Table 1. Dataset details

Dataset name	AD	2AA	4AA
Training set simulation time	100 ns	50 ns	50 ns
Test set simulation time	100 ns	1 μs	1 μs
MD integration step Δt	0.5 fs	0.5 fs	0.5 fs
Timewarp prediction time τ	0.5×10^6fs	0.5×10^6fs	0.5×10^5fs
No. of training peptides	1	200	1400
No. of training pairs per peptide	2×10^5	1×10^4	1×10^4
No. of test peptides	1	100	30

permutation equivariant coupling layer. If the flow only took z^p as input *without* z^v, then to maintain permutation equivariance, each coupling layer would have to unnaturally split the Cartesian components of z_i^p into two disjoint sets.

Translation and rotation equivariance Consider a transformation $T = (R, a)$ that acts on x^p as follows:

$$Tx_i^p = Rx_i^p + a, \quad 1 \leq i \leq N, \tag{14}$$

where R is a 3×3 rotation matrix, and $a \in \mathbb{R}^3$ is a translation vector. We would like the model to satisfy $p_\theta(Tx(t+\tau)|Tx(t)) = p_\theta(x(t+\tau)|x(t))$. We achieve translation equivariance by subtracting the average position of the atoms in the initial molecular state (Appendix A.2). Rotation equivariance is not encoded in the architecture but is handled by data augmentation: each training pair $(x(t), x(t+\tau))$ from $\mathcal{D}$ is acted upon by a random rotation matrix R to form $(Rx(t), Rx(t+\tau))$ in each iteration.

5. Training objective

The model is trained in two stages. During *likelihood training*, the model is trained via maximum likelihood on pairs of states from the trajectories in the dataset. During *acceptance training*, the model is fine-tuned to maximise the probability of MH acceptance. Let k index training pairs, such that $\{(x^{(k)}(t), x^{(k)}(t+\tau))\}_{k=1}^K$ represents all pairs of states at times τ apart in $\mathcal{D}$. During likelihood training, we optimise:

$$\mathcal{L}_{\text{lik}}(\theta) := \tfrac{1}{K} \textstyle\sum_{k=1}^K \log p_\theta(x^{(k)}(t+\tau)|x^{(k)}(t)). \tag{15}$$

Once likelihood training is complete, we add a fine-tuning stage to optimise the MH acceptance probability. Let $x^{(k)}(t)$ be sampled uniformly from $\mathcal{D}$. Then, we use the model to sample $\tilde{x}_\theta^{(k)}(t+\tau) \sim p_\theta(\,\cdot\,|x^{(k)}(t))$ using Equation (3). Note that the sample value depends on θ through f_θ. We use this to optimise the acceptance probability in Equation (7) with respect to θ. Let $r_\theta(X, \tilde{X})$ denote the model-dependent term in the acceptance ratio in Equation (7):

$$r_\theta(X, \tilde{X}) := \frac{\mu_{\text{aug}}(\tilde{X}) p_\theta(X \mid \tilde{X}^p)}{\mu_{\text{aug}}(X) p_\theta(\tilde{X} \mid X^p)}. \tag{16}$$

The acceptance objective is given by:

$$\mathcal{L}_{\text{acc}}(\theta) := \tfrac{1}{K} \textstyle\sum_{k=1}^K \log r_\theta(x^{(k)}(t), \tilde{x}_\theta^{(k)}(t+\tau)). \tag{17}$$

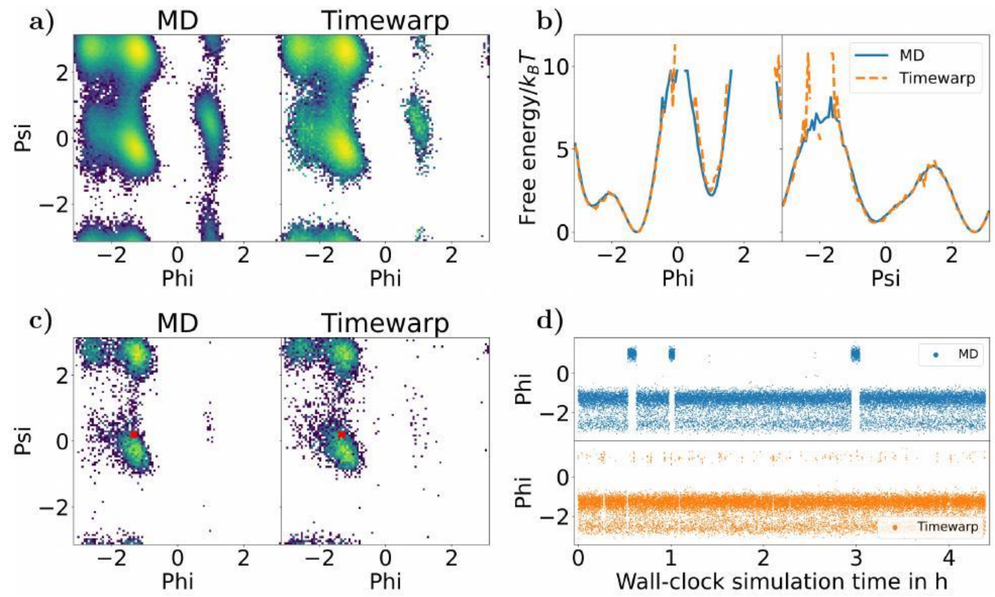

Figure 3. Alanine dipeptide experiments. (a) Ramachandran plots for MD and Timewarp samples generated according to Algorithm 1. (b) Free energy comparison for the two dihedral angles φ and ψ. (c) Ramachandran plots for the conditional distribution of MD compared with the Timewarp model. Red cross denotes initial state. (d) Time dependence of the φ dihedral angle of MD and the Markov chain generated with the Timewarp model.

Training to maximise the acceptance probability can lead to the model proposing changes that are too small: if $\tilde{x}_\theta^{(k)}(t+\tau) = x^{(k)}(t)$, then all proposals will be accepted. To mitigate this, during acceptance training, we use an objective which is a weighted average of $\mathcal{L}_{\text{acc}}(\theta)$, $\mathcal{L}_{\text{lik}}(\theta)$ and a Monte Carlo estimate of the average differential entropy,

$$\mathcal{L}_{\text{ent}}(\theta) := -\tfrac{1}{K} \textstyle\sum_{k=1}^K \log p_\theta(\tilde{x}_\theta^{(k)}(t+\tau)|x^{(k)}(t)). \tag{18}$$

The weighting factors for each term are hyperparameters.

6. Experiments

We evaluate Timewarp on small peptide systems. To compare with MD, we focus on the slowest transitions between metastable states, as these are the most difficult to traverse. To find these, we use *time-lagged independent component analysis* (TICA) (Pérez-Hernández et al., 2013), a linear dimensionality reduction technique that maximises the autocorrelation of the transformed coordinates. The slowest components, TIC 0 and TIC 1, are of particular interest. To measure the speed-up achieved by Timewarp, we compute the *effective sample size* per second of wall-clock time (ESS/s) for the TICA components. The ESS/s is given by:

$$\text{ESS/s} = \frac{M_{\text{eff}}}{t_{\text{sampling}}} = \frac{M}{t_{\text{sampling}}\,(1 + 2\sum_{\tau=1}^{\infty} \rho_\tau)}, \tag{19}$$

where M is the chain length, M_{eff} is the effective number of samples, t_{sampling} is the sampling wall-clock time, and ρ_τ is the autocorrelation for the lag time τ (Neal, 1993). The speed-up factor is defined as the ESS/s achieved by Timewarp divided by the ESS/s achieved by MD. Additional