calibrated dynamical trajectory; interpolation within the sampled data distribution is not evidence of generalization across the potential-energy landscape; and acceleration relative to molecular dynamics does not imply that the generated path can replace physics-based estimation of thermodynamic or kinetic quantities. We therefore discuss how kinetic evidence should be interpreted under finite sampling and heterogeneous simulation protocols, and provide practical guidance on how trajectory models can be combined with molecular dynamics in simulation and drug-discovery workflows.

7.1 What Constitutes Kinetic Evidence?

Kinetic interpretation requires more than a plausible ordering of molecular structures. Smooth transitions, realistic intermediates and agreement with known endpoint conformations can support the plausibility of a pathway, but they do not establish that successive frames correspond to a calibrated molecular timescale. This becomes consequential when generated trajectories are used to infer transition rates, residence times or relaxation processes, for which both temporal ordering and the mapping between generated steps and physical time must be supported by time-dependent observables.

The level of validation should therefore follow the intended use of the trajectory. For pathway discovery, recurrent intermediates or alternative transition routes may already provide useful hypotheses for subsequent simulation or experiment. Quantitative kinetic claims require stronger evidence, including consistency with the temporal observables discussed in Section 6.2 and, where relevant, a validated molecular clock. Generated trajectories can consequently remain useful even when this level of calibration has not been reached, provided that pathway hypotheses are distinguished from estimates of physical kinetics.

7.2 Finite Sampling and the Limits of Generalization

Finite sampling limits both what trajectory models can learn and how reliably their dynamics can be evaluated. Typical MD training data contain abundant local fluctuations but relatively few barrier-crossing events, leaving slow transitions weakly represented during learning. The same limitation applies to the reference trajectories used for validation: estimates of transition probabilities, implied timescales and dwell times become uncertain when metastable states are visited only a few times, and Chapman–Kolmogorov tests may be inconclusive when the underlying transition statistics are poorly sampled. A generated transition that is absent from the reference data is consequently difficult to interpret; it may reflect an unphysical extrapolation, but it may also correspond to a rare event missed by the finite simulation.

Kinetic benchmarks are therefore most informative for systems with long trajectories, repeated transitions or multiple independent replicas, such as well-sampled fast-folding systems and long-timescale Anton simulations[1,8]. The reference data should themselves support stable state populations and transition estimates across relevant lag times, with transition counts and uncertainty reported where possible to separate model error from sampling noise. For generated pathways that extend beyond events observed in the reference trajectories, independent MD, targeted sampling or experimental evidence is more informative than treating a finite simulation as an exhaustive ground truth.

7.3 Force-Field Heterogeneity and Protocol Dependence

Heterogeneous trajectory collections need not be avoided, but their role in model development should be defined more carefully than for static structural data. Mixing simulations generated under different physical models may be useful for learning transferable representations of molecular geometry and motion, yet the resulting model does not automatically inherit a single well-defined kinetic interpretation. This becomes particularly important for time-coarsened prediction, because the same nominal lag time can correspond to different transition statistics under different force fields, thermostats or sampling protocols. A model can therefore reproduce conformational diversity or MD-like paths within its training distribution without acquiring a transferable molecular dynamics. Evidence for simulator-like behaviour would require preservation of thermodynamic and kinetic observables on unseen systems, together with consistent responses to changes in temperature, solvent, ionic conditions

or molecular composition. Such validation remains limited, *leaving current trajectory models better viewed as proposal and acceleration tools than as general replacements for physics-based simulation.*

A practical strategy is therefore to separate broad representation learning from kinetic calibration. Diverse trajectories can support pretraining or structural generalization, followed by fine-tuning on data generated under a consistent physical regime when quantitative dynamics are required. If several simulation regimes are retained during training, force-field and protocol information can instead be provided explicitly as conditioning variables, with kinetic performance evaluated separately for each regime. Enhanced-sampling trajectories require additional caution because their temporal progression generally cannot be placed on the same molecular clock as unbiased MD without an appropriate correction or reweighting procedure. The physical meaning assigned to a generated trajectory should ultimately remain tied to the simulation regime from which its dynamical supervision was learned.

7.4 Drug Discovery: From Trajectory Proposal to Physical Validation

In AI-assisted drug discovery, molecular dynamics can serve as a structural and dynamical filter between computational design and experiment. Rather than relying on a predicted complex alone, MD can test whether the proposed binder remains associated with the receptor, whether key interactions persist, and how the surrounding protein adapts during simulation. NeoPep, for example, uses explicit-solvent MD to assess AI-designed peptide candidates[99], while AnewOmni evaluates generated biomolecular complexes through structural deviations and interaction persistence[100]. Similar workflows use MD-derived stability to examine AI-prioritized small molecules, as in 5F-MDL[101], with automated frameworks such as DynaMate aiming to scale these simulation and analysis steps[102]. In this setting, MD helps reduce a large computational candidate set to a smaller group with stronger physical support before downstream experimental characterization.

Trajectory AI models could add a dynamic screening stage before this physical validation. Rather than evaluating only a single predicted complex, they can propose alternative binding modes, induced-fit rearrangements, receptor conformational responses, transient intermediates, and ligand association or dissociation pathways. Such predictions are particularly useful when activity depends on conformational change or binding kinetics that cannot be inferred reliably from a static pose. Generated trajectories can then be used to rank dynamic hypotheses, select initial conformations for enhanced sampling, and identify contacts or collective motions that should be monitored. In this role, trajectory generation broadens the range of dynamical mechanisms considered during screening, while MD and experiment determine which proposals remain physically and functionally supported. As these models become more reliable, their speed could make such dynamic screening feasible across substantially larger candidate sets than can be examined routinely with classical MD alone.

7.5 When to Use AI and When to Use Molecular Dynamics

Classical MD remains the more appropriate choice when the objective is a quantitatively calibrated thermodynamic or kinetic observable under a specified physical model. Quantities such as binding free energy ΔG, equilibrium populations, diffusion coefficients and the dissociation rate constant k_{off} cannot generally be inferred from a plausible binding pose or transition pathway alone. Physics-based simulation is also preferable when predictions depend sensitively on solvent, ions, protonation, polarization or other conditions poorly represented in the training distribution. Its main constraint is computational cost: with a 2-fs timestep, a 100-ns trajectory requires approximately 5×10^7 integration steps and, for an explicit-solvent biomolecular system of roughly 10^5 atoms, can require on the order of hours on a modern GPU[103]. Direct sampling of slower rare events therefore remains difficult at screening scale.

Trajectory models offer a different trade-off by amortizing this cost across many predictions. Once trained, they can rapidly generate alternative conformations or pathways and thereby broaden the set of hypotheses considered before physical validation. MDGen, for example, generated a 100-ns-equivalent tetrapeptide trajectory in approximately 60 GPU-s, compared with about 3 GPU-h for the corresponding MD simulation[29]; such comparisons are system dependent but illustrate the potential gain in sampling throughput. In practice, the balance between AI and MD should follow the strength of the intended claim: rapid proposal and prioritization can rely more heavily on learned models, whereas estimates of thermodynamic or kinetic quantities require physics-based validation. Between these extremes, generated structures can seed MD or enhanced sampling, and simulation feedback can in turn refine or redirect subsequent generation. Table 3 summarizes these complementary modes of use.

8. Outlook: Toward Calibrated Molecular Dynamics Models

The next stage of molecular trajectory modelling will require progress along two related axes. Models must generate longer and more diverse trajectories without accumulating structural or dynamical errors, while also becoming more tightly coupled to physical simulation and experimental measurements. The objective is not simply to produce longer molecular movies, but to extend the range of systems and timescales for which generated paths support testable mechanistic or quantitative conclusions.

Table 3. Decision framework for choosing between AI trajectory models and traditional molecular simulation. Computational costs represent approximate orders of magnitude rather than standardized benchmarks.

Scientific objective	Primary bottleneck	Preferred workflow	Typical MD cost	AI role / cost
Rare-event pathway discovery[104,105]	Barrier crossing	AI → MD validation	μs–ms	Candidate paths (seconds–minutes)
Equilibrium conformational sampling[23]	Basin coverage	AI + MD	0.1 μs–ms	Rapid exploration (seconds–minutes)
Ligand or mutant screening[106,107]	Throughput	AI → MD/FEP	10–100 ns per candidate	Pre-screening (seconds–minutes per batch)
Binding free-energy estimation[106,107]	Thermodynamic convergence	MD/FEP	10–100 ns per transformation	Pose/state proposal; no direct ΔG
Residence-time estimation[105,108]	Rare-event kinetics	MD or enhanced sampling	μs–ms	Path proposal; no direct rate
Mechanistic interpretation[109,110]	Physical validation	AI → MD/experiment	0.1–10 μs	Hypothesis generation (seconds–minutes)

Notes: AI → MD denotes a sequential workflow in which AI-generated candidates are subsequently validated by MD, free-energy calculations, or experiments. AI + MD denotes iterative coupling in which MD results guide successive AI proposals. MD cost refers to aggregate simulated physical time. For reference, 100 ns of explicit-solvent all-atom MD for a system of approximately 10^5 atoms commonly requires wall-clock time on the order of 10 h on a single modern GPU[103]. AI costs exclude training and downstream physical validation.

8.1 Foundation Models and Closed-Loop Experimental Coupling

A long-term goal for molecular trajectory modelling is a foundation model that transfers across proteins, ligands, chemical environments and dynamical processes. Such transfer is unlikely to follow from model scaling or dataset aggregation alone, because trajectory data also encode the force fields, simulation protocols and regions of conformational space from which they were generated. A broadly transferable model will need to separate recurrent features of molecular dynamics from protocol-specific effects, recognize predictions that extend beyond the dynamical support of its training data, and account for changes in solvent, ions, temperature, pressure or membrane composition. Progress towards more general molecular simulators will therefore depend on treating these environmental variables explicitly or through physically calibrated conditioning, rather than tying the learned dynamics to a single simulation regime.

One route toward this broader transfer and calibration is to couple generation more closely to physical simulation. Proposed conformational jumps, transition intermediates or rare pathways can be examined by short MD simulations, enhanced sampling or energy evaluation, with the resulting information used to refine subsequent predictions. Experimental measurements provide complementary constraints at different structural and temporal resolutions, including time-resolved cryo-EM[111], single-molecule FRET[112], NMR relaxation[113–115], hydrogen–deuterium exchange[116], and kinetic binding measurements[117].

Simulation and experiment could in this way become conditioning signals or constraints within the generative process beyond their role as downstream validation tools. Each measurement provides an incomplete view of molecular motion, but their combination can restrict the range of admissible atomistic pathways and favour mechanistic hypotheses that remain compatible with observed dynamics.

8.2 Cross-Disciplinary Routes to Long-Horizon Generation

Long-horizon molecular trajectory generation shares an important challenge with long-form video and world models: small prediction errors can accumulate over repeated generation and progressively erode long-range consistency. Recent advances in these fields therefore offer several useful design principles for molecular trajectories, including long-context memory[118,119], rollout-aware generation[120–123], global–local temporal modelling[124,125], and context compression or sparse attention[126,127]. Adapted to molecular systems, these strategies could help retain information across extended trajectories, reduce rollout drift and preserve long-range temporal structure, while the additional requirements of thermodynamic and kinetic fidelity introduce constraints beyond temporal coherence alone.

Long-range consistency could also be combined with explicit guidance over which regions of path space are explored. Preference-based and physically guided generation, drawing on ideas from reinforcement learning from human feedback and direct preference optimization[128,129], has begun to appear in diffusion, flow-based and molecular generative models[24,130–133]. For molecular trajectories, such signals could encode stereochemical validity, energetic compatibility, preservation of key contacts, progress toward a rare event[58], committor-like scores or agreement with experimental restraints. Combined with long-context modelling, this guidance could stabilize extended rollouts while directing sampling toward ligand unbinding, receptor activation or folding transitions that are poorly represented in finite MD data. Because the guidance itself changes the generated path distribution, however, these trajectories should be interpreted as biased mechanism proposals unless they can be reweighted, refined through physics-based simulation or validated experimentally.

8.3 From Trajectory Generators to Perturbation-Aware Simulators

An important next step is to move beyond reproducing dynamics under a single fixed condition and ask how molecular behaviour changes under perturbation. In drug discovery and mechanistic studies, such perturbations include ligand substitution, mutation, protonation state and thermodynamic conditions, all of which can reshape metastable populations, transition pathways and kinetic observables. Models that capture these changes would offer a more informative notion of transfer than reproducing held-out trajectories from the same physical regime. Recent work on temperature-controllable generative sampling and transferable transition operators suggests that conditional dynamics of this kind may be tractable[53,134], although evaluation will need to focus on whether the induced changes in populations, rates and relaxation processes are reproduced correctly.

A second opportunity is to make accelerated trajectory generation statistically correctable. Many current generators can propose molecular paths efficiently, but their samples are not associated with an explicit correction to the path probability of a target dynamics, which limits quantitative interpretation. Accept–reject schemes, likelihood ratios and path-space reweighting offer possible routes to connect fast learned proposals with a defined physical ensemble. TimeWarp illustrates this idea through Metropolis correction of learned molecular transitions[25], while path-reweighting methods provide related machinery for biased dynamical ensembles[135]. Bringing such corrections into modern trajectory generators would make it possible to quantify, and in favourable settings reduce, the thermodynamic or kinetic bias introduced by accelerated generation.

9. Conclusions

Predictive and generative trajectory models are moving biomolecular modelling beyond static conformational ensembles toward learned dynamical processes. Across sequential predictors, trajectory-segment generators and latent-space models, their value lies in broadening the conformations and mechanisms that can be proposed and tested. The interpretation of these trajectories must nevertheless follow the strength of the evidence: structural plausibility and ensemble coverage can support exploration, whereas quantitative dynamics require validated transition statistics, thermodynamic consistency and a calibrated molecular clock. With such evidence still limited across molecular systems and physical conditions, current trajectory models are better regarded as learned surrogates of MD than as general molecular simulators; robust kinetic fidelity and transfer across environments remain central requirements for broader quantitative use.

Progress will depend on better characterized trajectory data, explicit treatment of simulation conditions and uncertainty, and closer coupling to physics-based simulation and experiment. These advances could make trajectory models more useful for prioritizing mechanisms, directing simulation effort and extending dynamic screening in drug discovery. The longer-term opportunity is to move from plausible molecular paths toward models whose generated trajectories remain transferable across conditions and can, where sufficiently validated, support quantitative dynamical inference.

References

1. Shaw, D. E. *et al.* Atomic-level characterization of the structural dynamics of proteins. *Science* **330**, 341–346 (2010).
2. Motlagh, H. N., Wrabl, J. O., Li, J. & Hilser, V. J. The ensemble nature of allostery. *Nature* **508**, 331–339 (2014).
3. Bhabha, G. *et al.* A dynamic knockout reveals that conformational fluctuations influence the chemical step of enzyme catalysis. *Science* **332**, 234–238 (2011).
4. Seo, M.-H., Park, J., Kim, E., Hohng, S. & Kim, H.-S. Protein conformational dynamics dictate the binding affinity for a ligand. *Nat. communications* **5**, 3724 (2014).
5. Karplus, M. & McCammon, J. A. Molecular dynamics simulations of biomolecules. *Nat. structural biology* **9**, 646–652 (2002).
6. Karplus, M. & Kuriyan, J. Molecular dynamics and protein function. *Proc. Natl. Acad. Sci.* **102**, 6679–6685 (2005).
7. Dror, R. O., Dirks, R. M., Grossman, J., Xu, H. & Shaw, D. E. Biomolecular simulation: a computational microscope for molecular biology. *Annu. review biophysics* **41**, 429–452 (2012).
8. Lindorff-Larsen, K., Piana, S., Dror, R. O. & Shaw, D. E. How fast-folding proteins fold. *Science* **334**, 517–520 (2011).
9. Jumper, J. *et al.* Highly accurate protein structure prediction with alphafold. *Nature* **596**, 583–589 (2021).
10. Abramson, J. *et al.* Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature* **630**, 493–500 (2024).
11. Stein, R. A. & Mchaourab, H. S. Speach_af: Sampling protein ensembles and conformational heterogeneity with alphafold2. *PLoS computational biology* **18**, e1010483 (2022).
12. Wayment-Steele, H. K. *et al.* Predicting multiple conformations via sequence clustering and alphafold2. *Nature* **625**, 832–839 (2024).

13. del Alamo, D., Sala, D., Mchaourab, H. S. & Meiler, J. Sampling alternative conformational states of transporters and receptors with alphafold2. *eLife* **11**, e75751, DOI: 10.7554/eLife.75751 (2022).

14. Latorraca, N. R., Venkatakrishnan, A. & Dror, R. O. Gpcr dynamics: structures in motion. *Chem. reviews* **117**, 139–155 (2017).

15. Hilger, D., Masureel, M. & Kobilka, B. K. Structure and dynamics of gpcr signaling complexes. *Nat. structural & molecular biology* **25**, 4–12 (2018).

16. Conflitti, P. *et al.* Functional dynamics of g protein-coupled receptors reveal new routes for drug discovery. *Nat. Rev. Drug Discov.* **24**, 251–275 (2025).

17. de Leon-Boenig, G. *et al.* The crystal structure of the catalytic domain of the nf-κb inducing kinase reveals a narrow but flexible active site. *Structure* **20**, 1704–1714 (2012).

18. Crawford, J. J. *et al.* Filling a nick in nik: Extending the half-life of a nik inhibitor through structure-based drug design. *Bioorganic & Medicinal Chem. Lett.* **89**, 129277 (2023).

19. He, X.-h. *et al.* Alphafold2 versus experimental structures: evaluation on g protein-coupled receptors. *Acta Pharmacol. Sinica* **44**, 1–7 (2023).

20. Torrens-Fontanals, M. *et al.* How do molecular dynamics data complement static structural data of gpcrs. *Int. journal molecular sciences* **21**, 5933 (2020).

21. Jing, B., Berger, B. & Jaakkola, T. Alphafold meets flow matching for generating protein ensembles. In *Proceedings of the 41st International Conference on Machine Learning*, 22277–22303 (2024).

22. Wang, Y. *et al.* Learning the all-atom equilibrium distribution of biomolecular interactions at scale. *bioRxiv* 2026–03 (2026).

23. Lewis, S. *et al.* Scalable emulation of protein equilibrium ensembles with generative deep learning. *Science* eadv9817 (2025).

24. Lu, J. *et al.* Aligning protein conformation ensemble generation with physical feedback. In *International Conference on Machine Learning*, 40436–40451 (PMLR, 2025).

25. Klein, L. *et al.* Timewarp: Transferable acceleration of molecular dynamics by learning time-coarsened dynamics. *Adv. Neural Inf. Process. Syst.* **36**, 52863–52883 (2023).

26. Shen, Y. *et al.* Simultaneous modeling of protein conformation and dynamics via autoregression. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems* (2025).

27. Shoghi, N. *et al.* Scalable spatio-temporal SE(3) diffusion for long-horizon protein dynamics. In *The Fourteenth International Conference on Learning Representations* (2026).

28. Shi, L. *et al.* Atomic trajectory modeling with state space models for biomolecular dynamics (2026). ArXiv preprint arXiv:2603.17633.

29. Jing, B., Stärk, H., Jaakkola, T. & Berger, B. Generative modeling of molecular dynamics trajectories. *Adv. Neural Inf. Process. Syst.* **37**, 40534–40564 (2024).

30. Feng, B., Zhang, J., Zhang, X., Liu, Z. & Li, Y. BioMD: All-atom generative model for biomolecular dynamics simulation. In *The Fourteenth International Conference on Learning Representations* (2026).

31. Zhang, Y., Tang, S. & Chatterjee, P. ScooBDoob: Schrödinger bridge with doob's h-transform for molecular dynamics. In *EurIPS 2025 Workshop on SIMBIOCHEM* (2025).

32. Sengar, A., Zhang, J., Vandergheynst, P. & BARTH, P. Beyond ensembles: Simulating all-atom protein dynamics in a learned latent space. In *The Fourteenth International Conference on Learning Representations* (2026).

33. Fan, Z., Zhu, J. & Chen, H.-F. Dynafold: A latent diffusion based generative framework for protein dynamic trajectory. *bioRxiv* 2025–09 (2025).

34. Zhang, J., Zhang, S., Vandergheynst, P. & BARTH, P. All-atom GPCR-ligand dynamics simulation via a residual isometric latent flow model. In *ICML 2026 AI for Science Workshop* (2026).

35. Adcock, S. A. & McCammon, J. A. Molecular dynamics: survey of methods for simulating the activity of proteins. *Chem. reviews* **106**, 1589–1615 (2006).

36. Hollingsworth, S. A. & Dror, R. O. Molecular dynamics simulation for all. *Neuron* **99**, 1129–1143 (2018).

37. Tang, H. *et al.* Learning structure, energy, and dynamics: A survey of artificial intelligence for protein dynamics. *arXiv preprint arXiv:2604.25244* (2026).

38. Olsson, S. Generative molecular dynamics. *Curr. Opin. Struct. Biol.* **96**, 103213 (2026).

39. Dao, T. M. & Rahman, T. Deep generative modeling of protein conformations: A comprehensive review. *BioChem* **5**, 32 (2025).

40. Prinz, J.-H. *et al.* Markov models of molecular kinetics: Generation and validation. *The J. Chem. Phys.* **134**, 174105, DOI: 10.1063/1.3565032 (2011).

41. Husic, B. E. & Pande, V. S. Markov state models: From an art to a science. *J. Am. Chem. Soc.* **140**, 2386–2396, DOI: 10.1021/jacs.7b12191 (2018).

42. Murtada, M. H., Brotzakis, Z. F. & Vendruscolo, M. Md-llm-1: A large language model for molecular dynamics (2025). ArXiv preprint arXiv:2508.03709.

43. Costa, A. d. S. *et al.* Equijump: Protein dynamics simulation via so (3)-equivariant stochastic interpolants. *arXiv preprint arXiv:2410.09667* (2024).

44. Xu, Y., Wang, D., Zhou, Z., Yu, T. & Chen, M. TEMPO: Temporal multi-scale autoregressive generation of protein conformational ensembles. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems* (2025).

45. Kapusniak, K., Gabellini, C., Bronstein, M. M., Tossou, P. & Di Giovanni, F. Mars-fm: Generative modeling of molecular dynamics via markov state models. In *The Fourteenth International Conference on Learning Representations* (2026).

46. Thiemann, F. L. *et al.* Force-free molecular dynamics through autoregressive equivariant networks. *Nat. Mach. Intell.* 1–13 (2026).

47. Wu, F. & Li, S. Z. Diffmd: A geometric diffusion model for molecular dynamics simulations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, 5321–5329 (2023).

48. Hong, H., Lin, W., Chusong, Z. & Tan, K. Geometric graph neural diffusion for stable molecular dynamics simulations. In *The Fourteenth International Conference on Learning Representations* (2026).

49. Kovács, D. P. *et al.* Linear atomic cluster expansion force fields for organic molecules: beyond rmse. *J. chemical theory computation* **17**, 7696–7711 (2021).

50. Kim, G. *et al.* Benchmark of machine learning force fields for semiconductor simulations: datasets, metrics, and comparative analysis. *Adv. Neural Inf. Process. Syst.* **36**, 51434–51476 (2023).

51. Liu, S. *et al.* A multi-grained symmetric differential equation model for learning protein-ligand binding dynamics. *Nat. Commun.* (2025).

52. Schreiner, M., Winther, O. & Olsson, S. Implicit transfer operator learning: Multiple time-resolution models for molecular dynamics. *Adv. Neural Inf. Process. Syst.* **36**, 36449–36462 (2023).

53. Antoniadis, P., Pavesi, B., Olsson, S. & Winther, O. Protein language model embeddings improve generalization of implicit transfer operators. In *Forty-third International Conference on Machine Learning* (2026).

54. Li, S. *et al.* F^3low: Frame-to-frame coarse-grained molecular dynamics with SE(3) guided flow matching. In *ICLR 2024 Workshop on Generative and Experimental Perspectives for Biomolecular Design* (2024).

55. Costa, A. d. S., Ponnapati, M., Rubin, D., Smidt, T. & Jacobson, J. Accelerating protein molecular dynamics simulation with deepjump (2025). ArXiv preprint arXiv:2509.13294.

56. Yu, Z., Huang, W. & Liu, Y. Unisim: A unified simulator for time-coarsened dynamics of biomolecules. In *International Conference on Machine Learning*, 73097–73114 (PMLR, 2025).

57. Yu, Z., Huang, W. & Liu, Y. Force-guided bridge matching for full-atom time-coarsened dynamics of peptides (2025).

58. Yu, Z., Huang, W. & Liu, Y. Unified biomolecular trajectory generation via pretrained variational bridge. In *The Fourteenth International Conference on Learning Representations* (2026).

59. Hu, W. *et al.* 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).

60. Smith, J. S. *et al.* The ani-1ccx and ani-1x data sets, coupled-cluster and density functional theory properties for molecules. *Sci. data* **7**, 134 (2020).

61. Li, M. *et al.* A novel 4-d dataset paradigm for studying complete ligand-protein dissociation dynamics. *arXiv preprint arXiv:2504.18367* (2025).

62. Petersen, M., Roig, G. & Covino, R. Dynamicsdiffusion: Generating and rare event sampling of molecular dynamic trajectories using diffusion models. In *NeurIPS 2023 AI for Science Workshop* (2023).

63. Xu, K. *et al.* Efficient generation of protein and protein–protein complex dynamics via se (3)-parameterized diffusion models. *J. Chem. Inf. Model.* **65**, 12366–12376 (2025).

64. Cheng, K. *et al.* 4d diffusion for dynamic protein structure prediction with reference and motion guidance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, 93–101 (2025).

65. Han, J., Xu, M., Lou, A., Ye, H. & Ermon, S. Geometric trajectory diffusion models. *Adv. Neural Inf. Process. Syst.* **37**, 25628–25662 (2024).

66. Iyengar, A. *et al.* Align your structures: Generating trajectories with structure pretraining for molecular dynamics. In *The Fourteenth International Conference on Learning Representations* (2026).

67. ten Brinke, K. B., Minartz, K. & Menkovski, V. STFlow: Data-coupled flow matching for geometric trajectory simulation. In *Forty-third International Conference on Machine Learning* (2026).

68. Phipps, H., Cagiada, M., Villalba, S. D. & Deane, C. Spectral diffusion for protein dynamics. In *The 2026 Workshop on Generative and Agentic AI for Biology* (2026).

69. Wang, X., Zhou, X., Duan, X. & He, J. SE(3)-equivariant flow matching with gaussian process priors for geometric trajectory prediction. In *Forty-third International Conference on Machine Learning* (2026).

70. Wu, J., Liu, Y., Yu, R. & Sun, J. Flow for future: Geometric SE(3)-equivariant flow matching for 3d trajectory prediction. In *Forty-third International Conference on Machine Learning* (2026).

71. Feng, B. *et al.* Physically grounded generative modeling of all-atom biomolecular dynamics. *bioRxiv* 2026–02 (2026).

72. Holdijk, L. *et al.* Stochastic optimal control for collective variable free sampling of molecular transition paths. In *Thirty-seventh Conference on Neural Information Processing Systems* (2023).

73. Du, Y. *et al.* Doob's lagrangian: A sample-efficient variational approach to transition path sampling. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems* (2024).

74. Seong, K., Park, S., Kim, S., Kim, W. Y. & Ahn, S. Transition path sampling with improved off-policy training of diffusion path samplers. In *The Thirteenth International Conference on Learning Representations* (2025).

75. Xu, K. *et al.* Tps-flow: Physics-guided flow-based generative modeling of protein transition paths. *J. Chem. Inf. Model.* (2026).

76. Pang, Y. T., Yang, L., Kuo, K. M. & Gumbart, J. C. Deeppath: Overcoming data scarcity for protein transition pathway prediction using physics-based deep learning. *Chem. Sci.* **17**, 12055–12073 (2026).

77. Wu, H., Mardt, A., Pasquali, L. & Noe, F. Deep generative markov state models. *Adv. Neural Inf. Process. Syst.* **31** (2018).

78. Sidky, H., Chen, W. & Ferguson, A. L. Molecular latent space simulators. *Chem. Sci.* **11**, 9459–9467 (2020).

79. Schütt, K. T., Sauceda, H. E., Kindermans, P.-J., Tkatchenko, A. & Müller, K.-R. Schnet–a deep learning architecture for molecules and materials. *The J. Chem. Phys.* **148**, 241722 (2018).

80. Batzner, S. *et al.* $e(3)$-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nat. Commun.* **13**, 2453, DOI: 10.1038/s41467-022-29939-5 (2022).

81. Batatia, I., Kovács, D. P., Simm, G. N. C., Ortner, C. & Csányi, G. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. In *Advances in Neural Information Processing Systems 35* (2022).

82. Lipman, Y., Chen, R. T. Q., Ben-Hamu, H., Nickel, M. & Le, M. Flow matching for generative modeling. In *ICLR* (2023).

83. Wu, H. *et al.* Variational koopman models: Slow collective variables and molecular kinetics from short off-equilibrium simulations. *The J. chemical physics* **146** (2017).

84. Noé, F., Tkatchenko, A., Müller, K.-R. & Clementi, C. Machine learning for molecular simulation. *Annu. Rev. Phys. Chem.* **71**, 361–390, DOI: 10.1146/annurev-physchem-042018-052331 (2020).

85. Chmiela, S. *et al.* Machine learning of accurate energy-conserving molecular force fields. *Sci. advances* **3**, e1603015 (2017).

86. Chmiela, S. *et al.* Accurate global machine learning force fields for molecules with hundreds of atoms. *Sci. Adv.* **9**, eadf0873 (2023).

87. Vander Meersche, Y., Cretin, G., Gheeraert, A., Gelly, J.-C. & Galochkina, T. Atlas: protein flexibility description from atomistic molecular dynamics simulations. *Nucleic acids research* **52**, D384–D392 (2024).

88. Mirarchi, A., Giorgino, T. & De Fabritiis, G. mdcath: A large-scale md dataset for data-driven computational biophysics. *Sci. Data* **11**, 1299 (2024).

89. Siebenmorgen, T. *et al.* Misato: machine learning dataset of protein–ligand complexes for structure-based drug discovery. *Nat. Comput. Sci.* 1–12 (2024).

90. Newport, T. D., Sansom, M. S. P. & Stansfeld, P. J. The memprotmd database: a resource for membrane-embedded protein structures and their lipid interactions. *Nucleic acids research* **47**, D390–D397 (2019).

91. Rodríguez-Espigares, I. *et al.* Gpcrmd uncovers the dynamics of the 3d-gpcrome. *Nat. Methods* **17**, 777–787 (2020).

92. Roy, A. *et al.* Mdrepo-an open data warehouse for community-contributed molecular dynamics simulations of proteins. *Nucleic Acids Res.* **53**, D477–D486 (2025).

93. Mokhtari, O., Bignon, E., Khakzad, H. & Karami, Y. Dynarepo: the repository of macromolecular conformational dynamics. *Nucleic Acids Res.* **54**, D393–D401 (2026).

94. Liu, C. *et al.* Dynamic pdb: A new dataset and a se(3) model extension by integrating dynamic behaviors and physical properties in protein structures (2024). ArXiv preprint arXiv:2408.12413.

95. Zhou, Y. *et al.* Proteinconformers: Benchmark dataset for simulating protein conformational landscape diversity and plausibility. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track* (2025).

96. Tan, C. B. *et al.* Amortized sampling with transferable normalizing flows. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems* (2025).

97. Aranda-García, D. *et al.* Large scale investigation of gpcr molecular dynamics data uncovers allosteric sites and lateral gateways. *Nat. communications* **16**, 2020 (2025).

98. Stansfeld, P. J. *et al.* Memprotmd: Automated insertion of membrane protein structures into explicit lipid membranes. *Structure* **23**, 1350–1361, DOI: 10.1016/j.str.2015.05.006 (2015).

99. Li, M. *et al.* Accurate de novo design of peptides from programming biophysical landscape. *bioRxiv* 2026–06 (2026).

100. Kong, X. *et al.* Programming biomolecular interactions with all-atom generative model. *bioRxiv* 2026–03 (2026).

101. Zhu, H.-Y., Du, S.-J., Xu, L. & Shi, W. Drug discovery guided by maximum drug likeness (2025). 2512.21895.

102. Guilbert, S., Masschelein, C., Goumaz, J., Naida, B. & Schwaller, P. Dynamate: An autonomous agent for protein-ligand molecular dynamics simulations (2025). 2512.10034.

103. Eastman, P. *et al.* Openmm 8: molecular dynamics simulation with machine learning potentials. *The J. Phys. Chem. B* **128**, 109–116 (2023).

104. Laio, A. & Parrinello, M. Escaping free-energy minima. *Proc. national academy sciences* **99**, 12562–12566 (2002).

105. Tiwary, P., Limongelli, V., Salvalaglio, M. & Parrinello, M. Kinetics of protein–ligand unbinding: Predicting pathways, rates, and rate-limiting steps. *Proc. Natl. Acad. Sci.* **112**, E386–E391, DOI: 10.1073/pnas.1424461112 (2015).

106. Wang, L. *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. *J. Am. Chem. Soc.* **137**, 2695–2703, DOI: 10.1021/ja512751q (2015).

107. Schindler, C. E. M. *et al.* Large-scale assessment of binding free energy calculations in active drug discovery projects. *J. Chem. Inf. Model.* **60**, 5457–5474, DOI: 10.1021/acs.jcim.0c00900 (2020).

108. Wolf, S., Lickert, B., Bray, S. & Stock, G. Multisecond ligand dissociation dynamics from atomistic simulations. *Nat. Commun.* **11**, 2918, DOI: 10.1038/s41467-020-16655-1 (2020).

109. Dror, R. O. *et al.* Pathway and mechanism of drug binding to g-protein-coupled receptors. *Proc. Natl. Acad. Sci.* **108**, 13118–13123, DOI: 10.1073/pnas.1104614108 (2011).

110. Dror, R. O. *et al.* Activation mechanism of the β_2-adrenergic receptor. *Proc. Natl. Acad. Sci.* **108**, 18684–18689, DOI: 10.1073/pnas.1110499108 (2011).

111. Harder, O. F., Barrass, S. V., Drabbels, M. & Lorenz, U. J. Fast viral dynamics revealed by microsecond time-resolved cryo-em. *nature communications* **14**, 5649 (2023).

112. Roy, R., Hohng, S. & Ha, T. A practical guide to single-molecule fret. *Nat. methods* **5**, 507–516 (2008).

113. Agback, T. *et al.* Combined nmr and molecular dynamics conformational filter identifies unambiguously dynamic ensembles of dengue protease ns2b/ns3pro. *Commun. biology* **6**, 1193 (2023).

114. Wieczorek, M. *et al.* Mhc class ii complexes sample intermediate states along the peptide exchange pathway. *Nat. communications* **7**, 13224 (2016).

115. Wayment-Steele, H. K. *et al.* Learning millisecond protein dynamics from what is missing in NMR spectra. *Nature* DOI: 10.1038/s41586-026-10989-4 (2026).

116. Zheng, J., Strutzenberg, T., Pascal, B. D. & Griffin, P. R. Protein dynamics and conformational changes explored by hydrogen/deuterium exchange mass spectrometry. *Curr. opinion structural biology* **58**, 305–313 (2019).

117. Hojjat Jodaylami, M., Masson, J.-F. & Badia, A. Surface plasmon resonance sensing. *Nat. Rev. Methods Primers* **5**, 47 (2025).

118. Po, R. *et al.* Long-context state-space video world models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2025).

119. Chen, S. *et al.* Context forcing: Consistent autoregressive video generation with long context. In *Forty-third International Conference on Machine Learning* (2026).

120. Chen, B. *et al.* Diffusion forcing: Next-token prediction meets full-sequence diffusion. In *Advances in Neural Information Processing Systems* (2024).

121. Huang, X., Li, Z., He, G., Zhou, M. & Shechtman, E. Self forcing: Bridging the train-test gap in autoregressive video diffusion. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems* (2025).

122. Huang, J. *et al.* LIVE: Long-horizon interactive video world modeling. In *Forty-third International Conference on Machine Learning* (2026).

123. Yi, J. *et al.* Deep forcing: Training-free long video generation with deep sink and participative compression. In *Forty-third International Conference on Machine Learning* (2026).

124. Ma, X. *et al.* Flex-forcing: Towards a unified autoregressive and bidirectional video diffusion model. In *Forty-third International Conference on Machine Learning* (2026).

125. Zhao, C. *et al.* Moviedreamer: Hierarchical generation for coherent long visual sequences. In *The Thirteenth International Conference on Learning Representations* (2025).

126. Samuel, D. *et al.* FAST-AR: Fast autoregressive video diffusion and world models with temporal cache compression and sparse attention. In *Forty-third International Conference on Machine Learning* (2026).

127. Lv, C. *et al.* Light forcing: Accelerating autoregressive video diffusion via sparse attention. In *Forty-third International Conference on Machine Learning* (2026).

128. Ouyang, L. *et al.* Training language models to follow instructions with human feedback. *Adv. neural information processing systems* **35**, 27730–27744 (2022).

129. Rafailov, R. *et al.* Direct preference optimization: Your language model is secretly a reward model. *Adv. neural information processing systems* **36**, 53728–53741 (2023).

130. Wallace, B. *et al.* Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8228–8238 (2024).

131. Jiao, R., Kong, X., Huang, W. & Liu, Y. 3d structure prediction of atomic systems with flow-based direct preference optimization. *Adv. Neural Inf. Process. Syst.* **37**, 110197–110217 (2024).

132. Gu, S. *et al.* Aligning target-aware molecule diffusion models with exact energy optimization. *Adv. Neural Inf. Process. Syst.* **37**, 44040–44063 (2024).

133. Sun, Y. *et al.* Epo: Diverse and realistic protein ensemble generation via energy preference optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, 1060–1068 (2026).

134. Moqvist, S., Chen, W., Schreiner, M., N"uske, F. & Olsson, S. Thermodynamic interpolation: A generative approach to molecular thermodynamics and kinetics. *J. Chem. Theory Comput.* **21**, 2535–2545 (2025).

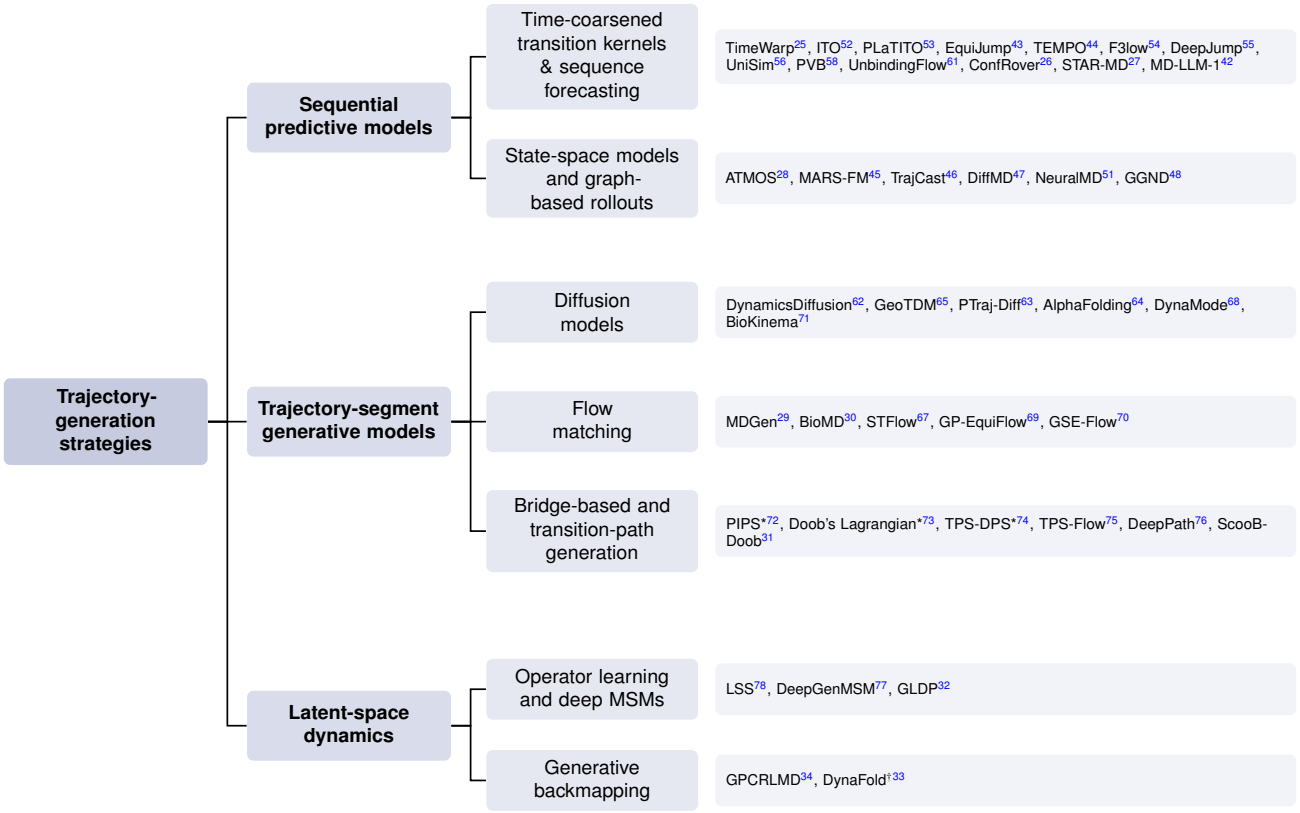
135. Donati, L., Hartmann, C. & Keller, B. G. Girsanov reweighting for path ensembles and markov state models. *The J. Chem. Phys.* **146**, 244112 (2017).

136. Song, Y. *et al.* Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations* (2021).

Acknowledgements

This work was supported by the EPFL AI Center PhD Fellowship Program 2026, a Swiss National Science Foundation grant (SNSF grant 31003A_182263), Swiss Cancer Research (KFS-4687-02-2019), funds from EPFL, and the Ludwig Institute for Cancer Research; Shenzhen Hetao Shenzhen-Hong Kong Science and Technology Innovation Cooperation Zone, under Grant No. HTHZQSWS-KCCYB-2023052.

Supplementary Material



Supplementary Figure 1. Taxonomy of trajectory-generation strategies and representative methods. Methods follow the organization used in the main text and Supplementary Table 1. Asterisks (*) denote path-based methods whose path-level formulation does not necessarily imply parallel frame generation; daggers (†) denote methods that combine mechanisms from more than one category and are placed according to their primary role.

S.1 More Details of the related work

S.1.1 Time-Coarsened Transition-Kernel-Methods

Time-coarsened transition models learn a stochastic propagator $p_\theta(x_{t+\tau} \mid x_t, c)$ at a lag time τ that is substantially larger than the numerical integration step used in classical molecular dynamics. Repeated application of this kernel produces an autoregressive trajectory and, in principle, replaces many short force-based updates with a single learned transition. TimeWarp combines a conditional normalizing-flow proposal with a Metropolis–Hastings correction to improve equilibrium consistency[25]. Meanwhile, EquiJump learns equivariant coarse-time transitions through stochastic interpolation[43]. TEMPO explicitly separates slow and fast molecular modes within a time-coarsened autoregressive formulation[44]. UnbindingFlow uses flow matching for predicting dissociation pathways[61] frame by frame. Other diffusion- and flow-based models such as ITO[52], PLaTITO[53], F3low[54] and DeepJump[55], also belong to this family when their primary output is a future state or finite-lag transition rather than a coupled multi-frame segment. In particular, Implicit Transfer Operator (ITO) learning parameterizes conditional transition densities at multiple lag times using equivariant diffusion models[52]. PLaTITO augments ITO with protein language-model embeddings to improve cross-protein generalization while retaining finite-lag transition prediction and iterative rollout[53]. F3low similarly learns a frame-to-frame transition using guided flow matching and constructs coarse-grained trajectories through autoregressive sampling conditioned on the preceding frame[54]. DeepJump[55] uses conditional flow matching to learn protein conformational transitions over multiple jump times, while UniSim[56] uses stochastic interpolants with force guidance to learn transferable time-coarsened state transitions across molecular domains.

S.1.2 Diffusion Models for Trajectory Segments

Trajectory diffusion models treat multiple temporally ordered conformations as a coupled generative object. Dynamics diffusion provides an early example by applying denoising diffusion directly to molecular trajectory segments, enabling both unconditional

trajectory generation and rare-event sampling[62]. PTraj-Diff applies $SE(3)$-parameterized diffusion to protein and protein–protein complex trajectories[63]; GeoTDM learns temporal distributions over geometric trajectories using equivariant diffusion kernels and temporal attention[65]; and AlphaFolding uses four-dimensional diffusion to generate multi-step all-atom protein trajectories with reference and motion guidance[64]. BioKinema extends spatiotemporal diffusion to continuous-time all-atom biomolecular trajectories and combines coarse forecasting with interpolation for long-horizon generation[71]. EGInterpolator combines diffusion-based structure pretraining with a temporal interpolation module trained on MD trajectories, supporting unconditional generation, forward simulation, and endpoint-conditioned trajectory interpolation across molecular systems [66]. DynaMode takes a complementary spectral approach, applying diffusion to discrete-cosine-transform representations of finite protein trajectories and inverse-transforming generated spectral volumes into temporally ordered conformations[68]. This representation provides an inductive bias for separating slow conformational modes from faster local fluctuations. Overall, these methods differ from state-transition diffusion models in that the diffusion variable contains multiple physical frames, allowing temporal correlations within a finite-horizon trajectory segment to be modelled jointly.

S.1.3 Flow Matching over Trajectory Segments

Flow matching provides a simulation-free objective for learning deterministic, time-dependent vector fields that transport a simple prior distribution to the distribution of molecular trajectory segments[82]. At inference, samples are generated by integrating the learned field with an ODE solver, potentially requiring fewer neural-network evaluations than iterative stochastic denoising and thereby enabling efficient trajectory-segment generation. MDGen provides an early general formulation of this approach by representing an entire trajectory segment as a spatiotemporal array and generating it using a stochastic-interpolant flow model[29]. BioMD extends segment-level flow matching to all-atom protein–ligand trajectories through hierarchical forecasting and interpolation[30]. These methods illustrate the extension of trajectory-level flow matching from general protein dynamics toward all-atom biomolecular complexes and longer-horizon generation. More recent methods further incorporate trajectory-aware priors, temporal structure, and geometric symmetries into the generative transport. STFlow introduces data-coupled flow matching with conditioned random-walk priors and spatiotemporal graph architectures to model geometric trajectory distributions[67]. GP-EquiFlow constructs $SE(3)$-equivariant, temporally correlated priors from observed trajectories using vector-valued Gaussian processes[69], whereas GSE-Flow combines $SE(3)$-equivariant flow matching with history-aware spatiotemporal representations to jointly generate future trajectory segments[70]. In each case, multiple frames are coupled within a generated segment, although longer trajectories may still be constructed through autoregressive extension across successive segments.

S.1.4 Bridge-Based and Transition-Path Generation

Bridge-based and transition-path models specialize trajectory-segment generation $p(X_{t+1:t+L-1} \mid x_t, x_{t+L}, c)$ to pathways connecting prescribed molecular states. PIPS connects transition-path sampling with Schrödinger bridges and stochastic optimal control, learning collective-variable-free neural bias policies for sampling rare molecular transitions[72]. Doob's Lagrangian learns a variational approximation to the Doob h-transform for endpoint-conditioned dynamics[73], and TPS-DPS formulates transition-path generation as amortized sampling from a target path distribution using off-policy diffusion-path training [74]. ScooBDoob combines Schrödinger bridges with Doob's h-transform in an MSM-based state space [31]. TPS-Flow uses physics-guided, endpoint-conditioned flow matching to generate multi-frame conformational paths between predefined protein states [75], and DeepPath uses progressive adversarial learning to predict intermediate conformations connecting known endpoint structures[76].

These methods share a path-level objective but differ in how paths are sampled. PIPS, Doob's Lagrangian, and TPS-DPS approximate conditioned path measures through controlled stochastic dynamics and therefore generate paths sequentially. TPS-Flow and DeepPath generate multiple intermediate conformations directly, and ScooBDoob operates on coarse metastable-state paths. Their common feature is that the transition path itself is the primary generative object.

S.1.5 Latent-Space Dynamics and Spatiotemporal Coarse-Graining

GLDP explicitly compares transition operators in a learned latent space[32], and MARS-FM places generative modelling on top of MSM-defined states[45]. GPCRLMD[34] generates latent trajectory segments for GPCR–ligand complexes with flow matching and decodes the latent frames to all-atom conformations; DynaFold generates latent trajectories by diffusion before atomistic reconstruction[33]; GLDP decodes learned latent dynamics to all-atom coordinates[32]; PVB learns variational bridges over latent and coordinate representations[58]; and MD-LLM-1 reconstructs dynamics from tokenized molecular states[42].

S.2 Sampling Efficiency Comparison

Because reported wall-clock speedups are difficult to compare across molecular systems, hardware, trajectory strides, and post-processing protocols, we summarize sampling efficiency in Supplementary Table 1 using architecture-level cost proxies. Specifically, we distinguish frame-to-frame generation from joint generation of trajectory blocks, indicate whether autoregressive rollout or iterative diffusion/flow sampling is required, and report the resulting scaling with trajectory length. Here, n denotes the

total number of generated frames, b the number generated jointly in one block, and K the number of diffusion- or flow-sampling iterations.

The comparison highlights a basic trade-off between sequential generation and temporal parallelism. Frame-wise predictors typically scale as $\mathcal{O}(n)$ or $\mathcal{O}(nK)$, while trajectory-segment generators can amortize iterative sampling across b jointly generated frames, giving an approximate cost of $\mathcal{O}(nK/b)$. Path-level modelling does not, however, imply parallel frame generation: several bridge-based methods define distributions over complete paths but still sample them sequentially. These proxies are therefore intended to compare generation strategies rather than absolute runtime, which also depends strongly on model size, molecular system, hardware, and any downstream refinement or correction.

S.3 Details of MD Datasets

Current trajectory resources cover several distinct regimes rather than a single benchmark setting. Small-molecule datasets such as MD17 and MD22 provide high-quality quantum or near-quantum trajectories and remain useful for force learning and geometric trajectory prediction, but their limited system size and timescale offer little access to rare biomolecular transitions[85,86]. Protein monomer and peptide datasets span a broader range of conformational dynamics. ATLAS provides standardized all-atom trajectories across representative proteins[87], while DynamicPDB and mdCATH increase the scale and diversity of available simulations[88,94]. Dense peptide resources such as ManyPeptidesMD support large-scale sampler training[96], whereas the DESRES fast-folding set and long Anton trajectories remain particularly useful for kinetic evaluation because they contain repeated transitions or unusually long dynamical events[1,8]. ProteinConformers serves a different purpose, emphasizing coverage of conformational landscapes rather than long equilibrium trajectories[95].

Protein–ligand and membrane systems add binding, dissociation, and environment-dependent dynamics that are largely absent from monomer benchmarks. MISATO provides simulated protein–ligand complexes[89], DD-13M concentrates on ligand-dissociation pathways[61], and GPCRmd and MemProtMD extend trajectory resources to membrane proteins and receptor signalling[90,91,98]. Cross-system repositories such as MDRepo and DynaRepo complement these curated datasets by organizing trajectories generated under a wider range of simulation protocols[92,93]. Table 2 summarizes their molecular scope, scale and simulation conditions.

S.4 Validation Coverage of Representative Molecular Trajectory Generation Models

Supplementary Table 2 summarizes the validation coverage of representative trajectory models across four levels: structural fidelity, thermodynamic fidelity, kinetic fidelity, and application-level validation. Structural evaluation includes geometric and conformational metrics such as RMSD, distance distributions, and contact recovery; thermodynamic evaluation considers ensemble populations, free-energy landscapes, or related equilibrium observables; and kinetic evaluation requires time-dependent quantities such as transition probabilities, implied timescales, first-passage times, or autocorrelation functions. Application-level validation records whether the generated trajectories were further examined in biologically relevant processes such as folding, conformational transitions, ligand binding, or unbinding.

The comparison reveals an uneven validation landscape. Structural fidelity is evaluated by most methods, and thermodynamic tests are increasingly common, but explicit kinetic validation remains substantially less widespread. Importantly, the table reports *validation coverage* rather than model performance: a filled circle indicates that a quantitative test was performed, not that the corresponding method necessarily achieved better agreement with reference dynamics.

S.5 Formulation of the Trajectory Generative models

S.5.1 Trajectory Segment Generation with Diffusion Model

Throughout this section and sec S.5.2, t indexes physical trajectory frames, whereas $s \in [0,1]$ denotes an artificial generative time step used by diffusion or flow models.

Let $X = X_{t+1:t+L}$ denote a reference trajectory segment. In the continuous-time formulation of score-based diffusion, a forward stochastic differential equation gradually transforms the trajectory distribution into a tractable prior[136]:

$$\mathrm{d}Y_s = f(Y_s, s)\,\mathrm{d}s + g(s)\,\mathrm{d}W_s, \qquad Y_0 = X, \quad s : 0 \to 1, \tag{1}$$

where f and g denote the drift and diffusion coefficients, respectively, W_s is a standard Wiener process, and Y_s is the evolving spatiotemporal tensor. The coefficients are chosen such that the terminal distribution $p_1(Y)$ is close to a simple prior, typically a Gaussian distribution. Here, Y_s retains the full spatiotemporal dimensions of the trajectory segment, so the diffusion process perturbs the segment as a coupled random object rather than independently generating one physical frame at a time.

The corresponding reverse-time process is also an SDE:

$$\mathrm{d}Y_s = \left[f(Y_s, s) - g^2(s)\nabla_{Y_s} \log p_s(Y_s \mid c)\right]\mathrm{d}s + g(s)\,\mathrm{d}\overline{W}_s, \qquad s : 1 \to 0. \tag{2}$$

Supplementary Table 1. Qualitative comparison of trajectory-generation strategies. Methods are grouped according to the taxonomy in Fig. 3. Here, n denotes the total number of generated trajectory frames, b denotes the number of frames generated jointly in one block, and K denotes the number of diffusion- or flow-sampling iterations. Method references follow Table 1.

Model	Frame-to-frame	Batch frames	Autoregressive	Diffusion/flow iterations	Scaling with trajectory length
Sequential predictive models					
Time-coarsened transition kernels & sequence forecasting					
TimeWarp[25]	✓	–	–	–	$\sim \mathcal{O}(n)$
ITO[52]	✓	–	–	✓	$\sim \mathcal{O}(nK)$
PLaTITO[53]	✓	–	✓	✓	$\sim \mathcal{O}(nK)$
EquiJump[43]	✓	–	–	–	$\sim \mathcal{O}(n)$
TEMPO[44]	✓	–	✓	–	$\sim \mathcal{O}(n)$
F3low[54]	✓	–	✓	✓	$\sim \mathcal{O}(nK)$
DeepJump[55]	✓	–	✓	✓	$\sim \mathcal{O}(nK)$
UniSim[56]	✓	–	–	✓	$\sim \mathcal{O}(nK)$
PVB[58]	✓	–	✓	✓	$\sim \mathcal{O}(nK)$
UnbindingFlow[61]	✓	–	✓	✓	$\sim \mathcal{O}(nK)$
ConfRover[26]	✓	–	✓	✓	$\sim \mathcal{O}(nK)$
STAR-MD[27]	✓	–	✓	✓	$\sim \mathcal{O}(nK)$
MD-LLM-1[42]	✓	–	✓	–	$\sim \mathcal{O}(n)$
State-space models and graph-based rollouts					
ATMOS[28]	✓	–	✓	✓	$\sim \mathcal{O}(nK)$
MARS-FM[45]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
TrajCast[46]	✓	–	✓	–	$\sim \mathcal{O}(n)$
DiffMD[47]	✓	–	–	✓	$\sim \mathcal{O}(nK)$
NeuralMD[51]	✓	–	–	–	$\sim \mathcal{O}(n)$
GGND[48]	✓	–	–	✓	$\sim \mathcal{O}(nK)$
Trajectory-segment generative models					
Diffusion models					
DynamicsDiffusion[62]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
GeoTDM[65]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
PTraj-Diff[63]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
AlphaFolding[64]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
DynaMode[68]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
BioKinema[71]	–	✓	✓	✓	$\sim \mathcal{O}(nK/b)$
Flow matching					
MDGen[29]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
BioMD[30]	–	✓	✓	✓	$\sim \mathcal{O}(nK/b)$
STFlow[67]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
GP-EquiFlow[69]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
GSE-Flow[70]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
Bridge-based and transition-path generation					
PIPS*[72]	✓	–	✓	–	$\sim \mathcal{O}(n)$
Doob's Lagrangian*[73]	✓	–	✓	–	$\sim \mathcal{O}(n)$
TPS-DPS*[74]	✓	–	✓	–	$\sim \mathcal{O}(n)$
TPS-Flow[75]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
DeepPath[76]	–	✓	–	–	$\sim \mathcal{O}(n/b)$
ScooBDoob[31]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
Latent-space dynamics					
Operator learning and deep MSMs					
DeepGenMSM[77]	✓	–	–	–	$\sim \mathcal{O}(n)$
LSS[78]	✓	–	–	–	$\sim \mathcal{O}(n)$
GLDP[32]	✓	–	✓	–	$\sim \mathcal{O}(n)$
Generative backmapping					
GPCRLMD[34]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$
DynaFold†[33]	–	✓	–	✓	$\sim \mathcal{O}(nK/b)$

✓: used by the primary generation procedure; –: not used. n is the total number of generated trajectory frames, b is the number of frames jointly generated per block, and K is the number of diffusion- or flow-sampling iterations. The scaling describes approximate sequential generation cost rather than total floating-point operations.
*Path-level modelling does not necessarily imply parallel frame generation: PIPS, Doob's Lagrangian, and TPS-DPS model conditioned path measures but sample trajectories sequentially through controlled stochastic dynamics.

Supplementary Table 2. Validation coverage of representative molecular trajectory generation models. Filled circles (●) indicate explicit quantitative evaluation, open circles (○) indicate partial or indirect evaluation, and dashes (–) indicate that the corresponding validation was not reported. Method references follow Table 1.

Model	Structural fidelity	Thermodynamic fidelity	Kinetic fidelity	App.-level validation
Sequential predictive models				
Time-coarsened transition kernels				
TimeWarp[25]	●	●	○	○
ITO[52]	●	○	●	○
PLaTITO[53]	●	●	●	●
EquiJump[43]	●	●	●	–
TEMPO[44]	●	●	●	○
F3low[54]	●	●	○	–
DeepJump[55]	●	●	●	●
UniSim[56]	●	●	○	○
PVB[58]	●	●	●	●
UnbindingFlow[61]	●	○	○	●
ConfRover[26]	●	●	●	○
STAR-MD[27]	●	○	●	–
MD-LLM-1[42]	●	○	–	○
State-space models and graph-based rollouts				
ATMOS[28]	●	○	○	–
MARS-FM[45]	●	●	○	○
TrajCast[46]	●	●	●	–
DiffMD[47]	●	○	–	○
GGND[48]	●	○	○	○
NeuralMD[51]	●	○	–	–
Trajectory-segment generative models				
Diffusion models				
DynamicsDiffusion[62]	○	●	○	○
PTraj-Diff[63]	●	○	–	○
AlphaFolding[64]	●	○	–	●
GeoTDM[65]	●	○	–	–
DynaMode[68]	●	●	○	●
BioKinema[71]	●	●	●	●
Flow matching				
MDGen[29]	●	●	○	○
BioMD[30]	●	●	○	●
STFlow[67]	●	–	–	–
GP-EquiFlow[69]	●	–	–	–
GSE-Flow[70]	●	–	–	–
Bridge-based and transition-path generation				
PIPS[72]	●	○	–	●
Doob's Lagrangian[73]	○	○	–	○
TPS-DPS[74]	●	○	–	●
TPS-Flow[75]	●	○	○	●
DeepPath[76]	●	○	–	●
ScooBDoob[31]	●	○	●	●
Latent-space dynamics				
DeepGenMSM[77]	●	●	●	–
LSS[78]	●	●	●	–
GLDP[32]	●	●	●	●
GPCRLMD[34]	●	●	○	○
DynaFold[33]	●	○	–	●

Structural validation includes metrics such as RMSD, structural similarity, distance distributions, and contact recovery. Thermodynamic validation includes conformational populations, free-energy landscapes, equilibrium distributions, and ensemble observables. Kinetic validation includes transition probabilities, implied timescales, mean first-passage times, autocorrelation functions, and other time-dependent observables. Application-level validation refers to biologically relevant processes, including folding, conformational transitions, ligand binding, and unbinding.

Here, $\overline{W}_s$ is a reverse-time Wiener process, p_s is the marginal distribution induced by the forward SDE, and $p_{0s}(Y_s \mid X)$ is its perturbation kernel. Because the marginal score $\nabla_{Y_s} \log p_s(Y_s \mid c)$ is unknown, a neural network $s_\theta(Y_s, s, c)$ is trained by denoising score matching using samples from the forward process. At inference, one draws $Y_1 \sim p_1$ and numerically integrates the reverse-time SDE from $s = 1$ to $s = 0$, replacing the unknown score with s_θ to obtain a generated trajectory segment $\widehat{X} = Y_0$. Initial conformation, endpoint structure, or molecular context may be incorporated through conditioning, masking or guidance during the reverse process.

S.5.2 Trajectory Segment Generation with Flow Matching

Let $X = X_{t+1:t+L}$ denote a reference trajectory segment. Flow matching considers a continuous probability path p_s that transports a tractable prior p_0 (typically a Gaussian distribution) at $s = 0$ to the target trajectory-segment distribution at $s = 1$. The corresponding evolution is governed by a time-dependent vector field u_s:

$$\frac{\mathrm{d}Y_s}{\mathrm{d}s} = u_s(Y_s, s, c), \qquad Y_0 \sim p_0, \qquad s \in [0,1], \tag{3}$$

where Y_s is the evolving spatiotemporal tensor. A neural vector field v_θ is trained to approximate u_s. At inference, a prior sample is transported from $s = 0$ to $s = 1$ by integrating the same ODE with v_θ in place of u_s, yielding a generated trajectory segment $\hat{X} = Y_1$. Because Y_s represents the complete trajectory segment, the learned field can couple all atoms and physical frames within the finite horizon. Longer trajectories may be obtained by conditioning each newly generated segment on previously generated frames. The learned vector field describes generative transport and should not automatically be interpreted as a molecular force field or physical probability current. Structurally plausible segments therefore still require validation against kinetic observables, including transition probabilities, committors, residence times and pathway fluxes.

S.5.3 Latent Trajectory Model formulation

More generally, a latent trajectory model first maps molecular conformations through an encoder E_ϕ, learns their evolution in latent space, and reconstructs sampled states with a decoder D_ψ. For a finite segment,

$$Z_{t:t+L} \sim p_\theta(Z_{t:t+L} \mid c), \qquad \widehat{x}_k = D_\psi(z_k, c), \quad k = t, \ldots, t+L,$$

with stochastic decoders instead sampling $\widehat{x}_k \sim p_\psi(x_k \mid z_k, c)$. Sequential models apply the same principle to repeated latent transitions $p_\theta(z_{t+\tau} \mid z_{\le t}, c)$ rather than a jointly generated segment.

S.6 Definitions of Kinetic Metrics

Transition probabilities. For a lag time $\tau = \ell \Delta t$, the transition count matrix is $C_{ij}(\tau) = \sum_{n=0}^{N-\ell-1} \mathbf{1}[s_n = i]\mathbf{1}[s_{n+\ell} = j]$, where N is the number of frames, ℓ is the lag measured in frames, and $\mathbf{1}[\cdot]$ is the indicator function. The row-normalized transition matrix is $T_{ij}(\tau) = C_{ij}(\tau)/\sum_k C_{ik}(\tau)$. Here, $T_{ij}(\tau)$ is the conditional probability of observing state j after time τ, given that the system starts in state i. State populations evolve according to $\mathbf{p}(t + n\tau) = \mathbf{p}(t)\mathbf{T}^n(\tau)$, where $\mathbf{p}(t)$ is the row vector of state populations at time t.

Continuous-time transition rates. In a continuous-time Markov model, the state populations satisfy $\mathrm{d}\mathbf{p}(t)/\mathrm{d}t = \mathbf{p}(t)\mathbf{K}$, where $\mathbf{K}$ is the generator or rate matrix. Its entries are $K_{ij} = k_{ij} \ge 0$ for $i \ne j$, and $K_{ii} = -\sum_{j \ne i} k_{ij}$. Here, k_{ij} is the instantaneous transition rate from state i to state j and has units of inverse time. The transition probability and rate matrices are related by $\mathbf{T}(\tau) = \exp(\tau\mathbf{K})$. Therefore, $T_{ij}(\tau)$ is a finite-time transition probability, whereas k_{ij} is an instantaneous transition rate. In general, $k_{ij} \ne T_{ij}(\tau)/\tau$ unless τ is sufficiently small.

Implied timescales. Let $1 = \lambda_1(\tau) > |\lambda_2(\tau)| \ge |\lambda_3(\tau)| \ge \cdots$ be the eigenvalues of $\mathbf{T}(\tau)$. The implied relaxation timescale associated with eigenmode k is $t_k(\tau) = -\tau/\ln|\lambda_k(\tau)|$ for $k \ge 2$. Large t_k values correspond to slowly relaxing conformational processes. A kinetically accurate model should reproduce both the magnitudes and ordering of the reference timescales. For an approximately Markovian state decomposition, $t_k(\tau)$ should also become approximately independent of the chosen lag time τ.

Dwell-time distributions. A dwell event in state i is defined as a maximal contiguous segment of frames assigned to that state. If event r contains $L_i^{(r)}$ consecutive time intervals, its duration is $D_i^{(r)} = L_i^{(r)}\Delta t$. For N_i observed dwell events, the empirical cumulative distribution is $\widehat{F}_i(t) = \frac{1}{N_i}\sum_{r=1}^{N_i} \mathbf{1}[D_i^{(r)} \le t]$, and the corresponding survival probability is $\widehat{S}_i(t) = 1 - \widehat{F}_i(t) = \frac{1}{N_i}\sum_{r=1}^{N_i} \mathbf{1}[D_i^{(r)} > t]$. The mean dwell time is $\overline{D}_i = \frac{1}{N_i}\sum_{r=1}^{N_i} D_i^{(r)}$.

For a continuous-time Markov state, the total escape rate is $q_i = -K_{ii} = \sum_{j \ne i} k_{ij}$. The theoretical dwell-time survival probability is $S_i(t) = \exp(-q_i t)$, and the mean dwell time is $\mathbb{E}[D_i] = 1/q_i$. For a discrete-time MSM sampled at interval τ, the dwell-time distribution is geometric: $P(D_i = n\tau) = T_{ii}^{n-1}(\tau)[1 - T_{ii}(\tau)]$ for $n = 1, 2, \ldots$, with $\mathbb{E}[D_i] = \tau/[1 - T_{ii}(\tau)]$. The full dwell-time distribution is generally more informative than its mean, because generated trajectories may reproduce the average residence time while missing long-lived events.

Mean first-passage times. For a target state or state set B, define the first hitting time as $H_B = \inf\{n\tau \geq 0 : s_{n\tau} \in B\}$. The mean first-passage time from state i to B is $m_i^B = \mathbb{E}[H_B \mid s_0 = i]$. For a discrete-time MSM, m_i^B satisfies:

$$m_i^B = \begin{cases} 0, & i \in B, \\ \tau + \sum_j T_{ij}(\tau) m_j^B, & i \notin B. \end{cases}$$

For a source state set A with initial distribution μ_i^A, the state-set mean first-passage time is $\mathrm{MFPT}_{A\to B} = \sum_{i\in A} \mu_i^A m_i^B$. A common choice is the equilibrium distribution conditioned on A: $\mu_i^A = \pi_i / \sum_{j\in A} \pi_j$ for $i \in A$, where the stationary distribution π satisfies $\pi = \pi\mathbf{T}$ and $\sum_i \pi_i = 1$. Both $\mathrm{MFPT}_{A\to B}$ and $\mathrm{MFPT}_{B\to A}$ should be evaluated because the forward and reverse transitions may occur on substantially different timescales.

TICA autocorrelation relaxation. Let $\mathbf{z}_t = \mathbf{x}_t - \mu$ denote centered molecular features, where μ is the mean feature vector. The instantaneous and time-lagged covariance matrices are $\mathbf{C}_{00} = \langle \mathbf{z}_t \mathbf{z}_t^\top \rangle_t$ and $\mathbf{C}_{0\tau} = \langle \mathbf{z}_t \mathbf{z}_{t+\tau}^\top \rangle_t$. The TICA directions are obtained from the generalized eigenvalue problem $\mathbf{C}_{0\tau}\mathbf{w}_k = \lambda_k^{\mathrm{TICA}} \mathbf{C}_{00}\mathbf{w}_k$, where $\mathbf{w}_k$ is the kth TICA direction and $\lambda_k^{\mathrm{TICA}}$ measures its lagged autocorrelation. The corresponding TICA coordinate is $y_k(t) = \mathbf{w}_k^\top \mathbf{z}_t$. Its normalized autocorrelation function is $A_k(t) = \langle y_k(t_0) y_k(t_0+t) \rangle_{t_0} / \langle y_k^2(t_0) \rangle_{t_0}$. The TICA basis should be fitted using the reference MD data and then applied unchanged to generated trajectories. Agreement of $A_k(t)$ indicates that the generated trajectory reproduces relaxation along the same slow collective coordinate. Matching only the distribution of TICA projections does not guarantee correct temporal dynamics.

Chapman–Kolmogorov consistency. For a Markovian state model, transitions over a longer lag time $n\tau$ should satisfy $\mathbf{T}(n\tau) \simeq \mathbf{T}^n(\tau)$ for $n = 2, 3, \ldots$. The left-hand side is estimated directly using transitions separated by $n\tau$, whereas the right-hand side is predicted by repeatedly applying the MSM estimated at lag time τ. A normalized matrix discrepancy can be defined as $E_{\mathrm{CK}}(n) = \|\widehat{\mathbf{T}}(n\tau) - \widehat{\mathbf{T}}^n(\tau)\|_{\mathrm{F}} / \|\widehat{\mathbf{T}}(n\tau)\|_{\mathrm{F}}$, where $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm. Lower values indicate better agreement between directly observed and MSM-predicted long-time dynamics.

Committor probabilities. For disjoint source and target state sets A and B, the forward committor probability is $q_i^+ = P_i(H_B < H_A)$, where H_A and H_B are the first hitting times of A and B, respectively. Thus, q_i^+ is the probability that a trajectory starting from state i reaches B before reaching A. It satisfies:

$$q_i^+ = \begin{cases} 0, & i \in A, \\ 1, & i \in B, \\ \sum_j T_{ij}(\tau) q_j^+, & i \notin A \cup B. \end{cases}$$

For a continuous-time rate matrix, the corresponding equation is $\sum_j K_{ij} q_j^+ = 0$ for $i \notin A \cup B$. States with $q_i^+ \approx 0.5$ are located near the dynamical transition region. Comparing committor probabilities tests whether generated trajectories reproduce transition progress and mechanisms, rather than only the endpoint populations.

Reactive flux and pathway diversity. Let π_i denote the stationary population, q_i^+ the forward committor, and q_i^- the backward committor. The reactive flux from state i to state j for transitions from A to B is $f_{ij}^{AB} = \pi_i q_i^- T_{ij}(\tau) q_j^+$. The effective directed flux is $f_{ij}^+ = \max(0, f_{ij}^{AB} - f_{ji}^{AB})$. These fluxes identify the states and transitions carrying the dominant reactive pathways. If the total flux is decomposed into pathways γ with pathway fluxes F_γ, the normalized pathway weight is $p_\gamma = F_\gamma / \sum_{\gamma'} F_{\gamma'}$. Pathway diversity can then be summarized by the pathway entropy $H_{\mathrm{path}} = -\sum_\gamma p_\gamma \ln p_\gamma$. A generated model should reproduce not only the highest-flux pathway but also the relative probabilities and diversity of alternative transition pathways.

Quantitative comparison with reference kinetics. In practice, those MSM-based metrics will be computed in a shared state space constructed from the reference MD trajectories. For positive kinetic quantities spanning several orders of magnitude, such as implied timescales, transition rates, and MFPTs, a symmetric logarithmic error can be used: $E_{\log}(z) = \left|\ln[(z_{\mathrm{gen}} + \varepsilon)/(z_{\mathrm{ref}} + \varepsilon)]\right|$, where z_{gen} and z_{ref} are the generated and reference values, respectively, and ε is a small constant for numerical stability. Lower values indicate better agreement, and zero represents exact agreement. Distributional quantities such as dwell times may instead be compared using the Wasserstein or Kolmogorov–Smirnov distance. Autocorrelation and Chapman–Kolmogorov curves may be compared using integrated curve errors. Confidence intervals should be estimated using trajectory-level or block bootstrap procedures because adjacent trajectory frames are temporally correlated.