

# Timewarp: Transferable Acceleration of Molecular Dynamics by Learning Time-Coarsened Dynamics

Leon Klein<sup>\*†12</sup> Andrew Y. K. Foong<sup>\*1</sup> Tor Erlend Fjelde<sup>\*†3</sup> Bruno Mlodozeniec<sup>\*†3</sup> Marc Brockschmidt<sup>†</sup>  
Sebastian Nowozin<sup>†</sup> Frank Noé<sup>124</sup> Ryota Tomioka<sup>1</sup>

## Abstract

Molecular dynamics (MD) simulation is a widely used technique to simulate molecular systems, most commonly at the all-atom resolution where the equations of motion are integrated with timesteps on the order of femtoseconds (1fs =  $10^{-15}$ s). MD is often used to compute equilibrium properties, which requires sampling from an equilibrium distribution such as the Boltzmann distribution. However, many important processes, such as binding and folding, occur over timescales of milliseconds or beyond, and cannot be efficiently sampled with conventional MD. Furthermore, new MD simulations need to be performed from scratch for each molecular system studied. We present *Timewarp*, an enhanced sampling method which uses a normalising flow as a proposal distribution in a Markov chain Monte Carlo method targeting the Boltzmann distribution. The flow is trained offline on MD trajectories and learns to make large steps in time, simulating the molecular dynamics of  $10^5 - 10^6$  fs. Crucially, *Timewarp* is *transferable* between molecular systems: once trained, we show that it generalises to unseen small peptides (2-4 amino acids), exploring their metastable states and providing wall-clock acceleration when sampling compared to standard MD. Our method constitutes an important step towards developing general, transferable algorithms for accelerating MD.

## 1. Introduction

Molecular dynamics (MD) is a well-established technique for simulating physical systems at the atomic level. When performed accurately, it provides unrivalled insight into

<sup>\*</sup>Equal contribution <sup>†</sup>Work done while at Microsoft Research <sup>1</sup>Microsoft Research AI4Science <sup>2</sup>Freie Universität Berlin <sup>3</sup>University of Cambridge <sup>4</sup>Rice University. Correspondence to: Ryota Tomioka <ryoto@microsoft.com>.

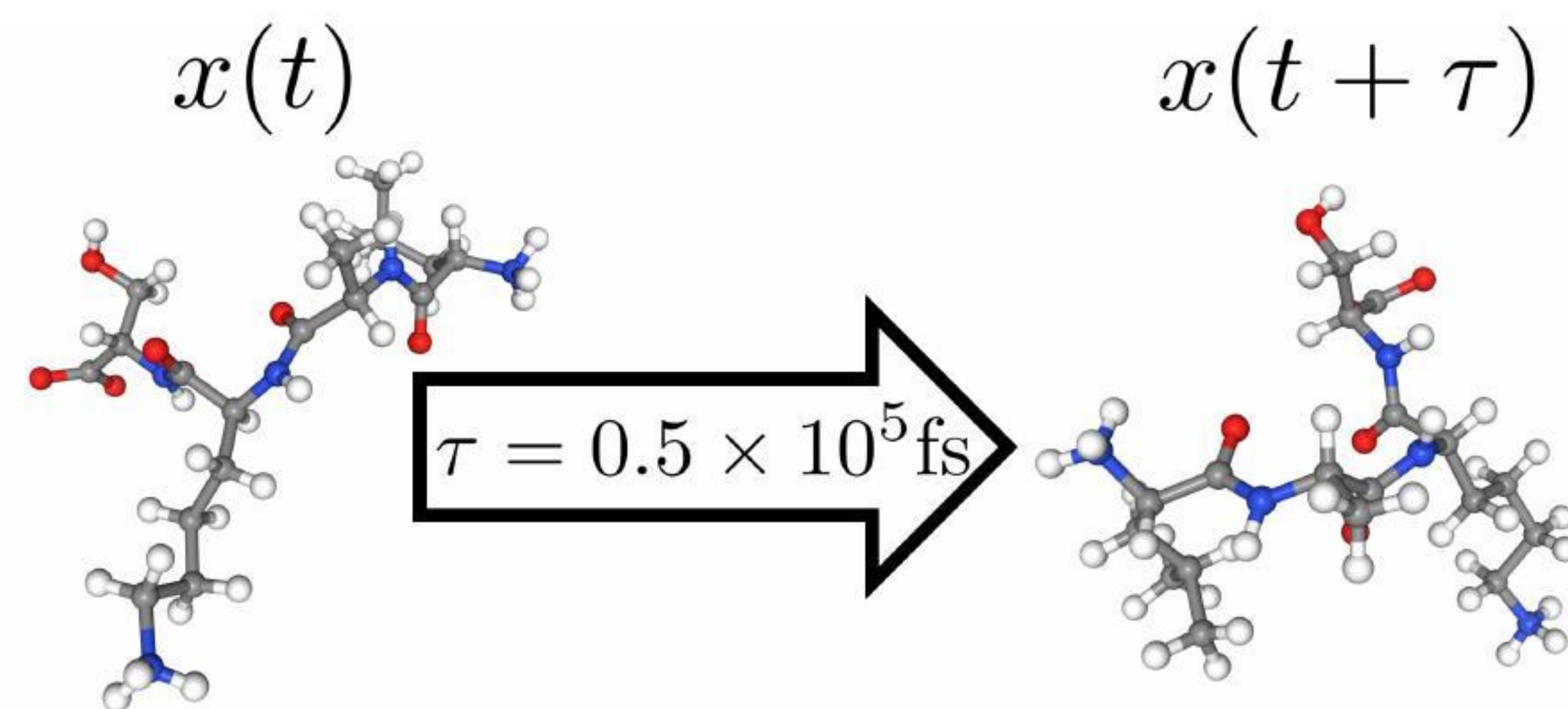


Figure 1. Initial state  $x(t)$  (Left) and accepted proposal state  $x(t + \tau) \sim p_\theta(x(t + \tau)|x(t))$  (Right) sampled from Timewarp for the tetrapeptide LAKS (not in trainset). The proposed state undergoes a large conformational change relative to the initial state, traversing from one metastable state to another.

the detailed mechanics of molecular motion, without the need for wet lab experiments. MD simulations have been used to understand processes of central interest in molecular biophysics, such as protein folding (Noé et al., 2009; Lindorff-Larsen et al., 2011), protein-ligand binding (Buch et al., 2011), and protein-protein association (Plattner et al., 2017). Many crucial applications of MD boil down to efficiently sampling from the *Boltzmann distribution*, i.e., the equilibrium distribution of a molecular system at a temperature  $T$ . Let  $(x^p(t), x^v(t)) := x(t) \in \mathbb{R}^{6N}$  be the state of the molecule at time  $t$ , consisting of the positions  $x^p(t) \in \mathbb{R}^{3N}$  and velocities  $x^v(t) \in \mathbb{R}^{3N}$  of the  $N$  atoms in Cartesian coordinates. The Boltzmann distribution is given by:

$$\mu(x^p, x^v) \propto \exp\left(-\frac{1}{k_B T}(U(x^p) + K(x^v))\right), \quad (1)$$

where  $U(x^p)$  is the potential energy,  $K(x^v)$  is the kinetic energy, and  $k_B$  is Boltzmann’s constant. In practice we are interested in the marginal distribution of the positions,  $\mu(x^p)$ . Many important quantities, such as the free energies associated with protein folding and protein-ligand binding, can be computed as expectations under  $\mu(x^p)$ .

A popular method in MD to sample from the Boltzmann distribution is *Langevin dynamics* (Langevin, 1908), which obeys the following stochastic differential equation (SDE):

$$m_i dx_i^v = -\nabla_i U dt - \gamma m_i x_i^v dt + \sqrt{2m_i \gamma k_B T} dB_t. \quad (2)$$