# Data-based Langevin modeling of biomolecular systems

Vorgelegt von:

Benjamin Lickert

Betreut durch:

Prof. Dr. Gerhard Stock

Dissertation

zur Erlangung des Doktorgrades
der
Fakultät für Mathematik und Physik
der
Albert-Ludwigs-Universität Freiburg

Freiburg im Breisgau
Juli 2021

| | |
|---|---|
| Dekan: | Prof. Dr. Wolfgang Soergel |
| Erstgutachter: | Prof. Dr. Gerhard Stock |
| Zweitgutachter: | Prof. Dr. Joachim Dzubiella |
| Tag der mündlichen Prüfung: | 16.09.2021 |

# Abstract

Understanding the dynamical behavior of proteins is a highly challenging area of current research. Based on the progress in algorithmic methods and the increase of computational power in the recent years, molecular dynamics simulations have emerged as powerful tool to access molecular motions on time scales from femto- to milliseconds. However, the resulting data is so overwhelming that a suitable interpretation framework is needed in order to detect and analyse the essential dynamics of the system under study. Frequently, following a dimensionality reduction to identify collective variables $\boldsymbol{x}$, the dynamics are described in terms of a diffusive motion on a low-dimensional free energy landscape $F(\boldsymbol{x})$. By using projection operator approaches, such as developed by Zwanzig, it is possible to derive coarse-grained equations of motions for the collective variables, such as the generalized Langevin equation. Going further, by assuming a time scale separation between the slow dynamics along the system coordinate $\boldsymbol{x}$ and the fast fluctuations of the bath, this equation can be simplified to the (memory-less) Markovian Langevin equation, which describes the system dynamics in terms of a deterministic drift, a Stokes' friction and a stochastic force. Alternatively, an additional step of coarse graining can be applied in order to account for the dynamics in terms of jumps between metastable conformational states. By furthermore assuming that those jumps are memory-free, a so-called Markov state model can be constructed.

In this thesis the virtues and shortcomings of data-based Markovian modeling are investigated. In particular, two modifications of the data-driven Langevin equation are presented: the rescaled and the binned data-driven Langevin equation. While the former approach allows for the rescaling of the dissipative force of the model, the latter concept enables the analysis of extensive MD data. In addition, it is investigated under which conditions the data-driven Langevin equation can be applied in the nonequilibrium regime. By considering molecular dynamics simulations of several systems with varying complexity it is shown that Markovian models can serve as powerful system descriptions of nontrivial dynamics. First, an one-dimensional model of sodium chloride in water and a five-dimensional model of the small Aib$_9$ peptide are constructed. Then, the Markovian framework is challenged by considering the dynamics of the 164-residue T4 lysozyme, the unbinding of benzamidine from trypsin and the unbinding of a resorcinol scaffold-based inhibitor from the N-terminal domain of heat shock protein 90. The latter two systems exhibit dynamics on the order of milliseconds or even seconds. To investigate the nonequilibrium regime, the enforced dissociation of sodium chloride in water and the pressure-jump induced nucleation and growth process in a liquid of hard spheres are considered.

# Contents

# List of publications and connection to thesis

The following publications are listed by date of publication.

[1] M. Biswas, B. Lickert, and G. Stock, "Metadynamics enhanced Markov modeling of protein dynamics". In *J. Phys. Chem. B* 122, 5508 (2018)

[2] D. Nagel, A. Weber, B. Lickert, and G. Stock, "Dynamical coring of Markov state models". In: *J. Chem. Phys.* 150, 094111 (2019).

[3] S. Wolf, B. Lickert, S. Bray, and G. Stock, "Multisecond ligand dissociation dynamics from atomistic simulations". In: *Nat. Commun.* 11, 2918 (2020).

[4] B. Lickert and, G. Stock, "Modeling non-Markovian data using Markov state and Langevin models". In: *J. Chem Phys.* 153, 244112 (2020).

[5] B. Lickert, S. Wolf, and G. Stock, "Data-driven Langevin modeling of nonequilibrium processes". In. *J. Phys. Chem. B* (in print) (2021).

Results presented in [4] are shown in Sec. 4.1.2, Sec. 5.1.1 and Sec. 6.1. Additionally, the findings in chapter 7 and many results from Sec. 5.2 are shown in [5]. The modeling of multisecond dynamics shown in Sec. 6.2 was published in [3] and Markov state models used for comparison in Sec. 5.2.4 are from [1].

# List of variables

The following list shows relevant variables used in the text in order of appearance.

| | |
|---|---|
| $F(\boldsymbol{x})$ | free energy landscape |
| $k_\mathrm{B}$ | Boltzmann constant |
| $\Gamma$, $\mathcal{K}(\boldsymbol{x})\xi$ | fiction matrix and stochastic force (Markovian Langevin equation) |
| $\mathcal{M}$ | system mass |
| $T(\tau)$ | transition matrix (Markov state model) |
| $C(\tau)$ | autocorrelation function |
| $\tau_\mathrm{wait}$ | waiting time |
| $t_i$ | i-th implied time scale (Markov state model) |
| $K(t)$, $N(t)$ | memory kernel and stochastic force (generalized Langevin equation) |
| $\delta t$ | time step numerical integration (Langevin equation) |
| $\hat{\boldsymbol{f}}(\boldsymbol{x})$, $\hat{\Gamma}(\boldsymbol{x})$, $\hat{\mathcal{K}}(\boldsymbol{x})$ | dLE fields (data-driven Langevin equation) |
| $\delta t_\mathrm{M}$ | lower limit on dLE time step (Markovianity) |
| $\delta t_\mathrm{R}$ | upper limit on dLE time step (Resolution and converged integration) |
| $S$ | (diagonal) rescaling matrix (rescaled dLE) |
| $s$, $N_\mathrm{max}$, $\omega_\mathrm{min}$, $\omega_\mathrm{max}$ | parameters pre-averaging (binned dLE) |
| $\tilde{\boldsymbol{f}}(\boldsymbol{x})$, $\tilde{\Gamma}(\boldsymbol{x})$, $\tilde{\mathcal{K}}(\boldsymbol{x})$ | Verlet-dLE fields (Verlet-dLE) |
| $\mathcal{F}(\boldsymbol{x})$ | biased energy landscape (nonequilibrium regime) |
| $V_\mathrm{ext}(\boldsymbol{x}, t)$ | potential external driving (nonequilibrium regime) |
| $\mathcal{P}(\boldsymbol{x})$ | time-dependent distribution (nonequilibrium regime) |

# List of acronyms

The following list shows the acronyms used in the text in alphabetical order.

| | |
|---|---|
| **dcTMD** | dissipation-corrected targeted MD |
| **dLE** | data driven Langevin equation |
| **FEL** | free energy landscape |
| **GLE** | generalized Langevin equation |
| **Hsp90** | heat shock protein 90 |
| **LE** | Markovian Langevin equation |
| **MD** | molecular dynamics |
| **MSM** | Markov state model |
| **NaCl** | sodium chloride |
| **PCA** | principal component analysis |
| **T4L** | T4 lysozyme |

# 1 Introduction

*"The purpose of (scientific) computing*
*is insight, not numbers."*
– Richard Hamming

Life in all its forms depends on many different types of biomolecules. One important class of biomolecules, called proteins, plays a crucial role in the regulation and execution of biological processes. Proteins were already discovered in 1789 by the French chemist Antoine Fourcroy, who observed their ability to flocculate under the influence of heat or an increase of the pH level [1]. Today, it is known that proteins form a highly diverse group of biopolymers. While the smallest members consist of only a few amino acids, the largest proteins are formed by multiple long chains of amino acids. Furthermore, the biological function of proteins is very diverse as well. Among other things, they determine the shape of cells, regulate the intra- and extracellular environment, perform signal transduction and catalysis or replicate the DNA. For a long time it was assumed that the function of any protein is solely determined by its three-dimensional structure which in turn is based on the respective sequence of amino acids [2]. Nevertheless, it was found by mutation studies [3] together with computational advances that the protein functionality also depends on its dynamics [4–8].

To study the structure of proteins, various experimental techniques, for example X-ray crystallography and different spectroscopic approaches, have been used [9–11]. Still, the direct experimental investigation of the microscopic protein conformation, which emerges for physiological conditions, suffers from noise and experimental limitations. Additionally, the dynamics of proteins appear on a wide range of time scales. While local bond vibrations appear at scales of tens of femtoseconds or picoseconds, conformational transitions and structural rearrangements occur in the range of nanoseconds, microseconds or even longer [8]. This diversity complicates experimental studies even further, since different techniques need to be combined to obtain a comprehensive picture.

Based on advances in theory and the steady increase of computational power in recent years, all-atom molecular dynamics (MD) simulations have become a powerful tool to complement experimental studies [12–14]. Here, the dynamics of interest are modeled using classical Newtonian equations of motion which are propagated using numerical methods. The equations of motion are defined by empirical potential energy functions called force fields [15–17], which account for microscopic effects like hydrogen bonds, electrostatic interactions or van der Waals forces in a classical framework. Additionally, experimental observations are used to refine the force fields. Still, although the numerical propagation of classical equations of motion is much simpler than exact (a priori quantum mechanical) calculations, even the most advanced computers are not able to reach time scales of seconds for significant biomolecular systems since integration time steps on the order of femtoseconds are needed to obtain reliable results. To overcome this limitation many different enhanced sampling schemes were proposed [18–21]. These

approaches bias the numerical simulations in such a way that the processes of interest appear more often, i.e., the needed computational times are reduced.
The result of MD simulations are extensive time series, so-called trajectories, which record the atomistic details of the dynamics in terms of discretized snapshots at a finite time resolution. This large amount of information is simultaneously a blessing and a curse. One the one hand, the possibility to follow the trajectory $\boldsymbol{r}(t)$ of every simulated atom can be very useful when investigating molecular details. On the other hand, the immense scope of recorded details can overshadow the essential information needed to understand the system dynamics. To facilitate the interpretation of the MD data, one often performs a dimensionality reduction. Here, we aim for a few (say, less then ten) collective coordinates $\{x_i(t)\} = \boldsymbol{x}(t)$ which are able to resolve the most important characteristics of the dynamics under study [22–27]. The choice of these coordinates defines the free energy "landscape" $F(\boldsymbol{x})$, which can be used to visualize the main properties of the system, see for example Fig. 1.1. The energy minima represent metastable states which interchange by passing the energy barriers between them [5–7]. Assuming that the system coordinates are chosen in such a way that we can apply Kramers rate theory [28], the height of the barriers directly accounts for the time scale of the interstate transitions. Small barriers indicate frequent, short-living oscillations, while large barriers represent sparse dynamics.
To further analyze the simulated data, it can be advantageous to construct a "post-simulation" model which maps the dynamics along the system coordinates on a theoretical framework. To this end, one may use projection operator techniques [29–31] to derive (in principle exact) equations of motions for these coordinates. The generalized Langevin equation (GLE) is one possible result. Under certain conditions [32], the GLE resembles a deterministic Newtonian equation of motion where the system dynamics are driven by three forces. The first force, the deterministic drift field, can be interpreted (depending on the projection [32]) as the gradient of the free energy in thermal equilibrium [14]. The second (dissipative) force includes the memory of the interactions of the system coordinates with the neglected degrees of freedom (called "bath"). In thermal equilibrium and under certain conditions [32] it can be related to the third force, which represents a stochastic driving of the system. It is worth noting that the stochastic nature of the latter force is a direct result of the dimensionality reduction where the bath degrees of freedom were actively projected out. By furthermore assuming that the time scales of the bath are much faster than the evolution of the system coordinates, it can be possible to simplify the GLE to the Markovian Langevin equation (LE) [31–33]

$$\mathcal{M}\ddot{\boldsymbol{x}} = -\nabla F(\boldsymbol{x}) - \Gamma(\boldsymbol{x})\dot{\boldsymbol{x}} + \mathcal{K}(\boldsymbol{x})\boldsymbol{\xi}, \tag{1.1}$$

with $\nabla F(\boldsymbol{x})$ representing the drift field, the Stokes' friction $\Gamma(\boldsymbol{x})\dot{\boldsymbol{x}}$ and the stochastic force $\mathcal{K}(\boldsymbol{x})\boldsymbol{\xi}$ which is usually called "noise". Here, the friction matrix $\Gamma(\boldsymbol{x})$ is often interpreted as indicator of the "roughness" of the free energy [34, 35]. The stochastic variable $\boldsymbol{\xi}$ is typically defined by $\langle\boldsymbol{\xi}\rangle = 0$ and $\langle\xi_i(t_1)\xi_j(t_2)\rangle = \delta(t_1 - t_2)\delta_{i,j}$. In equilibrium, friction and noise are related by the fluctuation-dissipation theorem $\mathcal{K}(\boldsymbol{x})\mathcal{K}(\boldsymbol{x})^T = 2k_{\mathrm{B}}T\Gamma(\boldsymbol{x})$.
In case the parameters $F(\boldsymbol{x})$, $\mathcal{M}$, $\Gamma(\boldsymbol{x})$ and $\mathcal{K}(\boldsymbol{x})$ are known, it is straightforward to integrate Eq. (1.1) numerically to obtain a time trace similar to an MD trajectory. On the other hand, in case we want to derive a model for the MD trajectory $\boldsymbol{x}(t)$, we need to find a way to estimate the Langevin forces from the data [36–42]. In this thesis we

will adopt a data-driven Langevin equation (dLE) [43–45] that calculates the Langevin forces locally in space and time. It will be shown that Eq. (1.1) represents a robust approach to model biomolecular motions.

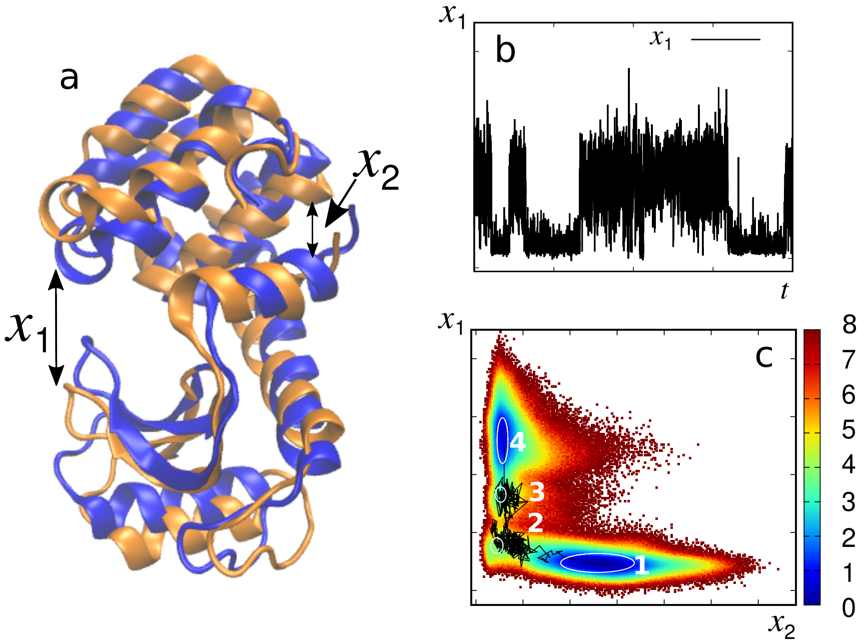

Figure 1.1: **Dynamics of T4 lysozyme.** (a) The 164-residue T4 lysozyme exhibits a prominent hinge bending motion which can be quantified by the coordinate $x_1$ and $x_2$. There are two main states, the open state (orange structure) and the closed state (blue structure). (b) The dynamics along $x_1$ alone indicate only two metastable states but (c) when considering the free energy (in units of $k_\mathrm{B}T$) in both coordinates, we observe actually four states (white numbers) explored by the MD trajectory (black line).

As more coarse-grained post-simulation model one can also aim for a Markov state model of the dynamics of interest [46–55]. Here, we describe the system dynamics in terms of memory-free jumps between its metastable states. Based on the assumption that the free energy barriers account for the relevant system transitions, these states can be defined via the minima of the free energy [56–61]. The condition of negligible system memory requires a separation of time scales between fast intrastate dynamics and rarely occurring interstate transitions. Note that this requirement is very similar to the separation of time scales of system and bath leading to the Markovian Langevin equation (1.1). Given that the state dynamics are truly memory-free, the system dynamics can be described in terms of the transition matrix $T(\tau)$ containing the probabilities $T_{ij}$ that the system jumps from state $i$ to $j$ within the so-called lag time $\tau$. By defining the state vector $\boldsymbol{P}(t) = (p_1(t), \ldots, p_k(t))^T$ with $p_i(t)$ representing the probability to be in state $i$ at time $t$, the time evolution of the system is given by

$$\boldsymbol{P}(t{=}n\tau) = T^n(\tau)\boldsymbol{P}(0). \tag{1.2}$$

Just as the applicability of Markovian Langevin framework depends on properly chosen system coordinates $\boldsymbol{x}$, Markov state models rely on the careful definition of the states. This modeling step is anything but trivial. In order to improve the Markovianity of a given state separation, several methods have been proposed. For example, the concept of "coring" demands that a transition must reach the core region of the target state to be considered as valid [47, 62–64]. Alternatively, it may be requested that the trajectory spends a minimum time in the new state to indicate a proper transition, i.e., coring can be performed in a dynamical way [65, 66].

In this thesis we investigate the virtues and shortcomings of the data-based Markovian modeling of several different biomolecular systems of varying complexity. Although Markov state models are considered as well, we will mostly rely on the Langevin equation (1.1) via the dLE approach. First, we introduce the basic theory and methods underlying our modeling framework in chapter 2 and 3 before the dLE is introduced in chapter 4. Based on the study of a model GLE, we will see that the temporal resolution $\delta t$ of the dLE plays a similar role as the lag time $\tau$ of the Markov state model. Once $\delta t$ is larger than the memory time of the system, the dLE correctly approximates the GLE dynamics, but if $\delta t$ is too small the dLE will underestimate the friction $\Gamma(\boldsymbol{x})$ and the kinetics will become too fast. These findings motivate the formulation of the rescaled dLE where we rescale the friction in such a way that the resulting Langevin dynamics correctly account for the initial decay of the position autocorrelation function of the data $\boldsymbol{x}(t)$. This correction resembles the concept of coring for Markov state models and coarse-grained MD approaches where the time scale of the model is rescaled by comparing the predicted diffusion behavior to the results of atomistic MD simulations [67–69]. By considering the dissociation and association of sodium chloride in water [42] as well as the dynamics of the small Aib$_9$ peptide [70] in chapter 5, we will see that the rescaled dLE allows for the construction of robust Markovian Langevin models. Then, we challenge the approach in chapter 6 by considering a 60 $\mu$s-long all-atom MD trajectory of T4 lysozyme [71] were optimal system coordinates and a reliable state partitioning are still unclear [72].

Additionally, this thesis introduces the concept of the binned dLE which allows to apply the dLE framework to extensive data sets. Based on the way the dLE estimates the Langevin forces, it is possible to "pre-average" the input data without harming the model dynamics (chapter 4). As a practical application we consider in chapter 5 a large data set [73] of AIB$_9$ where it will be possible to reduce the number of data points by a factor of $10^2$ without harming the Langevin estimates. This shows that the dLE approach represents a powerful tool to analyze, e.g., enhanced sampling data which consists of numerous short MD simulations [73].

Although Langevin dynamics are significantly less time consuming to propagate than all-atom MD simulations, it is still nontrivial to access time scales on the order of microseconds or larger. First, one needs to find a way to parameterize the Langevin model for such slow processes. At this point it is possible to use dissipation-corrected targeted MD simulations [42]. This approach was specifically designed to determine one-dimensional Markovian Langevin models of extremely slow dynamics based on constraint MD simulations. Second, assuming that the Langevin model could be determined, the direct integration of the Langevin equation would need prohibitively many time steps to provide converged estimates of the slow dynamics. This thesis will present an elegant way to circumvent this limitation based on Langevin simulations at high temperatures. Called T-boosting, this approach exploits the well-defined connection between Langevin kinetics and the temperature (chapter 3). By considering Langevin models of trypsin-benzamidine and a N-terminal domain of a heat shock protein 90 inhibitor complex in chapter 6, we will see that Langevin models are indeed able to predict times on the order of seconds with reasonable accuracy.

Finally, this thesis extends the application of the dLE framework to the nonequilibrium regime. Considering relaxation processes and externally driven dynamics, we will introduce the necessary modifications of the dLE formulation. As applications we inspect in chapter 7 the enforced dissociation of sodium chloride in water and the pressure-jump induced nucleation and growth process in a liquid of hard spheres. Since these systems were already studied with a nonstationary generalized Langevin equation [74, 75], it will be possible to compare the Markovian modeling to memory-based approximations.

# 2 Theory and methods

*"Every point of view is useful, even those that are wrong*
*- if we can judge why a wrong view was accepted."*
–Legion, Mass Effect 2

In this chapter we will inspect some fundamental topics which are not only important for the discrete and continuous Markov modeling in this work, but also for the study of biomolecular dynamics in general. First, molecular dynamics simulations are introduced. Here, force fields as well as their numerical integration are briefly touched to give an impression of the main ingredients of molecular dynamics. Going on it will be discussed how to enable the interpretation of simulated data by means of a dimensionality reduction. This concept describes the search for a few essential coordinates hidden in the, a priori high-dimensional, input data which makes it possible to interpret the data by, e.g., Markov models. Afterwards we will introduce the free energy landscape as well as the autocorrelation function. In the latter case it will be specified how to treat data given by multiple short trajectory and how to inspect nonequilibrium data. In the following section we will discuss the concept of states and transition statistics. Just as dimensionality reductions, the definition of states represents a coarse graining of the considered system dynamics, i.e., it simplifies the interpretation. Still, since the definition of states is in general nontrivial, the chapter concludes with the introduction of coring. This concept refines state separations by correcting wrong state assignments in a geometrical or temporal way.

## 2.1 Molecular dynamics simulations

To analyze the motion of biomolecular systems, molecular dynamics (MD) simulations provide a powerful framework to gather data in atomistic detail [13]. MD simulations are, in most cases, purely classical, i.e., they cover the dynamics of $N$ interacting atoms in terms of Newtonian equations of motion

$$\mathcal{M}_i \frac{\partial^2 \boldsymbol{r}}{\partial t^2} = -\nabla_i V(\boldsymbol{r}_1, ..., \boldsymbol{r}_N) \tag{2.1}$$

where $i = 1, ..., N$ holds and $\nabla_i V$ represents the gradient of the potential energy $V$ with respect to particle $i$. This gradient, often called force field, is parameterized in such a way that quantum effects can be incorporated into the classical equation (2.1). Several force fields are available, like the different versions of AMBER [15] or GROMOS [16] which can be employed using software packages like GROMACS [17]. The potential energy $V$ is typically a sum of the form

$$V(\boldsymbol{r}_1, ..., \boldsymbol{r}_N) = \sum_{i,j}^{\text{bond stretching}} \frac{K^b_{ij}}{2}(r_{ij} - r^0_{ij})^2 \tag{2.2}$$

$$+ \sum_{i,j,k}^{\text{bond bending}} \frac{K^\alpha_{ijk}}{2}(\cos(\alpha_{ijk}) - \cos(\alpha^0_{ijk}))^2 \tag{2.3}$$

$$+ \sum_{i,j,k,l}^{\text{proper dihedrals}} K^\phi_{ijkl}(1 + \cos(m_{ijkl}\phi_{ijkl} - \phi^0_{ijkl}))^2 \tag{2.4}$$

$$+ \sum_{i,j,k,l}^{\text{improper dihedrals}} \frac{K^\omega_{ijkl}}{2}(\omega_{ijkl} - \omega^0_{ijkl})^2 \tag{2.5}$$

$$+ V_{\text{non-bonded}}(\boldsymbol{r}_1, ..., \boldsymbol{r}_N) \tag{2.6}$$

where bond stretching (2.2), bond bending (2.3) and improper dihedral angles (2.5) are modeled by harmonic oscillators. The periodicity of proper dihedral angles motivates the cosine in (2.4). The long range interactions represented by $V_{\text{non-bonded}}$

$$V_{\text{non-bonded}}(\boldsymbol{r}_1, ..., \boldsymbol{r}_N) = \sum_{i,j}^{Lennard-Jones} \left(\frac{C_{ij,12}}{\Delta r^{12}_{ij}} - \frac{C_{i,j,6}}{\Delta r^6_{ij}}\right) \tag{2.7}$$

$$+ \sum_{i,j}^{\text{Coulomb}} \frac{1}{4\pi\epsilon_0}\frac{\delta_i\delta_j}{\Delta r_{ij}} \tag{2.8}$$

contain the Coulomb interaction (2.8) between the two atoms $i$ and $j$ with a distance of $\Delta r_{ij}$ and the Lennard-Jones potential (2.7) as approximation of the van der Waals interaction between $i$ and $j$. The different equilibrium values ($r^0_{ij}$, $\alpha^0_{ijk}$, $\phi^0_{ijkl}$, $\omega^0_{ijkl}$) as well as the force constants ($K^b_{ij}$, $K^\alpha_{ijk}$, $K^\phi_{ijkl}$, $K^\omega_{ijkl}$), partial charges $\delta_i$ and Lennard-Jones coefficients $C_{ij,12}$, $C_{ij,6}$ are determined by fitting them to experimental data and by evaluating ab-initio or semi-empirical quantum mechanical calculations.

Once $V(\boldsymbol{r}_1, ..., \boldsymbol{r}_N)$ is defined, we can simulate a system in the micro-canonical ensemble by numerically integrating the equations of motion. Here, some desired starting configuration, like an energy minimized structure with Maxwell-Boltzmann distributed velocities, is chosen. Then, a so-called integrator is used to propagate the dynamics. Many different integrators are available. The Leapfrog integrator

$$\dot{\boldsymbol{r}}_i\left(t + \frac{\delta t}{2}\right) = \dot{\boldsymbol{r}}_i\left(t - \frac{\delta t}{2}\right) - \frac{\nabla_i V(\boldsymbol{r}_1, ..., \boldsymbol{r}_N)}{\mathcal{M}_i}\delta t \tag{2.9}$$

$$\boldsymbol{r}_i(t + \delta t) = \boldsymbol{r}_i(t) + \dot{\boldsymbol{r}}_i\left(t + \frac{\delta t}{2}\right)\delta t \tag{2.10}$$

is a prominent example used in many MD simulations. Although it is relatively simple, this integrator has the advantages of being time-reversal as well as symplectic, i.e., it preserves the energy of the considered system. In case the system is coupled to some environment, like a surrounding fluid, some thermostat needs to be included to simulate an NVT-ensemble. The Bussi thermostat [76] is often used at this point. For NPT-ensembles a barostat, like for example the Berendsen barostat [77], is needed. Since the

details of MD simulations do not represent a key aspect of this thesis and since there are numerous comprehensive references like the GROMACS manual [17], we do not consider these concepts here in detail.
As result, MD simulations provide a time trace, or trajectory, $\boldsymbol{r}(n\delta t)$ which covers the simulated time evolution as snapshots at multiples of the integration time step $\delta t$. Typically, time steps on the order of fs are used, i.e., it is computationally expensive to reach time scales of ms or even $\mu$s by brute-force integration since $\nabla_i V(\boldsymbol{r}_1, ..., \boldsymbol{r}_N)$ needs to be evaluated for every particle $i$ at every time step. In consequence, enhanced sampling schemes [20] were proposed which, e.g., manipulate the force field $\nabla_i V(\boldsymbol{r}_1, ..., \boldsymbol{r}_N)$ to speed up the system dynamics so that shorter simulations need to be performed. There are plenty of enhanced sampling strategies like umbrella sampling [21], replica exchange MD [18] or metadynamics [19], to name a few.

## 2.2 Dimensionality reduction

While the access to atomistic scales is one key value of MD simulations, the extremely high dimensionality of the delivered data poses significant challenges. First, it is almost impossible for the human mind to understand the full complexity of the concerted motion of $N > 100$ single atoms by just inspecting the time traces of the different coordinates. Second and even more important, most parts of the full dimensional space explored by MD stay untouched or are only sparsely sampled since the high-dimensional volume scales $\propto \prod_{i=1}^{3N}(r_{\max,i} - r_{\min,i})$. This impedes reliable statistical analysis [78]. Luckily, it was found that protein dynamics, as considered in this thesis, can be projected to a rather low-dimensional manifold with just $5 \leq d \leq 10$ degrees of freedom [79–81]. Hence, the determination of these $d$ essential coordinates, also called reaction or collective coordinates, represents a natural first step to analyze MD data. It is also mandatory as preparatory step to apply the modeling approaches discussed later in this thesis. Only if the system is sufficiently simple or if one is already very familiar with it, the so-called dimensionality reduction can be done by human intuition. In general it is necessary to apply some computational approach, various elaborated nonlinear techniques [82–85] or approaches based on machine learning [72, 86–89] were developed. At this point we will only consider the relatively simple principal component analysis (PCA) [90, 91] since it is sufficient to understand the main concepts of dimensionality reduction.
Every dimensionality reduction is based on the assumption that some characteristic of the data, given in the input coordinates $\boldsymbol{x} = (x_1, x_2, ..., x_M)$, like particle positions or distances, can be used to find the essential coordinates. The PCA supposes that the eigensystem of the covariance matrix

$$\mathrm{Cov}_{i,j} = \langle (x_i - \langle x_i \rangle)(x_j - \langle x_j \rangle) \rangle \tag{2.11}$$

reveals the important system dynamics. By ordering the eigenvalues in descending order, $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_{3N}$, the eigenvector $\boldsymbol{v}_1$ represents the direction of maximal variance while $\boldsymbol{v}_2$ covers the second most variance orthogonal to $\boldsymbol{v}_1$ and so on. Fig. 2.1 illustrates the concept. The principal components (PCs) $\mathrm{PC}_i$ are given by projecting $\boldsymbol{x}$ on the eigenvectors

$$\mathrm{PC}_i = \boldsymbol{v}_i^T \cdot \boldsymbol{x} \tag{2.12}$$

so that the variance of $\mathrm{PC}_i$ equals $\lambda_i$.

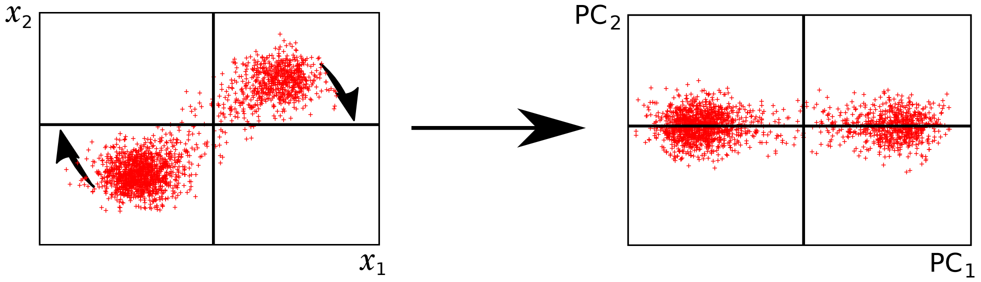

Figure 2.1: **Concept of the PCA.** Starting with some data set (red crosses) given in the coordinates $x_1$ and $x_2$, the PCA can be understood as the rotation to the new coordinates $\mathrm{PC}_1$ and $\mathrm{PC}_2$ where $\mathrm{PC}_1$ represents the direction of maximal variance.

The cumulative variance

$$\Lambda = \sum_{i=1}^{3N} \lambda_i \tag{2.13}$$

is often dominated by the first $d < 10$ eigenvalues. In consequence, a natural next step is to discard all PCs which do not contribute much to $\Lambda$ which leads to the wished low-dimensional system description. In case of Fig. 2.1, $\mathrm{PC}_2$ would be discarded since most variance is covered by $\mathrm{PC}_1$. As the most important feature of the distribution (the separation between the two point clouds) can be covered by $\mathrm{PC}_1$ alone, this dimensionality reduction works as intended. The most important characteristics of the data are preserved although one coordinate was discarded. Nevertheless, if the separation between the two point clouds would be smaller than the orthogonal noise, $\mathrm{PC}_1$ and $\mathrm{PC}_2$ would have changed the order. Consequently, we would have discarded the wrong coordinate and the two point clouds would have overlapped, i.e., the cumulative variance would have been misleading. Because of this, other dimensionality reduction methods choose the low-dimensional space based on the consideration of time scales. The time-lagged independent component analysis (TICA) [92–94], a variation of the PCA, represents an example for this approach. Interpreting the two point clouds in Fig. 2.1 as free energy minima separated by a lowly populated barrier which is rarely crossed, see Sec. 2.3, $\mathrm{PC}_1$ represents the direction of the slowest time scale and would be detected by TICA independent of its variance.

Besides the choice of a suitable dimensionality reduction method, it is also important to use the right input coordinates $\boldsymbol{x}$. The apparently most obvious choice of Cartesian coordinates proves problematic for flexible structures like proteins since it is difficult to separate internal motions from global rotations due to the lack of a single unique reference structure which could be used to identify the latter [95]. This problem can be circumvented by using internal coordinates like atom distances or dihedral angles [27]. While distances are unproblematic from perspective of the PCA, the circularity of dihedral angles makes it difficult to define means and variances. By using sine and cosine of the angles as input coordinates of the PCA, this problem can be avoided. To underline the use of periodic data, this variation of the PCA is often called dihedral PCA (dPCA) [91, 96]. Though, the nonlinearity of sine and cosine leads to distortion effects and the artificial doubling of the input coordinates is suboptimal as well. As solution the dPCA+ was developed [97]. It shifts the periodic boundary of the dihedral angles to the region of lowest point density so that the angles can be used directly.

## 2.3 Free energy

The free energy landscape (FEL) $F(\boldsymbol{x})$, with $\boldsymbol{x}$ representing an $d$-dimensional reaction coordinate, can be used to investigate statistical properties of the considered system, see [4, 5, 98]. It is defined by

$$F(\boldsymbol{x}) = -k_\mathrm{B}T\ln(P(\boldsymbol{x})) + \mathrm{const} \tag{2.14}$$

with the Boltzmann constant $k_\mathrm{B}$, temperature $T$ and the probability distribution $P(\boldsymbol{x})$. The added constant in Eq. (2.14) is typically used to set $F_\mathrm{min} = 0$. We note that $F(\boldsymbol{x})$ highly depends on the chosen $\boldsymbol{x}$. Assuming, for example, that $\boldsymbol{x} = (\mathrm{PC}_1, ..., \mathrm{PC}_d)$ was determined by a PCA, see Sec. 2.2, the free energy represents the projection

$$F(\boldsymbol{x}) \propto -k_\mathrm{B}T\ln\left(\int d\mathrm{PC}_{d+1}... \int d\mathrm{PC}_M \, P(\boldsymbol{x}, \mathrm{PC}_{d+1}, ..., \mathrm{PC}_M)\right) \tag{2.15}$$

with $\mathrm{PC}_{d+1}, ..., \mathrm{PC}_M$ representing discarded higher PCs. Given that $\boldsymbol{x}$ was chosen well, $F$ can be used to identify metastable states in form of local minima, see Sec. 2.5. The connectivity of the states, i.e., which transitions occur, and the energy barriers separating them can be seen as well. The resulting free energy should reveal all important barriers and minima so that we can conclude again that an appropriate choice of $\boldsymbol{x}$ is crucial. At the same time, $\boldsymbol{x}$ should be as low dimensional as possible to optimize the statistical reliability and the interpretability of the data.

## 2.4 Autocorrelation

While the FEL (2.14) provides many information on the statistics of the considered system, dynamical properties, like characteristic time scales of $\boldsymbol{x}$ or transition times between minima, are, in general, only approximately predictable by investigating the barriers. Hence, other observables need to be considered. One possible choice, which uses the temporal information stored in the trajectory $\boldsymbol{x}(t)$, is the (time lagged) autocorrelation $C(\tau)$. For some one-dimensional trajectory $x(t)$, given as discrete time series recorded in equilibrium with time step $\delta t$, the autocorrelation is defined by

$$C(\tau) = \frac{\langle(x(t) - \langle x\rangle_t)(x(t+\tau) - \langle x\rangle_t)\rangle_t}{\langle(x(t) - \langle x\rangle_t)^2\rangle_t} = \frac{\mathrm{Cov}(x(t), x(t+\tau))}{\mathrm{Cov}(x, x)} \tag{2.16}$$

with the lag time $\tau$ and the average $\langle ... \rangle_t$ taken over all times $t \in \{0, \delta t, 2\delta t, ..., (T - \tau)\}$ where $T$ denotes the length of the trajectory. It is assumed that mean and variance of $x(t)$ exist and do not change over time (because of the fact that equilibrium dynamics are considered) so that $C(\tau)$ will only depend on the lag time $\tau$. Since Eq. (2.16) includes the variance in the denominator, $C(\tau)$ starts at $C(0) = 1$.

In case the recorded data consists of a set of trajectories $x_1(t), x_2(t)..., x_N(t)$ with $N$

representing the number of trajectories, the averages $\langle ... \rangle_t$ in Eq. (2.16) need to be extended to $\langle ... \rangle_{i,t}$ with $i$ representing the average over the $N$ subtrajectories. Hence, global means and standard deviations have to be used instead of the individual values of the different trajectories. This results from the consideration that every subtrajectory of the set $x_1(t), x_2(t) ..., x_N(t)$ represents similar dynamics, i.e, mean and standard deviation should not depend on the individual subtrajectory. Note that this consideration only plays a role once the individual trajectories are too short to individually reach global equilibrium because at this point global and individual means and standard deviations are the same.

Considering the interpretation of $C(\tau)$, it can be stated that most biomolecular dynamics $x(t)$ are exposed to some sort of noise with accumulating influences for increasing $\tau$. Trajectory points separated by a large $\tau$ become statistically independent in consequence and $C(\tau) \to 0$ for $\tau \to \infty$ can be observed. The decay time can be seen as estimation of the time scale on which the coordinate $x$ varies. If $x$ resolves different processes at different time scales, the autocorrelation reveals various decay times. This can be covered by using a multi-exponential ansatz $C(\tau) = \sum c_i e^{-\tau/\tau_i}$ [99] where $\tau_i$ represents the time scale of the $i$-th process. If a single process $j$ has a much slower time scale than the others, this approach will again reduce to a mono-exponential function for large $\tau$

$$C(\tau) \approx c_j e^{-\tau/\tau_j}. \tag{2.17}$$

Although the different $\tau_j$ can be understood as estimations of the time scales of the system, it needs to be noted that the various $\tau_j$ represent highly averaged quantities. Individual transitions between two specific minima of the free energy are not resolved if the system shows more than those two minima, for example.

Up to this point it was assumed that we investigate equilibrium dynamics, i.e., $C(\tau)$ only depended on a single time $\tau$. Still, in case we investigate nonequilibrium dynamics, e.g., a system which is exposed to an external time-dependent force, see Sec. 7.1.1, the autocorrelation needs to include a second time variable

$$C(t, t+\tau) = \frac{\langle (x(t) - \langle x(t) \rangle_i)(x(t+\tau) - \langle x(t+\tau) \rangle_i) \rangle_i}{\langle (x(t) - \langle x(t) \rangle_i)^2 \rangle_i}. \tag{2.18}$$

Here, mean and standard deviation depend explicitly on the time $t$, i.e., the averaging $\langle ... \rangle_i$ cannot be done along a single trajectory $x(t)$ but needs to be performed using a set of independent trajectories $x_1(t), x_2(t) ..., x_N(t)$. The trajectories $x_i(t)$ need to have the same initial conditions since nonequilibrium dynamics are very sensitive at this point.

## 2.5 State definition and transition statistics

To deepen the analysis of system dynamics of interest, it might be advantageous to inspect dynamical observables which are more detailed than the autocorrelation. At this point it can be useful to partition the conformational space of the considered system into states so that transition statistics can be calculated and interpreted. Such states represent clusters of points which are sufficiently similar to be grouped together without losing important information, i.e., the definition of states can be understood as a coarse graining.

Assuming that the reaction coordinates $\boldsymbol{x}$ cover all important information of the system, it is straight forward to define states as shown in Fig. 2.2. The four minima of the free energy, see Sec. 2.3, resolved in the two coordinates $x_1$ and $x_2$ represent large clusters of similar points, i.e., they are suitable state centers. The sparsely populated barriers can be used as borders between the states since only a few points might be assigned to the wrong state if the border is not perfectly drawn.

While it is simple, or at least possible, to detect minima and barriers by visual inspection for $d < 4$ dimensions, computational assistance is needed once $d$ becomes larger. Most algorithms used today, also in other scientific fields, are of the k-means type [100]. In the simplest version of k-means the $N$ available data points are separated into $K$ clusters by minimizing the sum $\sum_{i=1}^{k} \sigma_i^2$ of the squared errors $\sigma_i^2 = \sum_{j=1}^{N_i}(x_j - \mu_i)^2$ with $\mu_i$ being the mean of cluster $i$ and $x_j$ representing the $N_i$ points assigned to this cluster. This approach works well if spherically shaped states can be found, like the ones in Fig. 2.2, but struggles if states are entwined so that cluster centers of different states overlap. In addition, the number of clusters $K$ needs to be given as input parameter which means that significant information must be known beforehand. If this is not the case, numerous different $K$ need to be tested without any guarantee that the best $K$ can be detected unambiguously.

Even though there are various modifications and improvements of this simple k-means ansatz, the mentioned problems are, due to the design of k-means, to some extend indispensable. Hence, many density-based clustering approaches have been formulated as alternative [56–61]. These algorithms do not establish all states at once, like k-means does, but use high density regions of the conformational space as "seeding" points of the different states to which data points from regions with less density are assigned successively. In this way the hierarchical nature of the considered free energy is exploited to circumvent the predefinition of the final number of states and to find a state separation which, in principle, does not struggle with irregularly shaped state borders.

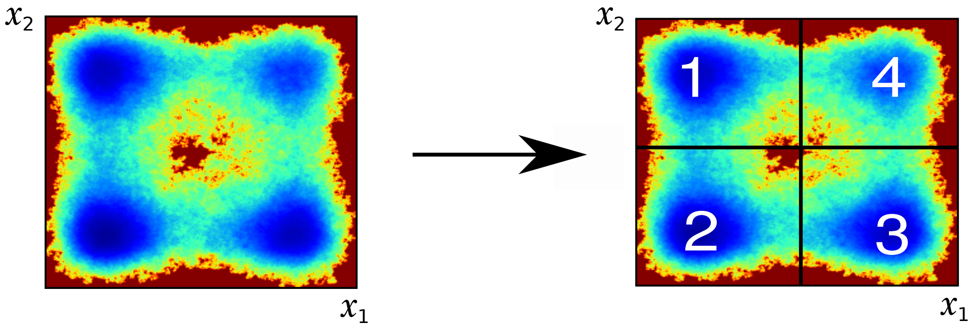

Figure 2.2: **Definition of states.** One possibility to define the states of some system is based on the investigation of the free energy. Since the minima represent sets of similar (from perspective of the chosen reaction coordinates) system configurations, i.e., configurations which appear relatively often, it makes sense to interpret them as the desired states.

Although the choice of a suitable clustering algorithm is undoubtedly very important, we should not forget at this point that the quality of the resulting clustering ultimately depends on the chosen reaction coordinates $\boldsymbol{x}$ as well. If some important state separations are not resolved from the very beginning, they cannot be detected by any clustering. Fig. 2.2 can be used to illustrate the problem. If we used only $x_1$ to describe the system, the states 1 and 2, as well as the states 3 and 4, would overlap completely. As consequence, when applying some clustering to this data, we could only separate two states. It could be argued that there are approaches to define states without the need of a prior dimensionality reduction [101] but practice shows that the computational costs are too high to process sufficiently large numbers of data points. Hence, the problem of choosing a suitable $\boldsymbol{x}$ which resolves all relevant minima and energy barriers can hardly be circumvented.

Assuming that we found a reasonable state separation, we can inspect the observed transition statistics to analyze the connectivity of the states. As first step to do so, the count matrix $N(\tau)$ is determined. The elements $N_{ij}(\tau)$ are just the number of transitions from state $i$ to state $j$ which are observed within a fixed lag time $\tau$. The transition matrix $T(\tau)$ follows from $N(\tau)$ by simple row-normalization, i.e., for the elements of $T(\tau)$ holds

$$T_{ij}(\tau) = \frac{N_{ij}(\tau)}{\sum_{l=1}^{K} N_{il}(\tau)} \tag{2.19}$$

with $K$ being the number of states. Obviously, $T_{ij}(\tau)$ can be understood as the probability to jump from state $i$ to state $j$ within $\tau$. Based on $T(\tau)$, the rate matrix

$$k(\tau) = \frac{1}{\tau} T(\tau) \tag{2.20}$$

follows immediately. We note that all three matrices $N(\tau)$, $T(\tau)$ and $k(\tau)$ heavily depend on the used lag time $\tau$. For small $\tau$ the matrices resolve fast oscillations between neighboring states, i.e., those lag times are prone to overestimate the real transition dynamics due to wrongly assigned data points on or close to the barrier. Large $\tau$, on the other hand, overlook many short-time dynamics which are maybe important to understand the process of interest. In consequence, it is important and often non-trivial to choose an appropriate lag time $\tau$ to resolve the observed state dynamics. Since the transition matrix $T(\tau)$ represents the cornerstone of the popular Markov state models (MSMs) [46–48, 50, 52–55] there are a lot of scientific studies on the question of finding the right $\tau$. MSMs play a prominent role in this thesis and are introduced below in Sec. 3.1.

As last dynamical observable associated with interstate dynamics we introduce the average waiting time $\tau_{\text{wait},i,j}$. It describes the average time between the first trajectory point assigned to state $i$ and the first following point that belongs to state $j$. While $T(\tau)$ includes (mostly) dynamics between neighboring states, $\tau_{\text{wait},i,j}$ measures long range dynamics if $i$ and $j$ are far apart, i.e, it provides more global insights. Being less microscopic, $\tau_{\text{wait},i,j}$ represents an experimentally accessible observable.

## 2.6 Coring of state dynamics

As already touched in the last section, separating a low-dimensional system space into discrete states can be problematic once the barriers are only sparsely sampled or the system coordinates $\boldsymbol{x}$ do not resolve the full state separation. For example, once the

free energy shown in Fig. 2.2 is projected on $x_1$, only two instead of four states can be resolved. But even if all states are still resolved in some low-dimensional state, it is not guaranteed that the state separation turns out to be trivial since the state connectivity can be problematic. Fig. 2.3 illustrates this problem. Here, a two-dimensional model system reveals two distinct states A and B which are connected by a curved transition path. Once the observed dynamics are projected on a single coordinate, the pathway crosses the projected free energy barrier several times per transition. In consequence, if the two states of the projected system are defined by simply cutting at the top of the barrier, the resulting transition matrix will overestimate the system dynamics since it counts too many jumps between the states.
To solve, or at least weaken, this problem, the concept of coring can be applied. Here, state cores are defined by identifying some region around the center of the state where a certain percentage of the total state population can be found. In Fig. 2.3, such regions are indicated by the shaded grey areas. Any transition from state A to state B has to reach the core region of state B before it is counted in the transition matrix. For the example considered in Fig. 2.3, this approach solves the problem of wrongly projected transition pathways.

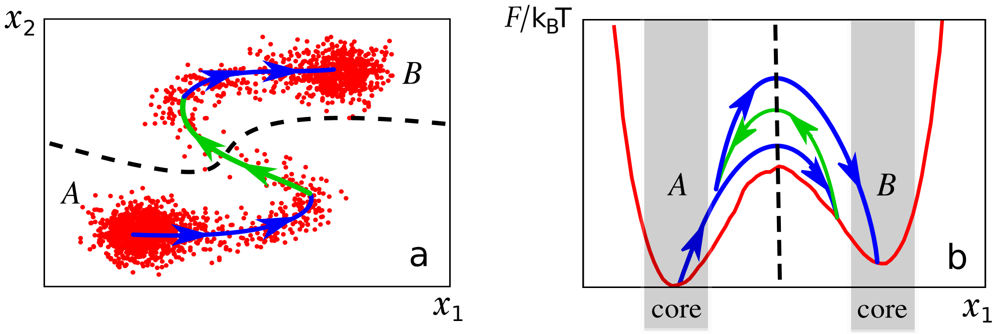

Figure 2.3: **Idea of coring.** (a) The two-dimensional model system consists of two states A and B connected by a single transition path. (b) Once the system dynamics are projected to a single coordinate $x_1$, the transition path cannot be resolved properly and the individual transitions apparently cross the central free energy barrier several times. Simply cutting the two states at the central barrier would lead to an overestimation of the inspected dynamics due to the recrossings of the barrier. Defining state cores (grey regions) or demanding a minimal lifetime to count state transitions (see text) can circumvent this problem. Panels are taken from [66].

While geometrical cores are often used to refine observed state dynamics [47, 62–64], it becomes cumbersome to define the state borders for high-dimensional systems especially if the states become relatively broad or show several subminima. Here, it can help to define state cores not in space but in time. This ansatz, called dynamical coring, scans all observed transition events and demands that the trajectory has to spend at least some minimal $\tau_{\text{lag}}$ in the reached state before the transition is considered as valid [65, 66]. Once this condition is not fulfilled, the trajectory points of the short-living transition are reassigned to the previously visited state. If we assume that the considered system behaves according to a Markov state model, see Sec. 3.1, we will have a simple heuristic to choose the so-called coring time $\tau_{\mathrm{lag}}$, i.e., it will not be arbitrary [65].

# 3 Discrete and continuous Markov models

*"Selig sind die Vergesslichen... [Blessed are the forgetful...]"*
–Friedrich Nietzsche, "Jenseits von Gut und Böse", (1886)

Having introduced several fundamental concepts, this chapter considers the more specialized theory of discrete and continuous Markov models. In the first section we consider the different characteristics of Markov state models. This approach separates the conformational space of the system under study into several discrete states and combines the observed transition statistics to a single transition matrix. This matrix represents the key observable of Markov state models and needs to fulfill several conditions. Afterward we will introduce the Langevin framework which covers the considered system dynamics in terms of free energy, friction and stochastic noise. In contrast to Markov state models, Langevin equations directly work on the system coordinates $\boldsymbol{x}$, i.e., their dynamics are continuous. Here, we will recapitulate the phenomenological derivation of the Langevin equation formulated by Paul Langevin [102] before a microscopic derivation following Robert Zwanzig [31] is presented. One version [32] of the generalized Langevin equation, which includes system memory, will be introduced and simplified to the Markovian Langevin equation which represents the main model framework used in this thesis. Since Langevin equations are only rarely accessible by analytical calculations, we will subsequently consider numerical integrators which can be used to generate trajectories $\boldsymbol{x}(t)$ for further analysis. At the end of this chapter the concept of T-boosting is introduced. This approach allows to reduce the computational time needed to obtain converged Langevin dynamics via numerical integration.

## 3.1 Markov state models

Having divided the conformational space into $K$ states as described in section 2.5, the question arises of how to interpret the observed dynamics. Markov state models (MSMs) are often used at this point due to their conceptional simplicity and the extensive theoretical developments established in the recent years [46–55]. In the following we will recapitulate the basic features of MSMs.

As already indicated by its name, an MSM assumes that the system dynamics observed in state space can be approximated by a Markov chain. Finding the system in state $S_t = \mathrm{i}$ at time $t$, this condition implies that the conditional probability to reach state j after some lag time $\tau$, $P(S_{t+\tau} = \mathrm{j})$, obeys

$$P(S_{t+\tau} = \mathrm{j}) = P(S_{t+\tau} = \mathrm{j}|S_t = \mathrm{i}, S_{t-\tau} = \mathrm{h}, ..., S_0 = \mathrm{g}) = P(S_{t+\tau} = \mathrm{j}|S_t = \mathrm{i}), \tag{3.1}$$

where we assumed equilibrium dynamics, i.e., no explicit dependence on the time. We see that all trajectory points preceding $S_t$ do not influence the probability of the next step $S_{t+\tau}$, the system exhibits no memory. This property makes MSMs very powerful

since, theoretically, numerous short data trajectories of length $\tau$ are sufficient to predict the long-time dynamics of the system under study, i.e., the data does not need to be in global equilibrium as long as all existing state transitions are observed often enough. In practice, however, the data trajectories should be significantly longer than the lag time since $\tau$ needs to be validated by comparing MSM dynamics to the data, see below.
Another consequence of Eq. (3.1) is that any given input trajectory recorded with the time step $\delta t = \tau/n$, $n \in \mathbb{N}$, can be analyzed highly efficiently by separating it into sub-trajectories $S_0 \rightarrow S_\tau ...$, $S_{\delta t} \rightarrow S_{\tau+\delta t} ...$, ..., $S_{\tau-\delta t} \rightarrow S_{2\tau-\delta t} ...$ to construct the MSM based on this ensemble. This "sliding window" approach allows to test different $\tau$ when constructing an MSM without the problem of losing a lot of data for large $\tau$.
When assembling the probabilities $P(S_{t+\tau} = \mathrm{j}|S_t = \mathrm{i})$ to the matrix $T'(\tau)$, it turns out that the maximum likelihood estimator of $T'$ is exactly $T(\tau)$ from Eq. (2.19) [52], just as it is intuitively expected. To provide a valid MSM, $T(\tau)$ needs to fulfill several conditions. First of all, the transition dynamics have to be ergodic. This means that all states need to be dynamically connected, i.e., each state i can be reached when starting at any other state j after waiting long enough and each state is visited infinitely often for $t \rightarrow \infty$. Secondly, based on the Markov condition Eq. (3.1), the transition matrix needs to fulfill the Chapman-Kolmogorov equation

$$T(\tau)^n = T(n\tau) \tag{3.2}$$

with $n = 1, 2, 3, 4, 5, ...$. This equation explains how an MSM predicts long-time observables based on short-time dynamics. Together with the ergodicity of the dynamics, Eq. (3.2) allows to define the unique stationary distribution $\boldsymbol{p}_{\mathrm{eq}}$ based on an arbitrary initial distribution $\boldsymbol{p} = (p_1, p_2, ..., p_K)$ with $\sum_{i=1}^{K} p_i = 1$ via

$$\lim_{n\rightarrow\infty} T^n(\tau)\boldsymbol{p} = \boldsymbol{p}_{\mathrm{eq}} \tag{3.3}$$

which additionally shows that $T(\tau)\boldsymbol{p}_{\mathrm{eq}} = \boldsymbol{p}_{\mathrm{eq}}$ holds. The elements $p_{i,\mathrm{eq}}$ of $\boldsymbol{p}_{\mathrm{eq}}$ describe the share of trajectory points belonging to state i when evaluating a data trajectory in global equilibrium. The stationary distribution allows for the formulation of the last condition on $T(\tau)$ considered in this section: the condition of detailed balance. Since the modeled equilibrium process evolves in thermal equilibrium, reversibility needs to be fulfilled which translates to

$$T_{ij}(\tau)p_{i,\mathrm{eq}} = T_{ji}(\tau)p_{j,\mathrm{eq}} \tag{3.4}$$

which simply means that the number of transitions from state i to j need to be equal to the number of transitions in the opposite direction. Eq. (3.4) prevents the emergence of "loops" in state space, i.e., patterns of shape i→j→k→i, which could act as perpetua mobilia by producing work without any influx of external energy.
Up to this point we assumed that the lag time $\tau$ used to construct the MSM is known or trivial to choose. In practice this is not true, $\tau$ needs to be long enough to overlook all memory effects due to, e.g., short-living oscillations at the state borders. Fortunately, Eq. (3.2) provides possibilities to find a suitable $\tau$ where the inspection of the implied time scales represents the most widely used approach. To calculate the implied time scales we first need to determine the eigenvalues $\lambda_i(\tau_j)$ of $T(\tau_j)$ with $i = 0, 1, ..., K-1$ for a range of lag times $\tau_j$. The largest eigenvalue $\lambda_0(\tau_j) = 1$ is associated with the

stationary distribution, as we see in Eq. (3.3), and the other eigenvalues can be ordered from large to small $\lambda_0(\tau_j) \geq \lambda_1(\tau_j) \geq \lambda_2(\tau_j) \geq ... \geq \lambda_{K-1}(\tau_j) \geq 0$. Then, the implied time scales are defined by

$$t_i(\tau_j) = -\frac{\tau_j}{\ln(\lambda_i(\tau_j))} \,. \tag{3.5}$$

They are ordered from large to small just like the eigenvalues. It should be noted that the different $t_i(\tau_j)$ can be associated with the relaxation time scales of the considered system which can be determined by experiments. When inspecting Eq. (3.2) we see that $\lambda_i(n\tau) = \lambda_i(\tau)^n$ is expected, i.e., the implied time scales of a valid MSM should be constant for varying $\tau$. In practice, one typically observes strongly increasing implied time scales for small $\tau$ followed by approximately constant plateaus for $\tau > \tau_0$. Assuming that the states were defined based on the free energy as described in Sec. 2.5, this can be explained by artifacts introduced by the projection on the low-dimensional system coordinates which might lead to misclassified trajectory points on top of the barrier [47]. Additionally, barrier regions are notoriously undersampled which impedes the correct assignments of the trajectory points even more. Another reason for an increase of the implied time scales could be that the state splitting is simply non-Markovian for time resolutions $\tau < \tau_0$. Independent of the actual reason, $\tau_0$ represents the smallest possible lag time to construct a valid MSM. While the implied time scales provide a way to choose a sufficiently large $\tau$, it needs to be noted that the upper bound on $\tau$ is practically defined by the smallest time which should be resolved by the model. Hence, it advisable to always use the smallest possible $\tau$ since this optimizes the informative value of the MSM.

As additional check of the reliability of a given MSM, it is possible to perform a so-called Chapman-Kolmogorov test [66]. This test calculates the probabilities $p_i(t;\tau_j)$ to be in state i after time $t$ given that the system started in the very same state i at $t = 0$. Once the left side of Eq. (3.2) is used to predict $p_i(t;\tau_j)$, i.e., $T(\tau_j)$ is repeatedly multiplied with $\boldsymbol{p}(0)$, and once the right side, i.e., $T(t)$ is calculated and multiplied a single time with $\boldsymbol{p}(0)$. $T(\tau_j)$ represents a valid MSM if both calculations yield the same prediction of $p_i(t;\tau_j)$ for all $t > \tau_j$. Compared to the implied time scales, the Chapman-Kolmogorov test has the advantage that it provides a direct observable for every state while $T(\tau_j)$ needs to be diagonalized to derive the eigenvalues $\lambda_i(\tau_j)$. This makes it simpler to identify problematic states via the Chapman-Kolmogorov test.

Another consequence of Eq. (3.2) is that the probability $P_{\text{stay},n}(t)$ to stay in state $n$ for a least the time $t$ is expected to decay exponentially. Still, once trajectory points on the barrier are misclassified, intrastate fluctuations are misinterpreted as short-living interstate dynamics, see Sec. 2.6, and $P_{\text{stay},n}$ reveals a fast initial decay. One possibility to optimize $P_{\text{stay},n}$ is to simply reassign the misclassified points via dynamical coring, see Sec. 2.6. Here, the removal of the fast initial decay of $P_{\text{stay},n}$ is used to calibrate the coring time $\tau_{\text{lag}}$, i.e., coring is applied in a self-consistent manner [65]. It is possible to define individual times $\tau_{\text{lag},i}$ for the different states $i$ to minimize the influences of coring. Alternatively, it is also possible to define geometrical state cores by optimizing $P_{\text{stay},n}(t)$.

In case the MSM of a given state separation fails in the different checks for the relevant range of $\tau_j$ even after coring, i.e., the MSM fails for lag times which are small enough to resolve the dynamics of interest, we have to find better states. One strategy to improve the state separation is to derive metastable states from a large number of microstate via an appropriate lumping [61, 64, 103, 104]. Afterwards, one may optimize the transition matrix of the resulting state separation such that an MSM reproduces some key observable like, e.g., the state populations [105]. Alternatively, in case $T(\tau_j)$ works for large $\tau_j$, we can use hidden Markov models to access smaller lag times [106, 107]. Those models take the past of the individual trajectory points into account, i.e., the Markov model is extended by some sort of rudimentary memory.

Assuming that we found a consistent $\tau$ for a suitable state separation of the system under study, we need to find a way to inspect the overall system dynamics, e.g., average waiting times, predicted by the MSM. To this end it is possible to generate Markov chain Monte Carlo (MCMC) simulations were uniformly distributed random numbers between zero and one are used to generate a surrogate trajectory in state space based on the transition matrix $T(\tau)$. This model trajectory can be analyzed and compared to the reference MD simulations.

To conclude this section, it is worth noting that it is also possible to apply the MSM framework to nonequilibrium dynamics [108, 109]. But since we are not using this generalization later in this thesis, we will not consider it here in detail.

## 3.2 Markovian Langevin equation

Having discussed the discrete Markov modeling of state dynamics in the last section, we will now inspect the Markovian Langevin framework. This approach allows for the continuous modeling of dynamics of interest directly based on a set of reaction coordinates $\boldsymbol{x}$, i.e., it circumvents the definition of states. First the phenomenological derivation of Paul Langevin [102] is considered before we inspect a more rigorous derivation following Robert Zwanzig [31].

### 3.2.1 Phenomenological derivation

The Langevin equation (LE) was formulated by Paul Langevin to describe the motion of a Brownian particle through some surrounding fluid [102]. The same problem was treated by Albert Einstein [110, 111] and Marian Smoluchowski [112] only a few years earlier using a different approach. Even today, more than hundred years later, many textbooks introduce the Langevin equation in similar ways [31, 33], i.e., it is worth starting at this point here as well. We will follow the argumentation of Langevin.

To set the stage we consider a spherical particle of radius $a$ and mass $\mathcal{M}$ in thermal equilibrium. It is surrounded by some isotropic fluid characterized by its viscosity $\eta$. The fluid constituents are assumed to be much lighter and smaller than the Brownian particle. Due to the isotropy of the problem, it is sufficient to only consider the motion along one direction $x(t)$ with momentum $p(t)$ since $y(t)$ and $z(t)$ behave in the same way. The equation of motion of $x$ is given by

$$\mathcal{M}\ddot{x} = \dot{p}(t) = F_{\mathrm{tot}}(t)$$

with $F_{\mathrm{tot}}(t)$ being the total force at time $t$. Assuming that external fields, like e.g., gravitation, can be neglected, $F_{\mathrm{tot}}(t)$ represents the interaction of particle and surrounding.

Assuming furthermore that the motion of the Brownian particle does not cause any turbulences, the first part of $F_{\mathrm{tot}}(t)$ can be approximated by the well known Stokes friction [113]

$$F_{\mathrm{fric}}(t) = -6\pi\eta a\dot{x}(t)$$

which acts opposed to the velocity $\dot{x}(t)$ and can be seen as the energy loss induced by collisions of the Brownian particle with the constituents of the fluid. Still, the Brownian particle does not only push the fluid particles away but there are also collisions the other way round. Since these collisions are numerous and random in direction and strength, they can be summed up to a random force

$$F_{\mathrm{random}}(t) = N(t)$$

which can be interpreted as system noise. Summing both force contributions gives the LE of the Brownian particle

$$\dot{p}(t) = -\gamma\dot{x}(t) + N(t) \tag{3.6}$$

with $\gamma = 6\pi\eta a$. The noise $N(t)$ shows different characteristics. Due to the isotropy of the fluid it is safe to state that the mean of $N$ needs to be zero. In addition, considering the mass difference between Brownian particle and fluid constituents, it can be assumed that the Brownian particle evolves on a much slower time scale than the fluid, i.e., the latter forgets any interaction with the former instantaneously. This means that $N(t)$ and $N(t')$ are uncorrelated. Hence, the noise is characterized by

$$\langle N(t)\rangle = 0, \tag{3.7}$$

$$\langle N(t)N(t')\rangle = 2B\delta(t-t'), \tag{3.8}$$

with $\langle ...\rangle$ describing the averaging over time and $B$ accounting for the width of the distribution. Due to the numerous collisions of Brownian particle and fluid constituents it is possible to apply the central limit theorem which means that Eqs. (3.7) and (3.8) are sufficient to describe the distribution of the random force since higher moments can be neglected. This type of noise is often denoted as "white" noise where the term "white" is motivated by the fact that the spectral density of the force

$$S(\omega) = 2\int_{-\infty}^{\infty} e^{i\omega\tau}\langle N(t)N(t-\tau)\rangle d\tau = 2\int_{-\infty}^{\infty} e^{i\omega\tau}2B\delta(\tau)d\tau = 4B$$

is constant just as it is the case for white light. The Wiener-Kinchin theorem [114] can be used to derive this relation. To calculate the constant $B$ describing the strength of the noise we can start with the equipartition theorem [115]. It states for the mean energy $\langle E\rangle$ of the Brownian particle that

$$\langle E\rangle = \frac{k_{\mathrm{B}}T}{2} = \frac{\mathcal{M}\langle\dot{x}(t)^2\rangle}{2} \tag{3.9}$$

is given in thermal equilibrium at temperature $T$ which means that

$$\langle\dot{x}(t)^2\rangle = \frac{k_{\mathrm{B}}T}{\mathcal{M}} \tag{3.10}$$

has to be fulfilled. Furthermore it is possible to deduce $\langle \dot{x}(t)^2 \rangle$ from the LE Eq. (3.6) where we get

$$\langle \dot{x}(t)^2 \rangle = e^{-2\gamma t/\mathcal{M}} \dot{x}(0)^2 + \frac{B}{\gamma \mathcal{M}}(1 - e^{-2\gamma t/\mathcal{M}}), \tag{3.11}$$

as can be seen in [31]. By neglecting the exponential terms for large times $t \to \infty$ and by equating Eq. (3.10) and Eq. (3.11),

$$B = \gamma k_\mathrm{B} T \tag{3.12}$$

can be concluded. This means that random force and friction are connected via

$$\langle N(t) N(t') \rangle = 2\gamma k_\mathrm{B} T \delta(t - t') \tag{3.13}$$

which is known as fluctuation-dissipation theorem (FDT). It is a very fundamental relation which appears in different shapes for many thermodynamical frameworks as long as equilibrium is given. For the studies of this thesis based on the LE it will be used for simulation setups and interpretation purposes.

To conclude this section it is worth noting that the Langevin equation (3.6) needs to be treated with some caution. Due to the stochastic nature of the noise, $p(t)$ is strictly speaking not differentiable, i.e., it makes only sense to interpret $\dot{p} = dp/dt$ in terms of finite $dp$ and $dt$. Sill, Eq. (3.6) is well behaving in the sense that it can be seen as extension of ordinary differential calculus [116].

### 3.2.2 Microscopic derivation

After the phenomenological motivation of the Langevin framework described in the previous section, we will now derive the LE for the unspecified system coordinate $x(t)$ by using a framework which is more abstract than the Brownian particle. This framework is known as Caldeira-Leggett model [117]. We will stick to a one-dimensional $x(t)$ for simplicity, the generalization to more dimensions will be explained at the end of this section.

First of all, we assume that the system coordinate $x$ couples bilinearly to some bath consisting of harmonic oscillators, see the book of Zwanzig [31]. Additionally, the potential $F(x)$, which does not interact with the bath, applies the Newtonian force $dF/dx$ to the system. This potential can be identified as the potential of mean force which is equivalent to the free energy landscape (FEL), see Sec. 2.3, from perspective of the Langevin equation [14, 118–120]. It can be derived from the total potential by integrating out all degrees of freedom assigned to the bath. Based on these two specifications the total microscopic Hamiltonian of system and bath takes the form

$$H_\mathrm{tot} = H_\mathrm{sys} + H_\mathrm{bath} = \frac{p(t)^2}{2} + F(x) + \sum_{i=1}^{N} \left( \frac{p_i(t)^2}{2} + \frac{\omega_i^2}{2} \left( q_i(t) - \frac{c_i}{\omega_i^2} x(t) \right)^2 \right) \tag{3.14}$$

with

$$H_\mathrm{sys}(x, p) = \frac{p(t)^2}{2} + F(x), \tag{3.15}$$

$$H_\mathrm{bath}(x, q_i, p_i) = \sum_{i=1}^{N} \left( \frac{p_i(t)^2}{2} + \frac{\omega_i^2}{2} \left( q_i(t) - \frac{c_i}{\omega_i^2} x(t) \right)^2 \right). \tag{3.16}$$

The momentum $p$ without index refers to the system while $q_i$ and $p_i$ describe the positions and momenta of the degrees of freedom of the bath. The transformation

$$p(t) = \frac{\bar{p}(t)}{\sqrt{\mathcal{M}}} \qquad x(t) = \bar{x}(t)\sqrt{\mathcal{M}} \qquad p_j(t) = \frac{\bar{p}_j(t)}{\sqrt{m_j}} \qquad q_j(t) = \bar{q}_j(t)\sqrt{m_j}$$

was used to make Eq. (3.14) less crowded. Based on $H_{\text{tot}}$, the equations of motion can be deduced

$$\dot{x}(t) = p(t), \tag{3.17}$$

$$\dot{p}(t) = -\frac{dF(x)}{dx} + \sum_{i=1}^{N} c_i \left( q_i(t) - \frac{c_i}{\omega_i^2} x(t) \right), \tag{3.18}$$

$$\dot{q_j}(t) = p_j(t), \tag{3.19}$$

$$\dot{p_j}(t) = -\omega_j^2 q_j(t) + c_j x(t). \tag{3.20}$$

The equations of the bath oscillators $q_j$ can be solved in terms of their initial values $q_j(0)$ and the influence of $x(t)$. The calculation is skipped at this point since it can be found in [31], for example. By inserting the result in the equation of motion of $p$, the generalized Langevin equation (GLE) can be derived

$$\dot{p}(t) = -\frac{dF(x)}{dx} - \int_0^t p(t-s)K(s)ds + N(t) \tag{3.21}$$

after defining the so-called memory kernel $K(t)$

$$K(t) = \sum_{i=1}^{N} \left( \frac{c_i^2}{\omega_i^2} \right) \cos(\omega_i t) \tag{3.22}$$

and the noise

$$N(t) = \sum_{i=1}^{N} c_i \left( p_i(0) \frac{\sin(\omega_i t)}{\omega_i} + \left( q_i(0) - \frac{c_i}{\omega_i^2} x(0) \right) \cos(\omega_i t) \right). \tag{3.23}$$

Compared to the Langevin equation of the Brownian particle, Eq. (3.6), we see that the right side of Eq. (3.21) does not only depend on $p(t)$ but also on prior momenta weighted by $K(t)$. This is the reason to associate this quantity with the memory of the system. It can be shown that a generalized FDT

$$\langle N(t_1) N(t_2) \rangle = k_{\mathrm{B}} T K(|t_1 - t_2|) \tag{3.24}$$

connects noise and memory kernel. To this end it is assumed that the starting values $q_i(0)$, $p_i(0)$ of the bath coordinates follow the Boltzmann distribution $f(q_i, p_i) \propto e^{-H_{\text{bath}}/k_{\mathrm{B}}T}$ [31]. This shows that the FDT holds only in thermal equilibrium, just as it was already noted above. Now, we claim that the system-bath couplings $c_i$ are distributed continuously, i.e., we interpret the bath as an ensemble of numerous oscillators with different frequencies. This allows to define the memory kernel continuously

$$K(t) = \int_0^\infty g(\omega) \frac{c(\omega)^2}{\omega^2} \cos(\omega t) d\omega = \int_0^\infty \frac{f(\omega)}{\omega^2} \cos(\omega t) d\omega \tag{3.25}$$

where $f(\omega)$ represents the spectral density. After inserting this into Eq. (3.21), we get

$$\dot{p}(t) = -\frac{dF(x)}{dx} - \int_0^t p(t-s) \int_0^\infty \frac{f(\omega)}{\omega^2} \cos(\omega t) d\omega ds + N(t) \tag{3.26}$$

Considering that the Langevin equation represents a stochastic model of the dynamics of $x$ it makes sense to assume that the memory kernel $K(t)$ decays to zero for large $t$ since the correlation between $p(t)$ and $p(t-\tau)$ decays for growing $\tau$ due to the permanent disturbance induced by the noise. This motivates the investigation of the case where $p(t)$ does not change on the time scale of the decay of $K(t)$. Inserting $p(s) \approx p(t)$ into the integral in Eq. (3.21) shows that $K(t)$ can be approximated by the $\delta$-function

$$K(t) \approx 2\gamma\delta(t) \tag{3.27}$$

with $\gamma = \int_0^t \int_0^\infty \frac{f(\omega)}{\omega^2} \cos(\omega t) d\omega ds$ without changing the result of Eq. (3.21). This yields the Markovian Langevin equation (LE)

$$\dot{p}(t) = -\frac{dF(x)}{dx} - \gamma p(t) + \sqrt{2k_\mathrm{B}T\gamma}\xi(t) \tag{3.28}$$

after using the FDT

$$\langle N(t_1)N(t_2)\rangle = k_\mathrm{B}TK(|t_1-t_2|) = 2\gamma k_\mathrm{B}T\delta(t_1-t_2) \tag{3.29}$$

The stochastic variable $\xi(t)$ follows a standard normal distribution with $\langle\xi(t_1)\xi(t_2)\rangle = \delta(t_1-t_2)$ and $\langle\xi(t)\rangle = 0$. Eq. (3.28) looks very similar to the LE of the Brownian particle Eq. (3.6), only the deterministic force $dF/dt$ was added. The term "Markovian" in the denomination of Eq. (3.28) illustrates that $\dot{p}(t)$ only depends on $p(t)$ and not on $p(t-\tau)$, just as the transitions of a Markov state model, Sec. 3.1, only depend on the actual system configuration and not on previously visited states.

For completeness we can inspect the limit of large friction. Here, $\langle\dot{p}\rangle$ can be neglected since the frictional force instantaneously damps $p(t)$ back to zero which leads to the overdamped Langevin equation

$$\dot{p}(t) = -\frac{1}{\gamma}\frac{dF(x)}{dx} + \sqrt{\frac{2k_\mathrm{B}T}{\gamma}}\xi(t) \tag{3.30}$$

after inserting $\dot{p} = 0$ in Eq. (3.28). While Eq. (3.30) has the advantage of being a first order differential equation whereas the Markovian Langevin equation is of second order, the restriction to large friction forces represents a severe limitation so that we stick to the Markovian Langevin equation for modeling purposes in this thesis.

In summary, we have seen that it is possible to derive the Langevin framework from a microscopic Hamiltonian. Although this foundation is, mathematically speaking, more robust than the phenomenological derivation by Langevin, Sec. 3.2.1, it needs to be noted that the system-bath interaction was very simple. Additionally, the bath consisted of simple harmonic oscillators. In consequence the presented derivation does not indicate that it is possible to rigorously derive the Langevin equation (might it be GLE (3.21), Markovian LE (3.28) or overdamped LE (3.30)) for an arbitrarily complicated Hamiltonian $H_\mathrm{tot}$.

To close this section we inspect the generalization of the LE for multidimensional systems $\boldsymbol{x}$. To be in line with the definitions used for the data-driven Langevin equation, Sec. 4.1, we first rewrite Eq. (3.28) to

$$\mathcal{M}\ddot{\boldsymbol{x}}(t) = -\nabla F(\boldsymbol{x}) - \Gamma(\boldsymbol{x})\dot{\boldsymbol{x}}(t) + \mathcal{K}(\boldsymbol{x})\boldsymbol{\xi}(t) \tag{3.31}$$

by switching back to explicitly appearing masses and by using $\Gamma(\boldsymbol{x}) = \gamma(\boldsymbol{x})\mathcal{M}$ and $\mathcal{K}(\boldsymbol{x})\mathcal{K}^T(\boldsymbol{x}) = 2k_\mathrm{B}T\Gamma(\boldsymbol{x})$. Please note that friction and noise might depend on $\boldsymbol{x}$ which is conceptionally unproblematic considering the Markovian nature of the equation. Due to $\boldsymbol{x}$ being a vector now, $\nabla F(\boldsymbol{x})$ and $\boldsymbol{\xi}$ represent vectors as well while $\Gamma(\boldsymbol{x})$, $\mathcal{K}(\boldsymbol{x})$ and $\mathcal{M}$ turn into matrices. Since $\mathcal{K}$ is only defined as $\mathcal{K}\mathcal{K}^T$, a way must be found to extract $\mathcal{K}$ from $2k_\mathrm{B}\Gamma(\boldsymbol{x})$. In this thesis we use the Choleskey decomposition at this point. We note that the GLE (3.21) can be extended to more than one dimension as well. Here, the memory kernel $K(t)$ becomes a matrix and the left side of the FDT changes to $\langle \boldsymbol{N}(t_1)\boldsymbol{N}(t_2)^T\rangle$.

## 3.3 Numerical integration

Having discussed the derivation of the Markovian Langevin equation, the next step is to use it to investigate system dynamics in applications. Due to the complicated interplay of $\nabla F(\boldsymbol{x})$, $\Gamma(\boldsymbol{x})$ and $\mathcal{K}(\boldsymbol{x})\boldsymbol{\xi}$, analytic calculations cannot access the Langevin dynamics of nontrivial systems which means that numerical methods must be applied. In this section it is assumed that the free energy $F(\boldsymbol{x})$, the friction $\Gamma(\boldsymbol{x})$ and the temperature $T$ are given and that the FDT is fulfilled. When we use these quantities to derive the trajectory $\boldsymbol{x}(t)$, we will speak of model-based Langevin equation (mLE) simulations. The inverted problem, i.e., a model is determined based on a given $\boldsymbol{x}(t)$, will be considered in Sec. 4.1 where the data-driven Langevin equation (dLE) is introduced.
To generate the trajectory $\boldsymbol{x}(t)$ of an mLE simulation we can choose one of numerous numerical integrators. The first method introduced in this thesis is the (relatively simple) stochastic Euler integrator. It represents the cornerstone of the dLE approach extensively used in this thesis. Afterwards, the more evolved OVRVO integrator is considered and its performance is compared to the Euler integrator by inspecting exemplary mLE simulations.

### 3.3.1 Euler integrator

The numerical approach presented at first is the so-called stochastic Euler or Euler-Maruyama integrator [114]. While there are other more elaborate approaches to solve the LE more precisely, like different versions of the Verlet integrator or the OVRVO scheme [121], the Euler integrator has the advantage of providing relations which are simple to interpret in terms of free energy, friction and temperature. This aspect is used below to derive the dLE approach.
The Euler scheme produces snapshots of $\boldsymbol{x}(t)$ separated by some time step $\delta t$. The value $\boldsymbol{x}(n\delta t)$ at time $t = n\delta t$ is approximated by

$$\boldsymbol{x}(t = n\delta t) = \boldsymbol{x}_n = \boldsymbol{x}_{n-1} + \dot{\boldsymbol{x}}_{n-1}\delta t \tag{3.32}$$

which represents the difference quotient of $\dot{\boldsymbol{x}}$. This approximation becomes exact once $\dot{\boldsymbol{x}} = const$ holds for the whole time $\delta t$ which is, of course, not true for any system influenced by forces and finite $\delta t$. Still, choosing $\delta t$ small enough minimizes the error. To propagate the LE (3.28), the velocity needs to be discretized as well

$$\dot{\boldsymbol{x}}_n = \dot{\boldsymbol{x}}_{n-1} + \mathcal{M}^{-1}\left(-\nabla F(\boldsymbol{x}_{n-1})\delta t - \Gamma(\boldsymbol{x}_{n-1})\dot{\boldsymbol{x}}_{n-1}\delta t + \sqrt{2k_\mathrm{B}T\delta t\Gamma(\boldsymbol{x}_{n-1})}\boldsymbol{\xi}_{n-1}\right) \tag{3.33}$$

so that Eq. (3.32) can be used to get the position $\boldsymbol{x}_n$ based on the parallel propagation of the velocity via Eq. (3.33). The noise $\boldsymbol{\xi}_{n-1}$ consists of random number distributed according to a standard normal distribution $\langle\xi_i(t_1)\xi_j(t_2)\rangle = \delta_{ij}\delta(t_1 - t_2)$. We note that the free energy gradient as well as the friction are multiplied by the time step $\delta t$ whereas the noise is multiplied by $\sqrt{\delta t}$, see Risken [114]. This shows that the noise induces a stochastic and not a deterministic motion of $\boldsymbol{x}(t)$. Considering that $\delta t$ is assumed to be very small, one might suspect that the terms $\propto \delta t$ can be neglected in Eq. (3.33) since $\sqrt{\delta t} > \delta t$. Still, due to its stochastic nature, the noise switches its sign a lot of times for short sequences of consecutive time steps which leads to significant cancellations so that the deterministic forces $\propto \delta t$ can, in general, catch up to provide roughly the same contribution to the dynamics [116].

Besides the Markovian LE we will inspect trajectories produced according the generalized Langevin equation (3.21) as well. To simplify the notation and since we will only use the GLE this way, the following equations assume an one-dimensional system. In addition, we assume that the memory kernel decays monoexponentially and stays independent of $x$, i.e, $K(t) = (\Gamma/\tau_\mathrm{K})e^{-t/\tau_\mathrm{K}}$ holds. This allows for the straightforward generation of the non-Markovian noise. The subsequent equations arise based on the Euler scheme. First, the friction force is discretized

$$f_{\mathrm{fric},n} = \int_0^{n\delta t} K(t-t')\dot{x}(t')dt' = \Gamma/\tau_\mathrm{K}e^{-n\delta t/\tau_\mathrm{K}}\sum_{k=0}^{n-1} e^{k\delta t/\tau_\mathrm{K}}\dot{x}_k\delta t \tag{3.34}$$

and propagated by

$$f_{\mathrm{fric},n} = \Gamma/\tau_\mathrm{K}e^{-\delta t/\tau_\mathrm{K}}\dot{x}_n\delta t + e^{-\delta t/\tau_\mathrm{K}}f_{\mathrm{fric},n-1}. \tag{3.35}$$

Second, the noise is produced using

$$N_n = e^{-\delta t/\tau_\mathrm{K}}N_{n-1} + \frac{\sqrt{2k_\mathrm{B}T\Gamma}}{\tau_\mathrm{K}}\sqrt{\delta t}\xi_n \tag{3.36}$$

which is based on the formal integration of the equation of motion of $N(t)$ [122]

$$\dot{N}(t) = -\frac{1}{\tau_\mathrm{K}}N(t) + \frac{\sqrt{2k_\mathrm{B}T\Gamma}}{\tau_\mathrm{K}}\xi(t)$$

which follows from the fluctuation-dissipation theorem after inserting the monoexponential $K(t)$. As always, $\langle\xi\rangle = 0$ and $\langle\xi(t_i)\xi(t_j)\rangle = \delta(t_i - t_j)$ hold. The velocity is propagated via

$$\dot{x}_{n+1} = \dot{x}_n + \frac{\delta t}{\mathcal{M}}\left(-\frac{dF}{dx} - f_{\mathrm{fric},n} + N_n\right), \tag{3.37}$$

and the position according to Eq. (3.32).

### 3.3.2 OVRVO integrator: Performance compared to Euler

Given that the Euler integrator represents a very simple numerical integrator, it is a good idea at this point to compare its performance to another, more elaborate approach. While the Euler integrator is expected to perform well for sufficiently small time steps $\delta t$, it is plausible that more complicated approaches allow for the use of larger $\delta t$. This is advantageous if long times need to be simulated since the computational costs will go down for larger $\delta t$. At least if the more elaborate approach does not perform significantly more calculations per time step than the Euler integrator.

The alternative approach considered now was developed by Bussi and Parrinello [121] and corresponds to the OVRVO splitting of the propagator $e^{\mathcal{L}\delta t}$ with the Liouville operator $\mathcal{L}$ [123, 124]. Here, the letter O, referring to the Ornstein-Uhlenbeck process [114], represents a stochastic propagation while V and R symbolize deterministic updates of velocity and position, respectively. For the sake of clarity, since this integrator includes exponential functions which are complicated to extend to more than one dimension, we will only consider a one-dimensional system $x$. In the one-dimensional case, the equations of the OVRVO integrator for the propagation of the Markovian LE (3.31) are [121]

$$\dot{x}(n\delta t + \frac{1}{4}\delta t) = \dot{x}_{n+\frac{1}{4}} = c_1(x_n)\dot{x}_n + \frac{c_2(x_n)}{\mathcal{M}}\xi, \tag{3.38}$$

$$\dot{x}_{n+\frac{1}{2}} = \dot{x}_{n+\frac{1}{4}} + \frac{\delta t}{2\mathcal{M}}\frac{dF(x_n)}{dx}, \tag{3.39}$$

$$x_{n+1} = x_n + \dot{x}_{n+\frac{1}{4}}\delta t + \frac{\delta t^2}{2\mathcal{M}}\frac{dF(x_n)}{dx}, \tag{3.40}$$

$$\dot{x}_{n+\frac{3}{4}} = \dot{x}_{n+\frac{1}{2}} + \frac{\delta t}{2\mathcal{M}}\frac{dF(x_{n+1})}{dx}, \tag{3.41}$$

$$\dot{x}_{n+1} = c_1(x_n)\dot{x}_{n+\frac{3}{4}} + \frac{c_2(x_n)}{\mathcal{M}}\xi', \tag{3.42}$$

with $c_1(x) = e^{-\Gamma(x)\delta t/2}$, $c_2(x) = \sqrt{(1 - c_1(x)^2)\mathcal{M}k_\mathrm{B}T}$ and $\xi$, $\xi'$ being two independent normal distributed random numbers. The sequence of the propagations illustrates the name of the integrator, OVRVO. Please note that the steps (3.39) and (3.41) can be combined in our case since their splitting would have been only important if the Hamiltonian had been updated (due to an explicit time dependence) after step (3.40) (which would have led to another update of $x$ directly after the Hamiltonian). It is also worth noting that the steps (3.38), (3.39) and (3.41) only provide virtual points needed to reach $t + \delta t$, they are not meant to describe real dynamics at fractions of $\delta t$.

Now we can compare the performance of Euler and OVRVO integrator. The system which will be considered is specified in Fig. 3.1. The free energy has two minima separated by a single barrier. Two different choices of $\Gamma(x)$ are considered, once $\Gamma = 150$ stays fixed and once $\Gamma(x)$ increases on the barrier. This increase of $\Gamma(x)$ on the barrier is motivated by the findings of Wolf et al. [125]. Here, Langevin models derived by dissipation-corrected targeted MD [42, 126] for sodium chloride, trypsin-benzamidine and the heat shock protein 90 showed exactly this behavior, see Sec. 6.2. Besides friction and free energy, $\mathcal{M} = 1$ ps and $T = 300$ K was set. Please note that we use the unit convention of the dLE at this point, see Sec. A.3, where friction factors are unitless, masses are given in ps and $k_\mathrm{B}T = 38$ ps$^{-1}$ holds at $T = 300$ K.

Based on these parameters, the two integrators, Euler and OVRVO, were used to generate a set of trajectories of 2 $\mu$s length each. The time step $\delta t$ was varied to assess the convergence behaviour of the integrator. Besides the resulting free energy, representing the statistics of the Langevin trajectories, the average waiting times, see last paragraph of section 2.5, of the transitions between the left and the right minimum can be calculated to inspect the model dynamics. The left minimum, called state L in the following, was defined by $x < -1.7$ while the right minimum, called state R, was defined by $x > 1.4$. The dashed lines in Fig. 3.1 show these borders.

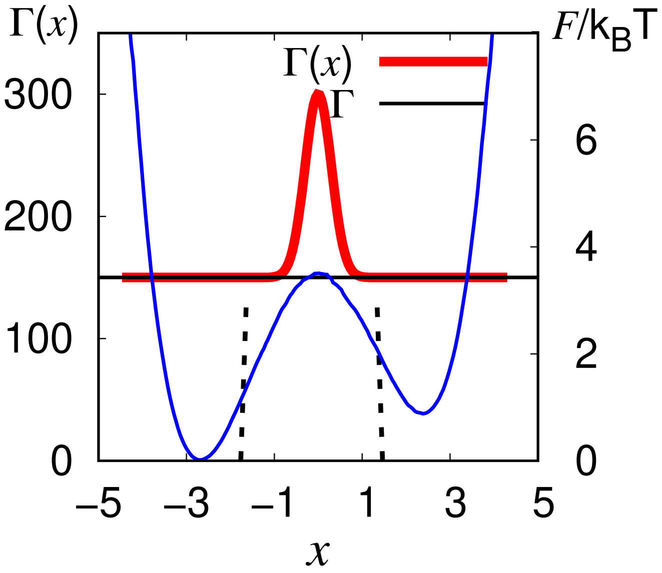

Figure 3.1: **System used to compare Euler and OVRVO integrator.** A free energy with two minima (blue) together with constant friction $\Gamma$ (black) or varying friction $\Gamma(x)$ (red) is used as test system to compare the performance of the two integrators. The two dashed black lines indicate the borders of the two states L and R, see text.

Considering the free energies of the various Langevin simulations in Fig. 3.2, it can be seen in Fig. 3.2a that the Euler integrator provides accurate results up to $\delta t = 0.01$ ps when using the constant $\Gamma = 150$. For larger time steps the barrier gets underestimated and right after $\delta t = 0.013$ ps the simulation produces nonsensical trajectories. For the varying $\Gamma(x)$ the Euler integrator even needs time steps of maximum $\delta t \approx 0.001$ ps to provide correct FELs (Fig. 3.2c), larger time steps result in landscapes featuring an artificial minimum on top of the barrier. The OVRVO integrator, on the other hand, produces accurate free energies up to $\delta t = 0.02$ ps (Fig. 3.2b). When accepting a slight underestimation of the main barrier even $\delta t = 0.05$ ps works satisfying, larger time steps (like $\delta t = 0.1$ ps) show significantly underestimated barriers. Interestingly, the OVRVO integrator does not show any significant difference between simulations using the constant $\Gamma = 150$ and simulations employing the varying $\Gamma(x)$. This observations indicate that the OVRVO integrator allows for the use of significantly longer time steps $\delta t$. However, the average waiting times relativize this conclusion. As shown in Fig. 3.3a, the Euler integrator provides good estimates for $\Gamma = 150$ up to $\delta t = 0.01$ ps, a time step where the OVRVO integrator already starts to fail. When considering the varying $\Gamma(x)$ (top right), in contrast, the Euler integrator fail earlier than the OVRVO integrator but the difference is less than an order of magnitude. As for the free energies we see that the latter integrator behaves the same for both choices of $\Gamma$.

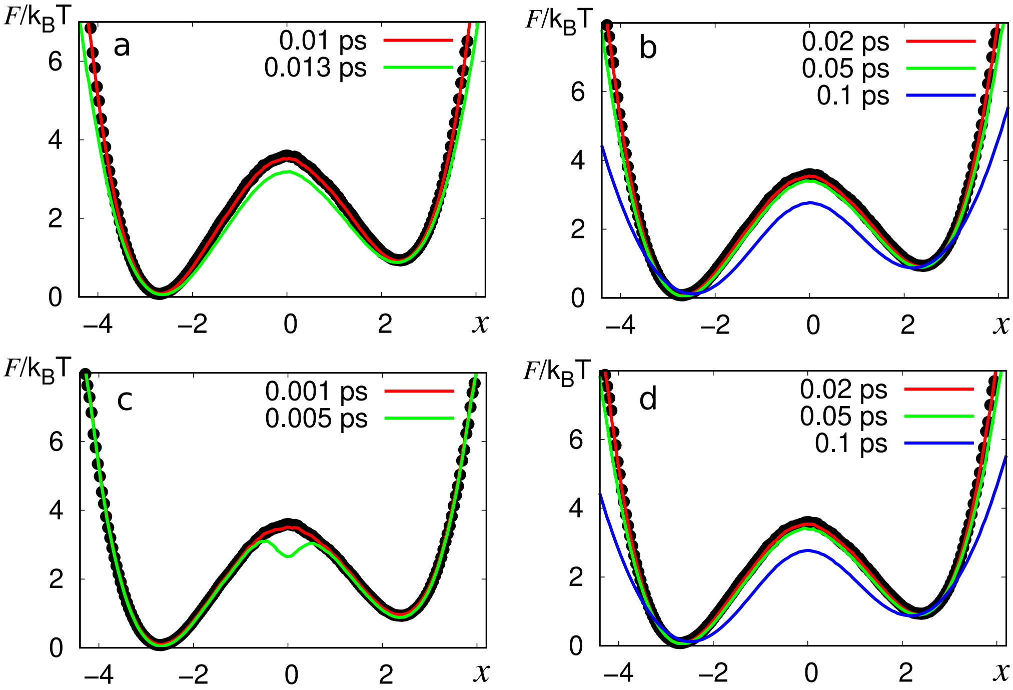

Figure 3.2: **Free energies with Euler and OVRVO integrator.** (a,b) Free energies of Langevin simulations with $\Gamma = 150$ are shown. (c,d) Here, the varying $\Gamma(x)$ is used. (a,c) show results for the Euler integrator while (b,d) consider the OVRVO integrator. Black dots indicate the input free energy while the other colors show Langevin simulations which use the integration time steps given by the legend in the different figures.

Hence, it cannot be stated that the OVRVO integrator is significantly more reliable than the Euler integrator. While this finding appears counterintuitive considering that the OVRVO integrator is significantly more involved than the Euler approach, it is worth noting that it is possible to improve the former by rescaling the time step [123]. The factor

$$b = \sqrt{\frac{2\mathcal{M}}{\Gamma\delta t}\tanh\left(\frac{\Gamma\delta t}{2\mathcal{M}}\right)}$$

can be determined by requiring that the OVRVO integrator preserves the right mean-square displacement of the free diffusive motion through a homogeneous medium [123]. The rescaling can be implemented by substituting $\delta t$ with $b\delta t$ at every explicit occurrence of the time step in the five equations of the OVRVO integrator above, i.e., $c_1$ and $c_2$ stay untouched. For our example this rescaling drastically improves the results for the OVRVO integrator when using $\Gamma = 150$. Time steps up to 0.2 ps can be used which clearly outclasses the Euler integrator, see Fig. 3.3c. This result is not unexpected considering that $\Gamma =$ const directly corresponds to a homogeneous medium for which the factor $b$ was derived. Still, in case of a varying $\Gamma(x)$ this improvement nearly vanishes, i.e., the maximally usable time step of the Euler integrator is not much smaller than the one of the OVRVO integrator.

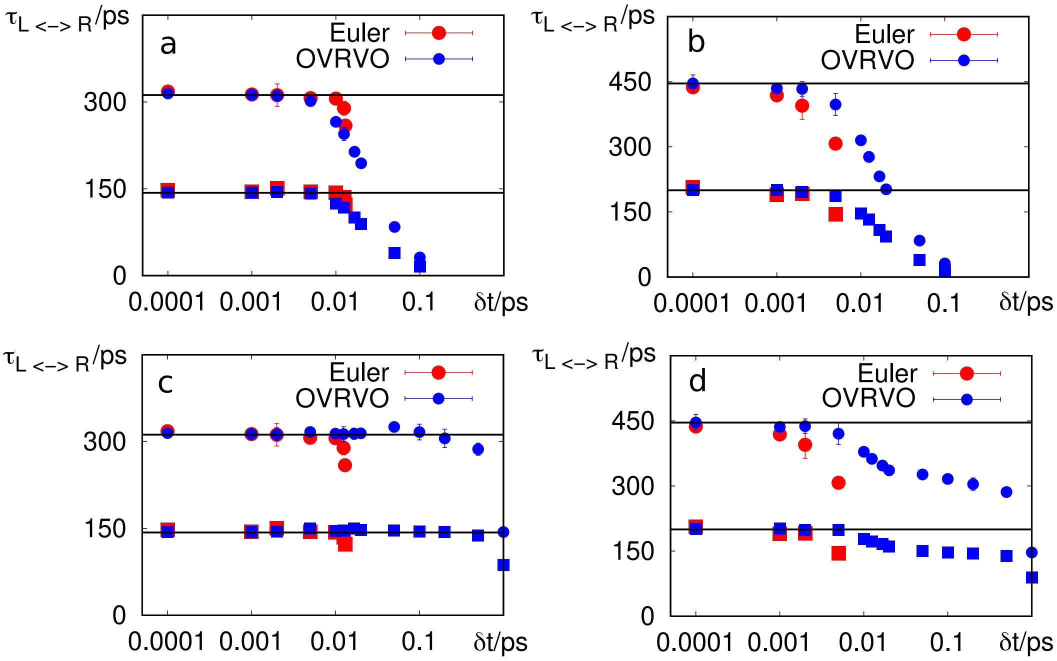

Figure 3.3: **Average waiting times with Euler and OVRVO integrator.** (a,b) Results for simulations based on the Euler and unmodified OVRVO integrator are shown (a) for $\Gamma = 150$ and (b) for varying $\Gamma(x)$. The results for the Euler integrator are shown in red while the OVRVO integrator is pictured in blue. Black lines represent the expected reference values. (c,d) Here, average waiting times of simulations with $\delta t$-rescaling in the OVRVO integrator are presented, see text. Error bars were defined as difference between the average waiting time estimates of first and second half of the respective trajectory.

In summation we have seen that the Euler integrator has its limits when compared to a more sophisticated approach. Nevertheless, just because an integrator works exceptionally well for one setup (here OVRVO with $\Gamma = 150$ and a rescaling of $\delta t$) this does not mean that it works just as well for any other setup (here when using the varying $\Gamma(x)$) so that the practical range of reliable $\delta t$ for a priori unknown Langevin forces, as for applications of the data-driven Langevin equation, will most likely stay comparable to the range of the Euler integrator.

## 3.4 Acceleration of Langevin dynamics via T-boosting

Compared to MD simulations, Langevin dynamics can be integrated much faster and cheaper because the system description $\boldsymbol{x}$ contains much less coordinates than the MD dynamics and the forces are simpler to calculate. This follows from assigning irrelevant degrees of freedom to the bath during dimensionality reduction, i.e, their explicit time evolution is ignored. The resulting simulation speedup motivates the application of Langevin dynamics in the prediction of long-time dynamics via analysis of limited MD data, see for example the applications of the data-driven Langevin equation in this thesis. Still, in case those long-time dynamics are of the order of millisecond or even seconds, see for example the studies in Sec. 6.2, the Langevin equation needs to be integrated up to $10^{17}$ times to get converged observables since, as we have seen in the last section, the time step of the numerical integration scheme has to be on the order of fs or maximally ps to provide accurate results. This is not feasible for standard computing resources which means that a way must be found to circumvent this limitation.

When inspecting Eq.

(3.28) we see that the temperature $T$ influences the Langevin dynamics via the stochastic force, i.e., a larger $T$ results in larger oscillations. When considering the rate $k$ of some Langevin process at two temperatures $T_1$ and $T_2$, it changes according to the Kramers relation [28]

$$k(T_2) = k(T_1) e^{-\Delta F\left(\frac{1}{k_\mathrm{B} T_2} - \frac{1}{k_\mathrm{B} T_1}\right)} \tag{3.43}$$

with $\Delta F$ representing the energy barrier which needs to be crossed during the process. Hence, if $T_2 > T_1$ holds we will observe for $T_2$ more event in some finite simulation time $t_\mathrm{sim}$ than for $T_1$, i.e., the statistics converge faster. Consequently, Eq. (3.43) can be exploited to accelerate Langevin simulations in the following way. First, free energy $F$ and friction $\Gamma$ (and mass $\mathcal{M}$ for completeness) are determined at the target temperature $T$. Second, various different temperatures $T_i > T$ are used to rapidly collect statistics of the considered dynamical process via Langevin simulations. Finally, $k(T)$ is deduced from the different $k(T_i)$ via a fit to Eq. (3.43). This procedure, called "T-boosting" by Wolf et al. [125], was successfully applied to correctly predict rates down to $k \approx 10^{-3}$ $\mathrm{s}^{-1}$ within factor of 5 to 20, see Sec. 6.2 in chapter 6, which is remarkably accurate considering the length of this time scale.

As note of caution it should be pointed out that we cannot expect to predict the real system dynamics at the larger temperatures $T_i$ due to the general temperature dependence of the Langevin fields in biomolecular applications. The produced "pseudo-dynamics" leading to $k(T_i)$ only allow for the determination of $k(T)$ based on Eq. (3.43) and not for any interpretation of the system dynamics at $T_i \neq T$. This becomes obvious when assuming that some biomolecular system is modeled at $T \approx 300$ K. While Eq. (3.43) can predict accelerated dynamics at, e.g., $T = 1000$ K based on the Langevin fields at $T = 300$ K, the real system would be destroyed at this temperature. This important point distinguishes T-boosting from, e.g, temperature accelerated MD [127] where the free energy $F(\boldsymbol{x})$ is determined at a high temperature and then rescaled to the target temperature $T$. This might by problematic considering that $F(\boldsymbol{x})$ probably depends on $T$. Our approach of T-boosting, in contrast, estimates the fields at $T$ and only uses the larger $T_i$ to predict the dynamics via Eq. (3.43).

## 3.5 Summary

This chapter introduced in Sec. 3.1 the concept of Markov state models which describe system dynamics in terms of memory-less jumps between discrete states. Afterwards, Sec. 3.2 introduced the Langevin equation which covers the system dynamics in terms of continuous motions driven by free energy, friction and stochastic noise. We considered the generalized Langevin equation and discussed the simplification to the Markovian Langevin equation. Sec. 3.3 considered numerical integrators which can be used to simulate Langevin trajectories. We compared the performance of the relatively simple Euler integrator to the more evolved OVRVO integrator and observed that the former, despite its simplicity, does not perform significantly worse than the latter. This finding indicates that it is reasonable to assume that the data-driven Langevin equation, introduced in the next chapter and based on the Euler integrator, will not be hindered by a impractical integrator. In the last section of this thesis we introduced the concept of T-boosting. Given a Langevin model at some temperature $T$, this approach integrates the Langevin equation at higher temperatures $T_i$ to accelerate the process of interest and extrapolates the dynamics at $T$ via a Kramers relation. As we will see in chapter 6, T-boosting allows us to access timescales of the order of seconds.

# 4 Data-driven Markov modeling

*"Die Erfahrung ist fast immer eine Parodie auf die Idee.*
*[The experience is almost always a parody of the idea.]"*
–Johann Wolfgang von Goethe, "Tagebücher. 3. Schweizer Reise", (1797)

We saw that it is possible to solve the Markovian Langevin equation numerically if free energy, friction and temperature are given. Still, those forces are in general not obvious if we want to model an unknown system. This means that some way to extract them from given data needs to be found and many attempts to do so have been formulated [38, 39, 128–131]. In this thesis we will mainly use the data-driven Langevin equation (dLE) approach introduced by Hegger and Stock [132] and further developed by Schaudinnus et al. [43–45]. The fundamental formulas and derivations are shown at the beginning of this chapter. Directly afterwards we will apply the dLE to exemplary data to inspect the influence of its crucial parameter: the time step $\delta t$. Based on our observations we will introduce the rescaled dLE which allows for the optimization of the dLE performance at small $\delta t$. To enable the study of extensive data sets by the dLE, we will afterward consider the question of the reliable reduction of large input data sets. It will be shown that the removal of redundant data points may spoil the model dynamics, instead, one can use an appropriate pre-averaging of the data. Having considered the established dLE formulation based on the Euler integrator, we will subsequently inspect the performance of a varied dLE formulation based on a Verlet integrator. As alternative approach of parameterizing the Markovian Langevin equation, we will afterwards consider the basic formulas of the dissipation-corrected targeted MD (dcTMD) framework established by Wolf and Stock [42]. This approach provides one-dimensional Langevin models based on constraint MD simulations. Subsequently, the chapter closes by presenting the results of the Markov state modeling of exemplary Markovian input data.

## 4.1 Data-driven Langevin equation (dLE)

The specific approach used in this thesis, called data-driven Langevin equation (dLE), generates Langevin trajectories $\boldsymbol{y}(t)$ based on given data points $\boldsymbol{x}(t)$ by iterating two steps. Assuming that the Langevin dynamics reached position $\boldsymbol{y}_n$ at time $t = n \cdot \delta t$, the first step is to search for the $k$ next neighboring data points $\boldsymbol{x}_i$. These neighbors can be used in the second step to estimate the Langevin fields "on-the-fly" in a local manner, i.e., the forces might depend on $\boldsymbol{y}_n$ and are constructed in parallel to the Langevin trajectory $\boldsymbol{y}_n$.

### 4.1.1 Fundamental equations

To introduce the dLE in detail following previous work [43–45, 132] we first recapitulate that the Markovian LE is determined by the drift field $\boldsymbol{f}$, which is the gradient of

the free energy, the friction field $\Gamma$ and the noise amplitude $\mathcal{K}$. $\Gamma$ and $\mathcal{K}$ are related via the fluctuation-dissipation theorem (3.29) in equilibrium. Assuming that the three fields are not known for the system under study, we need to extract the forces from some given stochastic trajectory $\boldsymbol{x}(t)$ recorded at a time resolution of $\delta t$. The resulting field estimates can be used to propagate a new Langevin trajectory $\boldsymbol{y}(t)$ which can be compared to $\boldsymbol{x}(t)$ to validate the Langevin model. Additionally, $\boldsymbol{y}(t)$ can be used to predict long-time dynamics which cannot be calculated from the data.
At this point it is obviously assumed that the dynamics of $\boldsymbol{x}(t)$ are Markovian, i.e., it needs to make sense to approximate the equations of motion by the Markovian LE (3.28). By approximating

$$\ddot{\boldsymbol{x}}_m = \frac{d\dot{\boldsymbol{x}}_m}{dt} \approx \frac{\dot{\boldsymbol{x}}_{m+1} - \dot{\boldsymbol{x}}_m}{\delta t} \tag{4.1}$$

with the finite time step $\delta t$ this indicates that the velocity of the data $\dot{\boldsymbol{x}}_m = d\boldsymbol{x}_m/dt$ obeying

$$\dot{\boldsymbol{x}}_{m+1} = \mathcal{M}^{-1}\boldsymbol{f}(\boldsymbol{x}_m)\delta t - (\mathcal{M}^{-1}\delta t\Gamma(\boldsymbol{x}_m) - \mathbb{1})\dot{\boldsymbol{x}}_m + \mathcal{M}^{-1}\mathcal{K}(\boldsymbol{x}_m)\boldsymbol{\xi}_m\delta t^{1/2} \tag{4.2}$$

can be reproduced by the velocities of the Langevin trajectory $\dot{\boldsymbol{y}} = d\boldsymbol{y}/dt$

$$\dot{\boldsymbol{y}}_{n+1} = \mathcal{M}^{-1}\boldsymbol{f}(\boldsymbol{y}_n)\delta t - (\mathcal{M}^{-1}\delta t\Gamma(\boldsymbol{y}_n) - \mathbb{1})\dot{\boldsymbol{y}}_n + \mathcal{M}^{-1}\mathcal{K}(\boldsymbol{y}_n)\boldsymbol{\xi}_n\delta t^{1/2} \tag{4.3}$$

with, e.g., $\boldsymbol{f}(\boldsymbol{y}_n) = \boldsymbol{f}(\boldsymbol{x}_m)$ if $\boldsymbol{y}_n = \boldsymbol{x}_m$ holds. Both equation, (4.2) and (4.3), coincide with the Euler integrator introduced in Sec. 3.3.1. The two different indices $m$ and $n$ should emphasize that $\boldsymbol{x}(t)$ and $\boldsymbol{y}(t)$ are not identical, i.e., $\boldsymbol{y}(t)$ does not simply follow $\boldsymbol{x}(t)$. Note that $\boldsymbol{f}$ represents a vector in case of more than one dimension, whereas $\Gamma$, $\mathcal{K}$ and $\mathcal{M}$ become matrices. Inserting the Euler discretization (4.1) again to reach $\boldsymbol{y}$ via

$$\dot{\boldsymbol{x}}_m \approx \frac{\boldsymbol{x}_m - \boldsymbol{x}_{m-1}}{\delta t} = \frac{\Delta\boldsymbol{x}_m}{\delta t} \tag{4.4}$$

allows us to write

$$\boldsymbol{x}_{m+1} = \boldsymbol{x}_m + \hat{\boldsymbol{f}}(\boldsymbol{x}_m) - \hat{\Gamma}(\boldsymbol{x}_m)\Delta\boldsymbol{x}_m + \hat{\mathcal{K}}(\boldsymbol{x}_m)\boldsymbol{\xi}_m \tag{4.5}$$

and

$$\boldsymbol{y}_{n+1} = \boldsymbol{y}_n + \hat{\boldsymbol{f}}(\boldsymbol{y}_n) - \hat{\Gamma}(\boldsymbol{y}_n)\Delta\boldsymbol{y}_n + \hat{\mathcal{K}}(\boldsymbol{y}_n)\boldsymbol{\xi}_n \tag{4.6}$$

where we defined the so-called dLE fields

$$\hat{\boldsymbol{f}}(\boldsymbol{y}_n) = \mathcal{M}^{-1}\delta t^2\boldsymbol{f}(\boldsymbol{y}_n), \tag{4.7}$$
$$\hat{\Gamma}(\boldsymbol{y}_n) = \mathcal{M}^{-1}\delta t\Gamma(\boldsymbol{y}_n) - \mathbb{1}, \tag{4.8}$$
$$\hat{\mathcal{K}}(\boldsymbol{y}_n) = \mathcal{M}^{-1}\delta t^{3/2}\mathcal{K}(\boldsymbol{y}_n). \tag{4.9}$$

As announced above we aim for a local estimate of the different fields. Due to the stochastic nature of the dynamics, some sort of averaging needs to be performed to eliminate the influence of the individual noise realizations. To this end we use the $k$ next neighbors $\boldsymbol{x}_i$ of the dLE point $\boldsymbol{y}_n$ and estimate

$$B(\boldsymbol{y}_n) = \langle B(\boldsymbol{x}_i)\rangle = \frac{1}{k}\sum_{i=1}^{k} B(\boldsymbol{x}_i), \tag{4.10}$$

which is valid as long as the neighbors $\boldsymbol{x}_i$ are sufficiently close to $\boldsymbol{y}_n$. The number $k$ can be seen as the counterpart of the radius $\epsilon$ of some hypersphere centered at $\boldsymbol{y}_n$

$$\sum_{i=1} \Theta(|\boldsymbol{x}_i - \boldsymbol{y}_n| - \epsilon) = k \tag{4.11}$$

with $\Theta$ representing the Heavyside function. Fig. 4.1 illustrates this comparison.

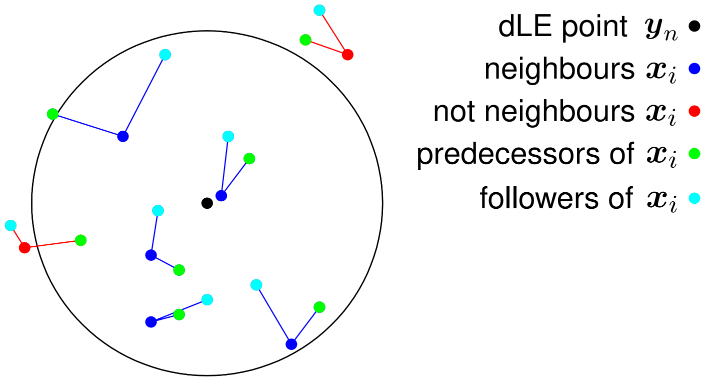
dLE point $\boldsymbol{y}_n$ ●
neighbours $\boldsymbol{x}_i$ ●
not neighbours $\boldsymbol{x}_i$ ●
predecessors of $\boldsymbol{x}_i$ ●
followers of $\boldsymbol{x}_i$ ●

Figure 4.1: **Illustration of the neighborhood estimation.** The dLE point $\boldsymbol{y}_n$ (black) is surrounded by its $k$ next neighbors $\boldsymbol{x}_i$ (blue). Each data point $\boldsymbol{x}_i$ has a predecessor (green) and a follower (cyan) which are used to estimate the dLE fields. Data points which are further away (red) are not considered.

The hypersphere representing the neighborhood of $\boldsymbol{y}_n$ grows and shrinks depending on the local density of input data points. Since local fields are desired, $\epsilon(k)$ needs to be sufficiently small. This means that a satisfying field locality, indicating a small $k$, needs to be balanced against a sufficiently reliable averaging which indicates a large $k$. When dealing in practice with data sets with more than $10^6$ points, $k \approx 10^2$ is typically chosen. Based on a reasonable choice for $k$, the three dLE fields can be determined. Detailed calculations can be found in Sec. A.1. The dLE friction can be deduced from the time-lagged covariance matrix

$$\begin{aligned}\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m) &= \mathrm{Cov}(\hat{\boldsymbol{f}}(\boldsymbol{x}_m) - \hat{\Gamma}(\boldsymbol{x}_m)\Delta\boldsymbol{x}_m + \hat{\mathcal{K}}(\boldsymbol{x}_m)\boldsymbol{\xi}_m, \Delta\boldsymbol{x}_m) \\ &= -\hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m)\end{aligned}$$

using the deterministic nature of the drift field $\hat{\boldsymbol{f}}$ and $\mathrm{Cov}(\boldsymbol{\xi}_m, \Delta\boldsymbol{x}_m) = 0$ which results from the white noise properties of $\boldsymbol{\xi}$. Eq. (4.10) is used to carry out the averages which are needed to calculate the covariances. Isolating $\hat{\Gamma}(\boldsymbol{y}_n)$, we get

$$\hat{\Gamma}(\boldsymbol{y}_n) = -\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m)\mathrm{Cov}^{-1}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m). \tag{4.12}$$

The drift field $\hat{\boldsymbol{f}}$ can be determined using the friction estimate via

$$\hat{\boldsymbol{f}}(\boldsymbol{y}_n) = \langle\Delta\boldsymbol{x}_{m+1}\rangle + \hat{\Gamma}(\boldsymbol{y}_n)\langle\Delta\boldsymbol{x}_m\rangle. \tag{4.13}$$

The noise amplitude is a bit more complicated to determine. We have to insert the drift estimate Eq. (4.13) into Eq. (4.5)

$$\Delta \boldsymbol{x}_{m+1} = \langle \Delta \boldsymbol{x}_{m+1} \rangle + \hat{\Gamma}(\boldsymbol{x}_m)(\langle \Delta \boldsymbol{x}_m \rangle - \Delta \boldsymbol{x}_m) + \hat{\mathcal{K}}(\boldsymbol{x}_m)\boldsymbol{\xi}_m$$

so that the covariance $\mathrm{Cov}(\Delta \boldsymbol{x}_{m+1}, \Delta \boldsymbol{x}_{m+1})$ can be rearranged to

$$\hat{\mathcal{K}}(\boldsymbol{y}_n)\hat{\mathcal{K}}^T(\boldsymbol{y}_n) = \mathrm{Cov}(\Delta \boldsymbol{x}_{m+1}, \Delta \boldsymbol{x}_{m+1}) - \hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}(\Delta \boldsymbol{x}_m, \Delta \boldsymbol{x}_m)\hat{\Gamma}^T(\boldsymbol{y}_n) \tag{4.14}$$

using again the white noise properties of $\boldsymbol{\xi}$ and the fact that $\langle \Delta \boldsymbol{x}_{m+1} \rangle$ as well as $\langle \Delta \boldsymbol{x}_m \rangle$ are local constants. To simplify Eq. (4.14) it is possible to use Eq. (4.12) together with $\mathrm{Cov}^{-1}(\Delta \boldsymbol{x}_m, \Delta \boldsymbol{x}_m) = (\mathrm{Cov}^{-1}(\Delta \boldsymbol{x}_m, \Delta \boldsymbol{x}_m))^T$ so that

$$\hat{\mathcal{K}}(\boldsymbol{y}_n)\hat{\mathcal{K}}^T(\boldsymbol{y}_n) = \mathrm{Cov}(\Delta \boldsymbol{x}_{m+1}, \Delta \boldsymbol{x}_{m+1}) + \hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}(\Delta \boldsymbol{x}_m, \Delta \boldsymbol{x}_{m+1}). \tag{4.15}$$

Using the fact that any covariance is positive-definite and Hermitian, the Choleskey decomposition can be employed to determine $\mathcal{K}(\boldsymbol{y}_n)$ which results in a noise amplitude of lower triangular shape for any multidimensional system.

Based on estimation equations above it is possible to determine the noise which would have been needed to generate the data trajectory $\boldsymbol{x}(t)$ in the LE framework. This means that severe contradictions to the Markovianity assumption can be detected by the dLE itself. To do so, the dLE only needs to estimate the fields at all data points $\boldsymbol{x}_m$. Since $\boldsymbol{x}_{m+1}$ is already defined, the noise trajectory $\boldsymbol{\xi}_m$ can be deduced via

$$\boldsymbol{\xi}_m = \hat{\mathcal{K}}(\boldsymbol{x}_m)^{-1}\left(\Delta \boldsymbol{x}_{m+1} - \hat{\boldsymbol{f}}(\boldsymbol{x}_m) + \hat{\Gamma}(\boldsymbol{x}_m)\Delta \boldsymbol{x}_m\right) \tag{4.16}$$

which means that it is possible to check whether or not $\boldsymbol{\xi}_m$ fulfills the Markovian requirements, i.e., it needs to be delta-correlated with mean zero. Besides the input noise it is additionally possible to use the dLE fields to derive the system mass $\mathcal{M}$ which is, of course, always dependent on the system description. To this end we remember that $\Gamma$ and $\mathcal{K}$ are connected via the fluctuation-dissipation theorem (3.29). By inserting the equations (4.8) and (4.9) into Eq. (3.29), it can be seen that

$$\hat{\mathcal{K}}(\boldsymbol{y}_n)\hat{\mathcal{K}}(\boldsymbol{y}_n)^T\mathcal{M}^T = 2k_\mathrm{B}T\delta t^2(\hat{\Gamma}(\boldsymbol{y}_n) + \mathbb{1}). \tag{4.17}$$

holds.

### 4.1.2 Influence of the dLE time step

Since the estimators of the three Langevin fields, (4.12), (4.13) and (4.15), depend on the displacements $\Delta \boldsymbol{x}_i$, the time step $\delta t$ chosen to evaluate $\Delta \boldsymbol{x}_i$ plays a major role in model consistency. Just as the lag time $\tau$ used to construct an MSM needs to be large enough to ensure the Markovianity of the state dynamics, $\delta t$ needs to be sufficiently large to validate the memory-less nature of the Markovian Langevin framework. On the other hand $\delta t$ must be chosen small enough to ensure that the estimated fields are truly local, especially the free energy landscape needs to be resolved sufficiently fine. This condition can also be interpreted as the requirement that the numerical integrator of the dLE needs to converge. Thus, in general we cannot expect to choose $\delta t$ completely free, there will be some valid value range $\delta t \in [\delta t_\mathrm{M}, \delta t_\mathrm{R}]$ where the subscripts M, R remind of

Markovianity and resolution as bounding conditions of $\delta t$.

This consideration raises the question of how to detect the two borders $\delta t_{\mathrm{M}}$ and $\delta t_{\mathrm{R}}$. The latter time step is relatively straightforward to detect in practice. Since the model dynamics and especially the free energy should be independent of $\delta t$, it is simply possible to compare the predicted free energies of dLEs at different $\delta t_i$. The upper bound $\delta t_{\mathrm{R}}$ reveals itself as the last time step where all free energy extrema are sharply resolved. Once $\delta t > \delta t_{\mathrm{R}}$, the predicted free energy starts to look blurry, i.e., minima are not as deep as for smaller $\delta t$ and the barriers are broadened. The lower bound $\delta t_{\mathrm{M}}$, on the other hand, is more complicated. Although Eq. (4.16) allows to calculate the noise observed in the data, it needs to be kept in mind that this calculation already assumes that the Langevin framework can be applied to the data. Hence, we can only use Eq. (4.16) to perform some self-consistency test, there is no guarantee that all possible contradictions between input data and Langevin model can be tracked. Consequently the estimation of $\delta t_{\mathrm{M}}$ via Eq. (4.16) should be treated with caution, it is possible that $\delta t$ needs to be chosen significantly larger due to hidden memory effects which stay undetected by the noise check due to some cancellation of errors.

In the following we are going to inspect the influence of the choice of $\delta t$ on the dLE performance. First of all, the ideal case is considered: perfectly Markovian data produced according to Eq. (4.5). To be more specific, the dynamics of the dimensionless coordinate $x(t)$, see Fig. 4.2, explores a standard double well superimposed by a sine function which aims to mimic a rugged energy landscape as it is observed in biomolecular dynamics. The main barrier at $x = 0$ is significantly larger than the secondary barriers caused by the sine overlay. As additional parameters we use a constant friction of $\Gamma = 3000$, the temperature $T = 300$ K and $\mathcal{M} = 400$ ps. The reference trajectory of $10^7$ points is produced by numerically integrating the Markovian Langevin equation with a time step of $\delta t_0 = 20$ fs by using the Euler integrator, i.e., we record system dynamics for 200 ns which provides converged statistics. This data constitutes the input for our following study of the influence of $\delta t$ on the performance of the dLE. To circumvent any influence of varying sampling quality, we do not simply take any $m$th point of the data to perform dLE simulations at $m\delta t_0$. Instead, the reference trajectory is separated into $m$ sub-trajectories (starting at $t = 0, \delta t_0, ..., (m-1)\delta t_0$ with time step $\delta t = m\delta t_0$) and the whole ensemble of trajectories is used as input for the dLE.

First, we want to detect the upper bound $\delta t_{\mathrm{R}}$. To this end the free energies of dLE simulations at different $\delta t$ are compared in Fig. 4.2a. It turns out that the sine overlay is only resolved for small time steps $\delta t \leq 0.2$ ps but the main barrier is reliably reproduced up to $\delta t \leq 20$ ps. Once $\delta t$ exceeds this limit, the overall shape of $F(x)$ is lost, i.e., $\delta t$ becomes so large that the respective displacements $\Delta x_n(\delta t)$ do not allow for a correct numerical integration of the Langevin equation. Considering the other two fields $\Gamma$ and $\mathcal{K}$, see Fig. 4.2b, we observe qualitatively correct estimates only for $\delta t \leq 4$ ps, both fields start to grow for larger time steps. We note that the field estimates develop some dependence on $x$ for $\delta t \geq 1$ ps, see Fig. 4.4b, but this is unimportant for the dLE predictions of free energy and dynamical quantities like transition times (see below). This demonstrates that not all features of the dLE field estimates are relevant for the performance of the Langevin model. After all, the consideration of the different Langevin fields indicates that $\delta t_{\mathrm{R}} = 4$ ps can be assumed as upper bound on the suitable time steps if we accept that the secondary barriers, i.e., the sine overlay, are not resolved. If we want to resolve those barriers we will have to set $\delta t \leq 0.2$ ps instead.

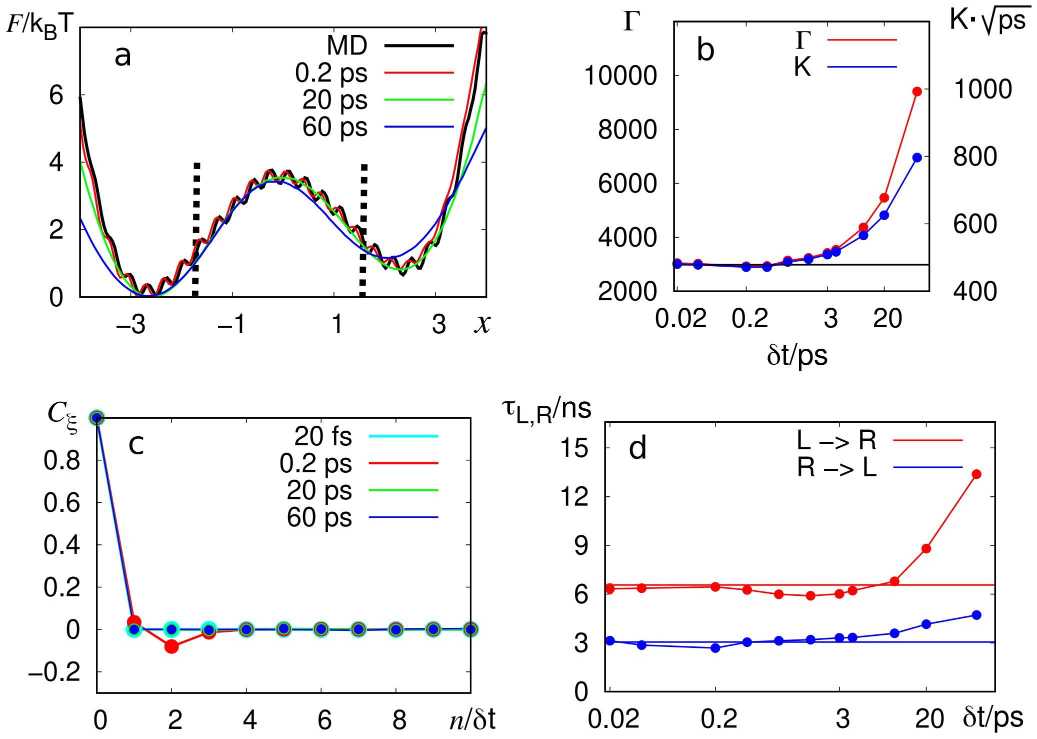

Figure 4.2: **Markovian double well model.** (a) Shown are the free energies explored by the input data (MD) and dLEs at different time steps. The dashed lines show the cores of the two states R and L (see text) which are used to quantify the system dynamics. (b) Here, the dLE estimates of friction $\Gamma$ and noise amplitude $\mathcal{K}$ for different time steps are presented. The black line shows the fields used in the input data. (c) The autocorrelation of the reconstructed noise at different $\delta t$. (d) We see the transition times $\tau_{\mathrm{L,R}}$ estimated by the dLE at different $\delta t$ (dots) compared to the expectations (lines). In red we see L→R while blue represents R→L.

While we see that $\delta t_{\mathrm{R}}$ can be complicated to choose, the lower bound $\delta t_{\mathrm{M}}$, on the other hand, is expected to be trivial for our specific example. Since there was no memory at all, $\delta t_{\mathrm{M}} = \delta t_0$ should hold. The noise autocorrelation shown in Fig. 4.2c reveals that this is indeed true, each curve decays in just one time step which indicates that the back-calculated noise is $\delta$-correlated. Averages, standard deviations and noise distributions (not shown) meet the expectations (normal distribution) as well. Hence, we expect that $\delta t \in [\delta t_0, 4\text{ ps}]$ represents a suitable value range for dLE modeling. To verify this assumption one can inspect the transition times $\tau_{\mathrm{L}}$ and $\tau_{\mathrm{R}}$ representing the average waiting times of the transition from the left minimum (state L) to the right minimum (state R) and backwards. To exclude spurious oscillations on top of the barrier, we only count transitions between the cores of the two states defined by $x_L \leq -1.7$ and $x_R \geq 1.4$, see the dashed lines in Fig. 4.2a. As can be seen in Fig. 4.2d, the dLE truly estimates the right times of $\tau_{\mathrm{L}} = \tau_{\mathrm{L}\to\mathrm{R}} = 6.6$ ns and $\tau_{\mathrm{R}} = \tau_{\mathrm{R}\to\mathrm{L}} = 3$ ns for $\delta t \in [\delta t_0, 4 \text{ ps}]$ which indicates, together with the correctly reproduced main barrier of the free energy, perfectly valid dLE models.

Up to now we have seen that the dLE reproduces free energy, friction and noise amplitude as long as $\delta t$ is sufficiently small. But by also considering the estimated mass $\mathcal{M}$, the situation becomes more complicated. As shown in Sec. 4.1.1, the mass estimate $\mathcal{M}$ of the dLE can be calculated based on $\hat{\Gamma}$ and $\hat{\mathcal{K}}$ using Eq. (4.17). Alternatively, the equipartition theorem

$$\mathcal{M} = \frac{k_{\mathrm{B}}T}{\langle \dot{x}^2 \rangle} \tag{4.18}$$

with $\dot{x} = (x_n - x_{n-1})/\delta t$ can be used. Fig. 4.3 shows that both mass estimates grow for an increasing $\delta t$, the equipartition-based mass is slightly smaller than the estimate based on $\hat{\Gamma}$ and $\hat{\mathcal{K}}$. Interestingly, using $\delta t = n \cdot \delta t_0$ and considering $\mathcal{M}/n$, it turns out that the masses rise approximately proportional to $\delta t$ in the value range $0.2 \leq \delta t \leq 20$ ps. Hence, the dLE apparently does not estimate the mass used to generate the data ($\mathcal{M} = 400$ ps) once time steps $\delta t \geq \delta t_0$ are employed. To understand this behavior one might consider the following line of thought. Based on the equations (4.5), (4.7), (4.8) and (4.9) we imagine some one-dimensional trajectory $x(t)$ with constant $\Gamma$ and $\mathcal{K}$ generated by integrating the Langevin equation with the basic time step $\delta t_0$

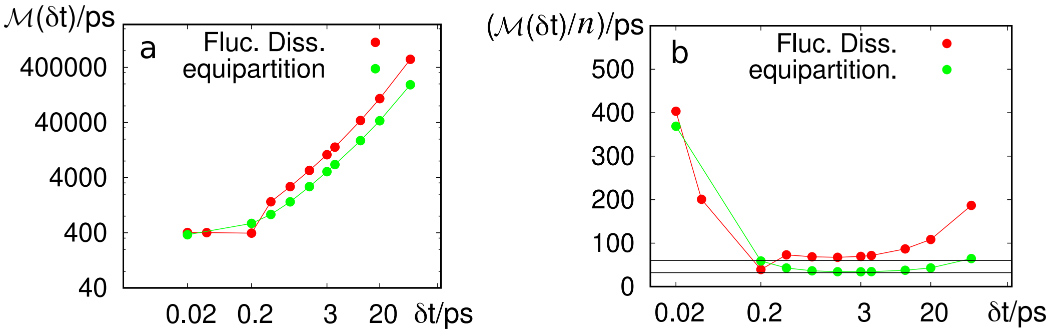

Figure 4.3: **Mass estimated by dLE for the Markovian double well data.** (a) We see that the mass estimate grows with $\delta t$. Both ways of estimating the mass, based on the dLE fields $\hat{\Gamma}$ and $\hat{\mathcal{K}}$ or based on the equipartition theorem, behave in the same way. (b) The growth is roughly proportional to the time resolution $\delta t$.

$$\delta x_{k+1} = \delta x_k - \frac{1}{\mathcal{M}} \frac{dF(x)}{dx_k} \delta t_0^2 - \frac{\Gamma}{\mathcal{M}} \delta t_0 \delta x_k + \frac{\mathcal{K}}{\mathcal{M}} \delta t_0^{3/2} \xi_k, \tag{4.19}$$

where $\delta x_k = x_k - x_{k-1}$ holds. Now we decrease the resolution from $\delta t_0$ to $\delta t = n \cdot \delta t_0$. As consequence the dLE detects the displacements $\Delta x_m = \sum_{j=0}^{n-1} \delta x_{m \cdot n + j}$, i.e, we investigate

$$\begin{aligned}\Delta x_{m+1} &= \sum_{j=0}^{n-1} \delta x_{(m+1)\cdot n + j} \\ &= \sum_{j=0}^{n-1} \delta x_{m\cdot n+j} - \sum_{j=0}^{n-1} \frac{1}{\mathcal{M}} \frac{dF(x_{m\cdot n+j})}{dx} \delta t_0^2 - \sum_{j=0}^{n-1} \frac{\Gamma}{\mathcal{M}} \delta t_0 \delta x_{m\cdot n+j} + \sum_{j=0}^{n-1} \frac{\mathcal{K}}{\mathcal{M}} \delta t_0^{3/2} \xi_{m\cdot n+j}.\end{aligned}$$

The first term on the right side is obviously $\Delta x_m$. Assuming that $\frac{dF}{dx}$ stays constant during $\delta t$, which represents the locality assumption of the dLE, the second term becomes $\frac{1}{\mathcal{M}}\frac{dF(x_m)}{dx}\delta t_0^2\sum_{j=0}^{n-1}1$. Since we assumed constant fields $\Gamma$ and $\mathcal{K}$, the third and fourth term are simple to interpret as well. We get

$$\Delta x_{m+1} = \Delta x_m - \frac{1}{\mathcal{M}}\frac{dF(x_m)}{dx}n\delta t_0^2 - \frac{\Gamma}{\mathcal{M}}\delta t_0\Delta x_m + \frac{\mathcal{K}}{\mathcal{M}}\delta t_0^{3/2}\sum_{j=0}^{n-1}\xi_{m\cdot n+j}. \tag{4.20}$$

The sum over the white noise can be replaced by a single normal distributed random number $\tilde{\xi}_m$ by considering that $\langle\sum_{j=0}^{n-1}\xi_{m\cdot n+j}\rangle = 0$ and $\langle\sum_{j=0}^{n-1}\xi_{m\cdot n+j}\sum_{j=0}^{n-1}\xi_{m\cdot n+j}\rangle = n$ holds. This leads to

$$\Delta x_{m+1} = \Delta x_m - \frac{1}{\mathcal{M}}\frac{dF(x_m)}{dx}n\delta t_0^2 - \frac{\Gamma}{\mathcal{M}}\delta t_0\Delta x_m + \frac{\mathcal{K}}{\mathcal{M}}\delta t_0^{3/2}\sqrt{n}\tilde{\xi}_m. \tag{4.21}$$

with $\langle\tilde{\xi}\rangle = 0$ and $\langle\tilde{\xi}\tilde{\xi}\rangle = 1$. Finally, we insert $\delta t = n\cdot\delta t_0$ and end up with the dynamics seen by the dLE at this time step

$$\Delta x_{m+1} = \Delta x_m - \frac{1}{n\mathcal{M}}\frac{dF(x_m)}{dx}\delta t^2 - \frac{\Gamma}{n\mathcal{M}}\delta t\Delta x_m + \frac{\mathcal{K}}{n\mathcal{M}}\delta t^{3/2}\tilde{\xi}_m. \tag{4.22}$$

It can be seen that $\mathcal{M}$ is substituted by $n\mathcal{M}$. This explains the behavior of $\mathcal{M}(\delta t)$ shown in Fig. 4.3 and indicates that the dLE does not necessarily estimates the mass used to generate the input data. Having seen that the dLE behaves as expected for perfectly Markovian data, we will now inspect the influence of memory. To this end, we consider input data following the GLE (3.21) with monoexponential memory, i.e., $K(t) = (\Gamma/\tau_K)e^{-t/\tau_K}$, integrated as described in Sec. 3.3. Free energy $F(x)$, friction $\Gamma$, temperature $T$ and mass $\mathcal{M}$ are the same as for the Markovian data, the decay time of

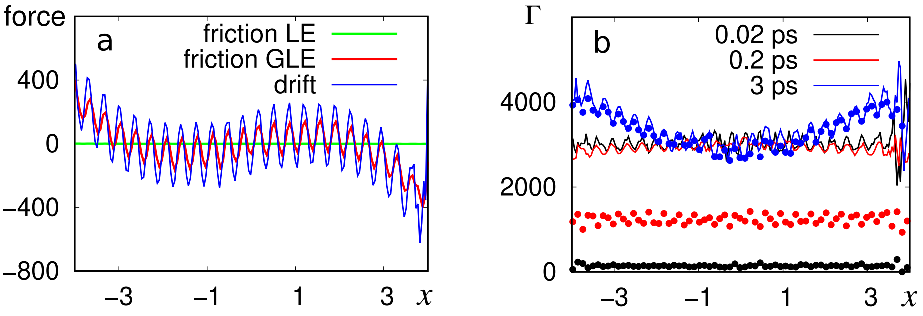

Figure 4.4: **Average friction force and friction estimates.** (a) We see that while the average of the Markovian LE friction force $\Gamma\dot{x}$ (green) is zero for all $x$, the GLE friction force $\sum_{m=0}^{M}K(m\delta t)\dot{x}(t-m\delta t)$ (red) shows a dependence on $x$ which closely resembles the drift force $dF/dx$ (blue). The Markovian data uses $\delta t = 20$ fs and the GLE data $\delta t = 2$ fs. (b) The dLE underestimates the friction for memory-based input data (dots) compared to Markovian input data (lines) as long as $\delta t < \tau_M \approx 3\tau_K$ holds. For larger $\delta t$, both input data sets lead to the same friction estimate.

the memory kernel is set to $\tau_K = 0.2$ ps. This time is chosen such that $\tau_K \ll \tau_{\mathrm{L,R}}$ which means that the GLE trajectory predicts the same long-time dynamics as the Markovian LE, considering that $\int_0^\infty K(t')dt' = \Gamma$ holds. An integration time step of $\delta t = 2$ fs was used. When comparing the average friction forces of this data set to the average friction force of the Markovian input data, Fig. 4.4a, we see that memory induces a clear position dependence. While the Markovian friction fulfills $\langle \Gamma \dot{x} \rangle = 0$ for all $x$, $\langle \sum_{m=0}^{M} K(m\delta t)\dot{x}(t - m\delta t)\rangle$ resembles the derivative of the free energy. Hence, there are clear differences between both data sets.

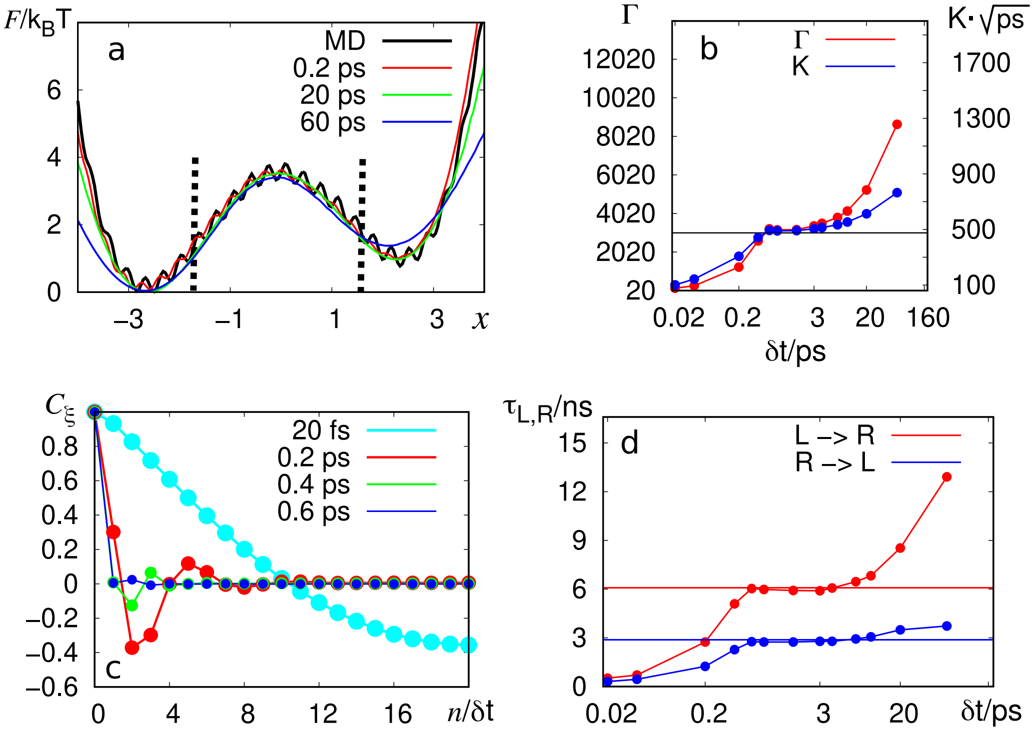

Figure 4.5: **Generalized double well model.** (a) Shown are the free energies explored by the input data (MD) and dLEs at different time steps. The dashed lines show the cores of the two states R and L (see text) which are used to quantify the system dynamics. (b) Here, the dLE estimates of friction Γ and noise amplitude $\mathcal{K}$ for different time steps are presented. The black line shows the fields used in the input data. (c) The autocorrelation of the reconstructed noise at different $\delta t$. (d) We see the transition times $\tau_{\mathrm{L,R}}$ estimated by the dLE at different $\delta t$ (dots) compared to the expectations (lines). In red we see L→R while blue represents R→L.

Now, we can inspect how memory influences the dLE. Fig. 4.5a shows that the main barrier of the free energy is, just as for the Markovian data, reproduced for $\delta t \leq 20$ ps while the sine overlay needs again $\delta t \leq 0.2$ ps. This is not surprising considering that the free energies of both GLE and LE are the same. When inspecting the estimates of $\Gamma$ and $\mathcal{K}$, see Fig. 4.5b, another similarity to the Markovian data can be seen: for $\delta t > 4$ ps the dLE overestimates both fields. Hence, the upper bound $\delta t_{\mathrm{R}}$ of suitable time steps stays unaffected by the non-Markovianity of the data at shorter times, just as expected. In contrast, the lower bound $\delta t_\mathrm{M}$ changes significantly. We see that $\Gamma$ and $\mathcal{K}$ are underestimated for $\delta t \leq 0.6$ ps. The autocorrelation of the noise, see Fig. 4.5c, indicates as well that small time steps are not well suited since the curves need more than one time step to decay when $\delta t \leq 0.4$ ps is used. An underestimation of $\Gamma$ and $\mathcal{K}$ is accompanied by an underestimation of the transition times $\tau_\mathrm{L}$ and $\tau_\mathrm{R}$, see Fig. 4.5d. In total, we can conclude that the dLE works correctly for time steps $\delta t \in [0.6 \text{ ps}, 4 \text{ ps}]$. In this value region the dLE field estimates and dynamical predictions cannot be distinguished from the dLE based on Markovian data, even the small (but practically irrelevant) dependence of $\Gamma$ and $\mathcal{K}$ on $x$ for large $\delta t$ can be found, see Fig. 4.4.

The lower bound $\delta t_\mathrm{M} = 0.6$ ps is highly consistent considering that the memory kernel decays approximately within $3\tau_K = \tau_M$. To understand why the friction is underestimated for $\delta t \leq \delta t_\mathrm{M}$ we can analyze the friction force $f_F(t)$ of a GLE with arbitrary memory kernel $K(t)$ decaying within the time $\delta t_\mathrm{M}$. Assuming that the GLE was integrated with a time step of $\delta t_0$ the discretized friction force is

$$f_F(t) = -\delta t_0 \sum_{m=0}^{M} K(m\delta t_0)\dot{x}(t - m\delta t_0) \tag{4.23}$$

with $M = t/\delta t_0$ for $t \leq \delta t_\mathrm{M}$ or $M = \delta t_\mathrm{M}/\delta t_0$ for $t > \delta t_\mathrm{M}$. Now, we insert the average velocity $\langle \dot{x}(t) \rangle = 1/M \sum_{m=0}^{M} \dot{x}(m\delta t_0)$ during memory decay to get

$$f_F(t) = -\Gamma\langle \dot{x}(t)\rangle - \delta t_0 \sum_{m=0}^{M} K(m\delta t_0)[\dot{x}(t - m\delta t_0) - \langle \dot{x}(t)\rangle] \tag{4.24}$$

with $\int_0^\infty K(t')dt' = \Gamma$. If the dLE is applied with a time step $\delta t \geq \delta t_\mathrm{M}$, it will only detect the average velocity $\langle \dot{x}(t)\rangle$ simply because it only detects the accumulated displacement $\Delta x(t) = \sum_{m=0}^{M} \Delta x(m\delta t_0)$ rather than the individual small displacements $\Delta x(m\delta t_0)$. The second term in Eq. (4.24) is not resolved. Hence, the friction force becomes Markovian for $\delta t \geq \delta t_\mathrm{M}$ and the dLE estimates the right friction $\Gamma$.

Now we can check the results if $\delta t = \delta t_0$ is used for the dLE. After rewriting the friction force to

$$f_F(t) = -\delta t_0 K(0)\dot{x}(t) - \delta t_0 \sum_{m=1}^{M} K(m\delta t_0)\dot{x}(t - m\delta t_0) \tag{4.25}$$

we can see that it contains the Markovian term $\delta t_0 K(0)\dot{x}(t)$ and an additional non-Markovian correction resulting from the long-running decay of $K(t)$. Since the dLE estimates the fields only based on the displacements $\Delta x$ accumulated within the time $\delta t$ and not based on the full history, it must be expected that it mainly detects the first term and overlooks the non-Markovian correction. If we assume $K(t)$ to be positive definite and if we furthermore assume that the sign of $\dot{x}$ does not change during $\delta t_\mathrm{M}$, i.e., $|f_F| > |\delta t_0 K(0)\dot{x}(t)|$, this indicates that the dLE detects only a fraction of the frictional force (the first term in Eq. (4.25)) which leads to an underestimated $\Gamma$. We note that a similar argumentation can be done for the noise amplitude $\mathcal{K}$.

In summary, we have seen in this section how the dLE performance depends on the chosen time resolution $\delta t$ and how a reasonable value range can be identified. The upper bound $\delta t_\mathrm{R}$ is defined by the locality condition of the dLE field estimation while the lower bound $\delta t_\mathrm{M}$ can be interpreted as the decay time of the system memory.

### 4.1.3 Rescaled dLE

In application, like for example Sec. 5.1.1, it might be possible to face the dilemma that the time step $\delta t_{\mathrm{M}}$, needed to observe Markovian dynamics by the dLE, is larger then the time step $\delta t_{\mathrm{R}}$, needed to resolve the dynamics accurately enough. Once $\delta t < \delta t_{\mathrm{M}}$ is chosen, we expect, based on the findings in the last section, that the friction $\Gamma$ (and by this the noise $\mathcal{K}$) are underestimated by the dLE, see Fig. 4.5. Still, this effect can be corrected by introducing the diagonal matrix $S$ with $S_{ii} > 1$ which is used to modify the two dLE fields $\hat{\Gamma}$ and $\hat{\mathcal{K}}$ via

$$(\hat{\Gamma} + \mathbb{1}) \rightarrow S(\hat{\Gamma} + \mathbb{1})S^T, \tag{4.26}$$

$$\hat{\mathcal{K}} \rightarrow S\hat{\mathcal{K}}, \tag{4.27}$$

which preserves the validity of the fluctuation-dissipation theorem (3.13). In this way we can apply the dLE at $\delta t < \delta t_{\mathrm{R}}$ and correct the most severe non-Markovian effects. The modification defined by the two equations above motivates the name of the new approach: rescaled dLE. In the following chapters we will see how to calibrate the matrix $S$ based on short-time information. We note that the rescaled dLE reminds of coarse-grained MD approaches [67–69] where an effective time scale of the coarse-grained model is defined by comparing the model dynamics to atomistic MD simulations.

### 4.1.4 Accelerating the dLE: Problems of data removal

When considering the computational costs of the whole dLE approach, it turns out that the search for the $k$ next neighbors represents the main bottleneck for data sets with $N \geq 10^6$ points. Although the use of a box-assisted search [133] helps by reducing the scaling from $\propto N^2$ to $\propto N \ln(N)$, the dLE propagation becomes tedious for $N \geq 10^7$ data points. Unfortunately, if the dLE should be suitable to interpret enhanced sampling data provided by MD, like the data in Sec. 5.2, it will be mandatory that it can work with data sets of this size. This means that we have to think of some pre-processing of the input data which allows for reasonable calculation times of the subsequently propagated dLE trajectories.

Considering that the free energy of the observed system dynamics allows for the differentiation of highly and lowly populated regions, i.e., minima and barriers, the first idea to solve this problem might be that it should be possible to remove a lot of data points in the minima of the free energy without harming the overall statistics since those minima are typically excessively sampled. Still, this concept of "data pruning" proposed by Schaudinnus et al. [43] turned out to be problematic for more than one system dimension. In the following we want to understand the reason for this observation before introducing a better data reduction scheme.

As first step we consider the simplest approach to implement data pruning. Here, we define value regions of $\boldsymbol{x}$ were data points should be removed, i.e., we specify the minima of the free energy. Afterwards, the given data trajectory is processed from the first to the last frame by removing trajectory pieces stochastically once the trajectory visits a free energy minimum. Three different parameters are defined to this end: $s_{\mathrm{min}}$ quantifies the minimal length of the trajectory pieces surviving the pruning, $p_1 \in [0, 1]$ represents the probability to start to remove points and $p_2 \in [0, 1]$ quantifies the probability to stop the removal once it has started. The parameter $s_{\mathrm{min}}$ helps to prevent the survival of excessively short trajectory pieces while $p_1$ and $p_2$ quantify the strength of the pruning. They are used in the following way: once the considered trajectory reaches the predefined value regions for more than $s_{\mathrm{min}}$ steps, uniformly distributed random numbers $\xi \in [0, 1]$ are generated until $p_1 < \xi$ is observed. At this point the removal of data points begins. Subsequent points are removed until one of the simultaneously generated random numbers $\xi \in [0, 1]$ fulfills $p_2 < \xi$ where the erasing is stopped. If the trajectory leaves the free energy region where data points should be removed the erasing will be stopped as well. Once the data removal has stopped in one of those two ways, we wait again until the trajectory spends at least $s_{\mathrm{min}}$ points in one of the predefined free energy regions before we allow for the next start of data removal as described above.

Additionally, the two parameters $p_{1,+} \in [0, 1]$ and $p_{2,+} \in [0, 1]$ were defined to make the pruning more flexible. $p_{1,+}$ is iteratively added to $p_1$ for every trajectory point after $s_{\mathrm{min}}$ which has not been removed, i.e., $p_{1,+} > 0$ successively increases the probability to remove trajectory points and leads to shorter surviving trajectory pieces. $p_{2,+}$ interacts in the same way with $p_2$. It should be noted that these two parameters are in practice relatively irrelevant considering the performance of the dLE applied to the pruned data.

To test this pruning approach we use a Markovian LE trajectory based on a one-dimensional double well potential, see Fig. 4.6b, with $\Gamma = 3000$, $T = 300$ K, $\mathcal{M} = 400$ ps and $\delta t = 0.2$ ps. The data set consists of $10^7$ points. Two different pruning setups are compared, the first setup uses $p_1 = p_{1,+} = 0.01$, $p_2 = p_{2,+} = 0.05$ together with $s_{\mathrm{min}} = 5$ and result in $7.8 \cdot 10^6$ surviving points, the second one uses $p_1 = 0.1$, $p_2 = 0.01$, $p_{1,+} = p_{2,+} = 0$ and $s_{\mathrm{min}} = 5$ and yields $1.9 \cdot 10^6$ remaining points, i.e., both prunings are relatively moderate considering the simplicity of the system. The value regions to be pruned were defined by $-3.3 \leq x \leq -1.75$ and $1.45 \leq x \leq 3.0$. Fig. 4.6a shows the distribution of data points before and after pruning. The second pruning shows two distinct discontinuities close to the barrier where the removal of points was stopped. When applying the dLE to this data set the resulting free energy slightly underestimates the barrier, see Fig. 4.6b, the deviations start exactly at $x = -1.75$ and $x = 1.45$. Dynamical observables, like, e.g., the autocorrelation, show deviations as well.

Therefore, we have to conclude that pruning might cause problems. We can try to understand their roots. To this end one might inspect the forward and backward displacements, $\Delta x_{m+1}(x_m) = x_{m+1} - x_m$ and $\Delta x_m(x_m) = x_m - x_{m-1}$, before and after pruning since this should directly reveal possible problems of the dLE which estimates its fields based on those displacements. The bottom row of Fig. 4.6 shows that both displacements are significantly influenced by pruning, they reveal pronounced peaks at the borders between pruned and unpruned regions. Hence, opposed to intuition which tells that all points are treated equally, the pruning based on constant probabilities treats different motion patterns in different ways. To make sense of this observation one can imagine two different trajectory pieces. Both pieces start close to one of the borders between pruned and unpruned regions but one of the pieces, called piece 1, stays the whole time inside of the pruned region while the other piece, piece 2, jumps back and forth over the border. Due to the way our version of pruning works it is straightforward to see that piece 1 will have a higher probability to loose points compared to piece 2 since all points outside of the pruned regions are by definition save from removal. This means that points with relatively small displacements, as found in piece 1 staying the whole time on one side of the border, are removed with a higher probability than points with large displacements like found in piece 2. This effect distorts the distributions of $\Delta x_{m+1}$ and $\Delta x_m$ and, ultimately, also the dLE dynamics which are based on those displacements.

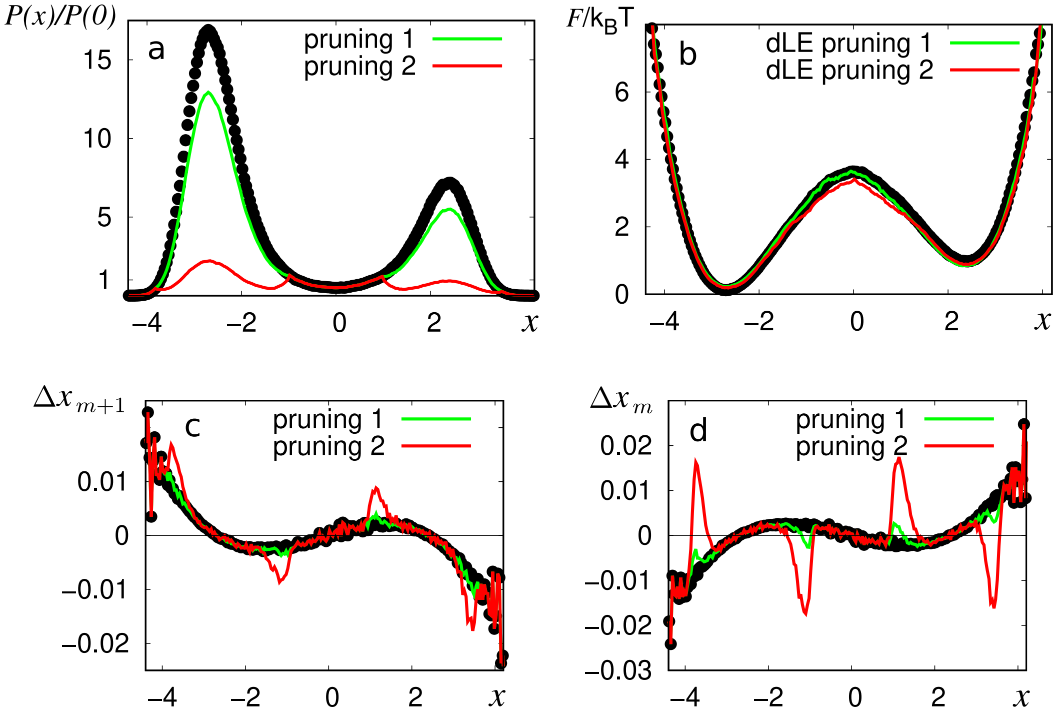

Figure 4.6: **Effects of data pruning based on constant probabilities.** (a) The distribution $P(x)/P(0)$ of the complete input data (black) is compared to $P(x)/P(0)$ of the two pruned date sets (see text). (b) Here, we see that the free energy of the dLE using the stronger pruned data (red) underestimates the barrier of the reference (black) while the dLE on the weaker pruned data (green) works flawless. The average forward displacements $\Delta x_{m+1}(x)$ (c) and the average backward displacement $\Delta x_m(x)$ (d) are significantly influenced by the stronger pruning.

By concluding that the main problem of pruning appears to be the transition between pruned and unpruned regions one can try to make this transition smoother. This can be achieved by replacing the constant probability $p_1$ by some varying function $p_1(x)$ which peaks in the minima. A possible choice could be a sum of Gaussian distributions $p_1(x) = \sum_{i=0}^{K} P_i e^{-(x-x_i)^2/2\sigma_i^2}$ with $K$ being the number of minima. Each minimum is defined by two parameters, $x_i$ and $\sigma_i^2$, representing center and width of the respective Gaussian distribution. To inspect whether this modification improves the pruning, two setups were tested for the double well data considered in this section. The first parameter set is given by $P_1 = P_2 = 0.01$, $x_1 = -2.7$, $x_2 = 2.4$, $\sigma_1^2 = \sigma_2^2 = 0.5$, $p_2 = 0.005$, $p_{1,+} = p_{2,+} = 0$ and $s_{\text{min}} = 5$. It results in $4.3 \cdot 10^6$ data points surviving the pruning. The second setup is defined by $P_1 = 0.07$, $P_2 = 0.03$, $x_1 = -2.7$, $x_2 = 2.4$, $\sigma_1^2 = \sigma_2^2 = 0.5$,

$p_2 = 0.005$, $p_{1,+} = p_{2,+} = 0$ and $s_{\min} = 5$. It yields $1.6 \cdot 10^6$ remaining points. Fig. 4.7 shows in the top row that the use of a varying $p_1(x)$ allows to preserve the forward displacements $\Delta x_{m+1}$ for both setups. Though, the backward displacements $\Delta x_m$ still deviate after pruning.

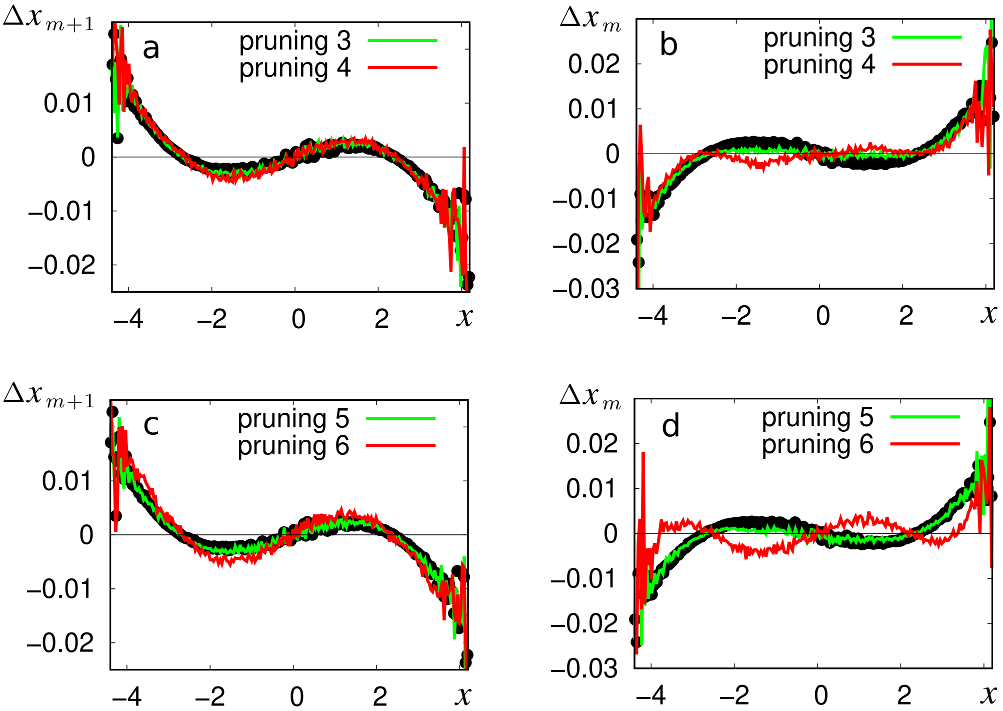

Figure 4.7: **Effects of data pruning based on other approaches.** Replacing the constant probability $p_1$ by a sum of Gaussian distributions, see text, results in the average forward $\Delta x_{m+1}$ and backward displacements $\Delta x_m$ shown in (a) and (b), respectively. The same quantities are shown in (c) and (d) for the pruning approach suggested by Schaudinnus et al. [43]

Considering that pruning approaches based on local probabilities might be generally problematic, we can inspect the approach presented by Schaudinnus et al. [43] which is conceptually different so that it might circumvent the observed problems. This pruning starts by cutting the coordinate range of the first system coordinate into $a$ discrete bins. Afterwards, $b$ points in each bin are randomly selected from the full trajectory to become the starting points of short trajectories. Each of these short trajectories consists of maximal $c$ consecutive points. Since it does not make sense to select any input data point more than once, it might be the case that individual short trajectories are cut before reaching length $c$. Since points on the barrier have, compared to points in the minima which are more numerous, a higher probability to be part of the $b$ selected starting points or the $c$ consecutive points, this approach removes more points in the minima than on the barrier, i.e., it achieves the desired data reduction in excessively sampled regions of the free energy. To test the performance of this approach, we inspect, again, two setups. Once the three parameters are set to $a = b = 100$ and $c = 700$ and once $a = b = c = 100$

is used. The former pruning results in $4.2 \cdot 10^6$ data points and the latter setup yields $0.86 \cdot 10^6$ points. When inspecting the pruned data, Fig. 4.7 shows in the bottom row that the fundamental problem of spoiled backward displacements $\Delta x_m$ cannot be solved. Again, this can be understood by considering the two trajectory pieces inspected above. Piece 1, which had a higher probability than piece 2 to lose points, is now very likely to be assigned to a single bin. The points of piece 2, in contrast, have good chances to belong to several bins since piece 2 is more dynamic. This means that the points of piece 1 have to compete with relatively many points found in its bin close to the free energy minimum while the points of piece 2 can escape to more sparsely populated bins at the barrier of the free energy where the chances are higher to survive the pruning. Hence, again, the pruning approach favors some distinct motion patterns.

### 4.1.5 Accelerating the dLE: Binned dLE

After all it must be concluded that the unbiased removal of redundant data is problematic to implement. To develop an alternative strategy to tackle the problem of extensive input data sets, one can recapitulate the formulation of the dLE itself. When inspecting the equations (4.12), (4.13) and (4.14) it becomes apparent that only local averages of $\Delta\boldsymbol{x}_{m+1}$ and $\Delta\boldsymbol{x}_m$ as well as averages of the products $\Delta\boldsymbol{x}_{m+1}\Delta\boldsymbol{x}_{m+1}^T$, $\Delta\boldsymbol{x}_m\Delta\boldsymbol{x}_m^T$ and $\Delta\boldsymbol{x}_{m+1}\Delta\boldsymbol{x}_m^T$ are needed to estimates the Langevin fields. This indicates that it is conceptually possible to formulate some "pre-averaging" which replaces the explicit $N$ input data points by $M \ll N$ grid points which record sufficiently local averages of $\Delta\boldsymbol{x}_{m+1}$, $\Delta\boldsymbol{x}_m$, $\Delta\boldsymbol{x}_{m+1}\Delta\boldsymbol{x}_{m+1}^T$, $\Delta\boldsymbol{x}_m\Delta\boldsymbol{x}_m^T$ and $\Delta\boldsymbol{x}_{m+1}\Delta\boldsymbol{x}_m^T$.

The approach we are proposing at this point starts by separating each of the $d$ dimensions of the system description $\boldsymbol{x}$ into $s$ coarse bins. All these $s^d$ cells are treated independently to account for the different sampling qualities of barrier and minima. First of all, the individual cells are once again separated into $b_{\text{min}}^d$ fine bins by evaluating the input parameter $\omega_{\text{max}}$ which quantifies the maximally allowed width of each fine bin. Here, the coordinate with the smallest value range is used to determine $b_{\text{min}}$. This step ensures that the pre-averages on the barrier stay sufficiently local. Then the number of data points $N_s$ found in each of the coarse bins is determined to calculate $b_{\text{points}} = (N_s/N_{\text{max}})^{1/d}$ by using the additional input parameter $N_{\text{max}}$. This parameter quantifies the average maximal number of points which should be averaged in lowly sampled regions of the free energy landscape. If $b_{\text{min}} < b_{\text{points}}$ holds in some coarse bin, it will separated into $b_{\text{points}}^d$ instead of $b_{\text{min}}^d$ fine bins. Finally, the last input parameter $\omega_{\text{min}}$ representing the minimally desired bin width is used to determine $b_{\text{max}}$, the maximally allowed number of fine bins per coarse bin. If $b_{\text{max}} < b_{\text{points}}$ holds, the coarse bin will be separated into $b_{\text{min}}^d$ instead of $b_{\text{points}}^d$ fine bins. This step prevents that unnecessarily many fine bins are used to average in the minima of the free energy.

This adaptive pre-averaging adjusts the binning depending on the local sampling, i.e., while only few points are averaged on the barrier, the minima are averaged with higher resolution but nevertheless larger sampling sizes $N_i$. By saving the averages of $\Delta\boldsymbol{x}_{m+1}$, $\Delta\boldsymbol{x}_m$, $\Delta\boldsymbol{x}_{m+1}\Delta\boldsymbol{x}_{m+1}^T$, $\Delta\boldsymbol{x}_m\Delta\boldsymbol{x}_m^T$ and $\Delta\boldsymbol{x}_{m+1}\Delta\boldsymbol{x}_m^T$ together with the number of averaged points $N_i$ in each of the $b^d$ fine bins, it is possible to rewrite the next neighborhood average Eq. (4.10) to

$$B(\boldsymbol{y}_n) = \langle B(\boldsymbol{x}_m)\rangle = \frac{1}{k}(N_1\langle B(\boldsymbol{x})\rangle_1 + ... + N_r\langle B(\boldsymbol{x})\rangle_r) \tag{4.28}$$

after assuming that $\sum_{i=i}^{r} N_i = k$ holds for the $r$ neighboring bins of point $\boldsymbol{y}_n$. Hence, only $r \ll k$ neighbors out of $s \cdot b^d = M \ll N$ candidates need to be found. Fig. 4.8 illustrates this procedure which drastically decreases the computation times of the dLE.

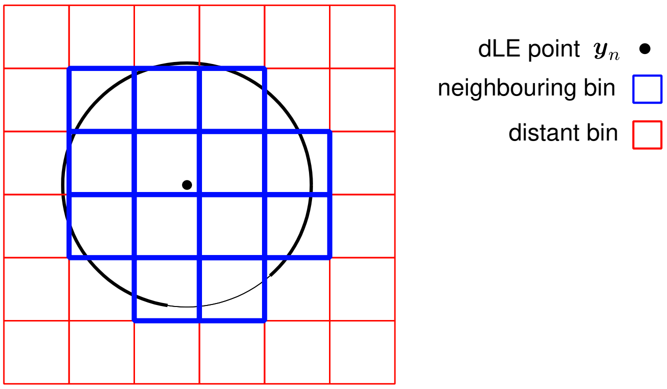

Figure 4.8: **Illustration of the binned dLE.** Instead of scanning the initial input trajectory for the $k$ next neighbors of dLE point $\boldsymbol{y}_n$ (black), the binned dLE detects the $r$ closest bins (blue) of the pre-averaged data. Bins with a larger distance (red) are not considered in the field estimation.

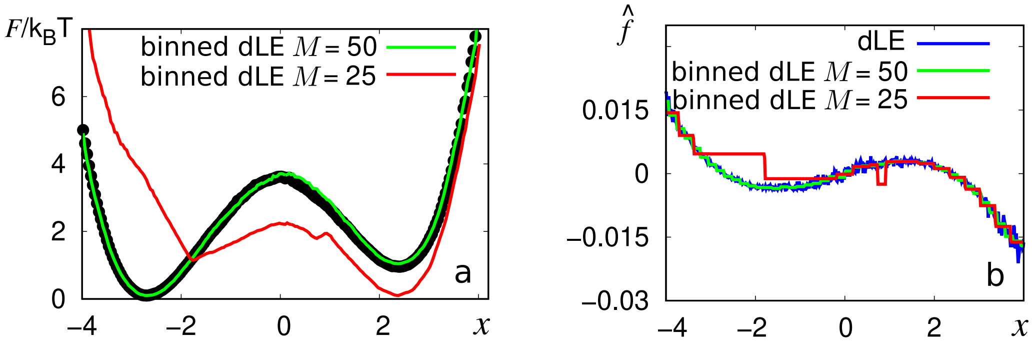

Figure 4.9: **Binned dLE for double well data.** (a) The binned dLE successfully reproduces the system dynamics of the reference (black dots) down to only $M = 50$ pre-averaged input points as can be seen exemplary for the free energy. Stronger pre-averaging leads to wrong dynamics as shown for $M = 25$ points. (b) This can be explained by wrong drift estimates $\hat{f}(x)$. While the binned dLE with $b = 50$ only overlooks unimportant oscillations but follows the right curve (compared to the normal dLE in blue), $M = 25$ spoils the overall shape.

We note that field estimates based on Eq. (4.28) are exactly the same as if they would have been determined based on Eq. (4.10) as long as the pre-averaging is sufficiently local, i.e., the method is does not influence the dLE field estimation. Data-driven Langevin simulations based on pre-averaged data are called binned dLEs in the following.

When considering the double well data used to investigate the different data pruning approaches, it turns out that it is possible to go down to only $M = 50$ pre-averaged input points, see Fig. 4.9. Less points spoil the model dynamics but the reduction from $10^7$ to only 50 is already massive and shows that the binned dLE approach is very stable. Hence, we can conclude that the pre-averaging of extensive input data sets is well suited to accelerate dLE studies. In addition, we note the advantage that this procedure is deterministic, i.e., the result of the pre-averaging is always the same if the parameters stay unchanged.

## 4.2 Alternative dLE formulation

Having considered the acceleration of the established dLE framework, we will now inspect alternative dLE implementations. As starting point we recapitulate that the dLE introduced in Sec. 4.1 is based on three assumptions. First, we suppose that the Markovian Langevin equation is suitable to describe the observed dynamics. Second, we demand that the $k$ next neighbors used to estimate the Langevin fields are sufficiently local everywhere in coordinate space. And third, it is claimed that the Euler integrator defined by Eq. (3.32) and Eq. (3.33) is sufficiently accurate at the considered time resolution $\delta t$ to produce meaningful trajectories. While the validity of the first assumption makes or breaks the whole dLE modeling framework, we can think of possible improvements for the other two points, i.e., we can think of better ways to estimate the fields and a more accurate integrator which extends the range of acceptable time steps $\delta t$.

Considering the fields estimators, it is possible to define the neighborhood by a fixed radius $r$ instead of a fixed size $k$. This means that we can search for all data points $\boldsymbol{x}_i$ which lie within a hypersphere of radius $r$ centered at the current dLE point $\boldsymbol{y}(t)$ instead of the $k$ next neighbors, just as it is done, e.g., in the density-based clustering approach of Sittel and Stock [60]. This has the advantage that the locality of the field estimates is actively ensured. On the other hand, the convergence of the field estimates becomes problematic if only few points are found. This indicates that especially barrier regions are problematic. While this is acceptable for the density-based clustering of Sittel and Stock [60], which primarily aims for the identification of the minima of the free energy, the dLE needs to be able to estimate reasonable fields on the barriers as well since it wants to cross them in a dynamical way. Hence, it makes sense to prioritize the statistical convergence of the field estimates over the locality on the barrier, i.e., the $k$-next-neighbors approach is advantageous. Additionally, it can be expected that the $k$ next neighbors found in free energy minima are local anyway, i.e., we do not gain anything at this point when replacing the neighborhood estimation. This consideration can be validated by exemplary dLE calculations using our, by now well known, double well data. Replacing the $k$ next neighbor average by an averaging employing a fixed neighborhood radius $r$ does not improve the dLE results in any way (not shown). Due to the excellent sampling of the one-dimensional data we do not observe problems on the barrier but this would certainly change if high-dimensional data with less ideal statistics is inspected.

Considering the integrator underlying the dLE framework, it is mandatory that it allows for the reconstruction of the Langevin fields only based on the discrete input trajectory $\boldsymbol{x}_n$. This means that the OVRVO integrator considered in Sec. 3.3 is not suitable since it does not simply perform the jump $t \to t + \delta t$ but instead uses three artificial intermediate points to propagate the velocity. Alternatively, we can use a Verlet integrator [134], which is, in theory, also superior to the Euler integrator, to formulate an alternative dLE implementation. To this end we start with the equations

$$\boldsymbol{x}(t+\delta t) = \boldsymbol{x}(t) + \dot{\boldsymbol{x}}(t)\delta t + \frac{1}{2}\ddot{\boldsymbol{x}}(t)\delta t^2 + \frac{1}{3!}\dddot{\boldsymbol{x}}(t)\delta t^3 + \mathcal{O}(\delta t^4), \tag{4.29}$$

$$\boldsymbol{x}(t-\delta t) = \boldsymbol{x}(t) - \dot{\boldsymbol{x}}(t)\delta t + \frac{1}{2}\ddot{\boldsymbol{x}}(t)\delta t^2 - \frac{1}{3!}\dddot{\boldsymbol{x}}(t)\delta t^3 + \mathcal{O}(\delta t^4), \tag{4.30}$$

representing the Taylor expansion of $\boldsymbol{x}(t)$ for $t \to t + \delta t$ and $t \to t - \delta t$, respectively. Adding both expansions yields

$$\boldsymbol{x}(t+\delta t) = 2\boldsymbol{x}(t) - \boldsymbol{x}(t-\delta t) + \ddot{\boldsymbol{x}}(t)\delta t^2 + \mathcal{O}(\delta t^4) \tag{4.31}$$

which leads to

$$\boldsymbol{x}_{m+1} - \boldsymbol{x}_m = \boldsymbol{x}_m - \boldsymbol{x}_{m-1} + \frac{1}{\mathcal{M}}(-\boldsymbol{\nabla} F(\boldsymbol{x}_m) - \Gamma(\boldsymbol{x}_m)\dot{\boldsymbol{x}}_m(t) + \mathcal{K}(\boldsymbol{x}_m)\boldsymbol{\xi}_m \delta t^{-1/2})\delta t^2 \tag{4.32}$$

after neglecting $\mathcal{O}(\delta t^4)$ and higher orders. Additionally, $\boldsymbol{x}_m = \boldsymbol{x}(m\delta t)$ was defined and $\mathcal{K}(\boldsymbol{x}_m) = \sqrt{2k_\mathrm{B}T\Gamma(\boldsymbol{x}_m)}$ was inserted. The equation above almost matches Eq. (4.5), only $\dot{\boldsymbol{x}}_m(t)$ is undefined so far. To get the Euler implementation from Sec. 4.1 one assumes $\dot{\boldsymbol{x}}_m \approx (\boldsymbol{x}_m - \boldsymbol{x}_{m-1})/\delta t$ but here we follow the Verlet approach [134] and subtract Eq. (4.30) from Eq. (4.29) to get

$$\dot{\boldsymbol{x}}_m \approx \frac{\boldsymbol{x}_{m+1} - \boldsymbol{x}_{m-1}}{2\delta t} \tag{4.33}$$

after neglecting $\mathcal{O}(\delta t^3)$ and higher orders. Inserting this expression of $\dot{\boldsymbol{x}}_m$ into the equation of motion leads to

$$\Delta\boldsymbol{x}_{m+1} = (\mathbb{1} + \tilde{\Gamma}(\boldsymbol{x}_m))^{-1}(\tilde{\boldsymbol{f}}(\boldsymbol{x}_m) - (\tilde{\Gamma}(\boldsymbol{x}_m) - \mathbb{1})\Delta\boldsymbol{x}_m + \tilde{\mathcal{K}}(\boldsymbol{x}_m)\boldsymbol{\xi}_m) \tag{4.34}$$

after isolating $\Delta\boldsymbol{x}_{m+1}$. Here, we defined the Verlet-dLE fields as

$$\tilde{\boldsymbol{f}}(\boldsymbol{x}_m) = \mathcal{M}^{-1}\delta t^2 \boldsymbol{\nabla} F(\boldsymbol{x}_m), \tag{4.35}$$

$$\tilde{\Gamma}(\boldsymbol{x}_m) = \mathcal{M}^{-1}\frac{\delta t \Gamma(\boldsymbol{x}_m)}{2}, \tag{4.36}$$

$$\tilde{\mathcal{K}}(\boldsymbol{x}_m) = \mathcal{M}^{-1}\delta t^{3/2}\mathcal{K}(\boldsymbol{x}_m), \tag{4.37}$$

in parallel to the Euler implementation in Sec. 4.1.

Now, equations need to be found to estimate the fields based on a given trajectory $\boldsymbol{x}(t)$ to be able to propagate the Langevin trajectory $\boldsymbol{y}(t)$ in parallel. The calculations are very similar to the ones of the Euler-dLE in Sec. 4.1 and can be found in the Sec. A.2. In the end we get

$$\begin{aligned}\tilde{\Gamma}(\boldsymbol{y}_n) =& \mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T)(\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m^T) + \mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T))^{-1} \\ &- \mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m^T)(\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m^T) + \mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T))^{-1},\end{aligned} \tag{4.38}$$

$$\tilde{\boldsymbol{f}}(\boldsymbol{y}_n) =(\mathbb{1} + \tilde{\Gamma}(\boldsymbol{y}_n))\langle\Delta\boldsymbol{x}_{m+1}\rangle + (\hat{\Gamma}(\boldsymbol{y}_n) - \mathbb{1})\langle\Delta\boldsymbol{x}_m\rangle, \tag{4.39}$$

$$\begin{aligned}\tilde{\mathcal{K}}(\boldsymbol{y}_n)\tilde{\mathcal{K}}(\boldsymbol{y}_n)^T =& (\mathbb{1} + \tilde{\Gamma}(\boldsymbol{y}_n))\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_{m+1}^T)(\mathbb{1} + \tilde{\Gamma}(\boldsymbol{y}_n))^T \\ &- (\tilde{\Gamma}(\boldsymbol{y}_n) - \mathbb{1})\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T)(\tilde{\Gamma}(\boldsymbol{y}_n) - \mathbb{1})^T,\end{aligned} \tag{4.40}$$

which can be calculated based on a local neighborhood like already known from the Euler-dLE.

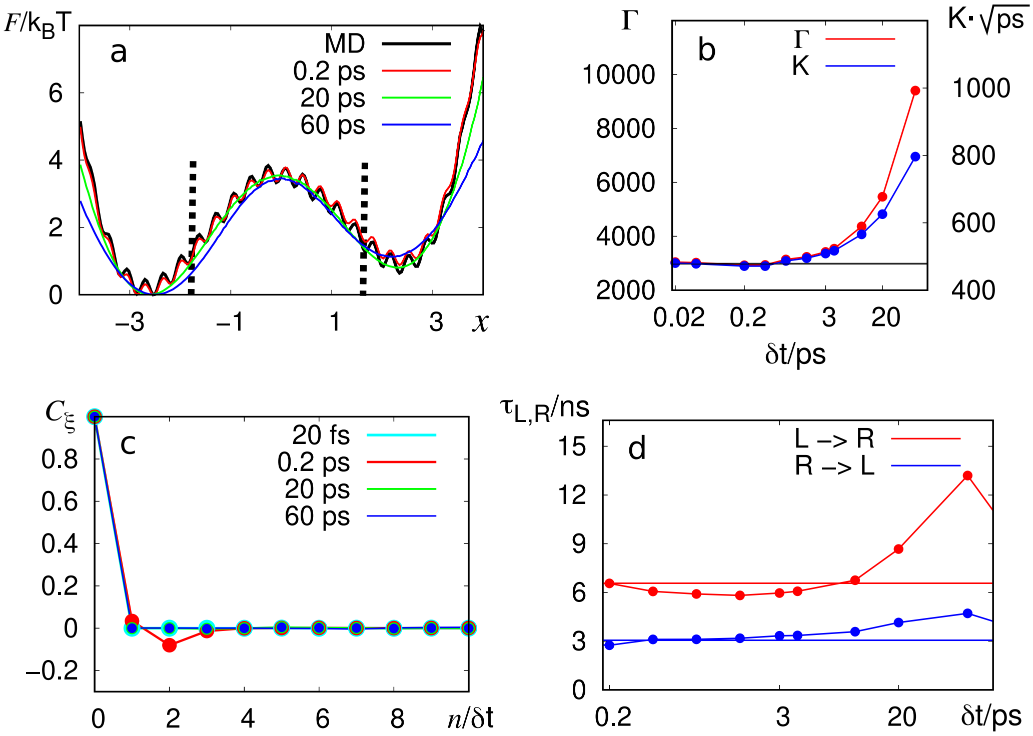

Figure 4.10: **Markovian double well model with Verlet-dLE.** (a) Shown are the free energies explored by the input data, called MD, and Verlet-dLEs at different time steps. The dashed lines show the cores of the two states R and L, see the text, which are used to quantify the system dynamics. (b) The Verlet-dLE estimates of friction Γ and noise amplitude $\mathcal{K}$ for different time steps. The black line shows the expected values. (c) Here, the autocorrelation of the reconstructed noise at different $\delta t$ is shown. (d) We see the transition times $\tau_{\mathrm{L}\leftrightarrow\mathrm{R}}$ estimated by the Verlet-dLE at different $\delta t$ (dots) compared to the expectations (lines). In red we see L→R while blue represents R→L.

Now, it can be checked if the Verlet-dLE performs better than the Euler-dLE. We are especially interested if the Verlet-dLE successfully works with larger $\delta t$. To be in line with Sec. 4.1.2, the known double well potential with sine overlay together with the friction $\Gamma = 3000$, the temperature $T = 300$ K and the $\mathcal{M} = 400$ ps is considered. The Markovian Langevin equation is integrated using the Verlet integrator (4.34) with a time step of $\delta t_0 = 20$ fs to record dynamics for 200 ns. This trajectory is used as input for the Verlet-dLE. We consider, again, the two borders $\delta t_{\mathrm{M}}$ and $\delta t_{\mathrm{R}}$ limiting the range of suitable time steps. While we expect again that $\delta t_{\mathrm{M}} = \delta t_0$ holds because of the Markovian nature of the data, it could be the case $\delta t_{\mathrm{R}}$ turns out to be larger than for the Euler-dLE due to a more robust definition of the velocity in the Verlet framework. A larger $\delta t_{\mathrm{R}}$ would be advantageous in practice since it would allow for the treatment

of systems where $\delta t_{\mathrm{M}} \geq \delta t_{\mathrm{R}}$ holds in the Euler framework.
Though, the results for our exemplary double well indicate that the Verlet-dLE shows the same $\delta t_{\mathrm{R}}$ than the established dLE approach based on the Euler integrator. Fig. 4.10 is mostly identical to Fig. 4.2, qualitatively correct estimates of $\Gamma$ and $\mathcal{K}$ can only be found for $\delta t \leq 4$ ps while the main barrier of the free energy is reproduced up to $\delta t = 20$ ps. The transition times $\tau_{\mathrm{L}\leftrightarrow\mathrm{R}}$ start to deviate from the expectations once the fields $\Gamma$ and $\mathcal{K}$ are overestimated. Considering the lower bound $\delta t_{\mathrm{M}}$, we note that we observe indeed Markovian noise for all possible $\delta t$, just as expected.
To understand why the Verlet-dLE does not allow for a larger $\delta t_{\mathrm{R}}$ we need to remember that the field estimation assumes that the neighborhood of the actual trajectory points behaves sufficiently local. This does not only indicate that there need to be sufficient close neighbors but also that the predecessors and followers of those points are not too far away. This follows from the fact that the field estimations are based on the displacements $\Delta\boldsymbol{x}_n$ and $\Delta\boldsymbol{x}_{n+1}$, i.e., the fields are detected in terms of their influence on the jump from $\boldsymbol{x}_{n-1}$ to $\boldsymbol{x}_n$ and onwards to $\boldsymbol{x}_{n+1}$. If the system jumps too far we will not be able to ensure that the system detects local values of the different fields and the dLE is propagated with wrong forces. However, the displacements $\Delta\boldsymbol{x}_n$ and $\Delta\boldsymbol{x}_{n+1}$ at large $\delta t \approx \delta t_{\mathrm{R}}$ are a priori physical quantities which should be independent of the actually used integration scheme which was used to generate the input data with the integration time step $\delta t_0 \ll \delta t_{\mathrm{R}}$. The long-time behavior of the whole system would depend on the used integrator if this did not hold. Hence, any lack of neighborhood locality at some large $\delta t$ cannot be corrected by simply switching the dLE integrator.

## 4.3 Markovian Langevin model via dissipation-corrected targeted MD

Another possibility to parameterize the Markovian Langevin equation in a data-driven way is provided by the dissipation-corrected targeted MD (dcTMD) framework. In contrast to the dLE which uses unbiased MD, dcTMD studies are based on targeted MD simulations (TMD) as developed by Schlitter et al. [126] where a constraint force $f_c$ evokes a moving distance constraint $x(t) = x_0 + v_v t$ of the one-dimensional system coordinate $x$. This allows to enforce the movement from the starting point $x_0$ to the end point $x_1$ which could describe, e.g., the unbinding of a ligand [125]. In the following we will have a look at the main dcTMD equations derived by Wolf and Stock in [42]. Results of dcTMD studies from [125] are shown in Sec. 6.2.
The main assumption of dcTMD is that the TMD system dynamics can be described by the Markovian Langevin equation via

$$0 = \mathcal{M}\ddot{x}(t) = -\frac{dF}{dx} - \Gamma(x)\dot{x}(t) + \sqrt{2k_{\mathrm{B}}T\Gamma(x)}\xi(t) + f_c(t), \tag{4.41}$$

which only deviates by the constraint $f_c$ from the Markovian LE in the sections above. Note that the constant velocity $\dot{x}(t) = v_c$ (preserved by $f_c$) allows us to set $\ddot{x} = 0$.
To derive an estimate for the free energy at point $x$, it is possible to perform an ensemble average over numerous dcTMD runs which yields based on $\langle\xi(t)\rangle = 0$ [42]

$$F(x) = \langle W(x)\rangle - v_c \int_{x_0}^{x} \Gamma(y)dy + \mathrm{const} \tag{4.42}$$

after integrating over $x$. $\langle W(x)\rangle = \int_{x_0}^{x}\langle f_c(y)\rangle dy$ represents the averaged external work performed on the system. The second term $W_{\text{diss}} = v_c\int_{x_0}^{x}\Gamma(y)dy$ corresponds to the dissipated work. To decouple free energy and friction, Jarzynski's identity [135]

$$e^{-F(x)/k_{\text{B}}T} = \langle e^{-W(x)/k_{\text{B}}T}\rangle \tag{4.43}$$

is used since it allows to calculate the free energy directly from dcTMD data. Since the exponential average on the right side shows convergence problems [136], Wolf and Stock invoked a second-order cumulant expansion which leads to

$$F(x) = \langle W(x)\rangle - \frac{\langle \delta W^2(x)\rangle}{k_{\text{B}}T}, \tag{4.44}$$

with $\delta W(x) = W(x) - \langle W(x)\rangle$. By comparing this equation with Eq. (4.42) and by expressing the work fluctuations $\delta W(x)$ in terms of force fluctuations $\delta f_c = f_c(x) - \langle f_c(x)\rangle$, it can be shown [42] that

$$\Gamma(x) = \frac{1}{k_{\text{B}}T}\int_{t_0}^{t(x)}\langle \delta f_c(t) f_c(t')\rangle dt', \tag{4.45}$$

holds which allows for the calculation of $\Gamma(x)$ from a set of dcTMD simulations. By inserting the result into Eq. (4.42) it is possible to extract the free energy estimate from the data.

As discussed in [42], the derivation of Eq. (4.41) based on the equilibrium fields $F(x)$ and $\Gamma(x)$ requires a slow pulling velocity $v_c$ compared to the time scales of the bath fluctuations. This makes it possible to interpret the effect of $f_c$ as a slow adiabatic change [137] so that the equilibrium observables $F(x)$ and $\Gamma(x)$ stay unperturbed. In consequence, we can use the dcTMD estimates of $F(x)$ and $\Gamma(x)$ to simulate equilibrium trajectories of the considered dynamics by numerically integrating the Markovian LE (3.28), just as we know it from the dLE. Still, the main downside of the dcTMD framework compared to the dLE is the fact that it can only deal with one-dimensional system descriptions due to the inherently one-dimensional nature of the constraint force $f_c$.

Additionally, it needs to be kept in mind that Eq. (4.45) is based on a cumulant expansion of Jarzynski's identity which is only valid as long as the work $W(x)$ is Gaussian distributed everywhere along $x$. This might be not fulfilled for systems which do not follow a single reaction pathway which means that it is mandatory for more complicated systems to extract the dominant pathway [138, 139] before analyzing the dcTMD runs.

## 4.4 Markov state model: Importance of state definition

Having considered the parameterization of the Markovian Langevin equation for exemplary data, see Sec. 4.1.2, we can now use the same double well system to discuss the calibration of a Markov state model (MSM), see Sec. 3.1. Here, the most important step is the definition of discrete states, see Sec. 2.5. For the double well system the states appear to be obvious: there are two minima, the state R and the state L, indicated by the minima of the free energy. Still, it plays an important role how the states are separated.

We will now test two state definitions. On the one hand, the states can be defined by their cores just as it was done to calculate the average waiting times for the dLE in Fig. 4.2 and Fig. 4.5. Here, R is defined by $x \leq -1.7$ and L is given by $x \geq 1.4$. Trajectory pieces outside of this region are assigned to the previously visited state to ensure that every trajectory point has a clear state assignment. One the other hand, one can use the more straightforward approach of cutting directly on top of the barrier at $x = 0$. Having chosen the states, we need to choose the lag time $\tau$ which plays the same role as the time step $\delta t$ for the dLE. It needs to be larger than intrastate fluctuations but at the same time smaller than the interstate dynamics which should be modeled by the MSM.

Based on our two state separations we can inspect the implied time scale derived from transition matrices $T(\tau)$ at various different $\tau$ to find appropriate lag times, see Sec. 3.1. As can be seen in Fig. 4.11a, using the first state definition results in a constant implied time scale which indicates that $\tau$ is free to choose. When looking at the $\tau$-dependence of the average waiting times predicted by MCMC runs, on the other hand, it can be observed that $\tau \leq 1$ ns is needed to get accurate results. This makes sense considering that the L↔R dynamics itself are of the order of nanoseconds, i.e., $\tau$ can maximally be of this order if it should be able to resolve the dynamics.

By instead using the top of the barrier as state border we see the same upper bound on $\tau$. Still, it can be additionally observed that $\tau < 0.4$ ns yields underestimated waiting times. The implied time scale is influenced as well, it rapidly decays for small $\tau$. This behavior can be understood by considering that we now include spurious recrossings on top of the barrier in our state definition, i.e., the separation of intra- and interstate dynamics is significantly worse.

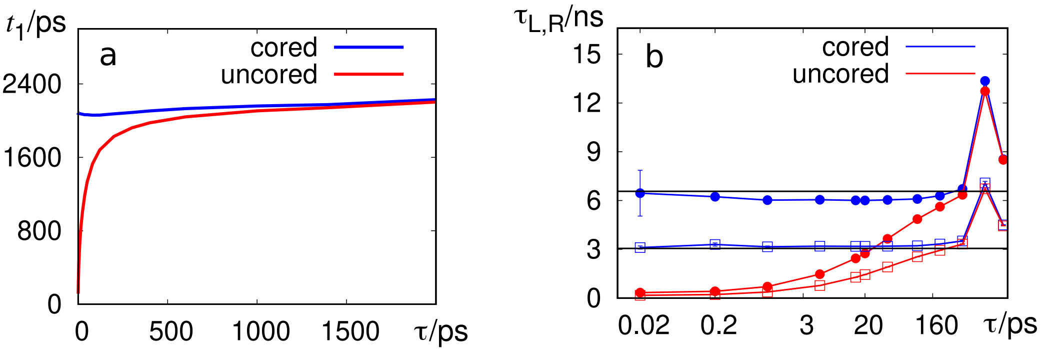

Figure 4.11: **Markov state model of Markovian double well data.** (a) The implied time scale of the MSM constructed for the Markovian double well data depends on the chosen state borders. (b) The average waiting times (circles represent L→R and squares R→L) calculated from MCMC runs based on MSMs at different $\tau$ underpin this observation. If cored states are used it is possible to use basically every lag time smaller than the observed R↔L dynamics. If the states are cut directly on top of the barrier, $\tau \geq 0.4$ ns will be needed.

Hence, we observe already for this objectively simple double well system that the state definition represents a crucial step of MSM construction. Just as we have seen here it is often advantageous to use some sort of coring when generating an MSM, see Sec. 2.6, may it be geometrical (like used here) or dynamical.

## 4.5 Summary

We recapitulated the main equations of the data-driven Langevin approach in Sec. 4.1 and applied the dLE to exemplary model data. It was observed that the dLE correctly approximates memory-based dynamics if the time step $\delta t$ used to evaluate the data is larger than the decay time of the system memory. On the other hand, $\delta t$ needs to be small enough to preserve the locality of the field estimates so that the dynamics can be resolved sufficiently fine. These two conditions define a value range for $\delta t$ where the dLE works successfully.

Still, in practice the resolution criterion might enforce a time step which is smaller than the decay time of the system memory. In this case we observed that the dLE tends to underestimate the friction $\Gamma$ and we suggested the rescaled dLE to compensate this effect. In the following chapters we will observe that the rescaled dLE can be very useful to derive robust Markovian Langevin models.

Subsequently, we discussed the application of the dLE to extensive input data where the standard field estimation becomes prohibitively slow. Based on the observation that the removal of (apparently redundant) data points proves complicated (Sec. 4.1.4), we proposed the binned dLE approach in Sec. 4.1.5. Here, we formulated a "pre-averaging" of the input data by exploiting how the dLE estimates the fields. This allows to drastically reduce the number of data points which need to be scanned by the dLE in every time step. Considering exemplary double well data, we saw that it was possible to reproduce one-dimensional dynamics only based on 50 input points.

Since the Euler integrator used to derive the established dLE is relatively simple, we derived the Verlet-dLE in Sec. 4.2 to inspect if a more elaborate integrator allows for the use of larger time steps. Still, due to the fact that the data displacements $\Delta \boldsymbol{x}$ used to estimate the Langevin fields are a priori physical quantities at large $\delta t$, we observed that the dLE integrator has only minor influences on the range of valid time steps.

Afterwards, we inspected in Sec. 4.3 an alternative approach to parameterize a Markovian Langevin model: dissipation-corrected targeted MD. This approach uses constraint MD simulations to derive one-dimensional Langevin models of the dynamics of interest. To conclude this chapter we inspected in Sec. 4.4 the performance of Markov state models for our exemplary double well model. Already for such simple data it was observed that the careful definition of states is very important.

# 5 Markov modeling of small systems

*"Facts are meaningless. You could use facts to prove anything that's even remotely true!"*
– Homer Simpson, "The Simpsons", season 9, episode 8

Having established the data-driven Markovian modeling framework in the last chapter, we can now apply it to small biomolecular systems to test its robustness. First, sodium chloride (NaCl) is inspected. Considering an one-dimensional system description, we will inspect the dLE performance for varying time steps and we will apply the rescaled dLE. It will be shown how to calibrate the rescaling matrix $S$ based on the initial decay of the position autocorrelation function. Afterward, we will inspect if a memory-based Langevin description improves the model consistency and compare the Langevin framework to the capabilities of a Markov state model and the results of a dcTMD-based parameterization of the Langevin fields.
Subsequently, we will inspect the small nine-residue peptide AIB$_9$. Considering a large enhanced sampling data set [73] and using a five-dimensional system description, this system is more complicated to model. First, dLE dynamics at several time steps are inspected to find a suitable $\delta t$. Afterwards we will see that the rescaled dLE can be used to optimize the Langevin model and that the pre-averaging of the binned dLE allows us to reduce the number of input points by a factor of 100 without harming the model dynamics. Having established a reasonable Langevin model, it will be possible to compare its predictions to a Markov state model derived by Biswas et al. [73]. Additionally, we will compare the model predictions to results of an alternative Markov state model based on the MELD (Modeling Employing Limited Data) protocol [140].

## 5.1 Study of sodium chloride

As first application of the modeling framework developed above, we are investigating the association and dissociation of sodium chloride (NaCl). The considered MD simulations of NaCl were performed and described by Wolf and Stock [42]. Details on the MD setup can be found in Sec. A.4.1. Two trajectories were simulated. The first trajectory has a length of 200 ns and was recorded at a resolution of $\delta t_0 = 10$ fs. It will serve as input for our Markov models. The second simulation collects the time evolution for 1 $\mu$s but only at a resolution of $\delta t_0 = 1$ ps. We will use this simulation as reference to assess the accuracy of the models.
The interionic distance $x$ is used as single reaction coordinate since it naturally resolves the process of interest. As can be seen in Fig. 5.1 the free energy along $x$ reveals a high barrier at $x \approx 0.4$ nm separating the bound state at $x \approx 0.27$ nm from the free state at $x \geq 0.5$ nm. The bound state is very narrow and the adjacent barrier turns out to be relatively steep. Looking closer at the free energy, we see a second smaller barrier

at $x \approx 0.6$ nm which represents the transition between a shared and two separated hydration shells [141]. The considered value range of the ion distance $x$ is restricted to $0.265$ nm $< x <$ $1.265$ nm [125] by removing trajectory points outside of this region completely from the trajectory. The idea behind this procedure is to mimic a more natural spherical system instead of the simulated cubic box, i.e., border artifacts at large $x$ should be removed. On the other hand, the lower border of $x$ is imposed to get rid of spurious behavior at artificially low ion distances.

For completeness it should be noted that it is somewhat bold to assume that $x$ represents a complete description of all important dynamics of NaCl. It is well known that the solvent plays an important role as well [141] which is not surprising considering that solvent and NaCl are of the same size. Still, it will be shown in the following that $x$ is sufficient to construct satisfactory models. As first step we will derive a dLE model of the dynamics of NaCl.

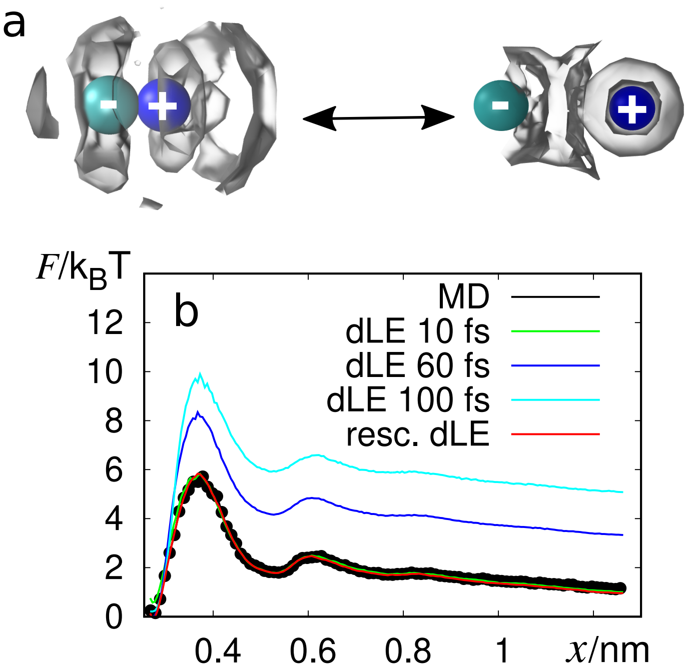

Figure 5.1: **Association and dissociation of NaCl in water.** (a) The two ions of sodium chloride associate and dissociate if solvated in water. The interionic distance $x$ can be used to resolve this process. (b) The free energy landscape reveals a pronounced barrier separating those states. A normal dLE and the rescaled dLE at $\delta t = 10$ fs (green and red, respectively) successfully reproduce the free energy while dLEs at $\delta t = 60$ fs (blue) and $\delta t = 100$ fs (cyan) overestimate the depth of the bound state. The illustrations of the associated and the dissociated state in the top figure are adopted from [42].

### 5.1.1 dLE modeling of sodium chloride

In parallel to Sec. 4.1.2, we start the dLE modeling with the determination of the two limiting time steps $\delta t_{\mathrm{M}}$ (following from the Markovianity condition) and $\delta t_{\mathrm{R}}$ (based on the need to resolve the dynamics sufficiently fine). To identify $\delta t_\mathrm{M}$ one can investigate the noise $\xi$ found in the data by the dLE as described in Sec. 4.1. Different time steps were tested starting at $\delta t = 10$ fs. Fig. 5.2a shows that this time step is too short to ensure that the noise autocorrelation $C_\xi$ decays in a single time step. When increasing $\delta t$, it turns out that $\delta t \geq 60$ fs is needed to observe the expected instantaneous decay. This observation is independent of position $x$, as can be seen in Fig. 5.2b, locally calculated noise autocorrelations show approximately the same first step $C_\xi(\delta t)$ everywhere along $x$. Another important aspect of $\xi$ is that it is expected to follow a normal distribution. Fig. A.1 in the Sec. A.6 reveals that the noise at $\delta t = 10$ fs meets the expectations while $\delta t = 60$ fs results in deviating distributions at small $x$, i.e., in the bound state. We see a clear bias to negative numbers, the peak of the distribution is shifted from $\xi = 0$ to $\xi \approx -0.5$. For larger time steps, like $\delta t = 100$ fs, this effect becomes even more severe.

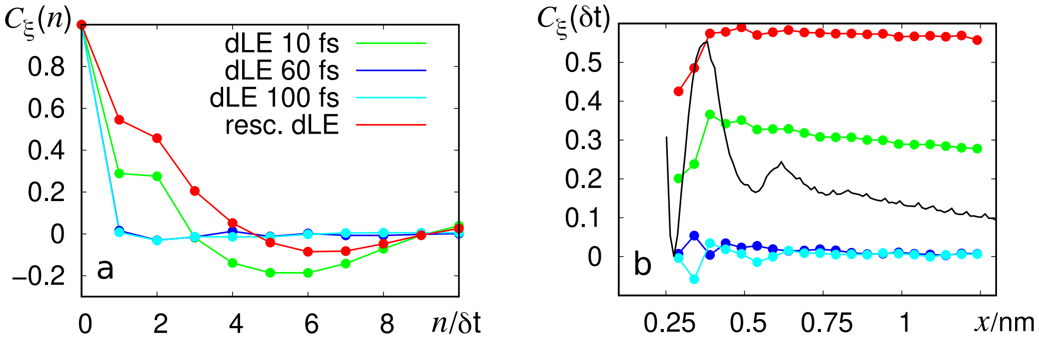

Figure 5.2: **Reconstructed noise for NaCl.** (a) The autocorrelation of the noise $\xi$ does not decay instantaneously for the smallest time step $\delta t = 10$ fs (green). The same holds for the rescaled dLE (red) since it uses the same $\delta t$. At larger time steps of $\delta t = 60$ fs (blue) and $\delta t = 100$ fs (cyan) the autocorrelation decays, as wished, immediately. (b) The locally calculated first step of the noise autocorrelation $C_\xi(\delta t)$ shows that it is approximately independent of the position.

We can conclude that the noise check indicates that NaCl might have the problem that $\delta t_\mathrm{R} < \delta t_\mathrm{M}$ could hold, i.e., the time step needed to observe Markovian dynamics might be larger than allowed by the resolution criterion. To check this suspicion dLE simulations at different time steps were performed. As shown in Fig. 5.1, we can, indeed, only reproduce the correct free energy for $\delta t = 10$ fs. Larger time steps lead to an overestimated depth of the bound state and by this to a higher barrier for the bound → unbound transition. The remaining parts of the free energy and the barrier of unbound → bound are well reproduced by all dLEs in contrast. This makes sense considering that the bound state is very narrow compared to the rest of the free energy, i.e., it is the most complicated feature to resolve by the dLE.

Hence, we observe that $\delta t_\mathrm{R} = 10$ fs (needed to resolve the dynamics by the dLE) is smaller then the Markovian limit $\delta t_\mathrm{M} = 60$ fs. To circumvent this problem it is possible to analyze the dLE estimates of friction $\Gamma$ and mass $\mathcal{M}$ aiming for a model which can be used by Langevin simulations integrated at a $\delta t$ which is small enough to resolve the bound state. As we see in Fig. 5.3, both fields are approximately constant at $\delta t = 10$ fs.

The mass approximately reproduces the reduced mass $\mathcal{M} \approx 211$ ps of the NaCl dimer. When increasing $\delta t$ to $\delta t = 60$ fs or $\delta t = 100$ fs, we can see that $\Gamma$ and $\mathcal{M}$ start to develop a pronounced peak in the bound state. Given that the dLE struggles at exactly this region of the free energy, it is reasonable to assume that those peaks are artificial, i.e., the constancy of the fields observed at $\delta t = 10$ fs appears to be more reliable than the dependence on $x$ for larger time steps.

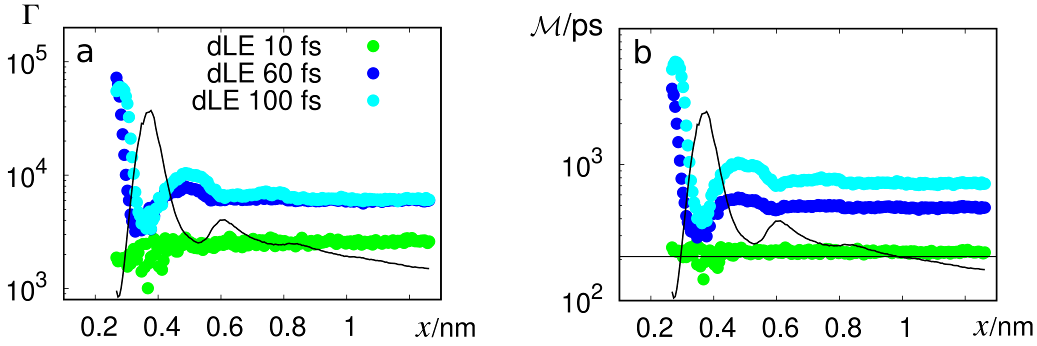

Figure 5.3: **Friction and mass estimates of the dLE.** The estimates of the friction $\Gamma$ (a) as well as the estimates of the mass $\mathcal{M}$ (b) show pronounced peaks in the bound state for larger time steps. The black curve in both plots shows the free energy for reference while the black horizontal line in (b) represents the reduced mass of the NaCl dimer.

Following this assumption, averages of $\Gamma$ and $\mathcal{M}$ were calculated based on dLEs at various different $\delta t$ by only taking points $x \geq 0.6$ nm into account. Fig. 5.4a shows that the means of friction and mass, called $\bar{\Gamma}$ and $\bar{\mathcal{M}}$, both increase with growing $\delta t$. In contrast to our double well model, Fig. 4.2 and Fig. 4.5, we do not see any plateau for $\bar{\Gamma}$ which would indicate a suitable friction estimate. This has the consequence that direct results of Langevin simulations using these fields need to be inspected to judge on the reliability of the field estimates. Hence, Langevin simulations using the Euler integrator were run employing an integration time step of $\delta t_{\text{integrate}} = 1$ fs to ensure that the bound state is sufficiently resolved. As dynamical observable to judge the performance of the different Langevin models we define the bound state by $x \leq 0.37$ and the free state by $x \geq 0.6$ and calculate the transition times between them, called association time $\tau_A$ and dissociation time $\tau_D$. Fig. 5.4b shows that the dLE estimates for $\delta t = 60$ fs and $\delta t = 100$ fs lead to good estimates of both times $\tau_D$ and $\tau_A$ and additionally to a good reproduction of the position autocorrelation $C_x(t)$. This indicates that the information $\tau_M \geq 60$ fs gained from inspecting the noise estimated by the dLE, see Fig. 5.2, is indeed accurate. If there had not been problems to resolve the bound state, the dLE would have predicted correct dynamics.

By comparing $\bar{\Gamma}$ estimated at $\delta t = 10$ fs to the estimate at $\delta t = 60$ fs we see that the dLE underestimates the friction at the smaller time step, just as it was observed and theoretically explained for the double well system above. Again, the dLE overlooks parts of the frictional force at $\delta t = 10$ ps which leads to an underestimation of the dynamics, i.e., $\tau_A$, $\tau_D$ and the decay time of the autocorrelation are predicted too small.

While the detour via Langevin simulations based on averaged dLE fields allows to create a reliable Langevin model, it would be less cumbersome if we could use the rescaled dLE, see Sec. 4.1.3, to correct for the underestimated friction at $\delta t = 10$ fs. Especially considering that the dLE modeling aims in general for high-dimensional system dynamics where it becomes very complicated to deduce an optimized Langevin model by hand.

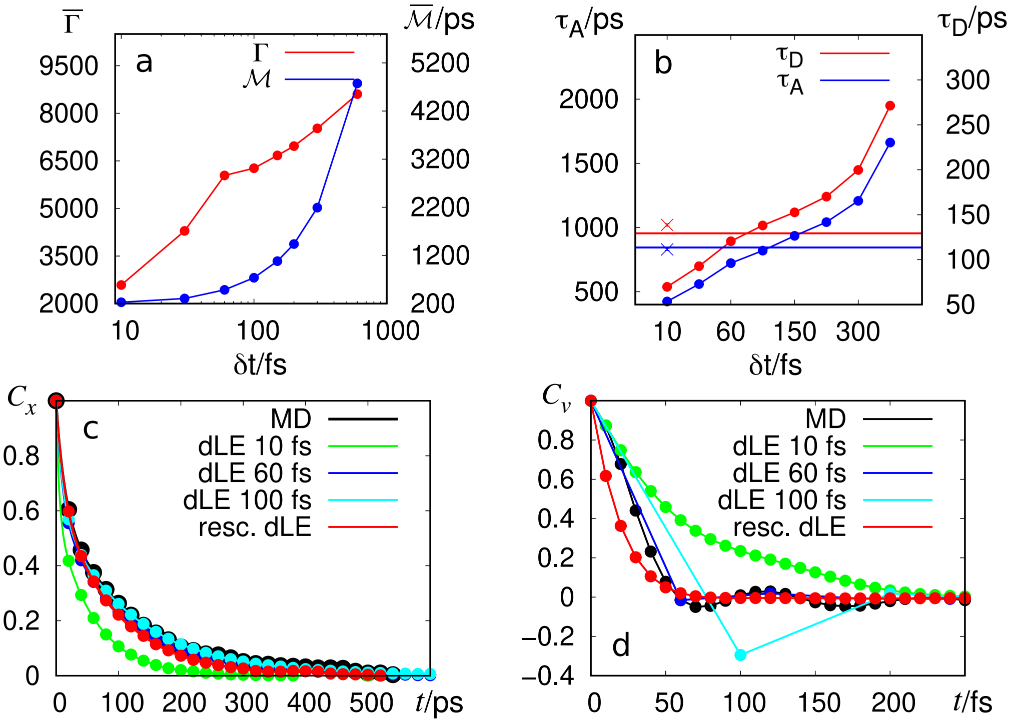

Figure 5.4: **Results of the dLE modeling of NaCl.** (a) Here, the averages of friction Γ and mass $\mathcal{M}$ calculated from dLEs at different time steps are shown. (b) Dissociation and association times, called $\tau_D$ and $\tau_A$, of Langevin simulations based on the averages in (a) and the given free energy. The two horizontal lines in this plot represent the MD values, the crosses estimates of the rescaled dLE. Additionally, the position autocorrelation $C_x(t)$ (c) as well as the velocity autocorrelation $C_v(t)$ (d) of MD and Langevin simulations are shown.

When applying the rescaled dLE we need to find a suitable rescaling factor $S$ which allows for reliable dLE dynamics at $\delta t = 10$ fs, i.e., we want to preserve the good resolution at this time step and correct for the underestimated friction. Using directly the transition times $\tau_A$, $\tau_D$ or the complete autocorrelation $C_x(t)$ to check the influence of different values of $S$ would massively harm the predictive power of the resulting model since we would only reproduce information which we put into the model beforehand. Hence, we will inspect if some short-time observable can be used to calibrate the rescaled dLE. One possible choice at this point is the initial decay of the position autocorrelation. As can be seen in Fig. 5.5, the Langevin models deduced from the dLEs at $\delta t = 60$ fs and $\delta t = 100$ fs approximately follow the initial decay of the MD autocorrelation when inspecting the time up to $t = 25$ ps. The dLE at $\delta t = 10$ fs decays too fast. When using $S > 1$ the rescaled dLE at $\delta t = 10$ fs decays slower than the normal dLE. At $S = 1.87$ the rescaled dLE coincides with the MD, i.e., this setup is a good candidate to predict long-time dynamics accurately. As shown in Fig. 5.4, the rescaled dLE indeed produces accurate transition times $\tau_A$ and $\tau_D$ and follows the complete decay of the autocorrelation $C_x(t)$ as well. Hence, the rescaled dLE successfully predicts long times only based on short-time information which means that it has predictive power.

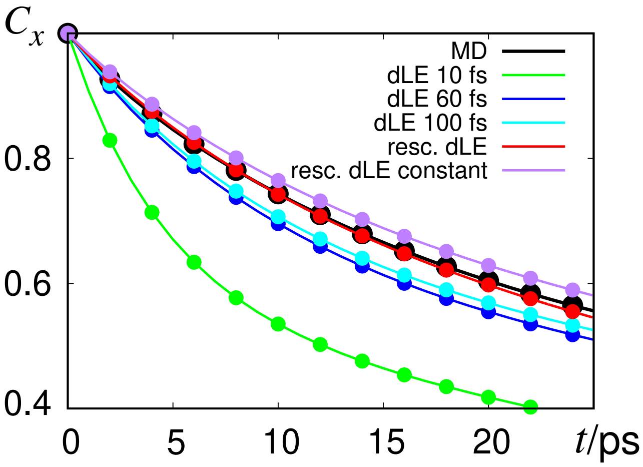

Figure 5.5: **Initial decay of the autocorrelation of MD and dLE.** Friction and mass estimated by dLEs at $\delta t = 60$ fs (blue) or $\delta t = 100$ fs (cyan) approximately reproduce the MD (black) while the dLE at $\delta t = 10$ fs decays too fast. The rescaled dLE with $S = 1.87$ (red) closely matches the MD. Averages of friction and mass deduced from the rescaled dLE lead to Langevin dynamics which deviate only marginally from the original rescaled dLE (purple).

Still, there are also some drawbacks. The noise deduced from the data via the rescaled dLE clearly reveals that we actively modify the Langevin model. The noise distribution (see Fig. A.1) becomes narrower than the normal distribution and the noise autocorrelation shown in Fig. 5.2 decays even slower than the unrescaled dLE at $\delta t = 10$ fs. Since the propagation of the rescaled dLE uses normally distributed white noise, we have to expect that the rescaled dLE will predict some characteristics of the MD data wrongly. Given that the rescaled dLE is especially aiming for correct long times and taking into account that the noise of the data decays on a short time scale of several tens of fs, it makes sense to assume that the rescaled dLE deviates from MD for dynamics which evolve on those fast time scales. When inspecting the velocity autocorrelation $C_v(t)$ with $v = (x_n - x_{n-1})/\delta t$, see Fig. 5.4, we see that the rescaled dLE indeed decays faster then the MD. Please note that $C_v$ decays on a time scale of 100 fs while $C_x$ decays on 100 ps, i.e., there is a clear separation of time scales between both observables. Given that $C_v(t)$ can be related to the memory kernel of an generalized Langevin description of the dynamics [142] it makes sense that the rescaled dLE, using perfectly Markovian noise, fails to reproduce the MD. Interestingly, the normal dLE at $\delta t = 10$ fs reproduces the first 20 fs of the reference $C_v(t)$ but fails for larger times. This shows that the rescaled dLE actively sacrifices short-time dynamics for correct long-time observables since it is

not possible to cover both by a Markovian Langevin model at $\delta t = 10$ fs.
Since friction and mass are nearly independent of $x$ for $\delta t = 10$ fs, see Fig. 5.3, it makes sense to assume that averages of both fields derived from the rescaled dLE provide a consistent model as well. When inspecting Langevin simulations using those averages it turns out that this is indeed true. Fig. 5.5 shows that the initial decay of $C_x(t)$ matches MD and rescaled dLE. The same holds for long-time observables like $\tau_A$ and $\tau_D$. Hence, we can conclude that a Langevin model consisting of the free energy, a constant friction $\Gamma$ and a constant mass $\mathcal{M}$ is sufficient to cover the long-time dynamics of NaCl.
As final point of this section we have to discuss a technical aspect. Considering the estimator of $\hat{\Gamma}$, see Eq. (4.12), it turns out that large values of $S$ may lead to consistency problems. When approximating the denominator of Eq. (4.12) by $\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m) = \sigma^2_{\Delta\boldsymbol{x}_m} \approx \sigma_{\Delta\boldsymbol{x}_{m+1}}\sigma_{\Delta\boldsymbol{x}_m}$ it turns out that the friction can be approximated by the negative velocity autocorrelation $\hat{\Gamma} \approx -C_v(\delta t)$ which means that it is restricted to $0 < |\hat{\Gamma}| < 1$. While relatively small rescaling factors, like $S = 1.87$ found for NaCl, are unproblematic, large $S$ may lead to friction factors contradicting the limits on $\hat{\Gamma}$. Still, there is a way to counter this effect. Considering Eq. (4.7) to (4.9) (for the purpose of generalization in the multidimensional case) we see that there is a clear rule how the time step $\delta t$ enters the different fields $\hat{\boldsymbol{f}}$, $\hat{\Gamma}$ and $\hat{\mathcal{K}}$. Since $\delta t$ represents the integration time step of the numerical scheme used to integrate the dynamics, it is possible to rescale it by a factor $0 \le \alpha \le 1$ so that $\delta t' = \alpha\delta t$ becomes the new integration time step. This idea leads to the dLE fields $\hat{\boldsymbol{f}}' = \alpha^2\hat{\boldsymbol{f}}$, $\hat{\Gamma}' + \mathbb{1} = \alpha(\hat{\Gamma} + \mathbb{1})$ and $\hat{\mathcal{K}}' = \alpha^{3/2}\hat{\mathcal{K}}$. Considering that $(\hat{\Gamma} + \mathbb{1})$ is multiplied by $S$ and $S^T$, we see that $\alpha = \max(S_{ii})^{-2}$ guarantees that $0 \le |\hat{\Gamma}| \le 1$ holds for all $S$.
In summation, the dLE modeling of NaCl consisted of the following steps. First, the unrescaled dLE was applied to detect the two borders $\delta t_\mathrm{M}$ and $\delta t_\mathrm{R}$. Observing that $\delta t_\mathrm{M} > \delta t_\mathrm{R}$ holds, we used in a second step the initial decay of the position autocorrelation to calibrate the rescaling factor $S$ of the rescaled dLE. This procedure provided a dLE model which could successfully predict the long-time dynamics of NaCl. We expect (and will show in the following chapters) that this procedure can be used for other systems as well. Still, it might be necessary to rescale the dLE integration time step by $\alpha = \max(S_{ii})^{-2}$ to ensure model consistency.

### 5.1.2 Influence of memory

Up to this point we modeled NaCl based on a Markovian model. Since it might be possible that the rescaling of the friction could be circumvented by including system memory in the Langevin model, it can be instructive to inspect the practical influence of memory on the model dynamics. To this end we will now parameterize and propagate a GLE with an exponentially decaying memory kernel $K(t) = (\Gamma/\tau_K)e^{-t/\tau_K}$, see Sec. 3.3. $\Gamma$ and $\mathcal{M}$ are taken as averages of the rescaled dLE fields found in the last section. This makes it possible to directly compare memory-based and Markovian Langevin dynamics. To choose the decay time $\tau_K$ we can inspect the velocity autocorrelation $C_v(t)$ since this observable can be related to the memory kernel of a GLE model [142]. Starting at the water time scale of approximately 10 fs [42], $\tau_K$ is successively increased until the GLE simulations match the initial decay of $C_v(t)$. Fig. 5.6 shows that $\tau_K = 30$ fs represents a good choice. GLEs with smaller decay times and especially the (Markovian) rescaled dLE show faster initial drops. The $C_v(t)$ of GLEs with $\tau_K$ on the order of tens of fs decay in general below zero, just as the MD, whereas the $C_v(t)$ of the rescaled dLE decays exponentially. Still, the MD deviates from all Langevin models for $t > 50$ fs, the GLE with $\tau_K = 30$ fs shows a deeper minimum than the MD, for example. Nevertheless, overall the GLE closer resembles $C_v(t)$ of the MD compared to the Markovian model. Assuming that $2 \cdot \tau_K$ represents approximately the decay time of the system memory, we note that $\tau_K = 30$ fs nicely fits to our dLE observations where we found that $\delta t \geq 60$ fs is needed to detect Markovian noise in the data.

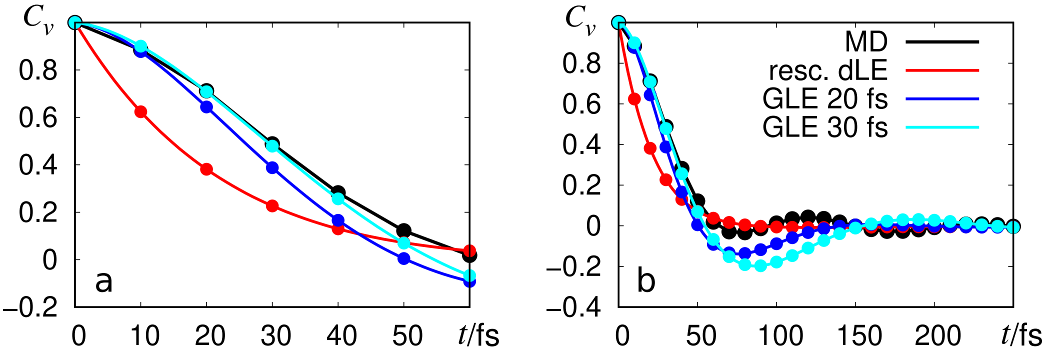

Figure 5.6: **Velocity autocorrelation of rescaled dLE and GLE.** (a) Here, we see that GLE simulations using $\tau_K = 30$ fs (cyan) reproduce the initial decay of the MD velocity autocorrelation (black) whereas GLE simulations with $\tau_K = 20$ fs (blue) and the rescaled dLE (red) deviate. (b) Still, all Langevin models deviate for $t > 50$ fs.

Considering long-time observables like $\tau_A$ and $\tau_D$, GLE simulations predict the same dynamics as the rescaled dLE. This makes sense considering that several orders of magnitude lie between $\tau_K = 30$ fs and $\tau_A$, $\tau_D$ and remembering that already the Markovian rescaled dLE performed well. Hence, we conclude that the explicit consideration of short-time memory only improves the Langevin model of NaCl if short-time observables like $C_v(t)$ are considered.

### 5.1.3 Other Markovian models

Given that the two-state dynamics of NaCl are not complicated to model by some rate matrix, it is not surprising that it is possible to construct a reasonable MSM. Using the state definitions $x \leq 0.37$ nm for the bound and $x \geq 0.6$ nm for the free state, we obviously prevent the observation of spurious, short-living transitions between the two states, see Sec. 4.4, since they are well separated. Considering that $\tau_A$ and $\tau_D$ are of the order of several hundreds of ps, we do not need a time resolution of the order of fs so that it is valid to work with the 1 $\mu$s long MD trajectory which has the minimal resolution of $\delta t_0 = 1$ ps. As shown in Fig. 5.7, the implied time scale of the two-state MSM is approximately constant which means that we can directly use the shortest possible lag time $\tau = \delta t_0 = 1$ ps to produce some MCMC trajectory to test the predictive accuracy of the MSM. As dissociation time we get $\tau_{D,\mathrm{MSM}} = 129$ ps and as association time $\tau_{A,\mathrm{MSM}} = 847$ ps which perfectly fits to the MD values of $\tau_{D,\mathrm{MD}} = 129$ ps and $\tau_{A,\mathrm{MD}} = 845$ ps.

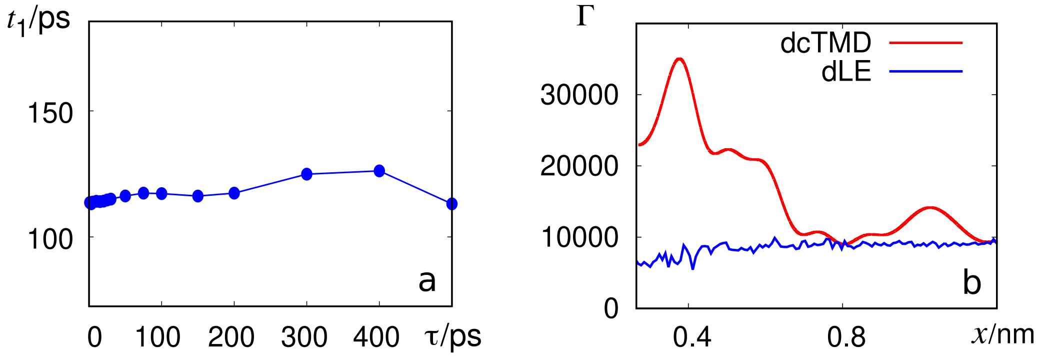

Figure 5.7: **Alternative Markov modelings of NaCl.** (a) The implied time scale of an MSM for NaCl. (b) Here, we see the comparison of the rescaled dLE friction estimate (blue) to the friction estimated by dcTMD (red) [42]. The right figure is taken from the supplementary information of [125].

As introduced in Sec. 4.3, it is also possible to derive a Markovian Langevin model from dissipation-corrected targeted MD simulations (dcTMD). This approach was recently used to derive a Langevin model for NaCl [42]. While the free energy estimate of dcTMD coincides with the dLE free energy (and by this the equilibrium MD), the friction estimated for small $x$ is systematically larger than the rescaled dLE estimate [125]. This is not surprising considering that it is known that constraints increase the effective friction of the system under study [143]. Nevertheless, we see that both friction estimates converge approximately to the same value for larger $x$. It can be assumed that the reason for the deviations at small $x$ lies in the nontrivial breaking of the water shell surrounding the bound NaCl system which is enforced by dcTMD when both ions are pulled apart [42]. For large $x$, in contrast, the fluid dynamics are much simpler so that constraints do not induce relevant artifacts. When integrating the Langevin equation using the dcTMD estimate of Γ, the transition dynamics are underestimated by a factor of 3 [125], i.e., $\tau_A$ and $\tau_D$ are too large. This makes sense since the comparatively larger friction leads to slower dynamics so that the deviations to MD are stronger than for the rescaled dLE which underestimates the dynamics only slightly, see Fig. 5.4.

## 5.2 Study of AIB$_9$

In this section we inspect the dynamics of the small AIB$_9$ peptide (H$_3$C-CO-(NH-C$_\alpha$(CH$_3$)$_2$-CO)$_9$-CH$_3$). In contrast to NaCl, the system description $\boldsymbol{x}$ will be multi-dimensional for this system, i.e., the modeling gets more complicated. Additionally, we face the problem that the enhanced sampling MD data used as modeling input [73] is very large so that a pre-averaging, see Sec. 4.1.5, needs to be done to be able to apply the dLE framework. But we inspect AIB$_9$ not only because it is more complex than NaCl. As will be explained in the next section, unbiased (long) MD simulations [70] and simulations based on an alternative enhanced sampling scheme [140] provided contradicting information on the nature of the most relevant conformational change of $AIB_9$. Hence, we can use the predictions of our models derived from the enhanced sampling data of Biswas et al. [73] to check which one of the two perspectives is supported. But before this investigation can be done we need to get familiar with the basic aspects of $AIB_9$

### 5.2.1 System characteristics

Although it consists of only nine residues, $AIB_9$ exhibits nontrivial dynamics [70]. As depicted in Fig. 5.8, the overall conformation is predominately left-handed (called L in the following) or right-handed (called R) with transitions on a time scale of 0.1 $\mu$s. Still, these transitions require that the individual residues do their own transition l↔r. Here, l and r refer to the left and right-handed states of the individual residue. The observation of individual transitions necessarily indicates that the exited states $l^*$ or $r^*$ are reached beforehand. This dynamics occur on time scales 1 ns, i.e., are significantly faster than L↔R. Though, the exited states are not randomly reached, certain H-bonds need to break at time scales of 10 ps to allow for the needed changes of the dihedral angles. This shows that $AIB_9$ exhibits hierarchical dynamics, i.e., slow conformational changes of the overall shape require previous changes on the level of individual residues which happen much faster.

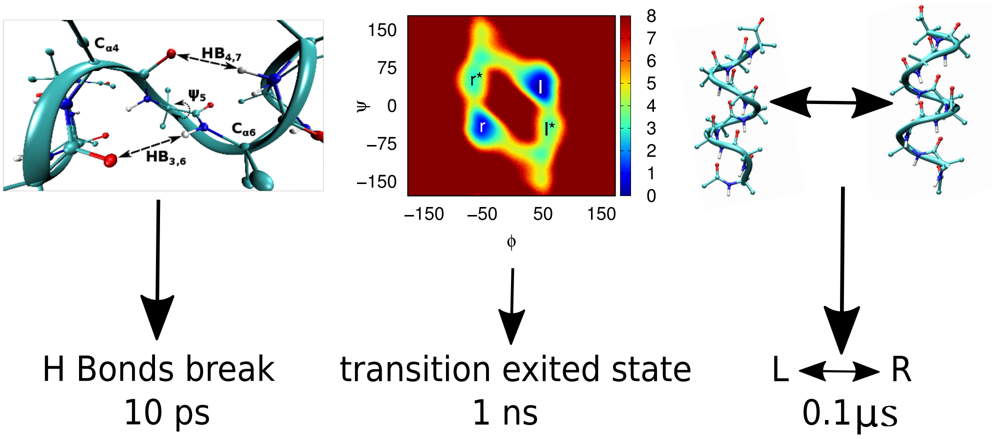

Figure 5.8: **Time scales of $AIB_9$.** The breaking of H-bonds represents the fastest observed time scale at 10 ps. Based on this process, it is possible that single residues of the system reach the exited states $r^*$ or $l^*$ at time scales 1 ns. By crossing these exited states the residue can reach the other ground state to finish the transition l↔r. Based on these transitions of the individual residues the whole system can change its collective conformation from state L to state R at a time scale of 0.1 $\mu$s. The figure on the left as well as the peptide representations on the right are taken from [70].

The reaction coordinates we are going to use were defined by Buchenberg et al [70] based on eight 2 $\mu$s MD trajectories at a temperature of 320 K. The simulation details can be found in Sec. A.4.2. Trajectory points were saved with a resolution of $\delta t = 1$ ps. It turned out that the L↔R dynamics of $\mathrm{AIB}_9$ are well described by the backbone dihedral angles $\phi_i$, $\psi_i$ of the inner five residues, the two residues at both ends of the system exhibit mainly fluctuations. By doing a principal component analysis with these angles (dPCA), it is possible to further reduce the dimensionality of the final system description. It should be emphasized that the original dPCA [144] based on the sine/cosine transformation of the angles is applied and not the improved dPCA+ method [97] which uses directly the angles. The first five principal components $x_1$ to $x_5$ (collecting 85% of the total variance) show multipeak distributions and slowly decaying autocorrelations which indicates that they represent a suitable system description.

After projecting the dynamics on the first two PCs, see Fig. 5.9, it becomes apparent that $x_1$ mainly separates the two main states L and R while $x_2$ resolves the two main pathways connecting them. By labeling the individual conformation of each of the inner five residues as l or r, the intermediate states can be identified as states of the shape "rrrll" and we see that the two pathways represent basically mirror images of each other, in both cases the conformational change leading to L=lllll↔rrrrr=R starts at one of the outer residues and propagates along the chain. The other combinatorily possible intermediate states, like, e.g., rlrrr, can be found between the two main pathways in $x_1$-$x_2$-projection. Although eight times 2 $\mu$s appears to be sufficient data for such a small system, it turns out that those states are only sparsely sampled which makes it hard to judge whether or not they contribute to relevant pathways of the transition L↔R in converged dynamics. Additionally, as already mentioned above, an MSM pathways analysis of short MD trajectories with implicit solvent seeded by the MELD (Modeling Employing Limited Data) protocol [140] indicated that transitions involving those states contribute up to ≈ 40% to the overall flux [145, 146].

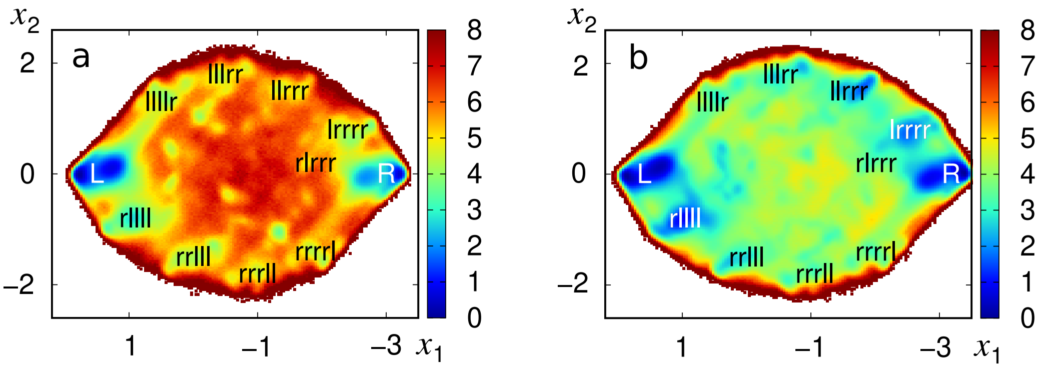

Figure 5.9: **Statistics of the two MD data sets.** (a) The unbiased MD data [70] reveals two main paths for the process R↔L but the overall free energy is only sparsely sampled. (b) The MD data based on the enhanced sampling scheme of Biswas et al. [73] covers the conformational space significantly more homogeneously. Note that the landscape in (b) does not represent the free energy of the system, it only shows the sampling of the data which consists of numerous short trajectories. The color code provides the energy in units of $k_\mathrm{B}T$.

To investigate the contradiction between unbiased MD and MELD studies, a second data set based on the enhanced sampling scheme presented by Biswas et al. [73] was produced. It consists of approximately 7700 short unbiased MD trajectories with in total $77.6 \cdot 10^6$ data points at a resolution of $\delta t = 1$ ps. Briefly summed up, the enhanced sampling scheme constructs a rough scan of the considered free energy landscape with metadynamics [19] followed by the generation of an extensive amount of (unbiased) short MD trajectories starting equally distributed over the free energy landscape. The metadynamics simulation was performed using the GROMACS program suite [17, 147] patched with PLUMED [148]. The simulation setup of both metadynamics and the short MD trajectories is similar to the setup used by Buchenberg et al. [70]. As ensemble, the short trajectories aim to homogeneously sample the full conformational space of the considered system. In case of AIB$_9$, the short trajectories with a length of 10 ns each indeed cover the complete energy landscape, see Fig. 5.9b. Biswas et al. [73] used Markov state models to interpret the dynamical information of the data, we will compare the dLE results to this analysis. We note that it is not possible to apply the conventional dLE to the full $77.6 \cdot 10^6$ points due to unreasonable calculation times which means that some pre-averaging, see Sec. 4.1.5, must be performed.

### 5.2.2 dLE modeling of AIB$_9$

Still, before the pre-averaging strategy is applied to the full AIB$_9$ data set we will first identify a suitable modeling time step $\delta t$. As shown for the exemplary data in Sec. 4.1.2 and for NaCl in Sec. 5.1.1, dLE fields and model dynamics strongly depend on the chosen $\delta t$. To apply the established dLE modeling steps, we use data subsets with time steps of 2, 6, 10, 20, 50 and 100 ps which are small enough to allow for acceptable calculation times. At $\delta t = 10$ ps, for example, we only take every tenth point and end up with $8 \cdot 10^6$ data points. This specific data set was also used in [73] to do the Markov state modeling.

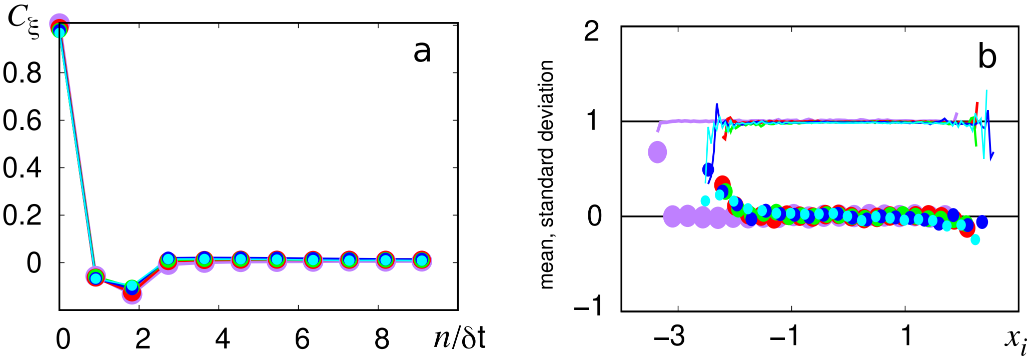

Figure 5.10: **Reconstructed noise for $\delta t = 2$ ps.** Noise autocorrelation (a) as well as (b) the mean (dots) and the standard deviation (lines) of the back-calculated noise behave as expected for $\delta t = 2$ ps. The different colors represent $\xi_1$ to $\xi_5$, i.e., all coordinates behave similar.

To derive a lower bound $\delta t_{\mathrm{M}}$ for $\delta t$, i.e., to exclude non-Markovian dynamics, one can reconstruct the noise $\boldsymbol{\xi}$ found by the dLE in the input data. Already the smallest used

time step $\delta t = 2$ ps provides noise estimates which fulfill the expectations. Fig. 5.10 shows that the autocorrelations of all components of $\boldsymbol{\xi}$ decay in one $\delta t$ and that mean and standard deviation reproduce the expected values 0 and 1 as well.
Subsequently, similar to NaCl, the dLE performance at all selected time steps $\delta t \geq 2$ ps was investigated by producing for every time step ten dLE trajectories. Every dLE run has a length of $3 \cdot 10^6$ frames. When inspecting the autocorrelations along the different coordinates, see Fig. 5.11, it turns out that $\delta t = 10$ ps appears to be the best candidate to get optimal dLE dynamics since the autocorrelation along $x_1$ is quantitatively covered while the other coordinates fit at least qualitatively.
This raises the question why $\delta t = 2$ ps does not provide suitable dLE dynamics. At this point it is important to recall that the observation of Markovian noise in the data represents a necessary but not a sufficient condition for an accurate dLE model. Some compensation of errors might overshadow the remaining system memory at $\delta t = 2$ ps so that the reconstructed noise appears Markovian while the overall dynamics are still influenced by memory effects.

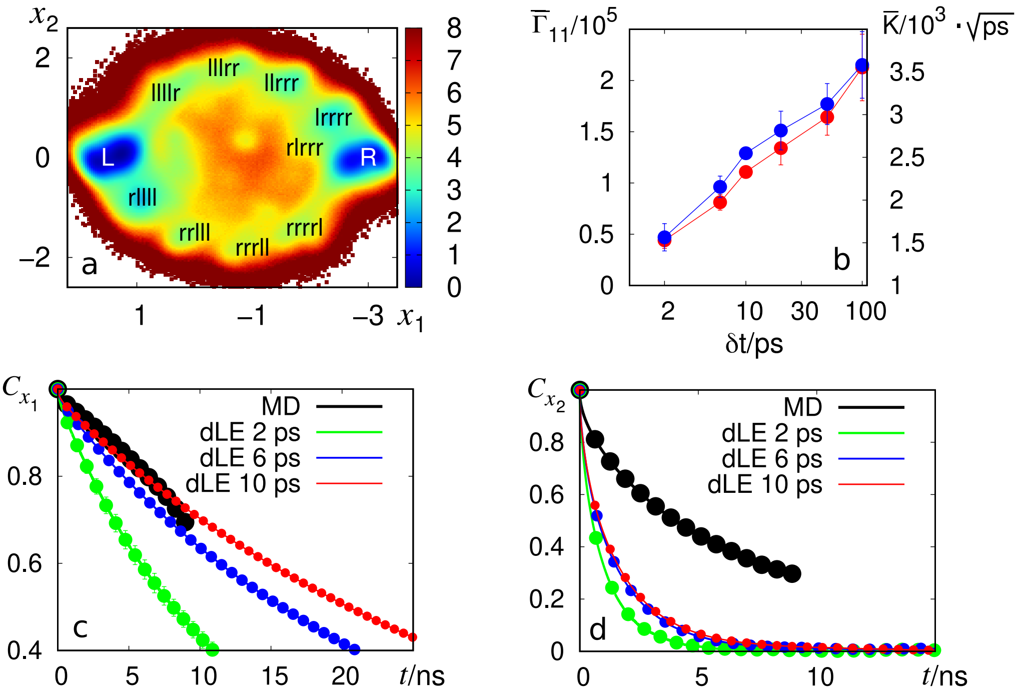

Figure 5.11: **dLE results for AIB$_9$.** (a) The dLE free energy (in units of $k_\mathrm{B}T$) at $\delta t = 10$ ps. (b) Here, we see the evolution of $\Gamma_{11}$ (red) and $\mathcal{K}_{11}$ (blue) with $\delta t$. The bottom row presents position autocorrelations along $x_1$ (c) and $x_2$ (d) of MD and dLEs at different $\delta t$. The MD autocorrelations are based on the data of Biswas et al. [73].

Inspecting the friction and noise field estimates $\Gamma$ and $\mathcal{K}$ reveals that the diagonal elements systematically increase with $\delta t$, just as we know it from NaCl. Interestingly, the states R and L show increased fields which shows their prominent role for the dynamics of AIB$_9$. The off-diagonal elements oscillate around zero and are significantly smaller.

Fig. A.2 to A.5 show the evolution of exemplary components of $\Gamma$ and $\mathcal{K}$. When calculating averages of the diagonal elements of $\Gamma$ and $\mathcal{K}$ for different $\delta t$, see Fig. 5.11b, we observe the same behavior as for NaCl, the fields grow monotonously with $\delta t$.

Since the off-diagonal elements of $\Gamma$ and $\mathcal{K}$ are significantly smaller than the diagonal parts it might be possible that the Langevin model does not depend on them at all. By defining $\Gamma_{i,j} = \mathcal{K}_{i,j} = 0, \; \forall \; i \neq j$ for an alternative dLE propagation, it is possible to check this hypothesis. After using again $\delta t = 10$ ps, we see in Fig. 5.12 that diagonal fields lead to slightly larger populations of the main states L and R compared to the normal dLE. Still, the autocorrelations, exemplarily shown along $x_1$, reveal that the overall dynamics are similar to the normal dLE.

Going one step further it can be tested if it is even possible to approximate the diagonal components of $\Gamma$ and $\mathcal{K}$ by constant values. This can be done by calculating averages $\hat{\Gamma}_{ii}$ and $\hat{\mathcal{K}}_{ii}$ via extending the neighborhood average (4.10) over the full input data. In this way we favour the field estimates in the minima over the values on the barrier which makes sense considering that the dynamics spend most time in the minima so that the statistical uncertainty is expected to be lower here. However, when combing $\hat{\Gamma}_{ii}$ and $\hat{\mathcal{K}}_{ii}$ with the (still local) estimate of the drift field $\hat{\boldsymbol{f}}$, the dLE dynamics are far off and the free energy becomes very faulty (not shown). This indicates that oscillations of the drift field in sparsely sampled regions need to be compensated by oscillations of the other two fields. If we constructed an analytically defined model of the free energy [45], not trivial in five dimensions with sparsely sampled barriers, it would be highly probable that we could find constant values for $\Gamma_{ii}$ and $\mathcal{K}_{ii}$ leading to a reliable Langevin model. Though, when relying on the oscillating dLE drift $\hat{\boldsymbol{f}}$ this is not possible.

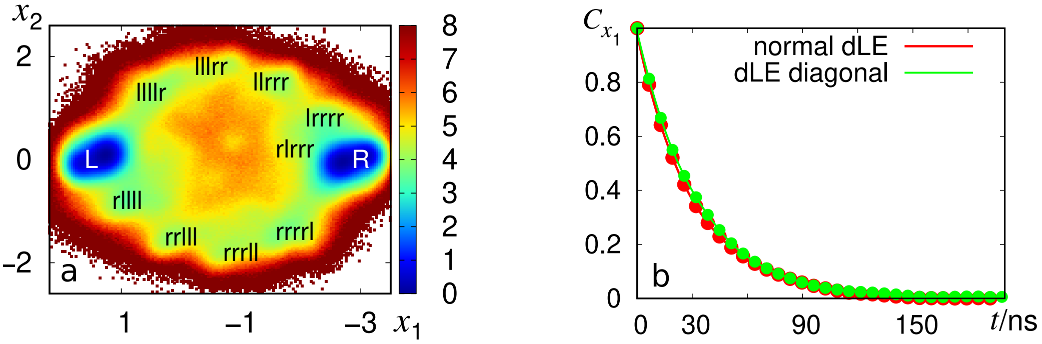

Figure 5.12: **dLE model with diagonal $\Gamma$ and $\mathcal{K}$ for AIB$_9$.** (a) The free energy (in units of $k_{\mathrm{B}}T$) of the dLE with diagonal fields emphasizes the states L and R stronger than the normal dLE. (b) Still, the overall dynamics are very similar as can be seen by the autocorrelation along $x_1$ compared to the normal dLE.

### 5.2.3 Binned dLE performance

Up to this point, we only used a subset of the enhanced sampling MD data for our Langevin studies. Still, it is always preferred to use all available MD data when constructing a Langevin model. This will be done now by using the binned dLE introduced

in Sec. 4.1.5.
Since we only tested the pre-averaging approach for simple model data so far, we need to make sure that five-dimensional MD data can be treated in the same way. To do so, it needs to be verified that the subset of $8 \cdot 10^6$ data points at $\delta t = 10$ ps can be pre-averaged without harming the dLE model. By testing different sets of pre-averaging parameters it turns out that it is possible to go down to only $0.97 \cdot 10^6$ input points without effecting the dLE dynamics. The green curve Fig. 5.13b represent the autocorrelation of the binned dLE based on this data. It nicely coincides with the result of the normal dLE shown in red. The underlying parameters of this pre-averaging are $s = 35$, $N_{\max} = 5$, $\omega_{\min} = 0.0015$ and $\omega_{\max} = 0.015$.

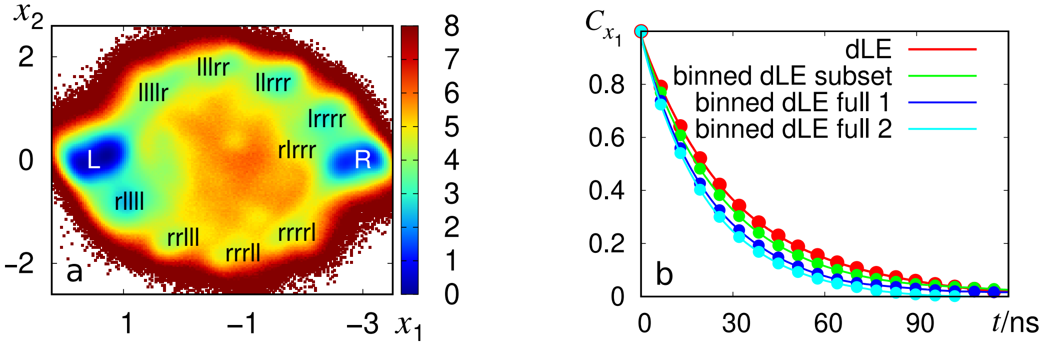

Figure 5.13: **Binned dLE results for AIB₉.** (a) The free energy (in units of $k_{\mathrm{B}}T$) of the binned dLE based on the maximally possible pre-averaging of the full input data. (b) Position autocorrelations of binned dLEs based on different pre-averagings are compared to the normal dLE (red) for $x_1$. In green we see the maximally possible pre-averaging of the subset of $8 \cdot 10^6$ data frames, in blue the result when applying this setup to the full input data set and in cyan the maximally possible pre-averaging of the full input data.

To use the full data set with a time step of 10 ps, we separate each of the 7732 short trajectories recorded at $\delta t_0 = 1$ ps into ten subtrajectories were the first subtrajectory consists of of the points $\boldsymbol{x}(0\text{ ps}), \boldsymbol{x}(10\text{ ps}), ...$, the second of $\boldsymbol{x}(1\text{ ps}), \boldsymbol{x}(11\text{ ps}), ...$ and so on. When applying the pre-averaging setup found for $8 \cdot 10^6$ data points to the full data of $77.6 \cdot 10^6$ data frames we get a reduction to $7.8 \cdot 10^6$ input points. The dynamics of a binned dLE constructed for this data varies only slightly, see the blue curve Fig. 5.13b. This shows that the subset of $8 \cdot 10^6$ data frames used to conduct the dLE studies above already contains enough information to allow for a valid modeling.
Since this pre-averaging results in relatively many points for the full data set, additional, more aggressive, pre-averagings were tested. It was possible to further reduce the number of data points to $0.7 \cdot 10^6$ frames without spoiling the dLE dynamics, see the cyan trace in Fig. 5.13c and the free energy Fig. 5.13a. The used parameters are $s = 25$, $N_{\max} = 120$, $\omega_{\min} = 0.0025$ and $\omega_{\max} = 0.025$. Please note that this pre-averaging harms the dLE dynamics when applied to the subset of $8 \cdot 10^6$ data frames which shows that optimal pre-averaging parameters do not only depend on the considered system but also on the available data set.

To summarize the results of the detailed study of the binned dLE, it can be concluded that the pre-averaging strategy works very well for the five-dimensional description of AIB$_9$, the modeling results coincide with the normal dLE. It is possible to reduce the full data set of $80 \cdot 10^6$ data frames down to $0.7 \cdot 10^6$, i.e, the amount of data points can be cut by a factor of 100. In addition, we have seen that the subset of $8 \cdot 10^6$ input frames used to conduct the conventional dLE analysis leads approximately to the same modeling results.

### 5.2.4 Rescaled dLE and comparison to Markov state model

If we inspect the free energy landscapes predicted by the dLE for $\delta t = 10$ ps a bit closer, see Fig. 5.11 top left, it turns out that the various minima look slightly blurred. Fortunately, the rescaled dLE can be used to construct a valid Langevin model already at $\delta t = 2$ ps. Based on the initial decay of the autocorrelation function of the MD (not shown), a rescaling matrix of

$$S = \mathrm{diag}(1.73, 2.24, 2.24, 2, 2) \tag{5.1}$$

can be determined. Using the $0.7{\cdot}10^6$ pre-averaged data points derived in the last section as input, ten 20 $\mu$s-long dLE trajectories were produced. These simulations represent the best dLE model we can get for AIB$_9$. The free energy predicted by the dLE is shown in Fig. 5.14 top left. The different minima are more constricted compared to the dLEs at $\delta t = 10$ ps and overall the dLE free energy is very similar to the free energy of the long MD simulations by Buchenberg et al., see Fig. 5.9. Still, we note that the main intermediate states at the two edges, e.g., rllll or rrlll, are more populated by the dLE than in reference MD, i.e., the enhanced sampling of the dLE input data has some influences. The center of the free energy landscape, i.e., states like rlrrr, are in contrast barely more populated by the dLE. Considering the dynamical predictions of the dLE model, we see that the position autocorrelations of the MD are, by design, reproduced by the rescaled dLE for all five coordinates, see Fig. 5.14 top right and Fig. A.6. To calculate average waiting times, we first assign the dLE trajectories to a density-based clustering of the MD (done by Biswas et al [73]) as described in Sec. A.8. This way we ensure maximal comparability between the different data sets. Based on this state assignment, Fig. 5.14 shows bottom right a selection of transitions between the ten most populated states. dLE and MD show an average deviation of 50%. To get a quantitative impression of the deviations between both data sets, Tab. 5.1 compares the average waiting times of the main system transition L↔R. The reference MD samples this transition 74 times which indicates good statistics. Interestingly, the deviation between reference MD and dLE is larger for R→L than for L→R.

Having determined a suitable rescaling matrix $S$, we can now investigate the stability of the dLE with respect to reduced input statistics. Biswas et al. found in their studies that the Markov state model stays stable as long as $\geq 2000$ short input trajectories are used or as long as the 7732 input trajectories have a length $\geq 4$ ns. By limiting the dLE input data in the same way, two additional dLE sets were produced called "dLE short data" (based on 7732 times $\geq 4$ ns) and "dLE less data" (2000 times 10 ns). Inspecting autocorrelations and average waiting times, we see that reduced input statistics lead to faster dLE dynamics. This holds in particular for shorter input trajectories, here we see a decrease of 20% in the waiting times of the L↔R transition, see Tab. 5.1.

| | $\tau_{\mathrm{L}\to\mathrm{R}}$ [ns] | $\tau_{\mathrm{R}\to\mathrm{L}}$ [ns] |
|---|---|---|
| MD | $161 \pm 23$ | $80 \pm 12$ |
| MSM | $132 \pm 0.2$ | $67 \pm 0.1$ |
| dLE (full data) | $200 \pm 8$ | $140 \pm 6$ |
| dLE (short data) | $162 \pm 6$ | $113 \pm 4$ |
| dLE (less data) | $178 \pm 6$ | $132 \pm 5$ |

Table 5.1: **Average waiting times of the L↔R transitions of Aib$_9$.** Compared are the reference MD of Buchenberg et al. [70] to different dLE setups using the data of Biswas et al. [73] as input data and the MSM constructed in [73]. The three dLE models are based on the full input data, shorter input trajectories (4 ns) or fewer input trajectories (2000), respectively. Errors are calculated as standard deviations of the mean.

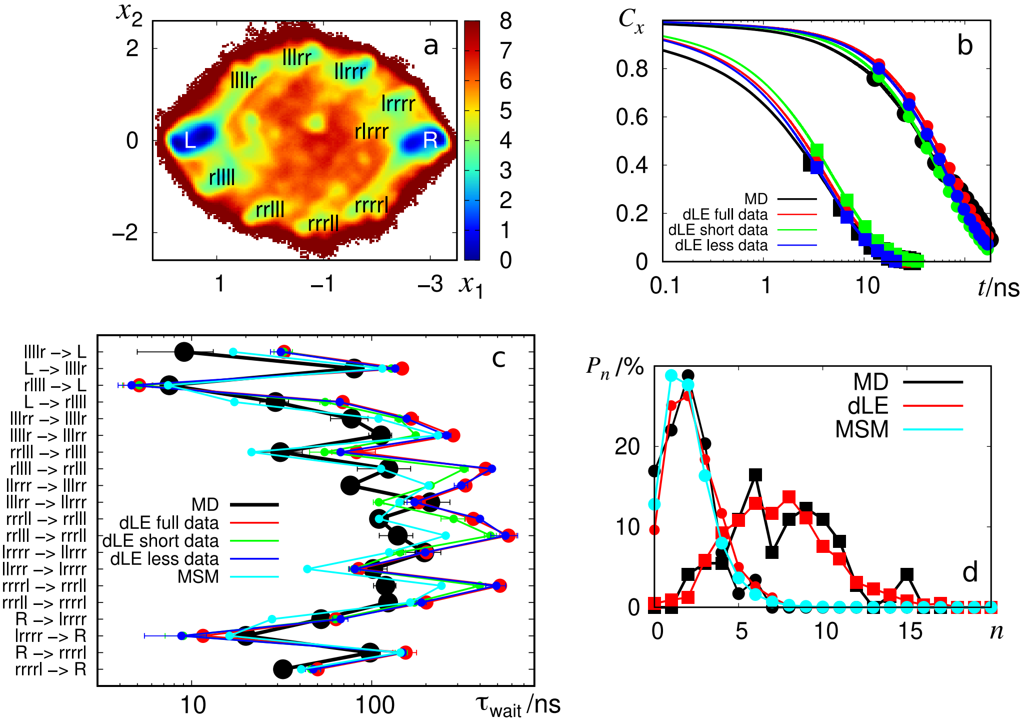

Figure 5.14: **Rescaled dLE results for AIB$_9$ and comparison to MSM.** (a) The minima of the free energy (in units of $k_{\mathrm{B}}T$) of the rescaled dLE are more constricted than for the normal dLE. (b) The autocorrelations of the long MD simulations are well covered by the rescaled dLE as we see for $x_1$ (●) and $x_2$ (■). (c) Here, we see exemplary average waiting times of MD, dLE and MSM. (d) This panel shows the probability of pathways to use exactly $n$ middle states given that they reach at least one main intermediate state. Here, time resolutions of 1 ns (●) and 2 ps (■) are compared.

For completeness, we can now compare the results of the dLE to the MSM model established by Biswas et al. [73]. Here, the subset of $8 \cdot 10^6$ data frames used in Sec. 5.2.2 was projected on the first five PCs and clustered using the density-based clustering [60] with a hypersphere radius of $R = 0.2$ yielding 102 states in total. The MSM was constructed with a lag time of $\tau = 1$ ns after inspecting the implied time scales. Based on the times shown in Fig. 5.14c, MSM and dLE show an average deviation of 43%. Considering the R↔L transition, Tab. 5.1, the MSM predictions deviate less from MD than the dLE, we find only differences $< 20\%$.

### 5.2.5 Comparison of dLE and MELD predictions

Having derived an optimized Langevin model via the rescaled dLE, we can now finally come back to the question which was raised by the studies of Perez et al. [140]: does the enhanced sampling scheme [73] used to generate our input data emphasizes the relevance of the center of the free energy, i.e., states like rlrrr? Was the sampling of Buchenberg et al. simply not good enough to see this relevance? Based on the free energy predicted by the rescaled dLE, Fig. 5.14a, we can already suspect that the Markov models do not confirm this assumption since it is very similar to the free energy of the reference MD of Buchenberg et al. To quantify the influence of the center of the free energy, one can isolate the L↔R pathways of reference MD, MSM and rescaled dLE and calculated the probability $P_n$ of pathways to use exactly $n$ states from the center region, like for example rlrrr, given that at least one main intermediate state, like, e.g, rrrrl, was reached. As we see in Fig. 5.14b, reference MD and rescaled dLE predict an unimodal distribution which peaks at $n \approx 6$ if the pathways are evaluated at a resolution of $\delta t = 2$ ps. When using 1 ns instead, the maxima are shifted to $n \approx 2.5$ and coincide with the prediction of the MSM. Hence, we can conclude that both models (which are based on the enhanced sampling data of Biswas et al.) predict pathways which are very similar to the reference MD. This indicates that MELD and unbiased MD lead to different dynamics of AIB$_9$, the former claims that states like rlrrr are very important for L↔R while the later emphasizes the main intermediate states like rllll or lrrrr.

## 5.3 Summary

Inspecting the modeling of NaCl, we saw in Sec. 5.1.1 that the normal dLE could not be applied successfully because the necessary time step was too large to allow for a sufficient resolution of the free energy. Nevertheless, the rescaled dLE could be used to derive an one-dimensional Langevin model based on the interionic distance $x$ which reproduces the MD dynamics [42] within an error of a few percent. The initial decay of the autocorrelation function could be used to calibrate the rescaling factor $S$. Additionally, when inspecting an alternative memory-based Langevin model in Sec. 5.1.2, we observed that the consideration of memory only improves the reproduction of short-time dynamics. Considering that it is known that the hydration shell dynamics are very important for NaCl [141], the excellent performance of a Markovian model is remarkable. We additionally observed in Sec. 5.1.3 that the rescaled dLE is not the only possibility to derive such a model, an MSM or a dcTMD-based Langevin model can be used as well.

Subsequently, we constructed a five-dimensional model of the dynamics of Aib$_9$ based on a large enhanced sampling data set [73] consisting of numerous short trajectories.

Considering a subset of the data, we determined a suitable dLE time step in Sec. 5.2.2 and tested carefully that the pre-averaging (needed to apply the binned dLE to the full data) did not harm the model dynamics in Sec. 5.2.3. The number of data points could be reduced by a factor of $10^1$ which shows that the binned dLE approach is very effective. Applying the rescaled dLE in Sec. 5.2.4, we were able to refine the Markovian Langevin model. Based on this optimization, we investigated the convergence behavior of the dLE with respect to number and length of the input trajectories. It was observed that the dLE frameworks tends to predict faster dynamics if the input trajectories are shorter or less numerous. When comparing the Langevin model predictions to those of an MSM constructed by Biswas et al. [73], we saw that both model frameworks predicted qualitatively the same dynamics. Compared to an alternative MSM based on the MELD (Modeling Employing Limited Data) protocol [140], we observed that Markovian models based on the data of Biswas et al. predict dynamics which emphasize the importance of the intermediate states at the "edges" of the free energy (like lrrrr) for the L↔R transition. This coincides with the results of long unbiased MD simulations performed by Buchenberg et al. [70]. MELD, in contrast, emphasizes the importance of states in the "center" of the free energy (like rlrrr).

# 6 Markov modeling of slow dynamics

*"One does not simply walk into Mordor!"*
– Boromir, "Lord of the Rings:
The Fellowship of the Ring"
(movie by Peter Jackson)

Although it is instructive to inspect small systems like NaCl and $\mathrm{AIB}_9$ to test the virtues and shortcomings of the Markovian modeling framework, we are eventually aiming for the modeling of more complicated dynamics where some simplified model is fundamentally needed to even have the chance to understand the system. Additionally, some model might be needed to predict system dynamics which are out of reach of MD simulations. Going in this direction, we will inspect the 164-residue T4 lysozyme in the first part of this chapter. Performing a prominent open↔closed transition on time scales of microseconds, the appropriate dimensionality reduction for T4 lysozyme is currently unclear. While not being able to solve this problem in this thesis, we will see how Markovian modeling can help to evaluate the informative value and completeness of low-dimensional system descriptions. First, we will inspect a set of coordinates based on a contact principal component analysis [71] which have been problematic for earlier Langevin-based studies of T4 lysozyme [149]. Here, we will try to understand the problems of this system description. Going further, we use Markov state models to inspect a two-dimensional system description derived earlier [71, 150] before we inspect the influence of additional coordinates on the accuracy of the MSM. Afterwards, the performance of the dLE framework for the two-dimensional system description is evaluated. We will have to use the rescaled dLE to derive a reasonable model which covers the long time scales based on a small time step.
Moreover, we will inspect even slower dynamics in the last section of this chapter. Considering the unbinding of benzamidine from trypsin and the unbinding of a resorcinol scaffold-based inhibitor from the N-terminal domain of heat shock protein 90 (Hsp90), it will be shown that dcTMD-based Langevin models are able to correctly predict time scales on the order of millisecond (trypsin) to half a minute (Hsp90) within a factor of ten [125]. At this point we have to use T-boosting (Sec. 3.4) to sufficiently accelerate the Langevin simulations.

## 6.1 Study of T4 lysozyme

In this section we will inspect the dynamics of bacteriophage T4 lysozyme (T4L) [151]. Lysozymes are a group of enzymes which can be found in animals, humans, plants, bacteria and viruses. For humans and animals they are part of the innate immune system and serve as defence against bacteria. Other organisms utilize it against bacteria as well, e.g., viruses use it to penetrate bacteria membranes. The first enzyme analyzed by a complete X-ray crystallography was hen egg lysozyme. This system was also the first enzyme for which a detailed mechanism of action was proposed [152]. In the following we will test the performance of the Markov state as well as the Langevin model for T4L.

### 6.1.1 System characteristics

The 164-residue T4L consists of two domains, see Fig. 6.1a, which perform a distinctive hinge-bending motion. This dynamics resembles the opening and closing of the mouth of a "Pac-Man" [153] when imagining that the two domains represent the upper and the lower jaw, respectively.

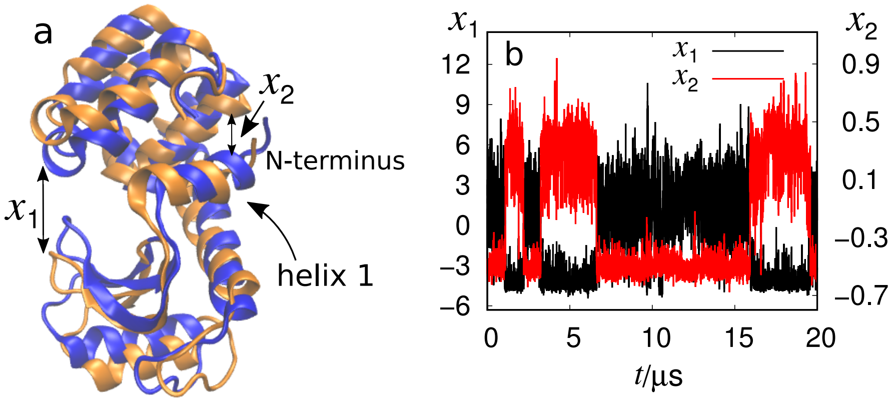

Figure 6.1: **Structure of T4 lysozyme and MD trajectory.** (a) A cartoon representation of the open (orange) and the closed (blue) state of T4L. The coordinate $x_1$ directly records the opening and locking while $x_2$ measures the dynamics of the Phe4 side chain (see text). (b) This panel shows the time evolution of $x_1$ and $x_2$ in fully atomistic MD simulations.

As model input we will use a 61 $\mu$s-long unbiased all-atom MD simulation at 300 K produced by Ernst et al. [71]. Details on the MD setup can be found in Sec. A.4.3. By closely inspecting the MD dynamics, Ernst et al. [71, 150] discovered that the transition between the open and the closed state of T4L is caused by a locking mechanism where the side chain of Phe4 moves from a solvent-exposed to a buried state (Phe4 is highlighted in Fig. 6.5). This observation motivates a two-dimensional system description of T4L. The first coordinate $x_1$ covers the transition between the open and the closed state and was defined by a contact-PCA based on the residues 20, 21, 22, 137, 141 and 142 of the N- and C-terminal domains. The second coordinate $x_2$ records the side chain motion of Phe4. It represents the scalar product of two distance vectors which monitor the side chain dynamics.

The first 20 $\mu$s MD simulation resolved along $x_1$ and $x_2$ can be seen in Fig. 6.1b. We see that transitions between open and closed are only rarely observed, the system remains in one of the two distinct conformations for several $\mu$s. Moreover, the transitions themselves take only a few ns (at least in this system description) which is surprisingly fast. This shows that the transition dynamics of T4L include information on several time scales from very long to rather short which indicates that it will be very challenging to derive a satisfying Markovian model, may it be a Markov state or a Langevin model.

### 6.1.2 Recrossing study

Still, before we consider T4L in terms of the coordinates $x_1$ and $x_2$, we go one step back and connect to earlier dLE studies [149]. Here, the first three principal components (PCs) $y_1$, $y_2$ and $y_3$ of a contact principal component analysis were used as system coordinates [71]. The first PC $y_1$ describes the opening and locking motion of the whole complex. It is dominated by two contacts acting as hinges at the N- and C-terminal domains. The second PC $y_2$ represents a twist-like rearrangement in the N-terminal domain while the third PC $y_3$ describes a rocking motion of helix 1 and the N-terminal domain (in Fig. 6.1 helix 1 and the N-terminus are marked). The free energy projected on $y_1$, $y_2$ and $y_3$ (see Fig. 6.2) reveals five relevant states in the form of energy minima[71, 149]. The states 1 and 2 account for the main open and closed conformation, respectively. The other three minima represent side states resolved along $y_2$ and $y_3$.

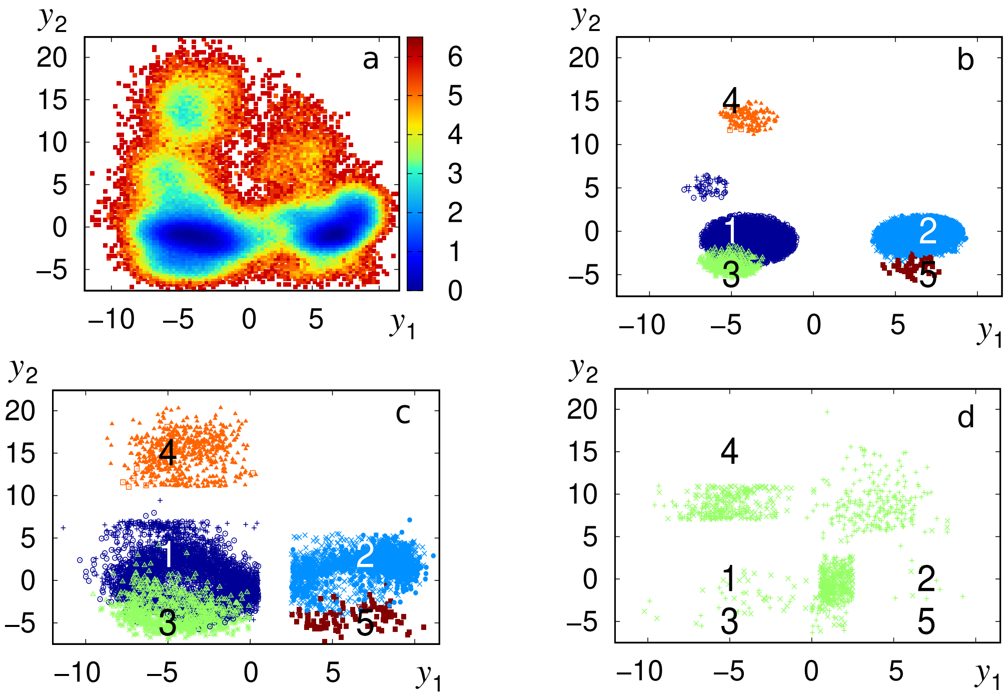

Figure 6.2: **Recrossing study based on contact distance.** (a) Here, we see the free energy projected on the first two PCs $y_1$ and $y_2$. The other three figures are related to the counting of recrossings. In (b), the state cores are shown, in (c) the state surroundings and in (d) the barrier region.

The free energy projection on $y_1$ and $y_2$ as well as the position of the states can be seen in Fig. 6.2a and b. Similar representations for the other projections can be found in Sec. A.10. We found that the dLE needs a time step of $\delta t = 5$ ns to become slow enough to account for the MD dynamics described by the three PCs [149]. This is problematic considering that, just as for the system description $x_1$, $x_2$, the actual transitions observed in MD have a duration of only a few ns, i.e., the dLE is not able to resolve them in detail. This observation motivated the conclusion that $y_1$, $y_2$ and $y_3$ are not suited to cover all essential dynamics of the system.

Now, we will find additional evidences for this assumption. Transition state theory tells that insufficient system descriptions can be detected by counting the number of barrier crossings observed during individual transitions [141, 154]. For ideal reaction coordinates, which allow for a Markovian model of the system dynamics, the barrier is crossed only a single time per transition. Suboptimal coordinates, i.e., coordinates which do not cover all important motions, show oscillations on the barrier in contrast. These oscillations can be explained by hidden coordinates which perform their own nontrivial motions and need to have a specific configuration to allow for successful transitions, i.e., the hidden coordinates give rise to system memory. Hence, if the hidden coordinates do not allow the transition, the trajectory will not reach the minimum of the target state and it will probably jump back over the barrier.

Now, let us see how T4L behaves. To count the barrier crossings we need to define the states and the barrier. State cores are defined by spheres in the three-dimensional space spanned by $y_1$, $y_2$ and $y_3$ as indicated in Fig. 6.2b. The barrier regions are specified by cubes located at the free energy maxima, see Fig. 6.2d. The remaining trajectory points are interpreted as surrounding area of the nearest state core, i.e., neither as state core nor as barrier. A transition is assumed to start once the present state core has been left. Reaching any other state core indicates its end. For the time in between start and end of the individual transition, one counts how often the state surrounding is changed. This allows to map the transition to a sequence like, e.g., 1→2→1→2 which indicates that the barrier was crossed three times.

Applying this procedure to the MD and a dLE with $\delta t = 100$ ps (this time step is also used to evaluate the crossings), revealed significant discrepancies between both data sets. Especially the main transition 1↔2 (representing the transition between open and closed conformation) deviates significantly. While the dLE shows approximately only one barrier crossing per transition, just as expected for Markovian dynamics, the MD reveals roughly 2.5 crossings. This can be seen as proof that the coordinates $y_1$, $y_2$, $y_3$ indeed miss some important dynamics. When inspecting dLE and MD at a resolution of $\delta t = 5$ ns the discrepancies vanish, both data sets cross the barrier only a single time per transition. However, this is not surprising since the details of the MD transitions are no longer resolved anyway.

In summary, counting of barrier crossings underpins the conclusion that the three coordinates $y_1$, $y_2$ and $y_3$ are not well suited to derive a satisfactory Markovian model. Considering that $x_1$ and $x_2$ are directly correlated to the opening and closing of T4L, it is likely that they will provide a more favourable system description.

### 6.1.3 Two-dimensional Markov state model

To evaluate if $x_1$ and $x_2$ allow for the derivation of more meaningful Markovian models, we will directly inspect the results of an MSM constructed for this set of coordinates. Inspecting the free energy resolved along $x_1$ and $x_2$, see Fig. 6.3a, reveals four minima, i.e., four metastable states [71]. State 1 represents the open/buried state while state 2 depicts the attempt of the system to close the mouth even though Phe4 is still buried. In state 3, Phe4 starts to be exposed to the solvent. State 4 is finally the closed/solvent exposed conformation. We note that state 1 and 4 are significantly more populated than the other two states, reflecting that the latter two can be interpreted as transition states. To identify the state borders we can use a density-based clustering [60] of the data. By setting the hypersphere radius to the lumping radius and by using a minimal population of $p_{\text{min}} = 2000$ (see [60] for details on the interpretation of these parameters) the density-based clustering finds a reasonable discretization into the expected four states. We note that the states 3 and 4 are merged for larger $p_{\text{min}}$, which reflects that they are separated by a low barrier. The density-based clustering defines populations of 69 % for state 1, 1.2 % for state 2, 2.3 % for state 3 and 28 % for state 4.

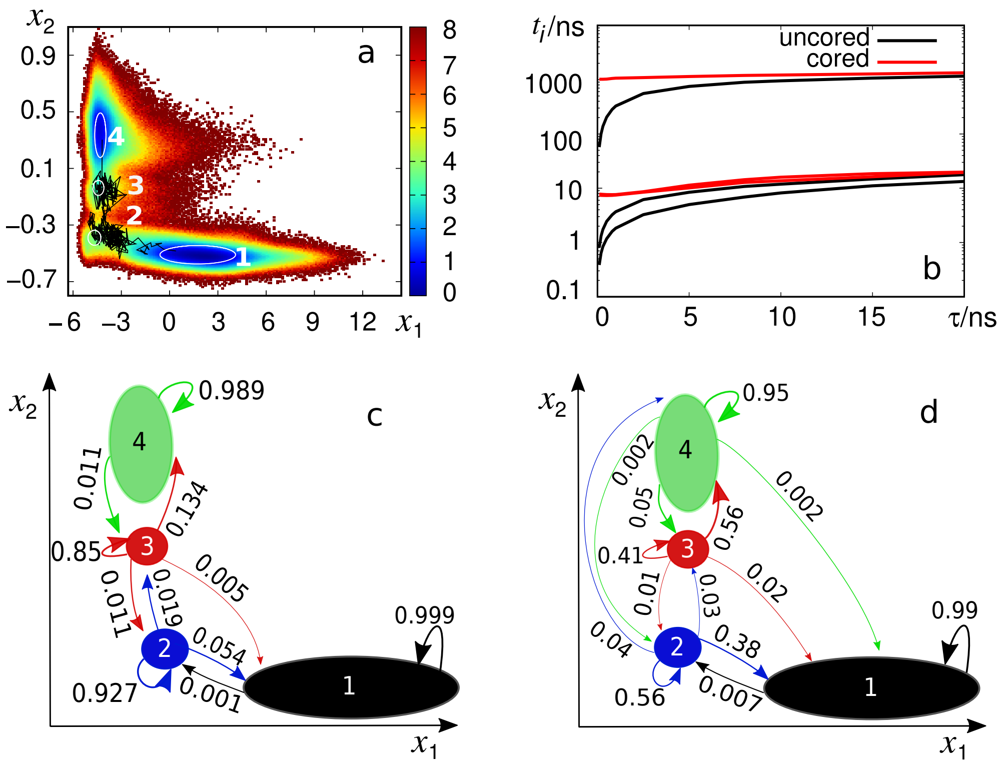

Figure 6.3: **Two-dimensional Markov state model for T4 lysozyme.** (a) The free energy (in units of $k_{\text{B}}T$) reveals four states (white numbers). The black line represents an exemplary 1→4 (open→closed) transition found in the MD simulation. (b) The implied time scales $t_1$, $t_2$ and $t_3$ of an MSM indicate that lag times $\tau \geq 5$ ns are needed to derive a valid MSM for the uncored data (black). After dynamical coring (red) the initial steep rise of the time scales disappears. Note that the two smaller time scales overlap after coring. In the bottom row we see illustrations of the transition matrix of the four-state splitting. The numbers at the arrows (connecting the states) indicate the probabilities to observe the transitions i→j within $\delta t = 10$ ps (c) and $\delta t = 5$ ns (d) when starting in state i.

The black line drawn in Fig. 6.3a represents a typical open → closed transition of the MD with a length of 4.4 ns. We see that it follows the path 1→2→3→4. Counting all transitions between the different states at a resolution of $\delta t = 10$ ps (see the graphical representation Fig. 6.3c) reveals frequent oscillations between states 1 and 2 as well as between states 3 and 4. Besides one direct 1→4 jump, all open → closed pathways include the states 2 and 3. The same holds true in the opposite direction. Unfortunately, the essential transition 2↔3 needed to perform the open ↔ closed transition occurs on a similar time scale as the mostly unproductive oscillations 1↔2 and 3↔4, i.e., failed and successful opening/closing attempts are not dynamically separated. Alternatively, when recalling the recrossings observed for $y_1$, $y_2$ and $y_3$ in Sec. 6.1.2, one might interpret the frequent oscillations 1↔2 and 3↔4 as symptom of similar problems. Either way, our observations indicate that it could be problematic to describe the open↔closed mechanism by Markovian dynamics.

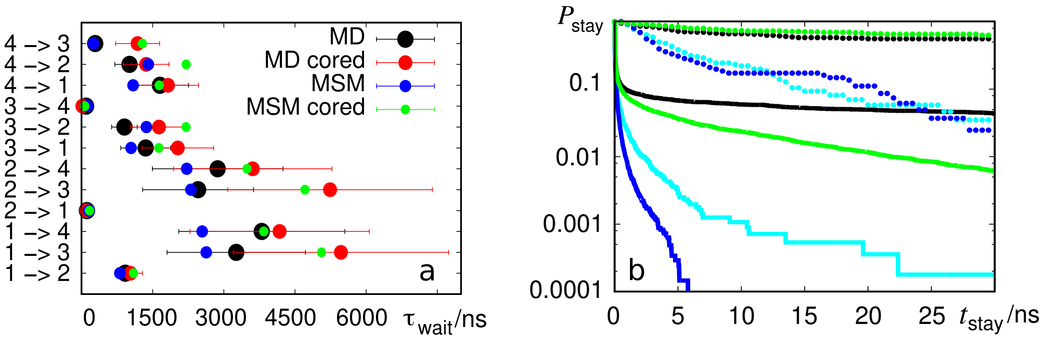

Figure 6.4: **Results of the two-dimensional Markov state model.** (a) Here, we see the average waiting times of MD (black), cored MD (red), the MSM on the uncored MD (blue) and the MSM on the cored data (green). Error bars are calculated as standard deviations of the mean. (b) The population probabilities $P_{\text{stay}}$ before (lines) and after (dots) coring. Black represents state 1, cyan state 2, blue state 3 and green state 4.

Indeed, when calculating the implied time scales of a four-state MSM, we see in Fig. 6.3b that lag times $\tau_{\text{lag}} \geq 5$ ns are necessary to observe approximately constant time scales. This implies that it is not possible to cover the transition events with a duration of the order of nanoseconds by means of the MSM framework. As a consequence, the transition network at $\delta t = 5$ ns shown in Fig. 6.3d, suggests that all four states are well connected. The open↔closed transition is dominated by direct 1↔4 jumps, i.e., the importance of the intermediate states 2 and 3 is hidden. Hence, while the MSM with $\tau_{\text{lag}}$ produces satisfactory average waiting times (see Fig. 6.4a) its interpretation of the T4L dynamics is misleading.

Considering that density-based clustering separates the states on top of the barrier, it is reasonable to suspect that wrongly assigned points on the barrier might disguise the true transition dynamics. Intrastate dynamics are maybe mistaken as interstate transitions, which leads to fast initial decays of the probability $P_{\text{stay},n}(t)$ to stay in state $n$ for at least the time $t$ (see Sec. 3.1). When calculating $P_{\text{stay},n}(t)$ for T4L (see Fig. 6.4b) we indeed observe such fast initial decays. Dynamical coring (see Sec. 2.6) can be used to remove those artifacts [65, 66]. To remove the initial decays of $P_{\text{stay},n}(t)$, we need coring times $\tau_{\text{cor},1} = \tau_{\text{cor},4} = 200$ ps for the states 1 and 4, $\tau_{\text{cor},2} = 500$ ps for state 2, and $\tau_{\text{cor},3} = 800$ ps for state 3, see Fig. 6.4b. When calculating the implied time scales based on the cored data, see Fig. 6.3b, we see that the initial steep rise was removed. The average waiting times of the most important transition 1↔4 found in MD is hardly influenced by the coring, see Fig. 6.4a, which indicates that coring does not affect the most important part of the dynamics. Still, the average waiting times of the transitions 1↔3, 4→3 and 2↔3 are significantly altered. Note that all those transitions include state 3. When counting all transitions observed in the cored data at $\delta t = 10$ ps, it turns out that state 3 only jumps to state 4, while all other states are interconnected. Even when counting the transitions at a resolution of $\delta t = 5$ ns, we do not observe a single 3→2 transition, and the 2→3 direction has the lowest count of all transitions. Hence, dynamical coring does not only remove short-living oscillations on top of the barrier but deletes the whole 1↔2↔3↔4 pathway. This shows that it is indeed not possible to combine this pathway with the fast oscillations 1↔2 and 3↔4 in the same MSM.

Still, when constructing an MSM with $\tau_{\text{lag}} = 5$ ns based on the cored data, the resulting average waiting times (see Fig. 6.4a) coincide with the MD data somewhat better than before coring. We can conclude that coring indeed improves the state definition from perspective of the MSM but removes at the same time important dynamical information. After all, while it is possible to reproduce the 1↔4 time scale by an MSM, the system description of T4L needs to be improved if we want a model which accounts for details of the open↔closed transition, i.e., the pathways.

### 6.1.4 Four-dimensional Markov state model

To enhance the system description of T4L, it can help to add additional coordinates so that newly resolved side states make the pathways more detailed and by this (hopefully) more suitable to be modeled by an MSM with a small lag time. Considering T4L, many different coordinate candidates were tested, but the constructed Markov state models did not allow to use of smaller lag time $\tau$ to cover the open↔closed pathways. To investigate the persistent problems of all tested system descriptions, we will now inspect one exemplary extended coordinate set.

Here, two additional coordinates are added. The first degree of freedom $x_3$ represents the distance between the carboxylate carbon atom of Glu5 and the ammonium nitrogen atom of Lys60, see Fig. 6.5a. These two atoms can form a salt bridge. $x_3$ might be interesting because the MD data showed that the salt bridge seals Phe4 in its buried cavity, i.e, $x_3$ needs to extend before Phe4 is able to switch its conformation to trigger the open→closed transition. In consequence, as can be seen in Fig. 6.6a, $x_3$ splits state 1 into three substates. The narrow state at $x_3 \approx 0.3$ represents the most stable open conformation where $x_3$ is so small that Phe4 is basically trapped. The attached, slightly broader, minimum at $x_3 \approx 0.55$ refers to the first step of the extension of $x_3$ where the salt bridge becomes unstable but Phe4 is still not able to switch. Only at the third substate centered at $x \approx 0.8$ (where the salt bridge is broken) is directly connected to the closed system configuration. This explains the broad arc that is spanned from the open to the closed state. However, although $x_3$ adds more details to the closing process, it needs to be noted that the transitions along $x_3$ is relatively fast (on the order of nanoseconds) so that we do not add an additional slow time scale to the system description, i.e., the open↔closed pathways are still fast.

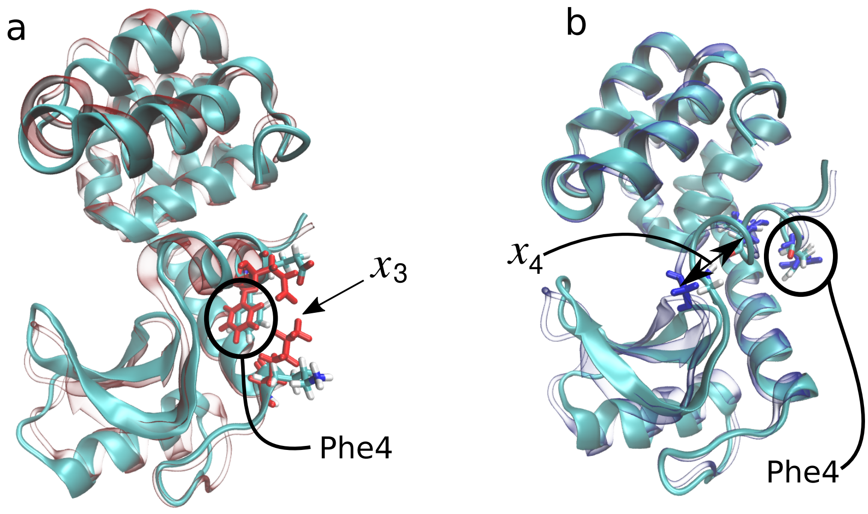

Figure 6.5: **Additional coordinates for T4 lysozyme.** (a) We see an overlay of the cartoon representations of the states 1A (cyan) and 1B (red-white). The coordinate $x_3$ is indicated by a black arrow, Phe4 is highlighted by a circle and Glu5 as well as Lys60 are shown as red sticks. (b) This panel shows an overlay of the cartoon representations of the states 4A (cyan) and 4B (blue-white). Again, Phe4 is highlighted and the coordinate $x_4$ is indicated. Leu7 and Gly12 are shown as blue sticks.

The second additional coordinate $x_4$ represents the distance between the backbone oxygen of Leu7 and the backbone nitrogen of Gly12, see Fig. 6.5b. Being the counterpart of $x_3$, it splits state 4 into substates, see Fig. 6.6b. When inspecting the MD trajectory in detail, it turns out that large $x_4$ indicate a stable closed conformationwhile small $x_4$ refer to a more unstable one.

A density-based clustering in the extended four-dimensional system space and subsequent manual state lumping leads to a simple and reasonable state separation: while the states 2 and 3 remain approximately unchanged, both large states 1 and 4 are split into two substates. Called 1A, 1B and 4A, 4B, the respective state populations are evenly shared. State 1A contains a population of 34 %, 1B collects 35 % of all data points while state 4A and 4B show populations of 14 % each. The states 1A and 4A can be interpreted as unstable system conformations while 1B and 4B describe stable states. An illustration of the overall state separation can be seen in Fig. 6.6c. Although this extended state space looks promising, the resulting MSM modeling does not reveal improvements compared to the MSM based on $x_1$ and $x_2$ alone. The implied time scales (see 6.6e) show again a steep initial rise which needs (for the larger time scales) approximately 5 ns to reach a plateau, i.e., the required lag time is not reduced. When inspecting the probabilities $P_{\text{stay},n}(t)$ to stay in state $n$ for at least the time $t$, on the other hand, we see that dynamical coring is needed to improve the state definition (see

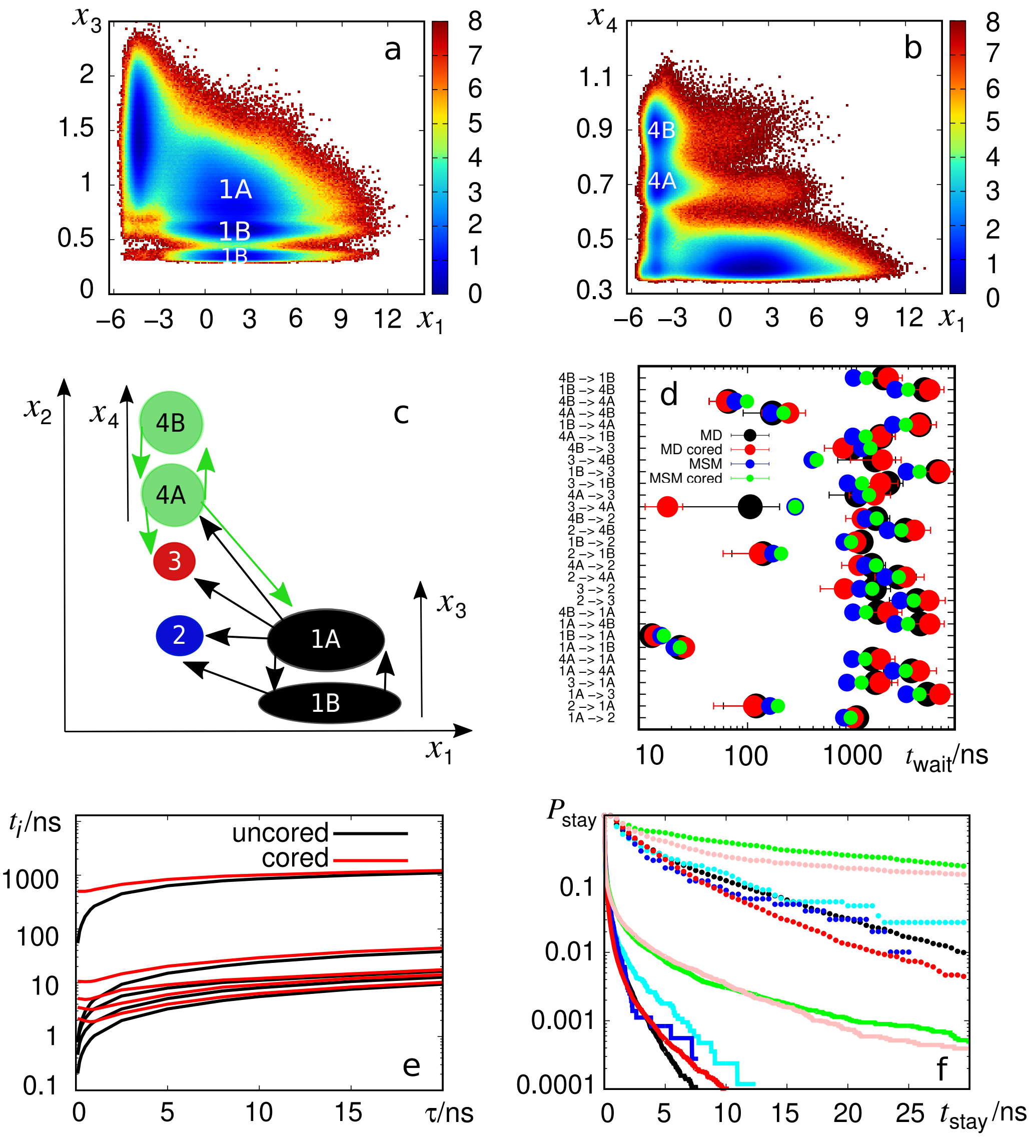

Figure 6.6: **Four-dimensional Markov state model for T4 lysozyme.** The free energy (in units of $k_\mathrm{B}T$) projected on $x_1$, $x_3$ (a) and projected on $x_1$, $x_4$ (b) reveals that state 1 and 4 are split (white numbers) when including the coordinates $x_3$ and $x_4$. Please note that 1B includes two minima due to the manual state lumping used to simplify the state separation. (c) Here, we see an illustration of the overall state splitting. (d) The average waiting times of MD (black), cored MD (red), the MSM on the uncored MD (blue) and the MSM on the cored data (green). Error bars are calculated as standard deviations of the mean. (e) The implied time scales $t_1$, $t_2$ and $t_3$ of an MSM, indicate again that lag times $\tau \geq 5$ ns are needed to derive a valid MSM for the uncored data (black). After dynamical coring (red) the initial steep rise of the time scales disappears. (f) Here, we see the population probabilities $P_\mathrm{stay}$ before (lines) and after (dots) coring. The different colors represent the six states.

6.6f). Still, we need again coring times on the order of 100 to 500 ps to remove the fast initial decay which is, again, similar to the observations for the two-dimensional system description. Consequently, it is possible to use an MSM with $\tau = 5$ ns to predict approximately correct average waiting times, see Fig. 6.6d, but the transition pathways itself are, again, not resolved.

Hence, $x_3$ and $x_4$ do not improve the capabilities of our T4L models. While there are numerous other coordinates which look just as promising as the two coordinates $x_3$ and $x_4$, this problem is very persistent. It is reasonable to speculate that this phenomenon is caused by a missing time scale separation between relevant and irrelevant system dynamics which makes the dimensionality reduction very complicated. Just because some motion appears to encode relevant side states of the open or the closed state, it is not clear that this motion is causally related to the open→closed transition. Additionally, it might be possible that system descriptions are needed which describe T4L more abstract than the simple distances which were used in this chapter. Maybe coordinates are needed which quantify, e.g., the disorder of or the structural stress at important regions of T4L. Still, these considerations are mere speculation at this point, future studies are needed to inspect their plausibility.

### 6.1.5 Two-dimensional dLE modeling

Having seen that the capabilities of MSMs to cover the dynamics of T4L are limited, we will now inspect the performance of the dLE at this point. Since the additional coordinates $x_3$ and $x_4$ did not improve the modeling situation significantly, we go back to the two-dimensional system description. As written in Sec. 6.1.2, it is known that $\delta t = 5$ ns is needed for sufficiently slow model dynamics when using the first three PCs of a contact-PCA [149]. Based on the MSM results presented above it is very likely that we obtain the same result for the coordinates $x_1$ and $x_2$ but we can fall back on the rescaled dLE to get at least an approximately consistent Langevin model at small $\delta t$.

But first we consider the normal dLE procedure, i.e., we adjust the time step $\delta t$ such that $\delta t_{\mathrm{M}} < \delta t < \delta t_{\mathrm{R}}$. The noise estimated by the dLE model for different $\delta t$, see Fig. 6.7c, reveals that already $\delta t = 10$ ps appears to be sufficient to observe Markovian noise. The free energy estimated at this time step (Fig. 6.7a) accurately reproduces the MD but the dLE dynamics are too fast as can be seen by inspecting the position autocorrelations in Fig. 6.8. This confirms that the two-dimensional system description apparently misses important dynamics of T4L, i.e., the conclusions from the MSM modeling are confirmed. Still, it is worth noting that the dLE successfully reproduces the velocity autocorrelation of the data at $\delta t = 10$ ps, i.e., good agreement for the velocity cannot be transferred to good accordance in the coordinate itself. Successively increasing $\delta t$ confirms the suspicion that, just as for the coordinate description based on contact-PCA or for the MSM above, a time step of $\delta t = 5$ ns is needed to get sufficiently slow dLE dynamics, see the autocorrelations in Fig. 6.8. Unfortunately, this $\delta t$ it too large to allow for the reproduction of the free energy, see Fig. 6.7b. It appears as if the dLE only observes two-state dynamics between the states 1 and 4 and overlooks the two intermediate states. When remembering that the 1↔4 pathways are typically shorter than 5 ns, this effect is perfectly reasonable.

To circumvent the problems of the dLE, we can use the rescaled dLE to construct a reasonable dLE model for $\delta t = 10$ ps. Using the initial decay of the autocorrelations along the two coordinates (Fig. 6.8c and d), a rescaling factor of $S = 5 \cdot \mathbb{1}$ was deduced. Similar to NaCl and AIB$_9$, the full decay of the autocorrelations is covered although we only use the first 100 ns to determine $S$. The free energy landscape of the rescaled dLE, Fig. 6.9a, reproduces the MD, just as expected at $\delta t = 10$ ps. When considering the

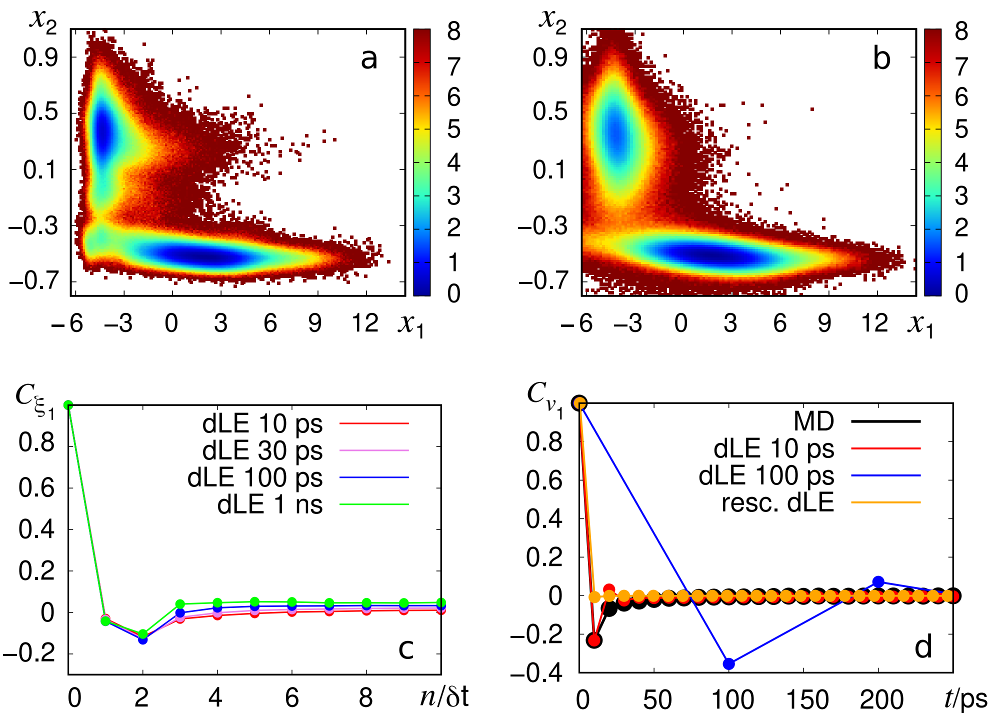

Figure 6.7: **Two-dimensional dLE modeling for T4 lysozyme.** In the top row we see the free energies of dLEs at $\delta t = 10$ ps (a) and $\delta t = 5$ ns (b) in units of $k_\mathrm{B}T$. (c) Here, the autocorrelation of the reconstructed noise at different time steps is shown. (d) The velocity autocorrelations of MD and dLEs (considering $x_1$).

average waiting times of the rescaled dLE trajectory (see Fig. 6.9b), we see that the transitions 1↔4 are accurate which makes sense considering that those dynamics, being the slowest transitions, dominate the autocorrelations we use to choose $S$. The other waiting times, in contrast, are only qualitatively covered. The transitions 1↔2 and 4↔3, for example, are estimated too slow. This observation can be understood by comparing transitions of MD and rescaled dLE, see Fig. 6.3b and Fig. 6.9a. While the MD shows numerous short-living 1↔2 and 3↔4 jumps before the transition is finished, the rescaled dLE evolves less dynamical (which makes sense considering that we increased the friction) and performs the transition more directly, i.e., the 1↔2 and 3↔4 oscillations are suppressed. This shows that we can interpret the oscillations 1↔2 and 3↔4 as recrossings which cannot be covered by Markovian models, i.e., there are most likely hidden degrees of freedom. In consequence, the rescaled dLE has to sacrifice some short-living dynamics (like the velocity autocorrelation shown in Fig. 6.7d) to cover the longer time scales. When considering the estimated Langevin fields $\Gamma$ and $\mathcal{K}$, we observe that the diagonal components $\Gamma_{nn}$ and $\mathcal{K}_{nn}$ mainly depend on the coordinate $x_n$ while the dependence on the other coordinate turns out to be only minor (see Sec. A.12). Additionally, the off-diagonal elements are comparatively small, i.e., it appears to be sufficient to approximate $\Gamma$ and $\mathcal{K}$ by diagonal matrices with $\Gamma_{11}(x_1)$, $\Gamma_{22}(x_2)$ and $\mathcal{K}_{11}(x_1)$, $\mathcal{K}_{22}(x_2)$, respectively. When simulating Langevin dynamics based on such a simplified model, it turns that it is even possible to drop the coordinate dependence of both fields completely by replacing the matrix elements by averaged values. As can be seen in Fig. 6.9b, the resulting Langevin dynamics produce very similar average waiting times. While especially 1→4 is better with the untouched rescaled dLE estimates, the simplified model is still qualitatively correct even though the field estimates vary by a factor of 10 or more. Thus, it is not necessary to have a very detailed model of friction and noise since the results of the Langevin model are relatively unaffected by this modeling aspect.

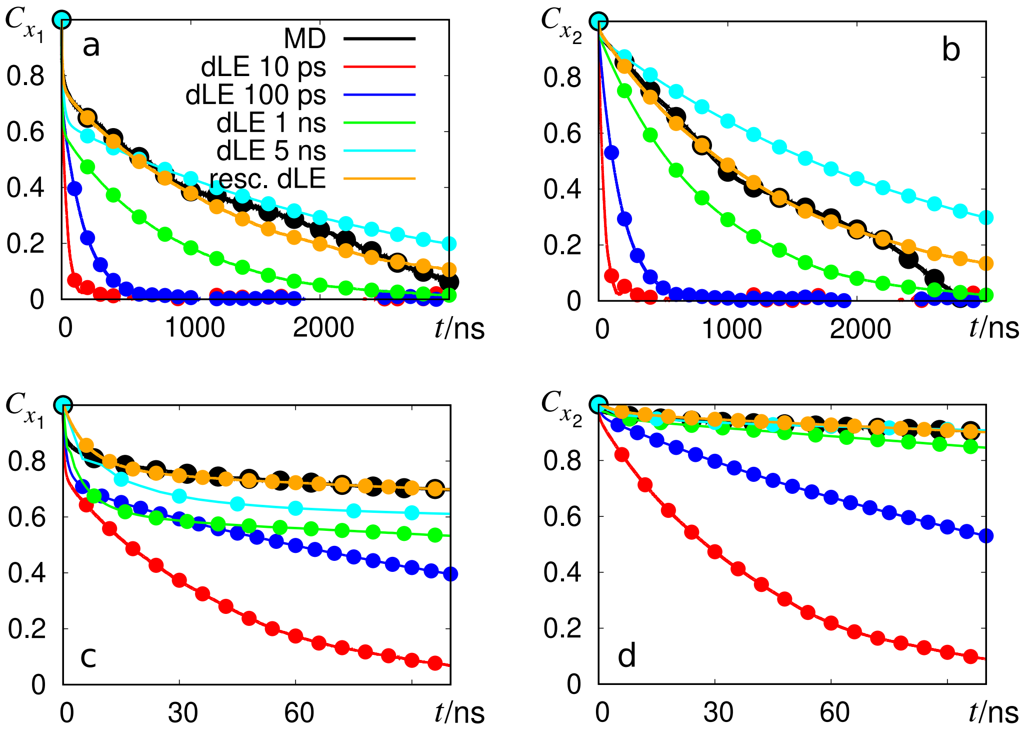

Figure 6.8: **Autocorrelations of MD and dLEs for T4 lysozyme.** The panels (a,c) show the $x_1$ autocorrelations of MD (black), normal dLEs at different time steps (red, blue, green, cyan) and the rescaled dLE at $\delta t = 10$ ps with $S^2 = 25$. The panels (b,d) show the same for $x_2$. In the top row, we can see the full decay of the autocorrelations, the rescaled dLE and the normal dLE at $\delta t = 5$ ns follow the decay at least qualitatively. The bottom row shows that it is sufficient to consider the first 100 ns of the autocorrelations to calibrate the rescaling matrix $S$.

For completeness we can investigate the influence of the additional coordinates used for MSMs in Sec. 6.1.4 on the capabilities of the dLE. Similar to the MSM framework, it can be observed that the resulting Langevin models do not benefit from the additional degrees of freedom. It is not possible to use $\delta t < 5$ ns for sufficiently slow (unrescaled) dLE trajectories, i.e., the friction needs to be rescaled again if we want to get the long time scales at small $\delta t$.

In summary, we can conclude that it is possible to construct a (rescaled) dLE model which is able to cover the long-time dynamics of T4L represented by $x_1$ and $x_2$. Additional coordinates do not improve the dLE performance. While short time scales need to be sacrificed, it is possible to drastically simplify friction and noise without losing the qualitative accuracy of the Langevin model. This shows that surprisingly simple Langevin models are able to cover the most important dynamics of T4L.

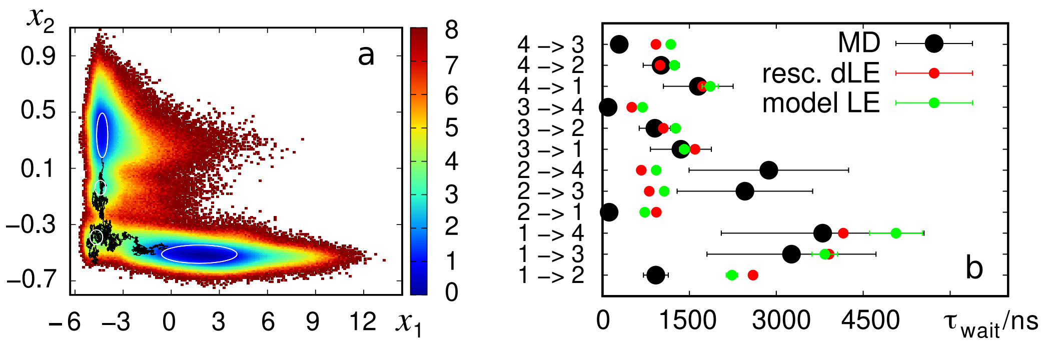

Figure 6.9: **Results of the rescaled dLE for T4 lysozyme.** (a) The free energy (in units of $k_\mathrm{B}T$) estimated by the rescaled dLE together with an exemplary 1→4 pathway. (b) The average waiting times of the rescaled dLE are compared to MD and a simplified Markovian Langevin model with constant and diagonal friction and noise (see text).

## 6.2 Langevin modeling of multisecond dynamics

As explained in Sec. 4.3, dissipation-corrected targeted MD (dcTMD) allows for the parameterization of a one-dimensional Langevin model by enforcing the transition of interest along $x$ via the constraint force $f_c$. Together with T-boosting, see Sec. 3.4, this makes it possible to access time scales of the order of tens of seconds. In the following, we will have a look at the results of the modeling of the dynamics of two protein-ligand complexes, trypsin-benzamidine and the N-terminal domain of a heat shock protein 90 inhibitor complex [125].

The unbinding of the inhibitor benzamidine from trypsin [155–157] is frequently used to test enhanced sampling techniques [139, 143, 158–161] due to its slow unbinding dynamics occurring on time scales of milliseconds [155]. The other considered process, the unbinding of a resorcinol scaffold-based inhibitor (**1j** in [162]) from the N-terminal domain of heat shock protein 90 (Hsp90) even shows time scales of half a minute. It has recently been established to investigate molecular effects influencing binding kinetics [162–165]. To parameterize Markovian Langevin models for both systems, TMD simulations were performed, the data set of trypsin consists of 200 trajectories, Hsp90 was sampled by 400 runs. At the start of each simulation, peptide and ligand were prepared in the bound state, the subsequent TMD run enforced the dissociation by increasing the distance $x$ between benzamidine/inhibitor and the binding site. The details of the MD setups for both systems can be found in Sec. A.4.4.

Using nonequilibrium principal component analysis [138], the dominant transition pathway of trypsin was determined. This pathway is used by 84 trajectories in total. For Hsp90, on the other hand, a path separation based on geometric distances between individual trajectories was performed [166], see also the Supplementary information of [125] for more details. Here, the dominant pathway is used by 93 trajectories. Employing the dcTMD equations from Sec. 4.3, free energy and friction estimates where calculated using these two trajectory sets. As can be seen in Fig. 6.10a, both systems show very similar free energy profiles. The bound state at $x = 0$ is followed by a single high barrier which peaks at $x \approx 0.46$ nm (trypsin) and $x \approx 0.5$ nm (Hsp90), respectively. Having passed this barrier, the dissociated state at $x \geq 0.75$ nm is reached. Interestingly, the single free energy barrier can be associated with very distinct dynamics for both systems. In case of trypsin, it reflects in line with [139] the breaking of the Asp189-benzamidine salt bridge representing the most important contact of the bound ligand. In case of Hsp90, on the other hand, the ligand is pushed between two helices to escape the binding site.

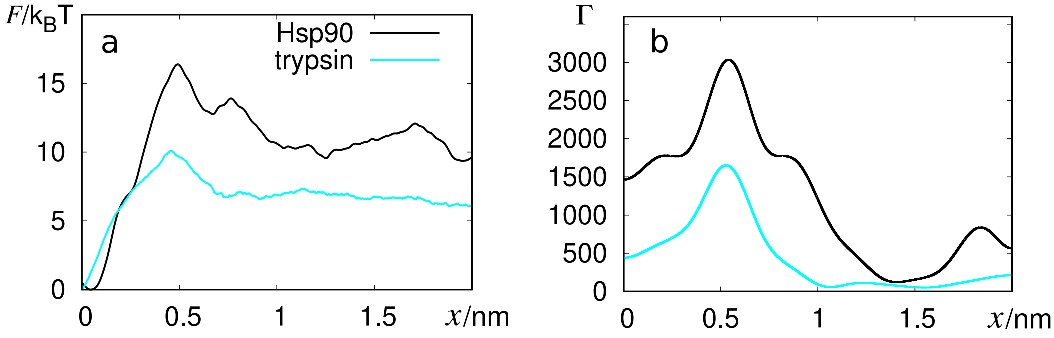

Figure 6.10: **dcTMD-based Langevin fields for trypsin and Hsp90.** (a) Here, we see the free energy estimate $F(x)$ of trypsin (cyan) and Hsp90 (black). (b) This panel shows the friction $\Gamma$ estimated by dcTMD.

When considering the two friction profiles (see Fig. 6.10b), we find the maxima directly behind the peaks of the free energy, i.e., both systems are again very similar. The local increase in friction can be related to rearrangements of the protein-ligand hydration shells [125].

To complete the Langevin model we need to choose suitable masses $\mathcal{M}$ for both systems since dcTMD does not provide any direct estimates at this point. Fortunately, the two friction profiles indicate overdamped Langevin dynamics at the main barrier which would mean that the mass does not influence the transition dynamics since it does not show up in the overdamped Langevin equation (3.30). Langevin calculations using the reduced masses of trypsin-benzamidine ($\mathcal{M} = 120.15$ g/mol) and Hsp90 ($\mathcal{M} = 288.73$ g/mol) as well as ten times larger masses show indeed that this suspicion is true (see Supplementary information of [125]). Hence, we can safely use the reduced masses for our Markovian Langevin model. Please note that we stick to the Markovian Langevin equation and do not switch to the (faster) integration of the overdamped version (3.30). Using the overdamped equation, we observed artifacts in the simulated free energy at larger $x$ where $\Gamma$ becomes small (for example at $x = 1$ nm for trypsin) and the system behaves no longer overdamped. Hence, we concluded that it is safer to use the non-overdamped Langevin equation although the transition statistics might be unaffected by these artifacts.

To estimate the binding and unbinding times, Langevin simulations using the OVRVO integrator from Sec. 3.3.2 were performed. Since the times should be of the order of milliseconds (trypsin) or even tens of seconds (Hsp90), it is necessary to use T-boosting (see Sec. 3.4) in order to collect enough transition statistics. Thus, 10 ms-long Langevin trajectories were produced for trypsin covering 13 temperatures ranging form 380 K to 900 K. For Hsp90, the simulations have a length of 5 ms and span a temperature range of 700 K to 1350 K. In both cases at least $10^2$ binding/unbinding events are recorded at the different temperatures. To account for the fact that the free energies are only defined for $x \in [0, 2]$ nm, fully reflective boundary conditions were implemented, i.e., once the trajectory jumped over one of the two borders $x_{\text{min}}$, $x_{\text{max}}$ by the distance $a$ it was set back to $x = x_{\text{max}} - a$, $x = x_{\text{min}} + a$ and the velocity was multiplied by $-1$.

To calculate the waiting times of interest, the bound state was set to $x < 0.3$ nm for both systems while the unbound state was defined by $x > 0.6$ nm (trypsin) and $x > 0.9$ nm (Hsp90), respectively. As can be seen in Fig. 6.11, the evolution of the average waiting times perfectly coincides with the expected Kramers relation (3.43) so that we can estimate binding/unbinding rates of $k_{\text{bind}} = 8.7 \cdot 10^6$ s$^{-1}$M$^{-1}$, $k_{\text{unbind}} = 2.7 \cdot 10^2$ s$^{-1}$ (trypsin) and $k_{\text{bind}} = 9.0 \cdot 10^4$ s$^{-1}$M$^{-1}$, $k_{\text{unbind}} = 1.6 \cdot 10^{-3}$ s$^{-1}$ (Hsp90) at the target temperature $T = 290.15$ K. Please note the native units s$^{-1}$M$^{-1}$ of the binding rates, it is assumed that both protein-ligand systems have a molarity of 50 mM (see the Supplementary Information of [125]).

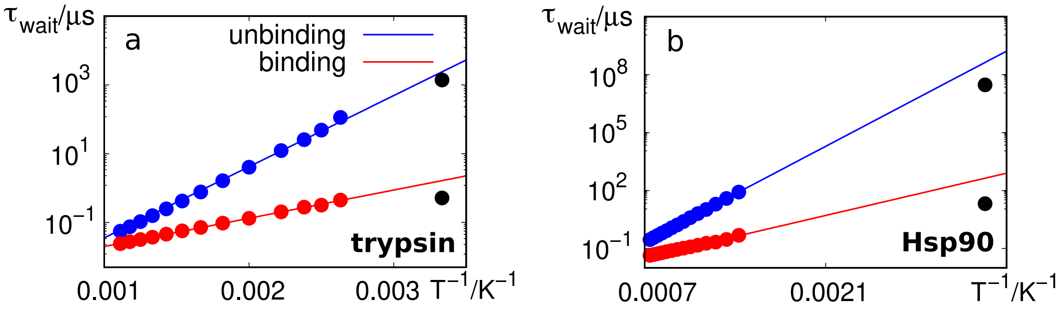

Figure 6.11: **Langevin predictions of binding and unbinding times.** Average binding (red) and unbinding (blue) times calculated from T-boosted Langevin simulations. Panel (a) presents trypsin and panel (b) shows Hsp90. The black dots represent experimental values taken from [155] (trypsin) and [163] (Hsp90).

When comparing these rates to experimental values, $k_{\text{bind}} = 2.9 \cdot 10^7$ s$^{-1}$M$^{-1}$, $k_{\text{unbind}} = 6.0 \cdot 10^2$ s$^{-1}$ [155] for trypsin and $k_{\text{bind}} = 4.8 \cdot 10^5$ s$^{-1}$M$^{-1}$, $k_{\text{unbind}} = 3.4 \cdot 10^{-2}$ s$^{-1}$

[163] for Hsp90, it turns out that we underestimate the true rates by a factor of $2-3$ for trypsin while the Hsp90 predictions deviate by factors of 5 (binding) and 20 (unbinding). Considering the fact that the errors caused by the T-boosting approach are marginal (see Sec. A.5) the deviations can be explained by limitations of the Langevin framework as well as the approximate calculation of free energy and friction by dcTMD. It is known, for example, that constraints lead in general to an overestimation of the friction [143]. The larger error of Hsp90 compared to trypsin is likely related to issues with the sampling of the correct unbinding pathway. Considering that the unbinding time scale of tens of seconds coincides with the slow conformational dynamics of host proteins [163], it can be assumed that the conformational space of the Hsp90 dynamics is significantly larger than the space of trypsin so that it would need a better sampling by significantly more TMD simulations to yield better predictions of free energy and friction. Still, it is worth noting that the deviations of our dcTMD-based Langevin model compete quite well with other computational methods, see [167] for a collection of modeling results for trypsin. Considering the Kramers relation Eq. 3.43, we see that an error of the rates on the order of ten can be related to an error of the energy barrier on the order of only $2.3k_{\mathrm{B}}T$ (due to $e^{2.3} \approx 10$) which is not much for the extremely long time scales of trypsin and Hsp90.

## 6.3 Summary

Considering the dynamics of T4 lysozyme, we have seen in Sec. 6.1.2 that coordinates based on a contact principal component analysis reveal barrier recrossings for the open↔closed transition. Since recrossings impede the Markovian system description, this observation explains why earlier Langevin studies did not provide accurate models at small time resolutions. Going further, we inspected in Sec. 6.1.3 the capabilities of an MSM to describe the dynamics for a two-dimensional system description derived earlier [71, 150]. While the MSM could reproduce the time scale of the open↔closed transition, it was not possible to account for the (relatively short) transition pathways since the lag time was too large to resolve them. Adding additional coordinates in Sec. 6.1.4 did not improve the capabilities of the Markov state model. Hence, Sec. 6.1.5 inspected the performance of the dLE framework only for the two-dimensional system description. The rescaled dLE was able to provide a Markovian Langevin model which reproduced the long time scales based on a small time step. Although we were able to simulate Langevin trajectories which resemble the MD simulation in the sense that they follow the free energy, we nevertheless had to sacrifice accuracy at small time scales.

Inspecting the unbinding of benzamidine from trypsin and the unbinding of a resorcinol scaffold-based inhibitor from the N-terminal domain of heat shock protein 90 (Hsp90) in Sec. 6.2, we observed that dcTMD-based Langevin models correctly predicted dynamics on the order of millisecond (trypsin) to half a minute (Hsp90) within a factor of ten via the T-boosting approach presented in Sec. 3.4. Compared to other computational methods which aim for such slow dynamics [167] our deviations are very competitive.

# 7 Langevin modeling of nonequilibrium dynamics

*Prof. Farnsworth: "Nothing is impossible! Not if you believe in it. That's what being a scientist is all about!"*
*Cubert Farnsworth: "No, that's what being a magical elf is all about!"*
– Professor Farnsworth, Cubert Farnsworth, "Futurama", season 2, episode 15

Up to now we only considered the modeling of equilibrium dynamics although many biomolecular processes belong to the nonequilibrium regime. The following studies will inspect the capabilities of the data-driven Langevin framework to model such dynamics. This approach can be motivated by the observation that standard Langevin theory can be generalized in many different ways [32, 40, 74, 75, 168–171]. For example, it is possible to derive nonstationary versions of the generalized Langevin equation based on projection operator techniques [32, 74, 75, 170]. Or we can deduce the equations of motions for low-dimensional collective coordinates based on the microscopic Hamiltonian of the full system [172] and consider nonequilibrium relaxation processes in terms of nonstationary initial conditions [171]. Before one can test the applicability of the data-driven Langevin approach to nonequilibrium dynamics, it is mandatory to discuss the considered processes since the expected modifications of the Langevin model depend on the characteristics of the inspected nonequilibrium scenarios. Here, we will consider two different situations. First, an externally driven system is considered, i.e., a system which is influenced by the time-dependent external force $\boldsymbol{f}_{\text{ext}}(t)$. We can relate this setup to atomic force microscopy experiments. It will discussed in the following which modifications of the Langevin framework are needed to account for these dynamics. To test our assumption we will inspect the enforced dissociation of sodium chloride.
Additionally, we will inspect relaxation processes, i.e., dynamics which can be associated with, e.g., photoexcitation. Here, the system generally starts in some nonstationary distribution $\rho(t_0)$ and relaxes into its equilibrium distribution $\rho_{\text{eq}}$. Again, we will discuss the expected modifications of the Markovian Langevin equations. The main observation will be that the finite MD simulation time $t_{\text{max}}$ is expected to modify the model results. By considering a hierarchical model energy, we will inspect the practical implications of this finding, i.e., we will inspect the convergence behavior of the dLE for limited data. Afterwards, we consider as exemplary relaxation process the crystal nucleation of a compressed liquid of hard spheres to verify our assumption that the dLE is indeed able to reproduce such dynamics.

## 7.1 Modeling of external driving

Regarding the concept of external driving discussed in the following, we might think of some situation as depicted in Fig. 7.1. Here, the binding/unbinding dynamics of NaCl is described by the interionic distance $x$, just as known from Sec. 5.1. In contrast to the equilibrium studies above, we now add the additional force $f_{\text{ext}}(t)$ to enforce the dissociation of the two ions, i.e., the jump out of the deep state at $x \approx 0.26$ nm is assisted by $f_{\text{ext}}(t)$. When constructing a Langevin model for such a process we have to understand how $f_{\text{ext}}(t)$ might influence the different Langevin fields.

First, it is obvious that $f_{\text{ext}}(t)$ needs to be added to the deterministic drift $f(x) \propto -dF(x)/dx$. On the level of all-atom MD simulations, where external forces can simply be added to the internal forces, this is enough to take the external driving into account but for the reduced dynamics of the Langevin equation it a priori not clear if the other two fields, friction and noise, remain unchanged. In fact, several studies [74, 75, 173] showed that it is possible that $\Gamma$ and $\mathcal{K}$ change significantly if external forces are applied. However, in the linear response regime where, following Onsager's regression hypothesis [118], nonequilibrium perturbations obey the same laws as equilibrium oscillations, $\Gamma$ and $\mathcal{K}$ stay unaffected.

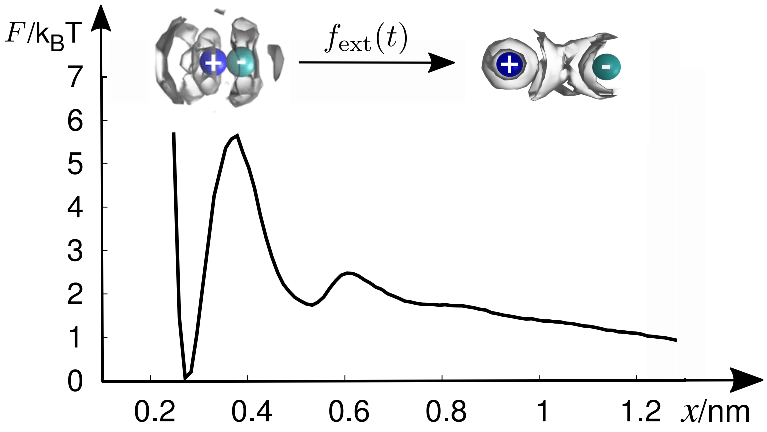

Figure 7.1: **External driving** When applying the external pulling force $f_{\text{ext}}(t)$ along the interionic distance $x$, it is possible to enforce the dissociation of solvated sodium chloride.

In the next section we will inspect if it is realistic to assume linear response for molecular systems like NaCl. This means that we will check if a given Langevin model (derived from equilibrium data) only needs to be modified by replacing the (equilibrium) free energy $F(x)$ by a biased energy landscape $\mathcal{F}(x,t)$ with

$$\mathcal{F}(x,t) = F(x) + V_{\text{ext}}(x,t). \tag{7.1}$$

Here, $V_{\text{ext}}(x,t)$ represents the external potential driving the system.

For completeness it should be noted that it is in principle also possible to go beyond linear response by constructing a dLE directly from data of the driven system. At this point we have to allow for explicitly time-dependent Langevin fields which means that the next-neighbor average Eq. (4.10) needs to be replaced by the explicitly time-dependent average $\langle B(x(t_n), t_n)\rangle$ with

$$B(y(t_n), t_n) = \langle B(x(t_n), t_n)\rangle = \frac{1}{N_{\text{traj}}} \sum_{i=1}^{N_{\text{traj}}} B(x^{(i)}(t_n), t_n)\delta(y(t_n) - x^{(i)}(t_n)). \tag{7.2}$$

Here it is assumed that the considered nonequilibrium process is sampled by $N_{\mathrm{traj}}$ individual trajectories $x^i(t_n)$. The function $\delta(y(t_n) - x^{(i)}(t_n))$ represents a boxing function which equals $1/k$ for the $k$ next neighbors of dLE point $y(t_n)$ and is zero otherwise.

### 7.1.1 Enforced dissociation of sodium chloride in water

We will now test if it is possible to describe the enforced dissociation of NaCl with a Langevin model which assumes linear response. In analogy to atomistic force microscopy experiments, the biasing force (see Fig. 7.1) is defined as harmonic potential [174–176]

$$V_{\mathrm{ext}}(x,t) = -\frac{C}{2}[x(t) - (x_0 + vt)^2] \tag{7.3}$$

with $C$ being the spring constant, $x_0$ the initial position of the spring and $v$ the pulling velocity. In case of a sufficiently small $v$ we expect that the enforced dynamics of NaCl take place close to equilibrium so that linear response can be assumed.

For studies of the Langevin modeling based on the dissipation-corrected targeted MD framework [42, 126] it was found that $v \approx 10$ m/s represents a reasonable upper bound for linear response behavior. Here, the pulling approaches the picosecond time scale of the solvation shell dynamics [42]. Due to this finding, our reference restrained MD simulations employ $v = 10$ m/s as well. The biasing force is exactly given by Eq. (7.3), two sets of $10^3$ trajectories each were created using $C = 100$ kJ/(mol nm$^2$) and $C = 1000$ kJ/(mol nm$^2$). The MD simulations used the same setup as the equilibrium MD simulations in Sec. 5.1, see Sec. A.4.1 for details. We note that the MD simulations were already used to construct a nonstationary generalized LE of this process [177].

The MD trajectories itself can be seen in the left column of Fig. 7.2. We see that $C = 100$ kJ/(mol nm$^2$)) represents a relatively weak spring, the transition time distribution is rather broad, such that transitions can be observed during the whole simulation time. The larger spring constant $C = 1000$ kJ/(mol nm$^2$)), on the other hand, restricts the system dynamics much stronger. Here, the system remains for $t \leq 20$ ps in the bound state before it rapidly moves over the barrier to the unbound state. We observe that the distribution of transitions times has a width of the order of 10 ps, i.e., it is relatively narrow. Having reached the unbound state, the MD trajectories closely follow the spring.

Now we construct a Langevin model based on the studies in Sec. 5.1. To emphasize that this model is based on equilibrium data, it will be called EQ-dLE model in the following. Its free energy can be taken from equilibrium MD or rescaled dLE, see for example Fig. 7.1. To choose the friction, we recall that the rescaled dLE estimated an approximately constant $\Gamma$, see Fig. 5.7. This motivates to set the friction of the EQ-dLE model to $\Gamma = 594$ kJ ps/(mol nm$^2$) by using the same unit convention as the spring constant $C$, see Sec. A.3 for details on the unit transformation. The mass estimate of the rescaled dLE coincides with the reduced mass of NaCl, i.e., $\mathcal{M} = 13.88$ u, and together with $k_{\mathrm{B}}T = 2.5$ kJ/mol at $T = 300$ K we get $\mathcal{K} = 54$ kJ ps$^{1/2}$/(mol nm) by using the fluctuation-dissipation theorem. As final step the EQ-dLE model is complemented with the external force Eq. (7.3) by simply adding it to the Markovian Langevin equation. Then, the equations of motion are integrated using a time step of $\delta t = 10$ fs and the bound state as starting conformation. $10^3$ trajectories are produced for each of the two springs. They closely resemble the MD for both spring constants, see the right column of Fig. 7.2. Again, the weak spring leads to an broad distribution of transition times, while the stiff spring restricts the trajectory much stronger.

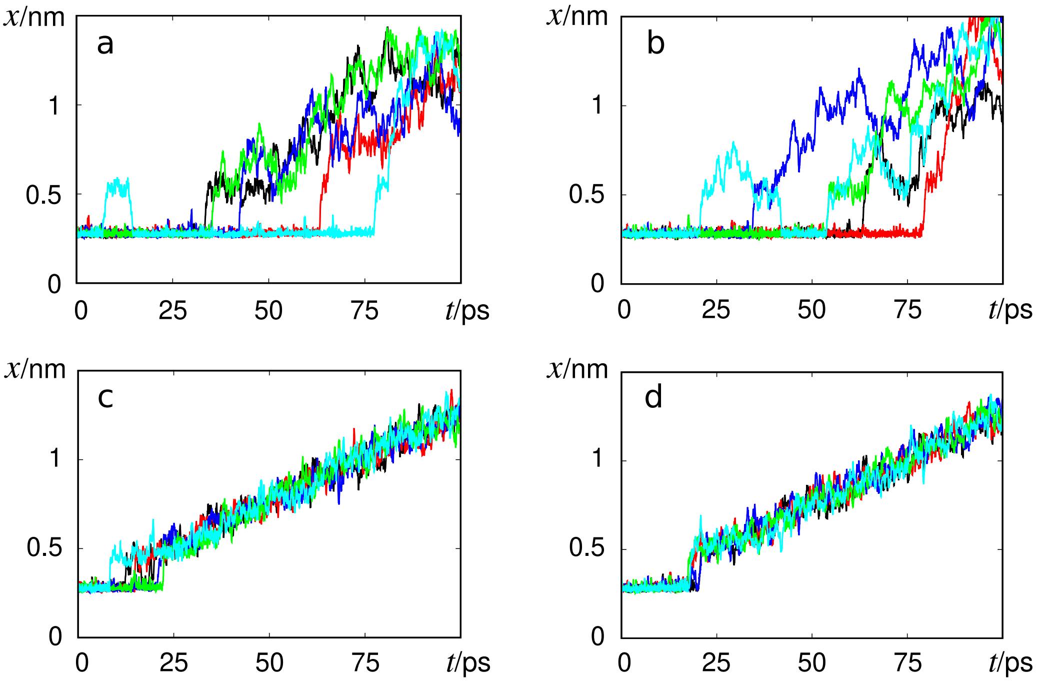

Figure 7.2: **Exemplary trajectories for MD and EQ-dLE model.** MD trajectories for $C = 100$ kJ/(mol nm$^2$)) (a) and $C = 1000$ kJ/(mol nm$^2$)) (c) are shown. The panels (b) and (d) show trajectories of the EQ-dLE model with similar spring constants.

To compare the dynamics of MD and EQ-dLE model in more detail, we first remove the systematic drift caused by the spring from the non-stationary trajectories $x(t)$. To this end the mean-free variable [177]

$$\delta x(t) = \frac{x(t) - \langle x(t) \rangle}{\langle (x(t) - \langle x(t) \rangle)^2 \rangle^{1/2}} \tag{7.4}$$

is introduced where the averages are taken over the different trajectory ensembles. The autocorrelation of this variable

$$C_x(t, t+\tau) = \langle \delta x(t) \delta x(t+\tau) \rangle \tag{7.5}$$

does not only depend on the lag time $\tau$ but also on the initial time $t$, see Sec. 2.4. As we see in Fig. 7.3a and b the position autocorrelations of both trajectory sets, MD and EQ-dLE model, decay very similarly for the weak spring. The decay time is of the order of $\approx 20$ ps independent of $t$.
When considering the stiff spring, Fig. 7.4a and b, we see that the position autocorrelations of MD and EQ-dLE model coincide as well but decay faster than for the weak spring. For $t = 10 - 15$ ps the decay constant is of the order of $\approx 5$ ps while the autocorrelations decay faster ($\approx 1$ ps) for larger $t$. The slower decay at early times can be

associated with forward and backward crossings over the main energy barrier while the fast decay at larger $t$ is tied to hydration shell dynamics around the two ions [42].
In total we can conclude that, at least for the inspected observables, the enforced dissociation of NaCl can be covered by a Markovian Langevin model constructed based on equilibrium data. This can be seen as conformation for the assumption that the NaCl dynamics evolve in the linear response regime as long as moderate pulling velocities $v$ are used. Still, just as for the rescaled dLE modeling of the equilibrium dynamics in Sec. 5.1, the EQ-dLE model has certain limitations. When considering the velocity autocorrelations $C_v$, shown in Figs. 7.3c and d and Figs. 7.4c and d, we observe for the MD a fast initial decay on a time scale of $\approx 25$ fs followed by damped oscillatory features with a period of $\approx 120$ fs. Besides a weak dependence on the spring constant, the amplitude of these oscillations changes with time $t$, for smaller $t$ it is higher.

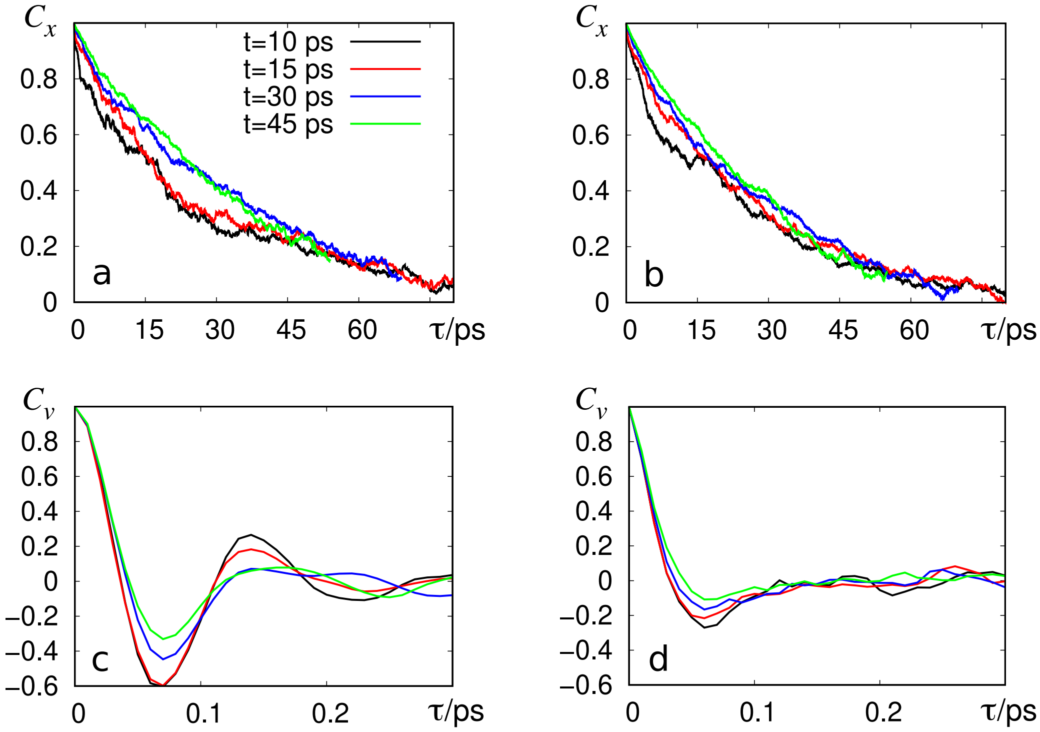

Figure 7.3: **Position and velocity autocorrelations for the weak spring.** The position autocorrelations $C_x$ of the MD (a) and the EQ-dLE model (b) are presented. Additionally, the velocity autocorrelations $C_v$ of the MD (c) and the EQ-dLE model (d) are shown. The considered spring constant is $C = 100$ kJ/(mol nm$^2$)).

When considering the EQ-dLE model it turns out that the Langevin model successfully covers the initial decay of $C_v$ but underestimates, or even misses, the following oscillations. Hence, just as for the equilibrium dynamics in Sec. 5.1, the Markovian Langevin model misses the dynamical details time scales of $\leq 0.1$ ps but successfully reproduces the long time scales at $\approx 1 - 10^3$ ps.
As final consideration we can have a look at the behavior of a dLE model directly con-

structed on the nonequilibrium data without modifying the next-neighbor estimation according to Eq. (7.2). As can be seen in Fig. A.12, this naive approach produces trajectories which completely miss the spring force. This can be explained by the lack of time information in the normal next neighbor estimation (Eq. (4.10)) were the driving force is falsely averaged out in the individual neighborhoods. This shows that we truly need to modify the next neighborhood estimation according to Eq. (7.2) if we want to derive a dLE model directly on the data.

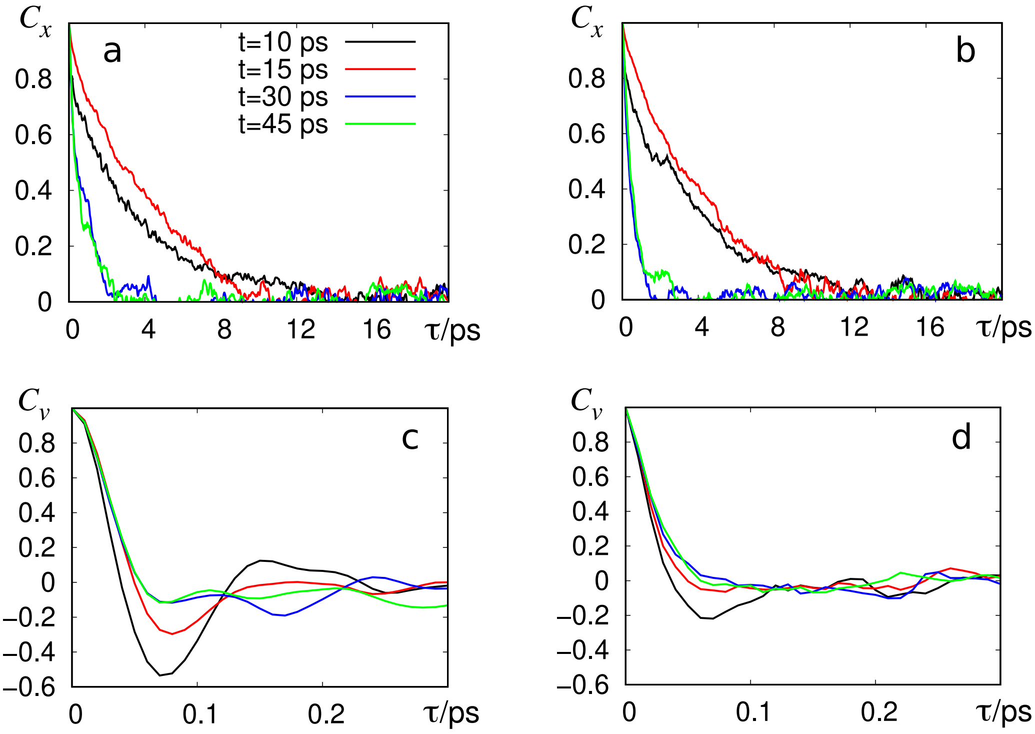

Figure 7.4: **Position and velocity autocorrelations for the stiff spring.** The position autocorrelations $C_x$ of the MD (a) and the EQ-dLE model (b) are presented. Additionally, the velocity autocorrelations $C_v$ of the MD (c) and the EQ-dLE model (d) are shown. The considered spring constant is $C = 1000$ kJ/(mol nm$^2$)).

## 7.2 Modeling of relaxation processes

In contrast to external driving, relaxation processes do not include explicitly time dependent forces. Instead, the nonstationary nature of such dynamics is a result of the chosen initial conditions. As depicted in Fig. 7.5, the system starts at $t_0$ in the high energy state $\rho(t_0)$ and relaxes for $t > t_0$ toward its low energy equilibrium state $\rho_{\text{eq}}$. Although the starting configuration of this process is by design nonstationary, the system itself is defined just the same as in equilibrium, i.e., the system Hamiltonian is time-independent. As consequence we see that the relaxation dynamics are governed by the free (equilibrium) energy landscape $F(\boldsymbol{x})$, just as shown in Fig. 7.5. From perspective of the Markovian Langevin equation these considerations imply that the drift field $\boldsymbol{f}(\boldsymbol{x})$ should be the same as in equilibrium. Additionally, identical Hamiltonians imply that equilibrium and relaxation dynamics are governed by the same $\Gamma$ and $\mathcal{K}$, i.e., all three Langevin fields are unchanged.

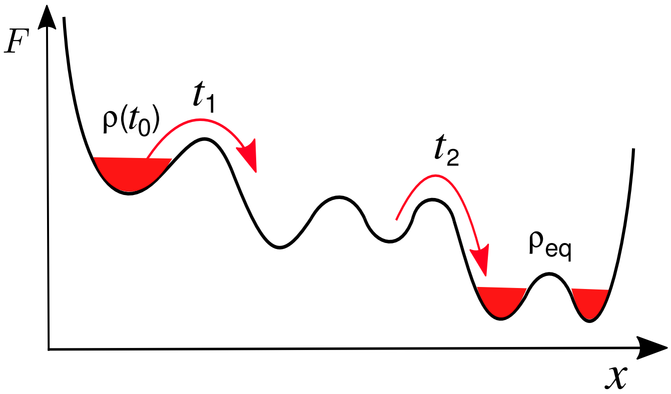

Figure 7.5: **Relaxation process** Initialized in the high-energy state $\rho(t_0)$, the system explores the free energy $F(x)$ by crossing the different barriers at time scales $t_1$ and $t_2$. The process ends once the low-energy equilibrium state $\rho_{\text{eq}}$ is reached.

Still, in case we want to determine the respective fields by, e.g., the dLE, it needs to be taken into account that there needs to be sufficient data everywhere in the conformational space to allow for converged estimations. Since the initial high energy state $\rho(t_0)$ is hardly visited again once the trajectory left it, one single reference time trace will not be sufficient to parameterize the LE. This means that we have to perform an ensemble average over numerous independent nonequilibrium trajectories $\boldsymbol{x}^r(t_n)$ of some length $t_{\max}$ where $r = 1, 2, ..., N_{\text{traj}}$ indicates the (arbitrary) numbering of the trajectories. In consequence, the convergence of the next-neighborhood estimation, and by this the accuracy of the Langevin fields, heavily depends on both parameters, $N_{\text{traj}}$ and $t_{\max}$. For example, considering again the system in Fig. 7.5, it is very likely that a short trajectory length $t_{\max} \approx t_1$ only allows to cover the initial dynamics $t \lesssim t_1$ but not the further relaxation to $\rho_{\text{eq}}$ with $t \gtrsim t_2$.
As consequence of a finite sampling time $t_{\max}$, the dLE will estimate a drift field $\boldsymbol{f}(\boldsymbol{x})$ which deviates from the equilibrium average $\boldsymbol{f}_{\text{eq}}(\boldsymbol{x}) = -\nabla F(\boldsymbol{x})$. Since $F(\boldsymbol{x}) = -k_{\text{B}}T \ln(P(\boldsymbol{x}))$ depends on the equilibrium distribution $P(\boldsymbol{x})$, which will not be completely sampled by the short nonequilibrium trajectories of length $t_{\max}$, the dLE observes a "biased energy landscape" [138]

$$\mathcal{F}(\boldsymbol{x}) = -k_{\text{B}}T \ln(\mathcal{P}(\boldsymbol{x})) \tag{7.6}$$

instead where $\mathcal{P}(\boldsymbol{x}) \propto \int dt \rho(\boldsymbol{x}, t)$ represents the distribution sampled within $0 \leq t \leq t_{\max}$. Only for $t_{\max} \to \infty$ (or more exactly for $\mathcal{P}(\boldsymbol{x}) \to P(\boldsymbol{x})$) it can be expected that $\mathcal{F}(\boldsymbol{x})$ approaches $F(\boldsymbol{x})$. It is worth to stress at this point that (concerning Eq. (7.6)) the term "biased" only refers to the nonstationary initial conditions and does not indicate any external biasing force like, e.g., given in Eq. (7.1).

In summary, our considerations up to this point indicate that it is in practice possible that $\mathcal{F}(\boldsymbol{x})$ only covers parts of the equilibrium free energy. Considering the example in Fig. 7.5, it may happen that only the initial and intermediate states are covered if $t_{\max}$ is chosen too small. In consequence, any dLE model constructed for this data will only cover these states as well.

Still, although the global observables $\mathcal{F}(\boldsymbol{x})$ and $F(\boldsymbol{x})$ can easily deviate, the dLE drift field $\boldsymbol{f}(\boldsymbol{x}) = -\nabla\mathcal{F}(\boldsymbol{x})$ is only weakly influenced by these deviations, as it is locally defined. Representing the slope of the energy landscape at $\boldsymbol{x}$, the dLE can estimate the drift based on local averages. These local averages converge much faster than $\mathcal{F}$ because the nonequilibrium trajectories $\boldsymbol{x}^{(r)}$ successively sample the consecutive energy minima by staying some time inside of them before performing the next jump. Since $\Gamma$ and $\mathcal{K}$ are also estimated based on local averages, their estimates should be unproblematic as well. Hence, we expect in general that the dLE might not sample the full configurational space once $t_{\max}$ is too small but the covered dynamics should be nevertheless accurately modeled. In the next section, we will inspect if this assumption holds true.

### 7.2.1 Hierarchical energy landscape

To investigate the effect of finite data on the resulting Langevin model, we consider the one-dimensional free energy landscape depicted in Fig. 7.6. Here, four states are separated by three energy barriers of similar height. The model is inspired by photo- or ligand-induced conformational transitions in proteins [178, 179] where the system is prepared in some nonstationary state (here state 1) from which it evolves via intermediate states (state 2 and 3) before it finally enters another, more stable, low-energy state (state 4). The hierarchical shape of the energy landscape effects that 1↔2 transitions occur

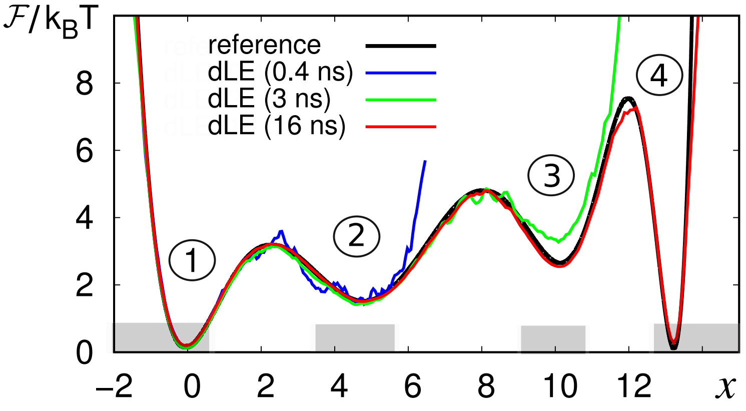

Figure 7.6: **Hierarchical free energy landscape.** The reference (black) reveals four states connected via energy barriers of similar height. The grey regions at the x-axis indicates the cores of the states used to calculate waiting times. State 1 represents the starting state of the dynamics we are observing. Three sets of trajectories of length $t_{\max} = 0.4$ ns, 3 ns and 16 ns are generated to serve as dLE input resulting in biased energy landscapes $\mathcal{F}(x)$, see Eq. (7.6), reaching state 2 (blue), 3 (green) and 4 (red), respectively.

earlier and more rapidly than 1↔3 dynamics which, in turn, emerge earlier and more frequent than 1↔4 transitions.
To complete the Langevin model we choose $\mathcal{M} = 400$ ps (which equals 26 u), $\Gamma = 2000$, $T = 300$ K and take an integration time step of $\delta t = 0.04$ ps. Twenty long trajectories of 16 $\mu$s each were simulated, every time series starts in state 1 at $x = 0$. As dynamical observable we consider the average waiting times $t_{\text{wait}}$ of the transitions $1 \rightarrow j$. As we see in Fig. 7.7, the transitions 1→2 occur on a time scale of ≈ 3 ns, the transitions 1→3 on a time scale of ≈ 30 ns and the dynamics 1→4 need times of ≈ 150 ns. Hence, as expected, there is a pronounced separation of time scales. The mean values of the different waiting times can be found in Tab. 7.1. We see that the reference data is large enough to minimize the errors on the estimates. The excellent convergence of the reference can also be seen in the estimates energy landscape which perfectly reproduces the input free energy.

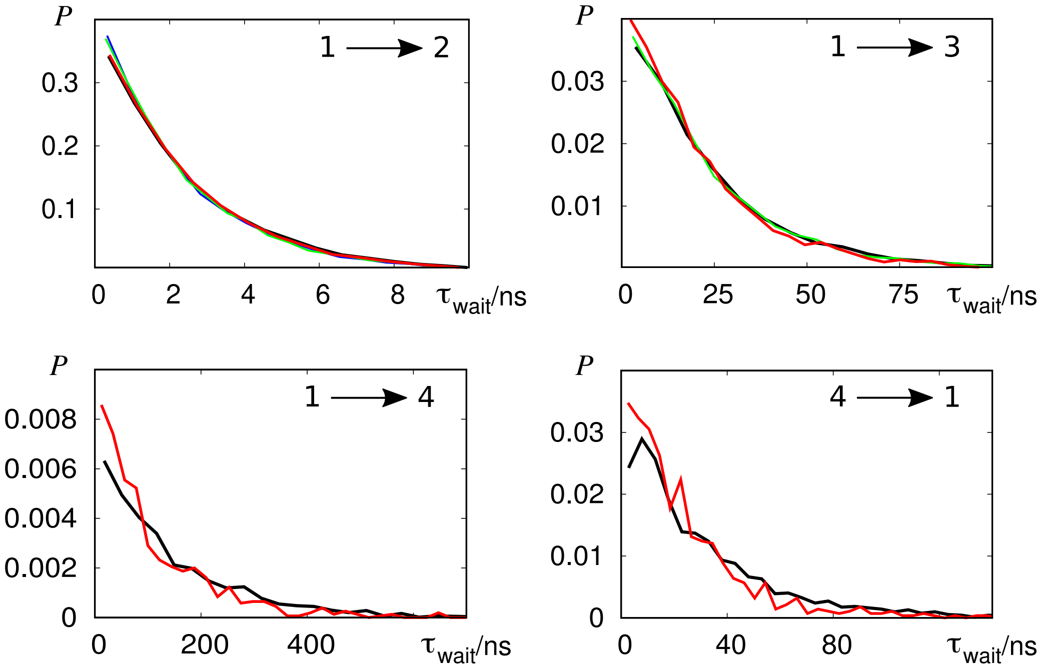

Figure 7.7: **Convergence of waiting times.** Shown are distributions of waiting times $\tau_{\text{wait},i\rightarrow j}$ from the reference (black) and dLE simulations using input data of length 0.4 ns (blue, the dLE only reaches state 2), 3 ns (green, the dLE reaches state 2 and 3) and 16 ns (red, the dLE reaches all state). Top left, we see the transition 1→2, top right the transition 1→3, bottom left the transition 1→4 and bottom right 4→1.

Having specified the reference dynamics, we can now inspect the effects of $t_{\text{max}}$ on the dLE predictions. To this end, three different input data sets were construct by choosing $t_{\text{max}}$ based on the waiting time distributions in Fig. 7.7. Each of the three data sets consist of $10^2$ Langevin trajectories which were produced with $\delta t = 0.04$ ps and started in state 1. By setting $t_{\text{max}} = 0.4$ ns, 3 ns and 16 ns, approximately ten transitions to state 2 ($t_{\text{max}} = 0.4$ ns), 3 ($t_{\text{max}} = 3$ ns) and 4 ($t_{\text{max}} = 16$ ns) are collected. To speed up the dLE dynamics, the data sets were pre-averaged, see Sec. 4.1.5, to apply the binned dLE. Each of the data sets was pre-averaged to ≈ $10^4$ points, see Sec. A.14 for details on the settings. For each of the pre-averaged data sets ten dLE trajectories with a length of 10 $\mu$s-long were produced. The dLE runs sample the different transitions at least $10^3$ times, i.e., their predictions of statistics and dynamics have converged. By considering the biased energy landscapes $\mathcal{F}(x)$ estimated by the dLE, see Fig. 7.6, we see that it

| data | $\tau_{\text{wait}1\to2}$ | $\tau_{\text{wait}1\to3}$ | $\tau_{\text{wait}1\to4}$ | $\tau_{\text{wait}4\to1}$ |
|---|---|---|---|---|
| reference | $2.5 \pm 0.01$ | $23 \pm 0.2$ | $143 \pm 3.3$ | $30 \pm 0.7$ |
| dLE (0.4 ns) | $2.4 \pm 0.02$ | – | – | – |
| dLE (3 ns) | $2.4 \pm 0.01$ | $22 \pm 0.3$ | – | – |
| dLE (16 ns) | $2.5 \pm 0.02$ | $21 \pm 0.3$ | $110 \pm 4.3$ | $24 \pm 0.9$ |

Table 7.1: Average waiting times (given in ns) of selected transitions $i \to j$ between states $i$ and $j$ for the hierarchical energy landscape in Fig. 7.6. The first row represents the reference data, the following rows show dLE results based on different data sets with varying length $t_{\max}$ (given in parenthesis). Errors are calculated as standard deviations of the mean. Note that transitions marked with "–" do not occur.

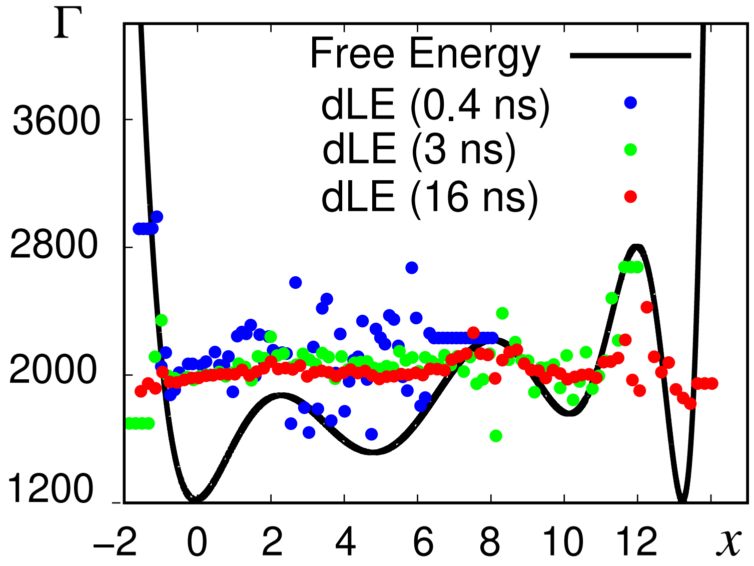

Figure 7.8: **Friction estimate for varying data length.** By increasing the length of the input trajectories, the scattering of the dLE friction estimate Γ can be reduced. Shown are the estimates for lengths of $t_{\max} = 0.4$ ns (blue), 3 ns (green) and 16 ns (red).

indeed depends on $t_{\max}$ if the dLE detects two, three or all four minima. Still, once a specific minimum is covered by the dLE, it is reproduced quite well, i.e., $\mathcal{F}$ and the equilibrium free energy $F(x)$ coincide. In consequence the derivative of $\mathcal{F}(x)$ covers the derivative of $F(x)$. When inspecting the friction estimate $\Gamma$, see Fig. 7.8, it can be seen that the dLE correctly estimates on average $\Gamma = 2000$ even for the smallest $t_{\max}$. We note that the same holds for the estimate of the noise amplitude $\mathcal{K}$ and the mass $\mathcal{M}$ (not shown). As consequence of the accurate estimations of the dLE fields we observe average waiting times for 1→2 and 1→3 which reproduce the reference very well, only a deviation of $\approx 4\%$ can be found, see Tab. 7.1. The waiting time distributions of reference and dLE are in line as well, see Fig. 7.7, the only difference is that the dLE tends to overestimate fast transitions for small $t_{\max}$. When considering the slowest transition 1→4 we see that the dLE with $t_{\max} = 16$ ns deviates by about 20%, i.e., the dLE prediction is not as accurate as for the other transitions. A similar relative error can be observed for the opposite direction 4→1.

In summary, we can nevertheless conclude that only a few transition events need to be covered by the input data to allow for qualitatively correct dLE predictions of the observed dynamics. The estimated "biased energy landscape" $\mathcal{F}(x)$ might deviate from the equilibrium free energy $F(x)$ but the (local) derivative as well as friction and noise are qualitatively reproduced by the dLE based on limited input statistics.

### 7.2.2 Pressure-induced nucleation of hard spheres

We now challenge the capabilities of the dLE to model relaxation processes by considering the crystal nucleation of a compressed liquid of hard spheres. This system was already used as test problem for the derivation of a nonstationary generalized Langevin equation [180, 181] which makes it instructive to inspect if the Markovian dLE can provide a reasonable dynamical model for this weak first-order phase transition [182].

The considered system consists of 16384 hard spheres defined by mass $\mathcal{M}$, diameter $\sigma$ and natural time step $\delta t = \sqrt{\mathcal{M}/k_{\mathrm{B}}T}\sigma$. Our reference MD data was simulated by Meyer at al. [181]. Here, the system was initially equilibrated in a liquid state at a volume fraction of $\eta_0 = 0.45$ before it was impulsively compressed to $\eta = 0.54$ at $t = 0$ by rescaling the simulation box as well as all positions. This compression induced the crystallization process. In total, $N_{\mathrm{traj}} = 580$ nucleation trajectories with a length of $t_{\max} = 214\ \delta t$ were produced.

As first step to apply the dLE one needs to define the system variable. We choose here the percentage $x$ of particles that completed crystallization as single collective coordinate since it directly accounts for the evolution of the nucleation. We note that $x$ can be readily calculated from the $Q_6$ order parameter [180, 181]. Fig. 7.9a displays eight exemplary MD trajectories $x(t)$. Due to the fact that nucleation seeds need to form before the crystallization can happen, the trajectories stay some time at $x = 0$ before the nucleation begins. The distribution of this "induction time" is quite wide, especially when comparing it to the fast, sigmoidal-shaped rise of $x(t)$ representing the crystallization itself. This process needs only about 20 $\delta t$. When inspecting the MD trajectories closer, we see that 357 trajectories reach values $x \gtrsim 0.8$ where they level off and slowly converge against the maximally possible $x = 1$. On the other hand, 125 trajectories, i.e., a significant fraction, get stuck at $x \approx 0.65$. This behavior reflects the occurrence of crystal defects hindering the successful nucleation of the full system [180, 181]. The remaining 98 trajectories start too late to decide whether they get stuck at $x \approx 0.65$ or reach $x = 0.8$. To get an overall idea of the time scale of the whole nucleation process we can average over all MD trajectories (see Fig. 7.9b) and find a time of $\approx 100 - 150\ \delta t$.

The different dynamical patterns observed for the individual trajectories can also be seen when inspecting the energy landscape $\mathcal{F}(x)$ defined by Eq. (7.6) (see Fig. 7.10a). The deep initial minimum at $x \approx 0.014$ represents the initial liquid state of the system, while the minimum at $x = 0.65$ accounts for the trajectories with defective clusters. The depth of the last minimum at $x \approx 0.95$ highly depends on $t_{\max}$ since it represents the (approximately) completed nucleation of the full system, i.e., if $t_{\max}$ gets larger, proportionally more trajectory points will contribute to this minimum and it will get deeper. This holds because more and more trajectories reach $x \approx 0.95$ for growing $t_{\max}$ and trajectories which already reached this state cannot leave it again.

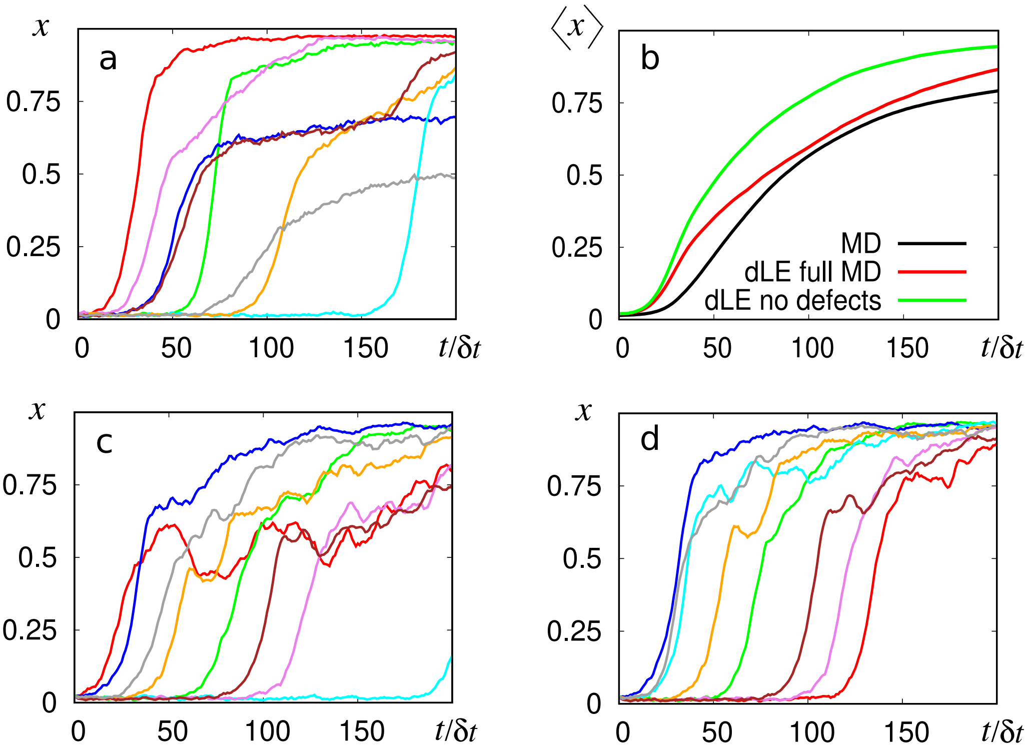

Figure 7.9: **Exemplary trajectories of the nucleation of hard spheres.** (a) Here, we see some exemplary MD trajectories. For comparison, the bottom row presents trajectories of the dLE based on the full data (c) and only based on the MD trajectories which reach $x = 0.8$ (d). (b) This panel depicts the average value $\langle x(t) \rangle$ of the MD and the two dLE sets based on 580 (MD) and 600 (dLE) trajectories.

As dynamical quantity to evaluate the dLE performance later on, we define the nucleation time $t_{\text{nuc}}$ as the time needed by the individual MD trajectories to cross $x = 0.8$. Fig. 7.10b shows the distribution calculated from the successfully crystallizing MD runs. Due to the relatively small statistics the resolution is quite low but we can nevertheless detect the peak at $t_{\text{nuc}} = 76\ \delta t$. The mean nucleation time is slightly larger and given

by $\langle t_{\mathrm{nuc}} \rangle = 102\ \delta t$.
Now, we can inspect the capability of the dLE to model the nucleation dynamics. It turns out that the noise detected by the dLE at $1 \cdot \delta t$ already covers the characteristics of white noise (see Sec. A.15). Hence, this time step was used to produce $10^4$ dLE trajectories of length $t_{\mathrm{max}} = 214\ \delta t$ where each run started at $x = 0.02$. When considering exemplary trajectories (Fig. 7.9c) we see that the dLE qualitatively reproduces the induction time distribution as well as the sigmoidal-shaped rise of $x(t)$. The average over the dLE trajectories (Fig. 7.9b) qualitatively coincides with the MD as well.

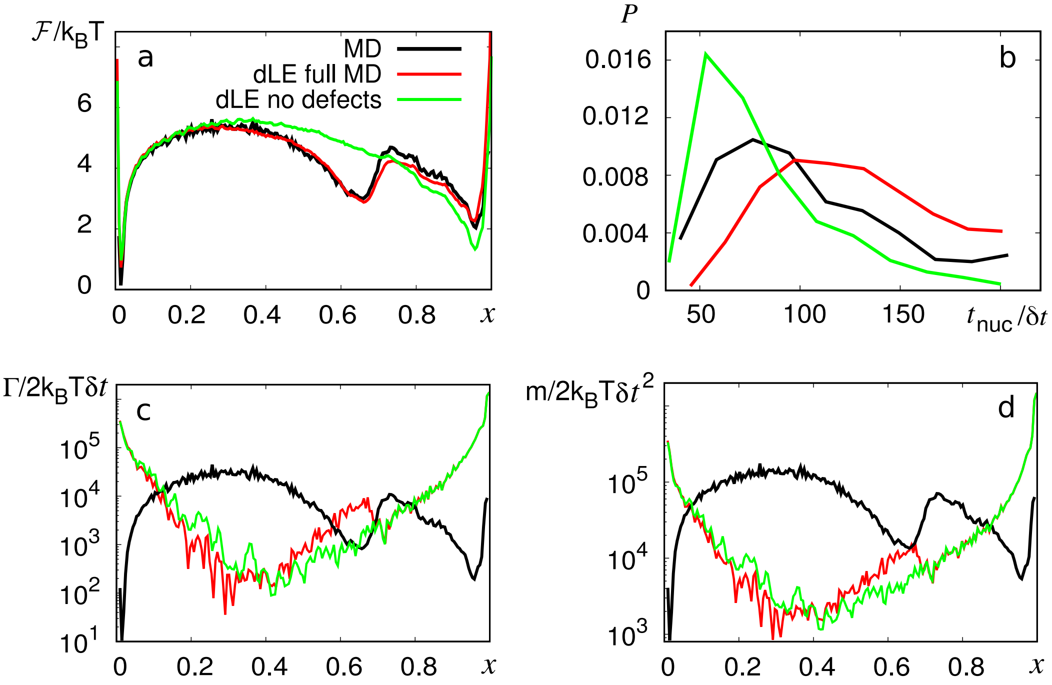

Figure 7.10: **Modeling results for the nucleation of hard spheres.** Shown are (a) the biased energy landscape $\mathcal{F}(x)$, (b) the distribution of nucleation times $t_{\mathrm{nuc}}$, (c) the friction estimate $\Gamma(x)$ and (d) the mass estimate $\mathcal{M}(x)$. The black curves in the top row show MD results while the black curves in the bottom row depict the MD energy landscape. The dLE results based on the full MD data are given in red while green represents the dLE estimates when only using successfully crystallizing MD trajectories as input data.

When inspecting the nonstationary energy landscape $\mathcal{F}(x)$ (see Fig. 7.10a) we find in addition that the dLE perfectly reproduces the MD. Still, there are some differences between MD and dLE. First, the dLE trajectories mostly do not directly rise to $x \approx 0.8$ but stay for some time around $x \approx 0.65$ before they are able to leave the local minimum again. In turn, we do not find any dLE trajectory which gets stuck at $x \approx 0.65$. Both observations result from the local estimation of the dLE fields where crystallizing and defective MD trajectories are simply mixed. As consequence, the dLE distribution of $t_{\mathrm{nuc}}$, Fig. 7.10b, is shifted compared to the MD. It peaks around $t_{\mathrm{nuc}} = 100\ \delta t$ and yields as mean nucleation time $\langle t_{\mathrm{nuc}} \rangle = 128\ \delta t$.

To quantify the influence of the defective MD trajectories on the dLE performance, we can produce dLE trajectories only based on those MD runs which successfully reach $x = 0.8$. Though it should be kept in mind that we do not just remove the defective MD runs this way but also trajectories which would successfully crystallize for $t > 214\ \delta t$. Still, based on the fact that the MD distribution of $t_{\text{nuc}}$ peaks much earlier, this problem should be minor. Hence, $10^4$ dLE trajectories based on the 357 successfully crystallizing MD runs were simulated. Exemplary trajectories shown in Fig. 7.9d indeed reveal that the dLE dynamics do not get stuck at $x \approx 0.65$, just as expected. The nonstationary energy landscape, Fig. 7.10a, confirms that the minimum at $x \approx 0.65$ can be associated with the defective trajectories since it vanished from perspective of the dLE. Still, this does not mean that the nucleation time $t_{\text{nuc}}$ of the MD can be perfectly reproduced by the dLE. Averaging over all $10^4$ dLE runs, see Fig. 7.9b, it turns out that the crystallization happens too fast when omitting the defective MD trajectories. Consequently the distribution of $t_{\text{nuc}}$, see Fig. 7.10b, peaks earlier at $t_{\text{nuc}} \approx 60\ \delta t$ and the mean nucleation time turns out to be predicted as $\langle t_{\text{nuc}} \rangle = 83\ \delta t$.

Hence, simply removing the defective MD trajectories does not perfect the dLE model. On the other hand it should be noted that the overall MD statistics are not optimal. Longer MD runs, for example, would give us the possibility to distinguish more reliably between crystallizing and defective trajectories since all slow (but successfully crystallizing) runs would gradually reach $x \approx 0.8$. Although this concerns only relatively few MD trajectories (given that the MD distribution of $t_{\text{nuc}}$ peaks at 76 $\delta t$), it might be possible that those few runs contribute essential information needed to slow down the dLE in the initial deep minimum.

To conclude this study we will take a look at the dLE estimate of $\Gamma$ as functions of $x$. Fig. 7.10c shows that the friction is small at the main barrier and large at the two main minima reflecting the liquid and crystallized state. To understand this behavior, we recall that the system coordinate $x$ represents the fraction of crystallized particles and that $\Gamma$ accounts for the fluctuations of the associated velocity. Considering the initial shock-compressed state and the final crystallized state, these fluctuations are expected to be high, since they reflect the highly restrained collective motion of the system. Meanwhile, the transition over the barrier is very different. It starts once a single crystal cluster (out of the numerous initial clusters existing after shock compression) exceeds a critical size [183] and evolves relatively rapidly. As we see that the crystallization occurs with a similar velocity in all simulations (Fig. 7.9a), the variance of the velocity and consequently the friction is low on the barrier. Compared to the other systems studied in this thesis where the friction is found to change only little, this pronounced variation of the friction by a factor of $\approx 10^3$ is remarkable. It presumably reflects the high cooperativity of the crystallization that involves essentially all particles, while typically only a comparatively small subset of atoms performs cooperative motion for our other systems. Interestingly, when considering $\mathcal{M}$ (Fig. 7.10d) calculated according to Eq. (4.17) based on $\Gamma$, we see that it behaves very similar to $\Gamma$.

In summary, it can be concluded that it is possible to construct a dLE model of the nucleation of hard spheres, which is able to cover the MD dynamics at least qualitatively. Given the highly pronounced nonstationarity of this process, this finding strongly supports the line of thought in Sec. 7.2, i.e., the dLE is truly able to model relaxation processes.

## 7.3 Connection to the nonstationary generalized Langevin equation

The Markovian Langevin equation (3.28) relies on a time scale separation between the fast bath degrees of freedom and the slow system coordinate $x$. To circumvent this limitation, which typically requires a high-dimensional system description, many implementations of the generalized Langevin equation (GLE), see Sec. 3.2.2, were proposed. One such implementation was presented by Meyer and coworkers [74, 75]. Based on a time-dependent projection operator formalism, this approach can account for nonequilibrium dynamics. In particular, this implementation of the GLE was used to model the nucleation of hard spheres [181]. Here, it was concluded that the memory kernel needed to cover the crystallization dynamics decays slowly, which appears to be incompatible with our results in Sec. 7.2.2 where it was shown that the (Markovian) dLE performs satisfactorily as well. Still, this apparent paradox can be solved.

The nonstationary GLE of Meyer et al. is, due to the used Mori-type approximation, different to the GLE (3.21) derived in Sec. 3.2.2 based on the work of Zwanzig [32]. Here, the equation of motion of some variable $A(t)$ is given by [74, 75]

$$\frac{dA}{dt} = \omega(t)A(t) + \int_0^t K(t,\tau)A(\tau)d\tau + \eta(t) \tag{7.7}$$

with memory kernel $K(t,\tau)$ and noise $\eta(t)$ related by a generalized fluctuation-dissipation theorem. In contrast to Eq. (3.21), we see on the left side of this equation $dA/dt$ and not $d^2A/dt^2$. Additionally, the drift force is set to $\omega(t)A(t)$ which vanishes for a mean-free variable $A(t)$.

Now we can compare our Markovian Langevin framework to this GLE. First, we note that, even for a one-dimensional system coordinate $x$, our framework is not defined by a single first-order differential equation like Eq. (7.7), but by the two equation

$$\dot{x}(t) = p(t)/\mathcal{M}, \tag{7.8}$$

$$\dot{p}(t) = f(x) - \Gamma p(t) + \mathcal{K}\xi(t), \tag{7.9}$$

where $\Gamma$ is assumed to be constant, for simplicity. To relate our two equations to Eq. (7.7) it is possible to follow the calculations of Zwanzig [31] by integrating Eq. (7.9) which yields

$$p(t) = \int_0^t e^{-\Gamma(t-\tau)/\mathcal{M}} f(x(\tau))d\tau + \int_0^t e^{-\Gamma(t-\tau)/\mathcal{M}} \mathcal{K}\xi(\tau)d\tau \tag{7.10}$$

when assuming $p(0) = 0$. After inserting this in Eq. (7.8) we get

$$\dot{x}(t) = \int_0^t K(t,\tau)x(\tau)d\tau + \eta(t), \tag{7.11}$$

with the memory kernel

$$K(t,\tau) = \frac{1}{\mathcal{M}} e^{-\Gamma(t-\tau)/\mathcal{M}} \frac{f(x(\tau))}{x(\tau)}, \tag{7.12}$$

and the corresponding colored noise

$$\eta(t) = \frac{1}{\mathcal{M}} \int_0^t e^{-\Gamma(t-\tau)/\mathcal{M}} \mathcal{K} \xi(\tau) d\tau. \tag{7.13}$$

Note that the memory kernel reduces to a simple exponential function for linear drift forces, i.e., $f(x(\tau))/x(\tau) = \text{const.}$ In this case, the kernel only depends on the time difference $t - \tau$ and becomes independent of the position $x$.

This brief calculation shows that it is indeed possible that the GLE framework defined by Eq. (7.7) and the Markovian Langevin approach defined by Eqs. (7.9) and (7.8) are simultaneously correct. The observation of apparently non-Markovian system features based on the GLE, like a slowly decaying memory kernel, could potentially be the result of a nontrivial multistate energy landscape with high barriers, i.e., slow state transitions. By explicitly including this energy landscape into the equations of motion it might be possible to construct a Markovian system description which relates to the GLE as sketched above. This explains why we could derive a reasonable Markovian Langevin model for the nucleation of hard spheres although other studies found a slowly decaying memory kernel for the GLE defined by Eq. (7.7).

## 7.4 Summary

We have seen that only a few modifications are needed to apply the data-driven Langevin approach to nonequilibrium dynamics. Two nonequilibrium scenarios were considered: external driving in the linear response regime (Sec. 7.1) and relaxation processes (Sec. 7.2). It was observed that nonequilibrium conditions mainly affect the deterministic drift. The free energy needs to be replaced by the biased energy landscape

$$\mathcal{F}(\boldsymbol{x}, t) = -k_{\mathrm{B}}T \ln(\mathcal{P}(\boldsymbol{x})) + V_{\mathrm{ext}}(\boldsymbol{x}, t). \tag{7.14}$$

Here, the term $-k_{\mathrm{B}}T \ln(\mathcal{P}(\boldsymbol{x}))$ includes the distribution sampled within a finite MD simulation time $t_{\max}$, i.e., it accounts for relaxation processes as well as finite sampling. The second term $V_{\mathrm{ext}}(\boldsymbol{x}, t)$, on the other hand, represents a time-dependent potential induced by the external driving. In contrast to the drift, friction and noise remain approximately unaffected.

To validate our assumptions, we considered in Sec. 7.1.1 the enforced dissociation of sodium chloride in water as example for external driving and in Sec. 7.2.2 the crystal nucleation of a compressed liquid of hard spheres as example for relaxation processes. In both cases, the Markovian Langevin models performed well. Additionally, we inspected the convergence behavior of the dLE for a finite MD simulation time $t_{\max}$ in Sec. 7.2.1. Here, we observed for a hierarchical model system that shorter $t_{\max}$ bias the dLE predictions toward too fast dynamics. We note that this finding coincides with our observations for Aib$_9$ in Sec. 5.2.4 were we observed that shorter input data leads to faster dLE dynamics.

In the last Sec. 7.3 we made the connection to a nonstationary generalized Langevin equation [74, 75] which predicted slowly decaying memory kernels for the enforced dissociation of sodium chloride and the nucleation of hard spheres. Although this seemed to be incompatible with our findings in this chapter, it was shown that both Langevin descriptions, generalized and Markovian, can be simultaneously correct due to different partitionings between system and bath.

# 8 Conclusion and outlook

*"The difference between your decision and ours is experience. But you don't have to rely on that."*
– Levi Ackerman, "Attack on Titan"

Gaining comprehensive insights into the dynamics of proteins is a long-standing scientific challenge due to their inherent structural complexity and the diversity of dynamical time scales. Although molecular dynamics (MD) simulations opened the door to microscopic descriptions of the dynamics of interest, it is often still unclear how to properly interpret and analyze the resulting extensive data sets.

In this thesis we investigated the capabilities of the data-based Markovian modeling to provide "post-simulation" models, which approximate the (high-dimensional) MD simulation by low-dimensional model dynamics. Here, we focused on the Markovian Langevin equation via the data-driven Langevin (dLE) approach. We saw that the Markovian Langevin equation represents a correct approximation of memory-based dynamics once the temporal resolution $\delta t$ of the dLE is chosen larger than the memory decay time of the system. Since this cannot always be ensured, we formulated the rescaled dLE, which compensates the effects of too short $\delta t$ by rescaling the friction.

Using this approach we were able to derive as a first example an one-dimensional model of the dissociation and association of water solvated sodium chloride. It accurately predicted association and dissociation times on the order of hundreds of picoseconds. Although it is known that the surrounding water plays a significant role in the dynamics of sodium chloride [141], we did not need to include any explicit water coordinate in our system description. Additionally, generalized Langevin dynamics with a memory decay time on the order of the water time scales did not improve the model significantly. This shows that the rescaled dLE is well suited to provide (Markovian) Langevin models also for suboptimal (or incomplete) system coordinates.

Given that the main bottleneck of MD analysis is often the definition of a suitable low-dimensional system description, this indicates that the rescaled dLE can significantly accelerate the modeling process for new, unknown biomolecular systems. Instead of extensively searching for the perfect coordinates we can simply take a selection of reasonable (and transparent) degrees of freedom, apply the rescaled dLE and investigate the resulting model dynamics which should correctly account for the long time scales of the system.

Considering the modeling of the dynamics of the 164-residue T4 lysozyme, we saw directly how this idea can be implemented for complicated systems. Although extensive studies were performed to derive appropriate system coordinates [71, 150], the currently established two-dimensional coordinate set did not allow for an accurate Markov state model (MSM). While it was possible to cover the time scale of the transition between the two main states, the MSM overlooks the pathways on which the transition is performed. Adding additional system coordinates did not allow for the use of smaller lag times in the

MSM construction. Hence, we applied the rescaled dLE to the two-dimensional system description and derived a reasonable Langevin model at a small time step. Although we had again to sacrifice accuracy at short time scales, we were able to provide model trajectories which follow the given free energy and reproduce the long time scales observed in MD.

As another important advancement, we introduced the binned dLE to allow for the analysis of extensive MD data sets via pre-averaging. Considering a large enhanced sampling data set ($8 \cdot 10^7$ data points [73]) of the small AIB$_9$ peptide, we carefully ensured that this approach does not harm the resulting five-dimensional Langevin model if done correctly. We were able to reduce the number of data points by a factor of $10^2$, which shows that the binned dLE opens the door to the analysis of other, more complicated, systems by means of an enhanced sampling schemes which produces a lot of short MD simulations in parallel [73].

Based on the above findings, we propose the following approach for future studies of (unknown) protein dynamics. First, enhanced sampling MD simulations are used to generate numerous short trajectories which homogeneously sample the free energy landscape along some coordinates of interest. Then, the data is pre-averaged at a time resolution $\delta t$ which is small enough to resolve the relevant dynamics. Finally, the rescaled dLE is applied to the pre-averaged data to construct a Markovian Langevin model which accounts for the long time dynamics of the system.

As an alternative to the sampling via many short trajectories, we considered the dissipation-corrected targeted MD approach [42]. This framework allows for the construction of one-dimensional Markovian Langevin models of extremely slow dynamics based on constrained MD simulations. Since it is not possible to effectively access time scales on the order of milliseconds or higher by direct Langevin simulations (which would need prohibitively many integration steps), we proposed the concept of T-boosting. Here, we integrate the Langevin equation at high temperature to reduce the needed simulation times and subsequently extrapolate to the dynamics at the real temperature. By considering Langevin models of trypsin-benzamidine and the ligand unbinding from the N-terminal domain of heat shock protein 90 (Hsp90), we saw that this approach correctly predicts time scales on the order of milliseconds (trypsin-benzamidine) and tens of seconds (Hsp90) within a factor of ten. This accuracy is very competitive compared to other modeling results for trypsin-benzamidine [167].

Considering the question if the dcTMD-based modeling or the dLE-based approach will be more promising for future Markovian Langevin studies, it needs to be kept in mind that both frameworks have different virtues and shortcomings. One the one hand, dcTMD is advantageous if the dynamics of interest can be enforced by constrained MD simulations and if it is sufficient to describe them by a single system coordinate (which is typically some distance). Once we are interested in multidimensional system dynamics (which might include angles or linear combinations of microscopic distances), on the other hand, the dLE can be more advantageous although the necessary MD simulations might be more complicated than for dcTMD. Hence, both methods have somewhat different scopes and complement each other.

Finally, we extended the applicability of the data-driven Langevin approach to the nonequilibrium regime. Two specific processes were considered: the external driving of a system under study and the relaxation from nonequilibrium initial conditions. We saw that nonequilibrium mainly effects the deterministic drift of the Langevin framework. In consequence, the free energy needs to be replaced by the biased energy landscape

$$\mathcal{F}(\boldsymbol{x},t) = -k_{\mathrm{B}}T\ln(\mathcal{P}(\boldsymbol{x})) + V_{\mathrm{ext}}(\boldsymbol{x},t), \tag{8.1}$$

where the first term account for relaxation processes (and finite sampling) and the second term represents a time-dependent potential added by external driving. The other two Langevin forces, friction and noise, stay approximately unaffected in the linear-response regime. By considering the enforced dissociation of sodium chloride and the crystal nucleation of a compressed liquid of hard spheres, we observed that an accordingly modified Markovian Langevin model is indeed able to reproduce the reference results. At first glance, this finding appears to be incompatible with recent studies based on a nonstationary generalized Langevin equation [74, 75] which predicted slowly decaying memory kernels for the very same problems [177, 181]. Still, we saw that both Langevin frameworks, the Markovian as well as the generalized, can be correct since they are based on different partitionings between system and bath.

In summary, we conclude that Markovian (Langevin) models can be very useful for future studies of biomolecular dynamics in both equilibrium and nonequilibrium conditions. Despite their conceptional simplicity they can provide robust low-dimensional approximations of high-dimensional MD dynamics, which will serve well to deeper understand the complicated dynamics of proteins.

# Appendix

## A.1 Derivation of field estimates for the dLE

To derive the three field estimates of the normal dLE, we start with Eq. (4.5) and first aim for $\hat{\Gamma}$ by calculating

$$\begin{aligned}
\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m^T) &= \mathrm{Cov}(\hat{\boldsymbol{f}}(\boldsymbol{x}_m) - \hat{\Gamma}(\boldsymbol{x}_m)\Delta\boldsymbol{x}_m + \hat{\mathcal{K}}(\boldsymbol{x}_m)\boldsymbol{\xi}_m, \Delta\boldsymbol{x}_m^T) \\
&= \hat{\boldsymbol{f}}(\boldsymbol{y}_n)\mathrm{Cov}(\mathbb{1}, \Delta\boldsymbol{x}_m^T) - \hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T) + \hat{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m, \Delta\boldsymbol{x}_m^T) \\
&= -\hat{\boldsymbol{f}}(\boldsymbol{y}_n)0 - \hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T) + \hat{\mathcal{K}}(\boldsymbol{y}_n)0 \\
&= -\hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T)
\end{aligned}$$

where we used that $\boldsymbol{\xi}_m$ represents white noise, i.e., $\mathrm{Cov}(\boldsymbol{\xi}_m, \Delta\boldsymbol{x}_m^T) = 0$ holds. Since we assumed that the covariances are calculated based on a local neighborhood, it is possible to assume that the Langevin fields are the same for all neighboring points, e.g., $\hat{\boldsymbol{f}}(\boldsymbol{x}_m) \approx \hat{\boldsymbol{f}}(\boldsymbol{y}_n)$. This allows us to pull them out of the covariances. In total, we get

$$\hat{\Gamma}(\boldsymbol{y}_n) = -\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m^T)\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T)^{-1}$$

as friction estimate. To get the estimate of $\hat{\boldsymbol{f}}(\boldsymbol{y}_n)$, we calculate

$$\begin{aligned}
\langle\Delta\boldsymbol{x}_{m+1}\rangle &= \hat{\boldsymbol{f}}(\boldsymbol{y}_n)\langle\mathbb{1}\rangle - \hat{\Gamma}(\boldsymbol{y}_n)\langle\Delta\boldsymbol{x}_m\rangle + \hat{\mathcal{K}}(\boldsymbol{y}_n)\langle\boldsymbol{\xi}_m\rangle \\
&= \hat{\boldsymbol{f}}(\boldsymbol{y}_n) - \hat{\Gamma}(\boldsymbol{y}_n)\langle\Delta\boldsymbol{x}_m\rangle + \hat{\mathcal{K}}(\boldsymbol{y}_n)0 \\
&= \hat{\boldsymbol{f}}(\boldsymbol{y}_n) - \hat{\Gamma}(\boldsymbol{y}_n)\langle\Delta\boldsymbol{x}_m\rangle
\end{aligned}$$

and use again the white noise properties of $\boldsymbol{\xi}$, i.e., $\langle\boldsymbol{\xi}_m\rangle = 0$. In consequence we get

$$\hat{\boldsymbol{f}}(\boldsymbol{y}_n) = \langle\Delta\boldsymbol{x}_{m+1}\rangle + \hat{\Gamma}(\boldsymbol{y}_n)\langle\Delta\boldsymbol{x}_m\rangle.$$

where we can insert the estimate of $\hat{\Gamma}(\boldsymbol{y}_n)$ from above. To derive the estimate of the amplitude, we can as first step insert the drift estimate into Eq. (4.5) to get

$$\Delta\boldsymbol{x}_{m+1} = \langle\Delta\boldsymbol{x}_{m+1}\rangle + \hat{\Gamma}(\boldsymbol{x}_m)(\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m) + \hat{\mathcal{K}}(\boldsymbol{x}_m)\boldsymbol{\xi}_m$$

which can be used to calculate

$$\begin{aligned}
\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_{m+1}^T) = \mathrm{Cov}(&\langle\Delta\boldsymbol{x}_{m+1}\rangle + \hat{\Gamma}(\boldsymbol{x}_m)(\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m) + \hat{\mathcal{K}}(\boldsymbol{x}_m)\boldsymbol{\xi}_m, \\
&\langle\Delta\boldsymbol{x}_{m+1}\rangle^T + (\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m)^T\hat{\Gamma}(\boldsymbol{x}_m)^T + \boldsymbol{\xi}_m^T\hat{\mathcal{K}}(\boldsymbol{x}_m)^T) \\
&= \langle\Delta\boldsymbol{x}_{m+1}\rangle\mathrm{Cov}(\mathbb{1},\mathbb{1})\langle\Delta\boldsymbol{x}_{m+1}\rangle^T + \langle\Delta\boldsymbol{x}_{m+1}\rangle\mathrm{Cov}(\mathbb{1}, (\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m)^T)\hat{\Gamma}(\boldsymbol{y}_n)^T \\
&+ \langle\Delta\boldsymbol{x}_{m+1}\rangle\mathrm{Cov}(\mathbb{1}, \boldsymbol{\xi}_m^T)\hat{\mathcal{K}}(\boldsymbol{y}_n)^T + \hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}((\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m), \mathbb{1})\langle\Delta\boldsymbol{x}_{m+1}\rangle^T \\
&+ \hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}((\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m), (\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m)^T)\hat{\Gamma}(\boldsymbol{y}_n)^T \\
&+ \hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}((\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m), \boldsymbol{\xi}_m^T)\hat{\mathcal{K}}(\boldsymbol{y}_n)^T + \hat{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m, \mathbb{1})\langle\Delta\boldsymbol{x}_{m+1}\rangle^T \\
&+ \hat{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m, (\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m)^T)\hat{\Gamma}(\boldsymbol{y}_n)^T + \hat{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m, \boldsymbol{\xi}_m^T)\hat{\mathcal{K}}(\boldsymbol{y}_n)^T
\end{aligned}$$

which looks more complicated than it is. To simplify this expression, we need to consider that all terms including $\mathrm{Cov}(\mathbb{1}, ...)$ and $\mathrm{Cov}(..., \mathbb{1})$ are zero. Additionally, $\boldsymbol{\xi}_m$ is uncorrelated to $(\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m)$ since it is white noise. In consequence, we have

$$\begin{aligned}\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_{m+1}^T) &= \hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}((\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m), (\langle\Delta\boldsymbol{x}_m\rangle - \Delta\boldsymbol{x}_m)^T)\hat{\Gamma}(\boldsymbol{y}_n)^T \\ &+ \hat{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m, \boldsymbol{\xi}_m^T)\hat{\mathcal{K}}(\boldsymbol{y}_n)^T \\ &= \hat{\Gamma}(\boldsymbol{y}_n)(-\mathrm{Cov}(\langle\Delta\boldsymbol{x}_m\rangle, \Delta\boldsymbol{x}_m) + \mathrm{Cov}(\langle\Delta\boldsymbol{x}_m\rangle, \langle\Delta\boldsymbol{x}_m\rangle^T) \\ &- \mathrm{Cov}(\Delta\boldsymbol{x}_m, \langle\Delta\boldsymbol{x}_m\rangle^T) + \mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T))\hat{\Gamma}(\boldsymbol{y}_n)^T \\ &+ \hat{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m, \boldsymbol{\xi}_m^T)\hat{\mathcal{K}}(\boldsymbol{y}_n)^T \\ &= \hat{\Gamma}(\boldsymbol{y}_n)(-\langle\Delta\boldsymbol{x}_m\rangle\mathrm{Cov}(\mathbb{1}, \Delta\boldsymbol{x}_m) + \langle\Delta\boldsymbol{x}_m\rangle\mathrm{Cov}(\mathbb{1}, \mathbb{1})\langle\Delta\boldsymbol{x}_m\rangle^T \\ &- \mathrm{Cov}(\Delta\boldsymbol{x}_m, \mathbb{1})\langle\Delta\boldsymbol{x}_m\rangle^T + \mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T))\hat{\Gamma}(\boldsymbol{y}_n)^T \\ &+ \hat{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m, \boldsymbol{\xi}_m^T)\hat{\mathcal{K}}(\boldsymbol{y}_n)^T \\ &= \hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T))\hat{\Gamma}(\boldsymbol{y}_n)^T + \hat{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m, \boldsymbol{\xi}_m^T)\hat{\mathcal{K}}(\boldsymbol{y}_n)^T\end{aligned}$$

where we used again that $\mathrm{Cov}(\mathbb{1}, ...)$ and $\mathrm{Cov}(..., \mathbb{1})$ are zero. $\mathrm{Cov}(\boldsymbol{\xi}_m, \boldsymbol{\xi}_m^T)$ is by definition $\mathbb{1}$ since $\boldsymbol{\xi}_m$ represents white noise. Hence, we end with

$$\hat{\mathcal{K}}(\boldsymbol{y}_n)\hat{\mathcal{K}}(\boldsymbol{y}_n)^T = \mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_{m+1}^T) - \hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T)\hat{\Gamma}(\boldsymbol{y}_n)^T$$

where we can use a Choleskey decomposition to extract $\hat{\mathcal{K}}(\boldsymbol{y}_n)$. Note that we can use

$$\hat{\Gamma}(\boldsymbol{y}_n)^T = -(\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T)^{-1})^T\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m^T)^T$$

with $(\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T)^{-1})^T = \mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T)^{-1}$ and $\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m^T)^T = \mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_{m+1}^T)$ to get

$$\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T)\hat{\Gamma}(\boldsymbol{y}_n)^T = -\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_{m+1}^T)$$

which means that

$$\hat{\mathcal{K}}(\boldsymbol{y}_n)\hat{\mathcal{K}}(\boldsymbol{y}_n)^T = \mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_{m+1}^T) + \hat{\Gamma}(\boldsymbol{y}_n)\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_{m+1}^T)$$

represents an alternative equation to determine $\hat{\mathcal{K}}(\boldsymbol{y}_n)\hat{\mathcal{K}}(\boldsymbol{y}_n)^T$.

## A.2 Derivation of field estimates for the Verlet-dLE

Starting with Eq. (4.34), we can derive $\tilde{\Gamma}$ via

$$\begin{aligned}\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m^T) &= (\mathbb{1} + \tilde{\Gamma}(\boldsymbol{y}_n))^{-1}\mathrm{Cov}(\tilde{\boldsymbol{f}}(\boldsymbol{x}_m) - (\tilde{\Gamma}(\boldsymbol{x}_m) - \mathbb{1})\Delta\boldsymbol{x}_m + \tilde{\mathcal{K}}(\boldsymbol{x}_m)\boldsymbol{\xi}_n, \Delta\boldsymbol{x}_m^T) \\ &= (\mathbb{1} + \tilde{\Gamma}(\boldsymbol{y}_n))^{-1}(\tilde{\boldsymbol{f}}(\boldsymbol{y}_n)\mathrm{Cov}(\mathbb{1}, \Delta\boldsymbol{x}_m^T) - (\tilde{\Gamma}(\boldsymbol{y}_n) - \mathbb{1})\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T) \\ &+ \tilde{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m, \Delta\boldsymbol{x}_m^T)) \\ &= (\mathbb{1} + \tilde{\Gamma}(\boldsymbol{y}_n))^{-1}(\tilde{\boldsymbol{f}}(\boldsymbol{y}_n)0 - (\tilde{\Gamma}(\boldsymbol{y}_n) - \mathbb{1})\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T) + \tilde{\mathcal{K}}(\boldsymbol{y}_n)0) \\ &= -(\mathbb{1} + \tilde{\Gamma}(\boldsymbol{y}_n))^{-1}(\tilde{\Gamma}(\boldsymbol{y}_n) - \mathbb{1})\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T)\end{aligned}$$

where we used that $\boldsymbol{\xi}_m$ represents white noise, i.e., $\mathrm{Cov}(\boldsymbol{\xi}_m, \Delta\boldsymbol{x}_m^T) = 0$ holds. Hence, we get

$$\tilde{\Gamma}(\boldsymbol{y}_n) = (\mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T) - \mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m^T))(\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1}, \Delta\boldsymbol{x}_m^T) + \mathrm{Cov}(\Delta\boldsymbol{x}_m, \Delta\boldsymbol{x}_m^T))^{-1}$$

To derive the drift estimate we calculate

$$\begin{aligned}
\langle \Delta\boldsymbol{x}_{m+1}\rangle &= (\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^{-1}(\tilde{\boldsymbol{f}}(\boldsymbol{y}_n)\langle\mathbb{1}\rangle - (\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})\langle\Delta\boldsymbol{x}_m\rangle + \tilde{\mathcal{K}}(\boldsymbol{y}_n)\langle\boldsymbol{\xi}_m\rangle)\\
&= (\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^{-1}(\tilde{\boldsymbol{f}}(\boldsymbol{y}_n) - (\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})\langle\Delta\boldsymbol{x}_m\rangle + \tilde{\mathcal{K}}(\boldsymbol{y}_n)0)\\
&= (\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^{-1}(\tilde{\boldsymbol{f}}(\boldsymbol{y}_n) - (\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})\langle\Delta\boldsymbol{x}_m\rangle)
\end{aligned}$$

and exploit again the properties of the white noise, i.e., $\langle\boldsymbol{\xi}_m\rangle = 0$. This leads to

$$\tilde{\boldsymbol{f}}(\boldsymbol{y}_n) = (\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))\langle\Delta\boldsymbol{x}_{m+1}\rangle + (\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})\langle\Delta\boldsymbol{x}_m\rangle$$

The noise amplitude needs, just as for the Euler-dLE, the most complicated calculation.

$$\begin{aligned}
\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1},\Delta\boldsymbol{x}_{m+1}^T) &= (\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^{-1}\mathrm{Cov}(\tilde{\boldsymbol{f}}(\boldsymbol{x}_m) - (\tilde{\Gamma}(\boldsymbol{x}_m)-\mathbb{1})\Delta\boldsymbol{x}_m + \tilde{\mathcal{K}}(\boldsymbol{x}_m)\boldsymbol{\xi}_m,\\
&\quad \tilde{\boldsymbol{f}}(\boldsymbol{x}_m)^T - \Delta\boldsymbol{x}_m^T(\tilde{\Gamma}(\boldsymbol{x}_m)-\mathbb{1})^T + \boldsymbol{\xi}_m^T\tilde{\mathcal{K}}(\boldsymbol{x}_m)^T)((\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^{-1})^T\\
&= (\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^{-1}(\tilde{\boldsymbol{f}}(\boldsymbol{y}_n)\mathrm{Cov}(\mathbb{1},\mathbb{1})\tilde{\boldsymbol{f}}(\boldsymbol{y}_n)^T - \tilde{\boldsymbol{f}}(\boldsymbol{y}_n)\mathrm{Cov}(\mathbb{1},\Delta\boldsymbol{x}_m^T)(\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})^T\\
&\quad + \tilde{\boldsymbol{f}}(\boldsymbol{y}_n)\mathrm{Cov}(\mathbb{1},\boldsymbol{\xi}_m^T)\tilde{\mathcal{K}}(\boldsymbol{y}_n)^T - (\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})\mathrm{Cov}(\Delta\boldsymbol{x}_m,\mathbb{1})\tilde{\boldsymbol{f}}(\boldsymbol{y}_n)^T\\
&\quad + (\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})\mathrm{Cov}(\Delta\boldsymbol{x}_m,\Delta\boldsymbol{x}_m^T)(\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})^T - (\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})\mathrm{Cov}(\Delta\boldsymbol{x}_m,\boldsymbol{\xi}_m^T)\tilde{\mathcal{K}}(\boldsymbol{y}_n)^T\\
&\quad + \tilde{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m,\mathbb{1})\tilde{\boldsymbol{f}}(\boldsymbol{y}_n)^T - \tilde{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m,\Delta\boldsymbol{x}_m^T)(\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})^T\\
&\quad + \tilde{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m,\boldsymbol{\xi}_m^T)\tilde{\mathcal{K}}(\boldsymbol{y}_n)^T)((\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^{-\mathbb{1}})^T\\
&= (\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^{-1}(\tilde{\boldsymbol{f}}(\boldsymbol{y}_n)0\tilde{\boldsymbol{f}}(\boldsymbol{y}_n)^T - \tilde{\boldsymbol{f}}(\boldsymbol{y}_n)0(\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})^T\\
&\quad + \tilde{\boldsymbol{f}}(\boldsymbol{y}_n)0\tilde{\mathcal{K}}(\boldsymbol{y}_n)^T - (\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})0\tilde{\boldsymbol{f}}(\boldsymbol{y}_n)^T\\
&\quad + (\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})\mathrm{Cov}(\Delta\boldsymbol{x}_m,\Delta\boldsymbol{x}_m^T)(\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})^T - (\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})0\tilde{\mathcal{K}}(\boldsymbol{y}_n)^T\\
&\quad + \tilde{\mathcal{K}}(\boldsymbol{y}_n)0\tilde{\boldsymbol{f}}(\boldsymbol{y}_n)^T - \tilde{\mathcal{K}}(\boldsymbol{y}_n)0(\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})^T\\
&\quad + \tilde{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m,\boldsymbol{\xi}_m^T)\tilde{\mathcal{K}}(\boldsymbol{y}_n)^T)((\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^{-1})^T\\
&= (\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^{-1}((\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})\mathrm{Cov}(\Delta\boldsymbol{x}_m,\Delta\boldsymbol{x}_m^T)(\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})^T\\
&\quad + \tilde{\mathcal{K}}(\boldsymbol{y}_n)\mathrm{Cov}(\boldsymbol{\xi}_m,\boldsymbol{\xi}_m^T)\tilde{\mathcal{K}}(\boldsymbol{y}_n)^T)((\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^{-1})^T
\end{aligned}$$

Now, we can insert the width of the white noise $\mathrm{Cov}(\boldsymbol{\xi}_m,\boldsymbol{\xi}_m^T) = 1$ which leads to

$$\begin{aligned}
\tilde{\mathcal{K}}(\boldsymbol{y}_n)\tilde{\mathcal{K}}(\boldsymbol{y}_n)^T =&(\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))\mathrm{Cov}(\Delta\boldsymbol{x}_{m+1},\Delta\boldsymbol{x}_{m+1}^T)(\mathbb{1}+\tilde{\Gamma}(\boldsymbol{y}_n))^T\\
&- (\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})\mathrm{Cov}(\Delta\boldsymbol{x}_m,\Delta\boldsymbol{x}_m^T)(\tilde{\Gamma}(\boldsymbol{y}_n)-\mathbb{1})^T
\end{aligned}$$

The Choleskey decomposition can be used to get $\tilde{\mathcal{K}}(\boldsymbol{y}_n)$. We note that it is possible to reconstruct the noise found in the data by the Verlet-dLE via

$$\boldsymbol{\xi}_m = \tilde{\mathcal{K}}(\boldsymbol{x}_m)^{-1}((\mathbb{1}+\tilde{\Gamma}(\boldsymbol{x}_m))\Delta\boldsymbol{x}_{m+1} - \tilde{\boldsymbol{f}}(\boldsymbol{x}_m) + (\tilde{\Gamma}(\boldsymbol{x}_m)-\mathbb{1})\Delta\boldsymbol{x}_m)$$

## A.3 Transformation of units

In the following, we will specify how to transform the unit system used in the dLE framework to another one. First, we need to write down the Langevin equation

$$\ddot{x} = \frac{k_\mathrm{B}T}{\mathcal{M}}\frac{d\ln P(x)}{dx} - \frac{\Gamma}{\mathcal{M}}\dot{x} + \frac{\sqrt{2k_\mathrm{B}T\Gamma}}{\mathcal{M}}\xi$$

where we inserted $F(x) = -k_\mathrm{B}T\ln P(x)$. The quantities which have to be independent of the used unit system are $\ddot{x}$, $\dot{x}$, $d\ln P(x)/dx$ and $\xi$. Consequently, the three terms on the right side of the equation need to be independent of the unit system as well.
Now, the conversion of units can be defined by the value of $k_\mathrm{B}T$. Assuming that $k_{B,1}T$ defines the unit system we are starting in and $k_{B,2}T$ the unit system we want to transform to, the factor

$$a = \frac{k_{B,2}T}{k_{B,1}T},$$

can be used to specify the unit conversion by

$$\mathcal{M}_2 = a \cdot \mathcal{M}_1$$

$$\Gamma_2 = a \cdot \Gamma_1$$

since this multiplications enforce the conservation of the three force terms in the Langevin equation.
As example, we can use as starting unit system the dLE convention with $k_{B,1}T = 38$ ps$^{-1}$ at $T = 300$ K, $\mathcal{M}_1 = 1$ ps and $\Gamma_1 = 100$ which we want to transform to $k_{B,2}T = 4.494$ kJ/mol at $T = 300$ K. By using the equations above, we get $\mathcal{M}_2 = 0.0656$ g/mol and $\Gamma_2 = 6.56$ g/mol/ps in the new unit system.

Note that both unit systems considered in this example measure the time in ps. If the time should be treated in units of, e.g., ns, all parameters which include the time need to be transformed. This might depend on the used unit convention.

## A.4 Details of MD simulations

### A.4.1 MD simulation of NaCl

The GROMACS 4.6.7. force field was used [184]. 895 TIP3P water molecules [185] were placed in a cubic box of 3 mm side length together with one Na$^+$ and one Cl$^-$ ion. The two ions were described by the Amber99 ion parameters [186] and the simulation was integrated at a time step of 1 fs. The water constrained bonds and angles were determined by the SETTLE algorithm [187], the van-der Waals interaction and particle mesh Ewald [188] real space was cut off at a distance of 1 nm. The Parinello-Rahman barostat [189] with isotropic pressure coupling was used to regulate the pressure to 1 bar. The coupling time was 0.5 ps and compressibility $4.5 \cdot 10^{-5}$ bar. The reference temperature of 293.15 K was preserved by the Bussi velocity rescaling thermostat [76] with a coupling time of 0.2 ps. Before the simulation started, an equilibration of 1 ns was done. Afterwards, a trajectory of 200 ns with a write out frequency of 100 frames/ps was produced. Additionally, the simulation was extended to 1 $\mu$s to get a reliable reference but here the saving time step was 1 frame/ps.

### A.4.2 MD simulation of AIB$_9$

The simulations were performed using the GROMACS program suite with the GROMOS96 43a1 force field [16] and explicit chloroform solvent [190]. A leapfrog algorithm with a time step of 2 fs was used to perform the integration. Additionally, the particle-mesh Ewald method for electrostatics with a minimal cutoff of 1.4 nm was applied [188] and the bonds were constrained by LINCS [191], including a hydrogen atom. The Bussi velocity rescaling thermostat [76] was used to preserve the temperature.

### A.4.3 MD simulation of T4 lysozyme

The GROMACS package (version 4.6.7) [192] was used to perform the simulation employing the Amber ff99sb*-ILDN force field [186, 193, 194] and TIP3P water [185]. Following Hub and de Groot [24], the M6I mutant of T4L (PDB 150L [195] chain D) was used and the residues 163 and 164 were omitted as they are not resolved in the crystal structure. Based on a triclinic box with a NaCl salt concentration of 150 mmol $L^{-1}$, roughly 29400 atoms had to be considered in MD. The LINCS algorithm [196] constrained the bonds, including hydrogen, which enabled the usage of an integration time step of 2 fs in the Verlet integrator. Electrostatic interactions were treated by Particle-Mesh Ewald (PME) summation [197]. The cut-offs of neighbor search, Lennard-Jones forces and the real space grid of PME were set to 0.12 nm. For preparation, a steepest descent energy minimization of T4L was performed in vacuo to remove sterically unfavorable interactions. Afterwards, the solvation box was built and a second energy minimization was done taking the solvent into account. The equilibration started with a 100 ps NVT run with position restraints and the Bussi thermostat [76] at 300 K, followed by a 1 ns NPT run using position restraints and the Berendsen barostat [77]. A 5 ns free NPT simulation was done afterwards, the last 4 ns were used to calculate the averaged box volume. Then, a 10 ns free NVT simulation was performed. Finally, the equilibrium simulation was performed with a write-out frequency of 1 frame/ps.

### A.4.4 MD simulation of trypsin and Hsp90

The Amber99SB* force field [186, 193], was used to describe protein and ion interactions. Water molecules were described with the TIP3P model [185]. Simulations were carried out using GROMACS v2018 [147] in a CPU/GPU hybrid implementation. Protein protonation states were determined with propka [198]. Van der Waals interactions were calculated with a cut-off of 1 nm, electrostatic interactions used the particle mesh Ewald method [188] with a minimal real-space cut-off of 1 nm. All covalent bonds with hydrogen atoms were constrained using LINCS [191]. After an initial steepest descent minimization with positional restraints of protein and ligand heavy atoms, an initial 0.1 ns equilibration MD simulation in the NPT ensemble was performed using a time step of 1 fs and positional restraints of protein and ligand heavy atoms. A temperature of 290.15 K was kept constant by the Bussi (v-rescale) thermostat [76], the coupling time constant was set to 0.2 ps. The pressure was kept constant at 1 bar with the Berendsen barostat [77], the coupling time constant was 0.5 ps. The equilibration was followed by a second steepest descent minimization without restraints and a short 0.1 ns equilibration MD simulation in the NPT ensemble.

The dcTMD calculations [42] were performed using the PULL code implemented in Gromacs employing the "constraint" option with a SHAKE implementation [199]. 200-400 statistically independent start points of simulations were obtained by generating different atomic velocity distributions after the 10 ns unbiased simulations, all corresponding to a temperature of 290.15 K. After a preequilibration of 0.1 ns using parameters as described above with positional restraints on protein and ligand heavy atoms and a constant distance constraint of all simulation systems, constant velocity calculations were performed with $v_c = 1$ m/s covering a distance of 2 nm. Hereby, the barostat was switched to the Parrinello-Rahman barostat [189]. The constraint pseudo-force $f_c$ was written out each time step.

**Trypsin-benzamidine**:
Benzamidine parameters were derived using AnteChamber [200] and ACPYPE [201] with atomic parameters deduced from GAFF parameters [202]. Atomic charges were obtained as RESP charges [203] based on QM calculations at the HF/6-31G* level using ORCA [204] and Multiwfn [205]. Trypsin (PDB ID 3PTB) [156] was put into a dodecahedral box with side lengths of 7.5, 7.5 and 5.3 nm and it was solvated with 8971 water molecules. 16 sodium and 25 chloride ions were added to obtain a charge neutral box with a salt concentration of 0.1 M [155]. After the initial equilibration, an additional 10.0 ns unbiased MD simulation was added to get a converged protein structure. As pulling coordinates, the distance between the center of mass of all benzamidine heavy atoms and the one of the $C_\alpha$ atoms of the central $\beta$-sheet of trypsin was used.

**Hsp90-inhibitor**:
The parameters of the resorcinol inhibitor were taken from Ref. [162]. Here, inhibitor parameters were generated using AnteChamber [200] and ACPYPE [201] with atomic parameters derived from GAFF parameters [202] and AM1-BCC atomic charges [206, 207]. The solvated simulation boxes of the Hsp90-inhibitor complex are the same as in [162] (compound **1j**), which are based on the 2.5 Å X-ray crystal structure with PDB ID 6FCJ [208]. Just as for trypsin, the distance between the center of mass of all ligand heavy atoms and the one of the C$\alpha$ atoms of the central $\beta$-sheet of Hsp90 served as pulling coordinate.

## A.5 T-boosting: Uncertainty of rate prediction

To get an idea of the uncertainties associated with $T$-boosting, we consider the waiting time $t_{\text{wait}}$ of some process of interest. We assume that $t_{\text{wait}}$ is exponentially distributed,

$$P(t_{\text{wait}}) = \frac{1}{\langle t_{\text{wait}} \rangle} e^{-\frac{t_{\text{wait}}}{\langle t_{\text{wait}} \rangle}}, \tag{A.1}$$

where $\langle t_{\text{wait}} \rangle$ is a function of temperature $T$. In consequence, the expectation value $\bar{t}_{\text{wait}}$ is given by the mean of the distribution $\langle t_{\text{wait}} \rangle$ plus/minus the error of the mean

$$\bar{t}_{\text{wait}}(T) = \langle t_{\text{wait}}(T) \rangle \pm \frac{\langle t_{\text{wait}}(T) \rangle}{\sqrt{N(T)}}, \tag{A.2}$$

where $N(T)$ denotes the number of recorded events.
By going to the dimensionless rate $k = t_0 / \langle t_{\text{wait}} \rangle$ (where $t_0$ represents some time scale,

e.g., ns) and after using Gaussian error propagation to lowest order [209], we get

$$\ln\left(\bar{k}(T)\right) = \ln\left(k(T)\right) \pm \frac{1}{\sqrt{N(T)}}. \tag{A.3}$$

Considering that the rate expression behaves like $k \propto e^{-\Delta F/k_\mathrm{B}T}$ (with transition barrier $\Delta F$), $\ln(k)$ depends linearly on $1/T$, i.e.,

$$\ln(k(T)) = \frac{a}{T} + b. \tag{A.4}$$

Linear regression theory [209] provides estimates for $a$ and $b$ as well as uncertainties

$$\sigma_b = \sqrt{\frac{\sum_i \frac{N(T_i)}{T_i^2}}{\Delta}} \tag{A.5}$$

and

$$\sigma_a = \sqrt{\frac{\sum_i N(T_i)}{\Delta}} \tag{A.6}$$

with

$$\Delta = \sum_i N(T_i) \sum_i \frac{N(T_i)}{T_i^2} - \left(\sum_i \frac{N(T_i)}{T_i}\right)^2, \tag{A.7}$$

where $T_i$ denotes a discrete set of temperatures at which simulations are performed. Using error propagation, we can calculate the uncertainty of $\ln(k)$ at $T_\mathrm{ref} = 300$ K via

$$\sigma_{\ln(k(T_\mathrm{ref}))} = \sqrt{\left(\frac{\sigma_a}{T_\mathrm{ref}}\right)^2 + \sigma_b^2}. \tag{A.8}$$

This indicates for the error of the average waiting time

$$\bar{t}_\mathrm{wait}(T_\mathrm{ref}) = \langle t_\mathrm{wait}(T_\mathrm{ref})\rangle \pm \langle t_\mathrm{wait}(T_\mathrm{ref})\rangle \cdot \sigma_{\ln(k(T_\mathrm{ref}))}. \tag{A.9}$$

To get an idea of the quantitative scale of this uncertainty, we can imagine that we performed 10 Langevin simulations of length $t_\mathrm{LE}$ at different $T_i$ ($i = 0, \dots, 9$)

$$T_i = T_0 + i\left(25\frac{T_0}{T_\mathrm{ref}}\right)\mathrm{K}. \tag{A.10}$$

The first three temperatures ($T_0$, $T_1$ and $T_2$) are chosen such that we observe $\approx 10^2$ transitions during the simulation time $t_\mathrm{LE}$. Additionally, we assume to collect $10^3$ transitions at the subsequent three temperatures $T_3$, $T_4$ and $T_5$ as well as $10^4$ transitions for $T_6$, $T_7$ and $T_8$. For $T_9$ we assume that we observed $10^5$ transitions.
Considering the case of $T_0 = T_\mathrm{ref} = 300$ K, the observed rate at 300 K is $k = 10^2/t_\mathrm{LE}$ to fulfill our assumptions above. Choosing $t_\mathrm{LE} = 5$ ms, this results in $k(300\mathrm{K}) = 1/50\mu\mathrm{s}$. We get as the error of the rate $\sigma_{\ln(k(T_\mathrm{ref}))} = 7.7\%$.
Alternatively, assuming that we need to set $T_0 \approx 450$ K in order to achieve $10^2$ transitions, employing the boosting relation Eq. (3.43) and $t_\mathrm{LE} = 5$ ms, the observed rate at 300 K is $k(300\mathrm{K}) = 0.063\ \mathrm{ms}^{-1}$ with an error of 10.6%.

Considering the Langevin simulations of trypsin described in Sec. 6.2, where we use $T$-boosting at 13 temperatures from $T \in [380, 900]$ K, the error at 300 K can be estimated to be $\sigma_{\ln(k(T_{\mathrm{ref}}))} = 3.3\%$. For Hsp90, were we reach time scales of tens of seconds, we obtain $\sigma_{\ln(k(T_{\mathrm{ref}}))} = 11.0\%$ at 300 K based on Langevin simulations at 14 temperatures from $T \in [700, 1350]$ K.

As it is known that constraints lead to an overestimation of the friction [143] and considering that erroneous free energy estimations enter Eq. (6.11) in the exponent, we can conclude that the extrapolation error due to $T$-boosting can easily be made negligible in comparison to theses error sources.

## A.6 Langevin modeling of NaCl

The following figure shows the distribution of the back-calculated noise $\xi$ for different value ranges of $x$ at different time steps.

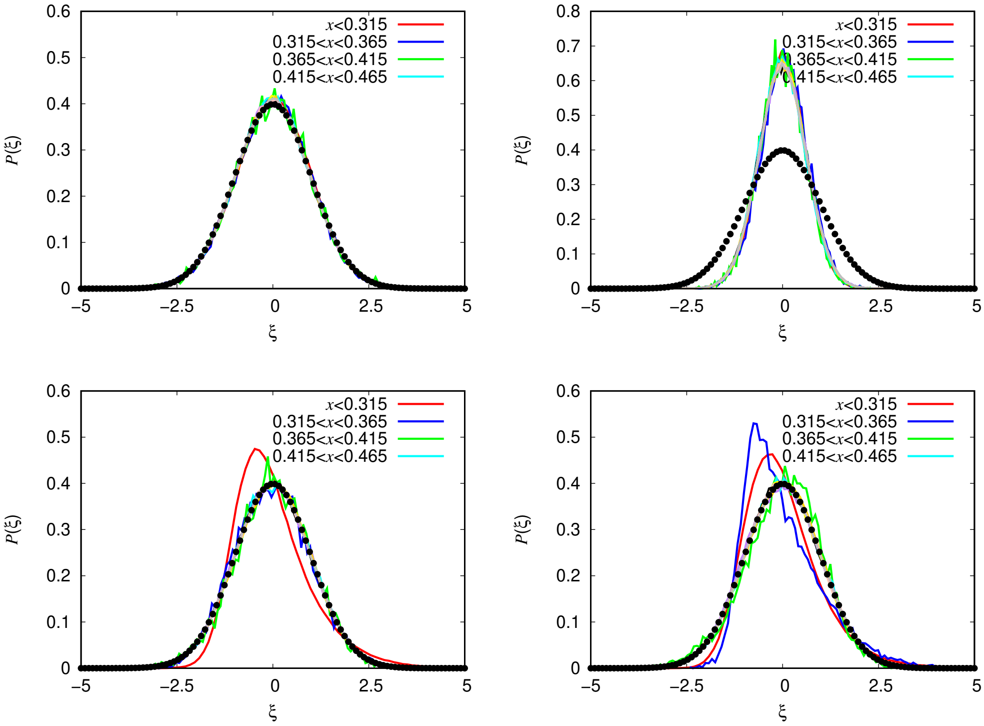

Figure A.1: **Noise distribution for different time steps.** We see top left that the expected Gaussian shape (black dots) of the noise distribution is found for the dLE at $\delta t = 10$ fs. The different colors represent noise binnings at different value ranges of $x$. We do not see any dependence on $x$. Top right, it is shown that the noise determined by the rescaled dLE deviates from the standard distribution. The noise at $\delta t = 60$ fs, shown bottom left, reveals deviations from the expected distribution for small values of $x$, which goes in line with the problems in reproducing the correct free energy via dLE at this $\delta t$. The deviations increase at $\delta t = 100$ fs.

## A.7 Field estimates for AIB

The following figure shows the friction estimate $\Gamma_{11}$ at different time steps.

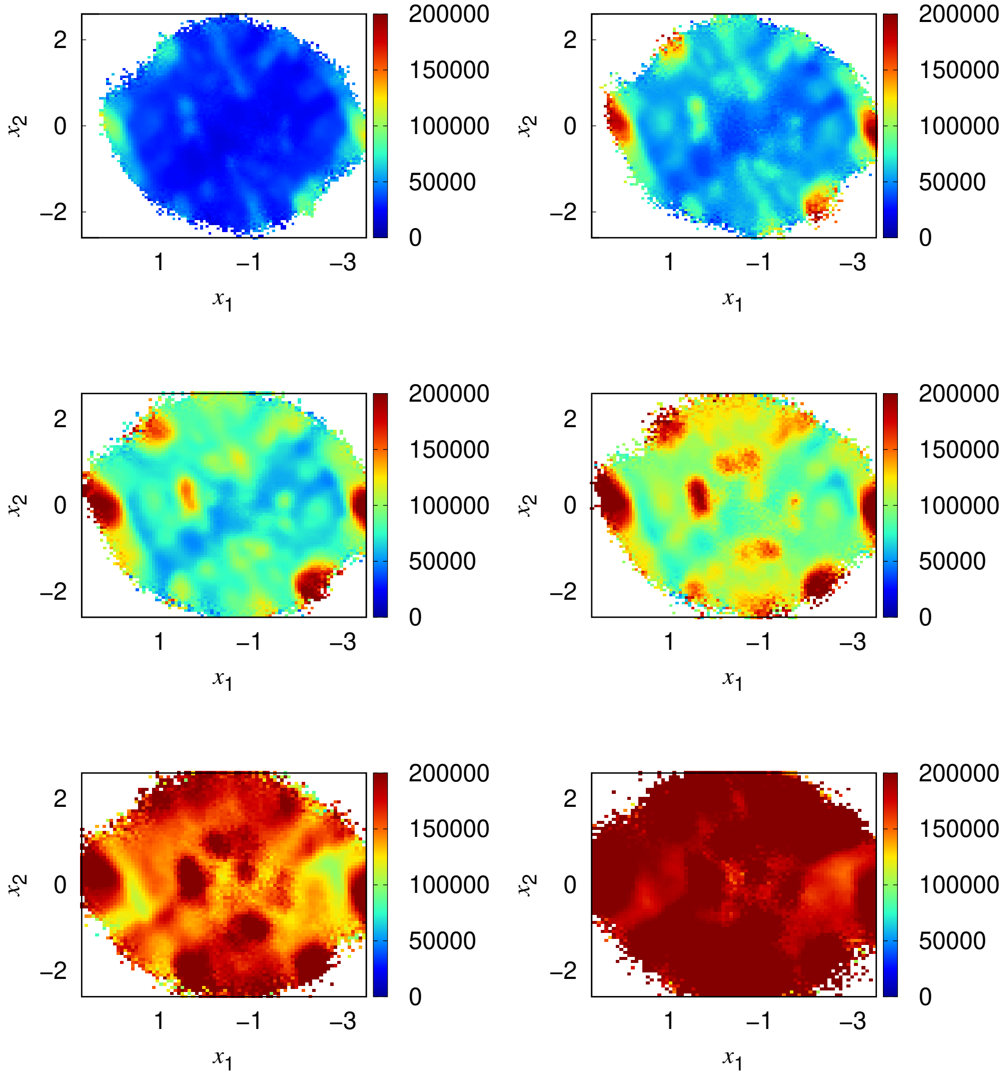

Figure A.2: **Friction estimate. $\boldsymbol{\Gamma_{11}}$ binned in the $\boldsymbol{x_1}$-$\boldsymbol{x_2}$ plane for different time steps.** Shown are $\delta t = 2$ ps (top left), $\delta t = 6$ ps (top right), $\delta t = 10$ ps (middle left), $\delta t = 20$ ps (middle right), $\delta t = 50$ ps (bottom left) and $\delta t = 100$ ps (bottom right).

The following figure shows the friction estimate $\Gamma_{12}$ at different time steps.

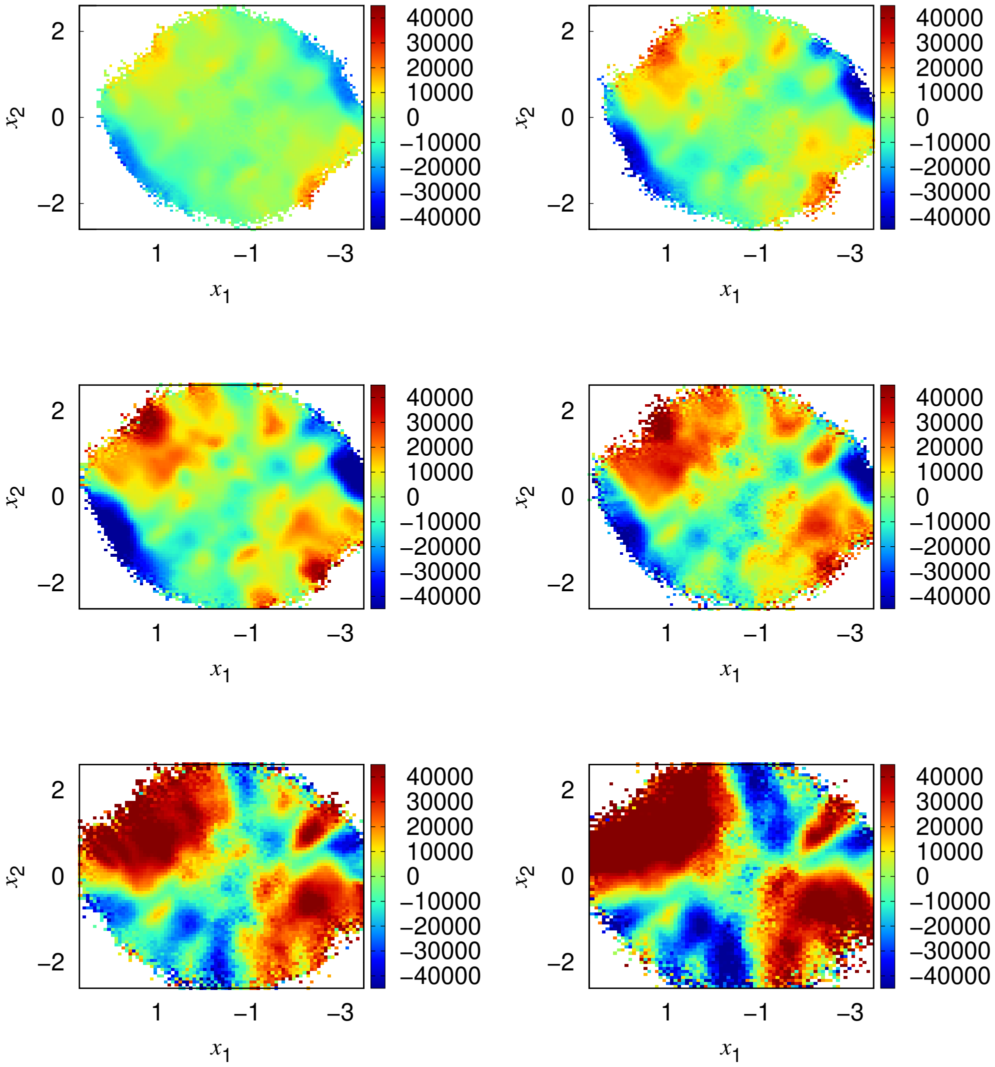

Figure A.3: **Friction estimate. $\boldsymbol{\Gamma_{12}}$ binned in the $\boldsymbol{x_1}$-$\boldsymbol{x_2}$ plane for different time steps.** Shown are $\delta t = 2$ ps (top left), $\delta t = 6$ ps (top right), $\delta t = 10$ ps (middle left), $\delta t = 20$ ps (middle right), $\delta t = 50$ ps (bottom left) and $\delta t = 100$ ps (bottom right).

The following figure shows the noise field estimate $\mathcal{K}_{11}$ at different time steps.

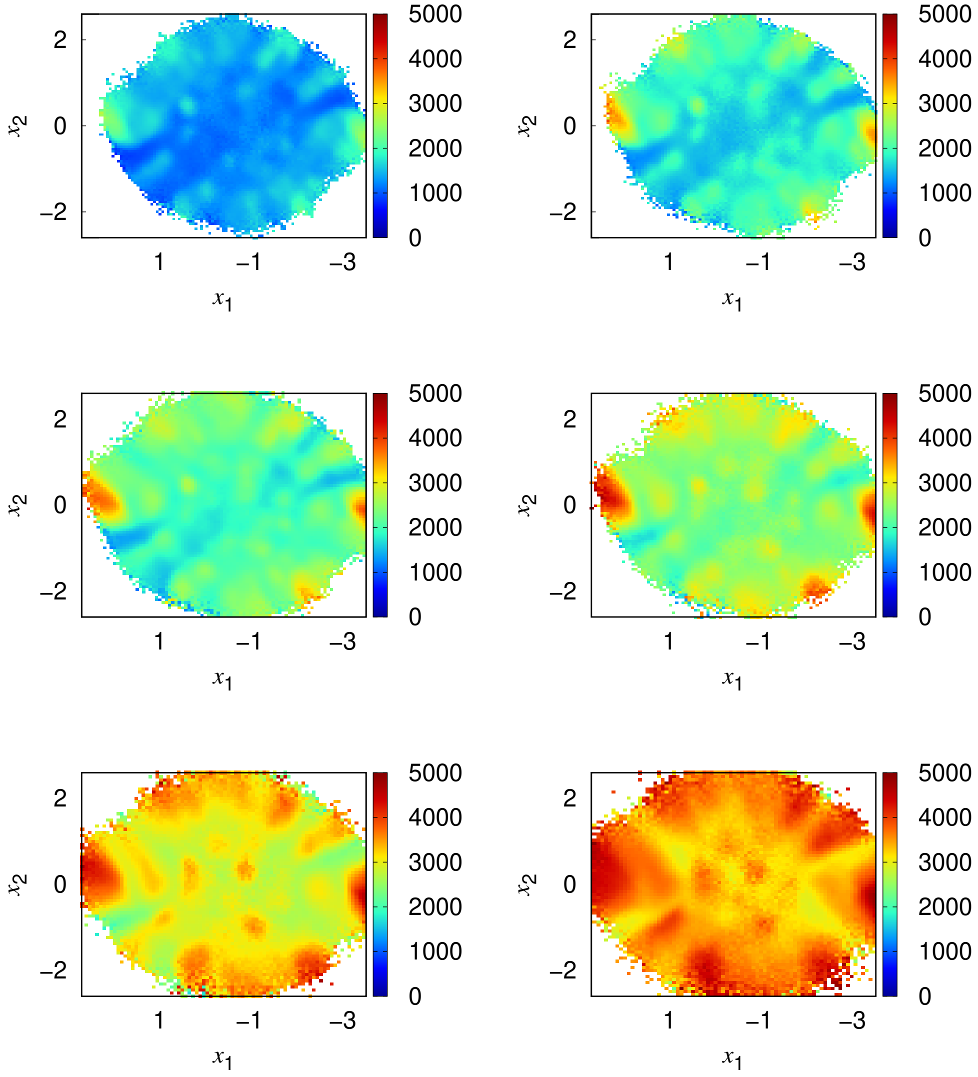

Figure A.4: **Noise field estimate. $\mathcal{K}_{11}$ binned in the $x_1$-$x_2$ plane for different time steps.** Shown are $\delta t = 2$ ps (top left), $\delta t = 6$ ps (top right), $\delta t = 10$ ps (middle left), $\delta t = 20$ ps (middle right), $\delta t = 50$ ps (bottom left) and $\delta t = 100$ ps (bottom right).

The following figure shows the noise field estimate $\mathcal{K}_{12}$ at different time steps.

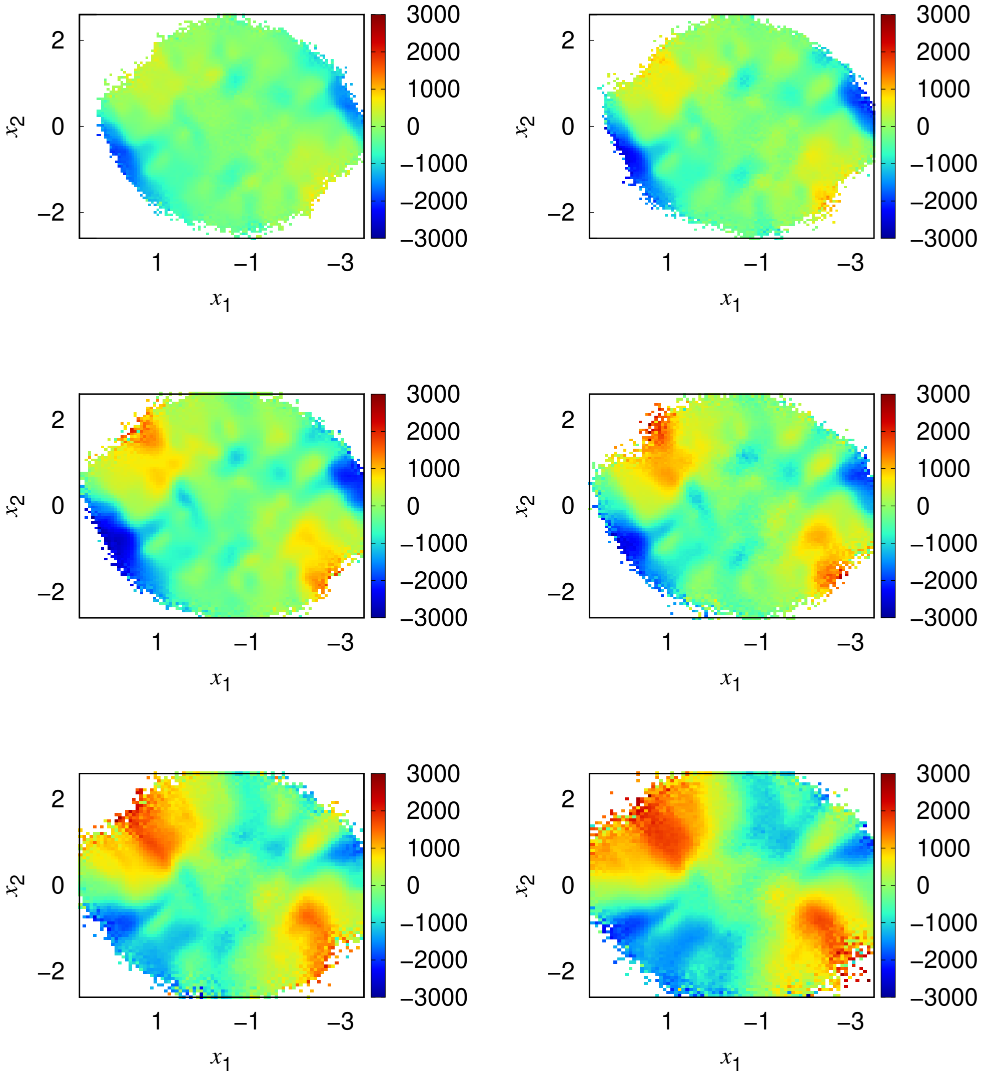

Figure A.5: **Noise field estimate. $\mathcal{K}_{12}$ binned in the $x_1$-$x_2$ plane for different time steps.** Shown are $\delta t = 2$ ps (top left), $\delta t = 6$ ps (top right), $\delta t = 10$ ps (middle left), $\delta t = 20$ ps (middle right), $\delta t = 50$ ps (bottom left) and $\delta t = 100$ ps (bottom right).

## A.8 Assignment dLE to states

Biswas et al. [73] provided a density-based clustering for the subset of $7.35 \cdot 10^6$ points. We want to use this information to get a finer insight into the dynamics estimated by the dLE, i.e., we want to consider single transitions like $rrrrr \leftrightarrow lllll$. To this end, we have to assign the dLE trajectories to the various states. This was done in the following way.

- The system space is binned into multiple coarse bins, each coarse bin is again partitioned in smaller bins.
- For each coarse bin, the reference trajectory is used to determine the dominate state per fine bin in the following way:
  - If one state has the most entries in the fine bin, it gets this bin.
  - If two states have the same number of entries, the bin is assigned to the "barrier", i.e., the bin is not assigned to a special state but to a intermediate region.
- If the coarse bin gets not a single entry at all, it is completely assigned to the "barrier"
- In case there are entries in the coarse bin, empty fine bins are assigned in the following way:
  - Each empty fine bin checks whether one of his direct neighbors is already assigned. If this is the case, it is assigned to this state or the "barrier".
  - If two direct neighbors are assigned to different states, the empty bin is assigned to the "barrier"
  - After having checked all empty bins, the checking is repeated by using the bins assigned in the last round as well.
  - This checking is repeated until all empty bins were assigned.
- Finally, the trajectory which should be assigned to states gets a state trajectory by using the assignment of the fine bins.

Out of the 102 states found by the clustering [73], only the first 42 states were taken, the rest of the states was assigned to the "barrier". This was done to be sure that the considered states have a reasonable sampling. By partitioning the space in $4^5 = 1024$ coarse bins and each coarse bin in $15^5 = 759375$ finer bins, it was possible to assign 87 % of the dLE points to states. For the binned dLE, 87 % of the dLE points were assigned to states as well.

## A.9 Additional autocorrelations of the rescaled dLE for Aib$_9$

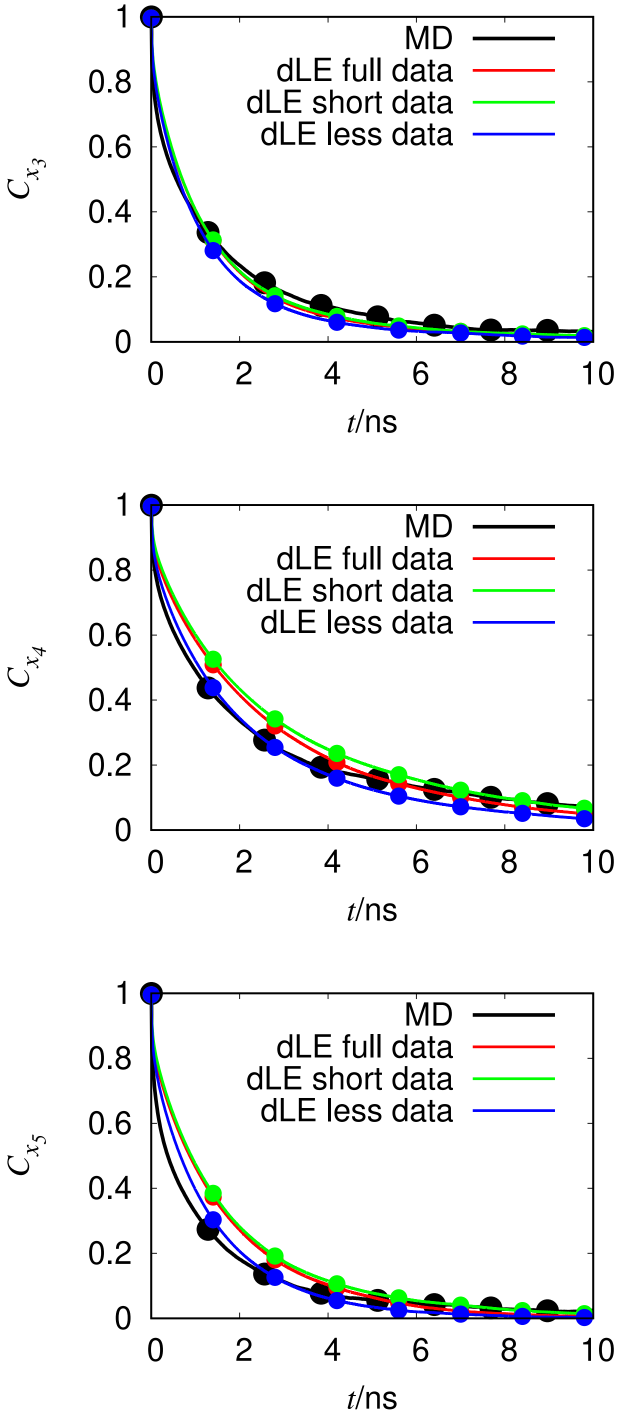

Figure A.6: **Autocorrelations of remaining coordinates.** The dLE estimates match the MD for all three coordinates $x_3$ (top left), $x_4$ (top right) and $x_5$ (bottom).

## A.10 T4 lysoyzyme: recrossing study for coordinates from contact PCA

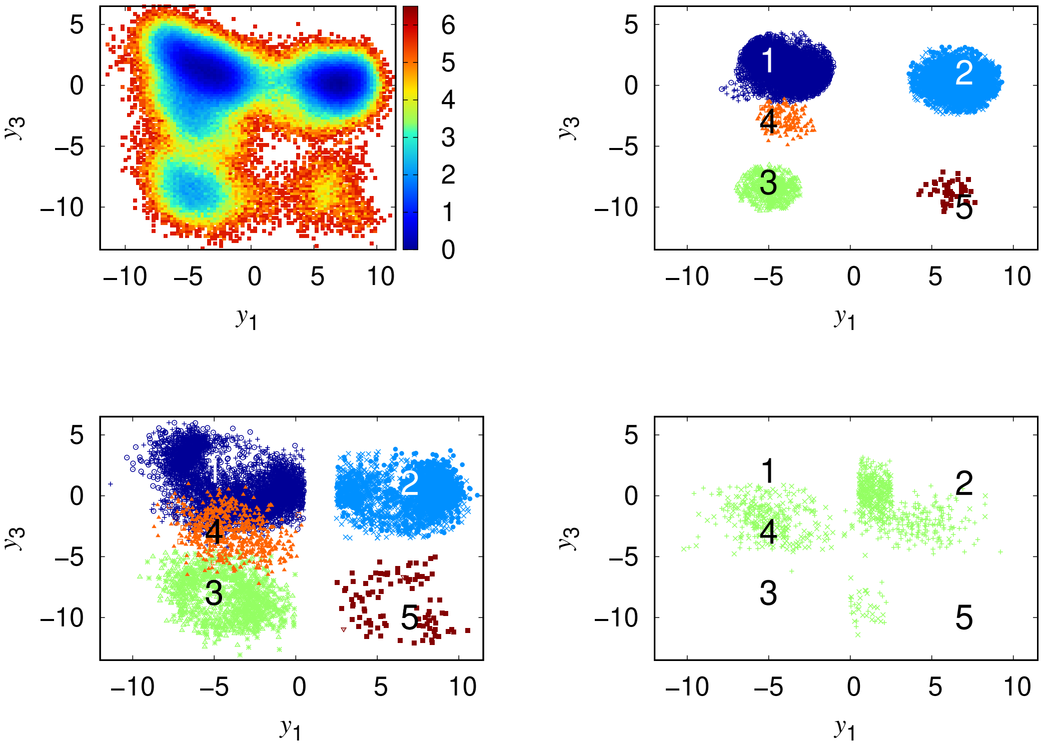

Figure A.7: **Definition of states, state surroundings and barrier.** Top right, we see the free energy projected on the two PCs $y_1$ and $y_3$. The other three figures are related to the counting of recrossings. Top right, the state cores are shown, bottom left the state surroundings and bottom right the barrier region.

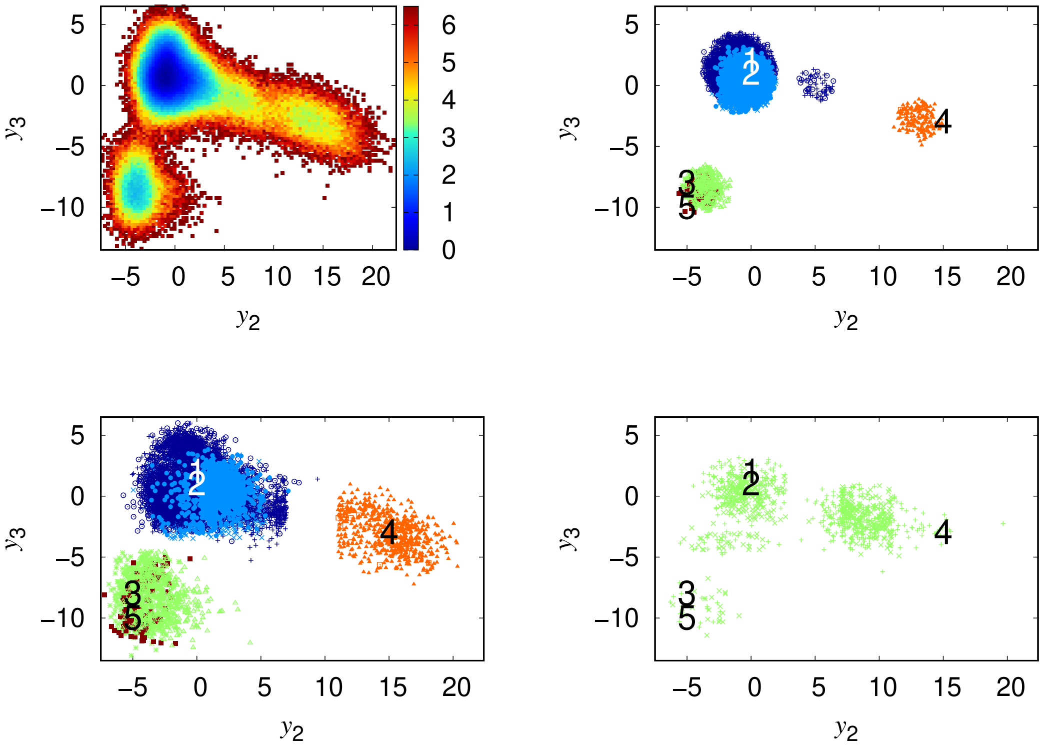

Figure A.8: **Definition of states, state surroundings and barrier.** Top right, we see the free energy projected on the two PCs $y_2$ and $y_3$. The other three figures are related to the counting of recrossings. Top right, the state cores are shown, bottom left the state surroundings and bottom right the barrier region.

## A.11 T4 lysoyzyme: Chapman-Kolomogorov test

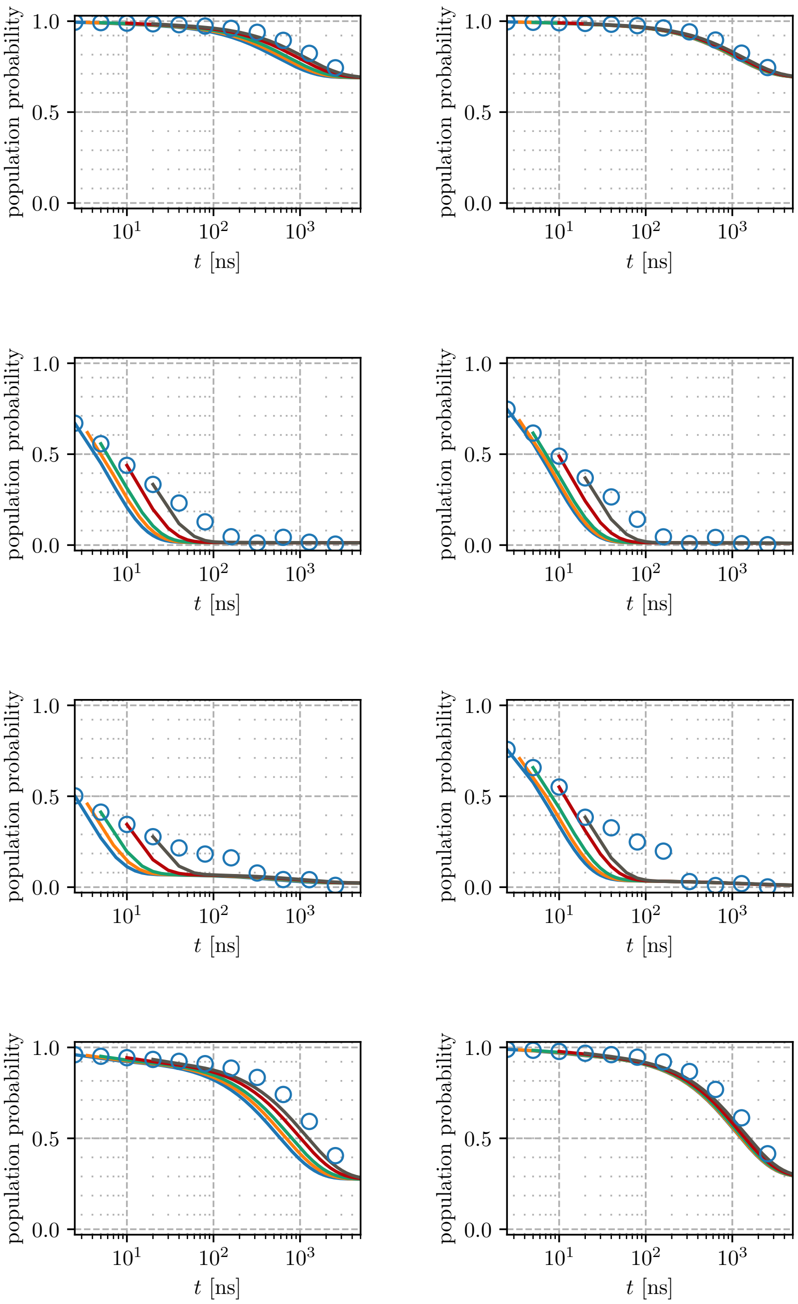

Figure A.9: **Chapman-Kolmogorov test for the MSM on the two-dimensional system description.** On the left we see the uncored and on the right the cored data. The first line shows state 1, the second line state 2, the third line state 3 and the fourth line state 4.

## A.12 T4 lysozyme: field estimates for the rescaled dLE model

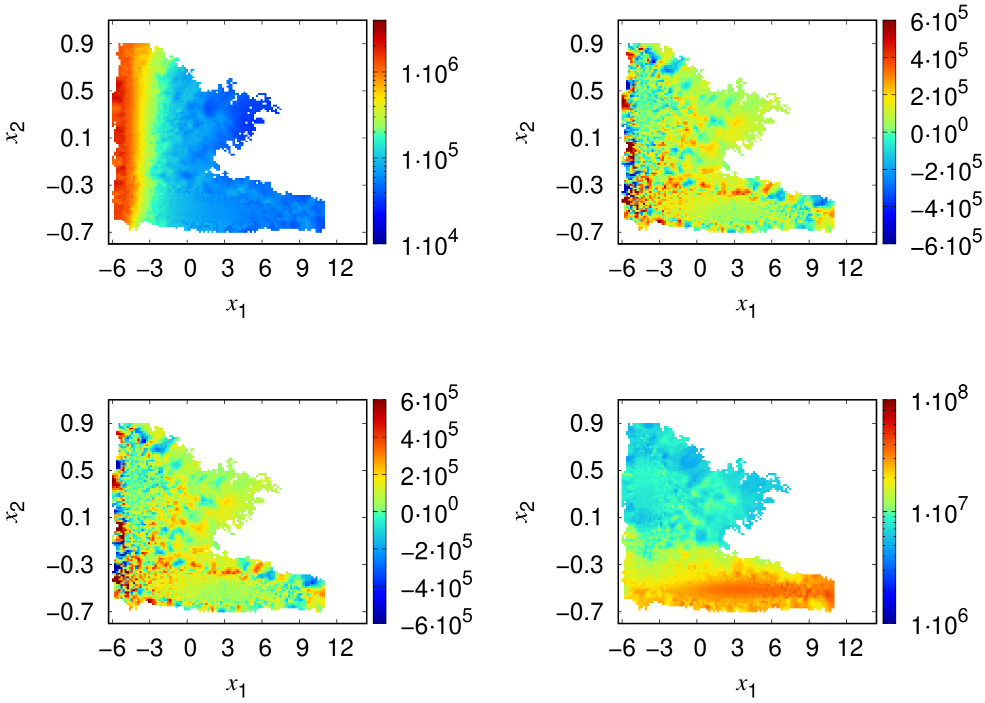

Figure A.10: **Friction estimate of the rescaled dLE.** Shown are $\Gamma_{11}$ (top left), $\Gamma_{21}$ (top right), $\Gamma_{21}$ (bottom left) and $\Gamma_{22}$ (bottom right) estimated by the rescaled dLE.

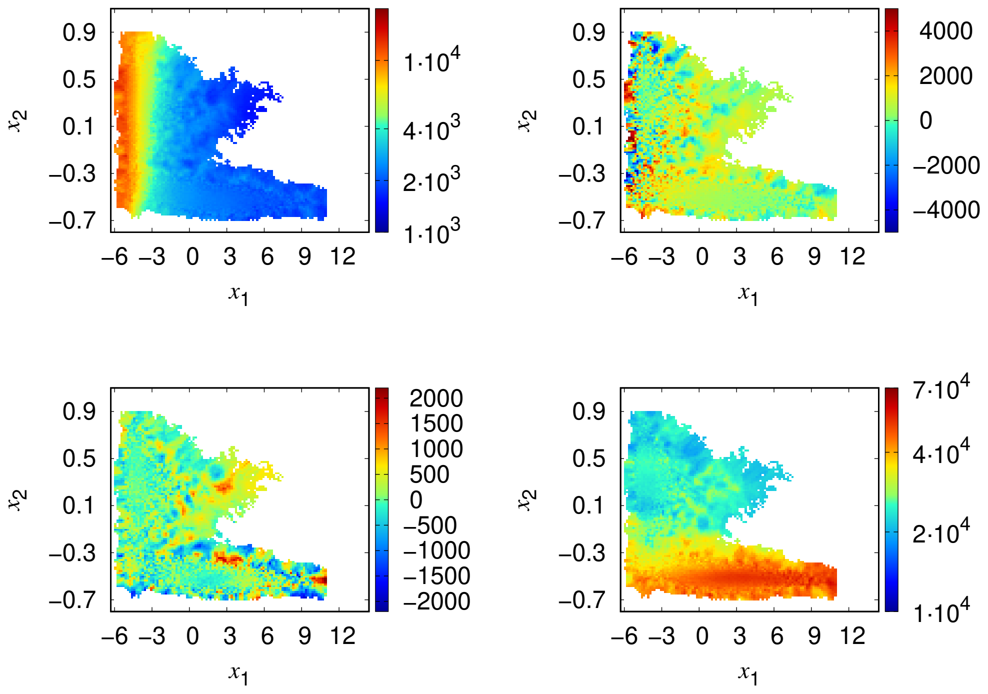

Figure A.11: **Estimates of the noise amplitude of the rescaled dLE.** Shown are $\mathcal{K}_{11}$ (top left), $\mathcal{K}_{21}$ (top right), $\mathcal{K}_{21}$ (bottom left) and $\mathcal{K}_{22}$ (bottom right) estimated by the rescaled dLE.

## A.13 dLE trajectories for the enforced dissociation of sodium chloride

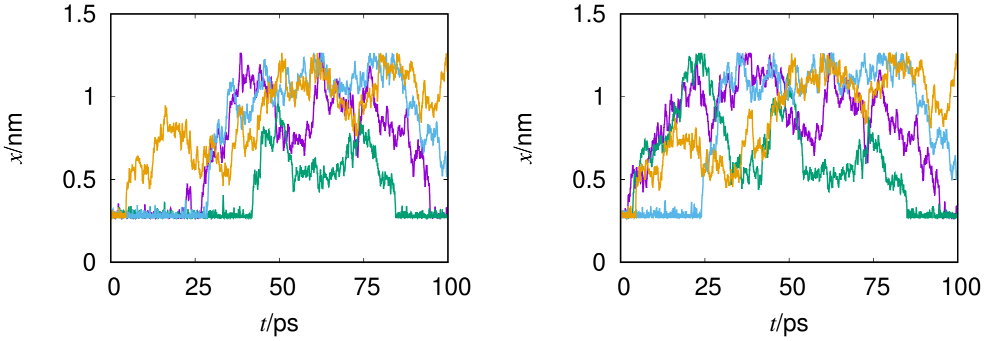

Figure A.12: **dLE trajectories.** Exemplary dLE trajectories for $C = 100$ kJ/(mol nm$^2$)) (left) and $C = 1000$ kJ/(mol nm$^2$)) (right) are shown. Here, the standard dLE was applied to the MD data.

## A.14 Parameters of per-averaging for the hierarchical model system

The following parameters were used to pre-average the data for the hierarchical model system in Sec. 7.2.1:

- $s = 10^3$ coarse bins
- $N_{\max} = 10^4$ defines the variability of the adaptive averaging
- $\omega_{\min} = 10^{-4}$ defines the highest resolution used mainly in the minima
- $\omega_{\max} = 10^{-3}$ defines the minimal resolution used mainly on the barriers

## A.15 Noise test for nucleation of hard spheres

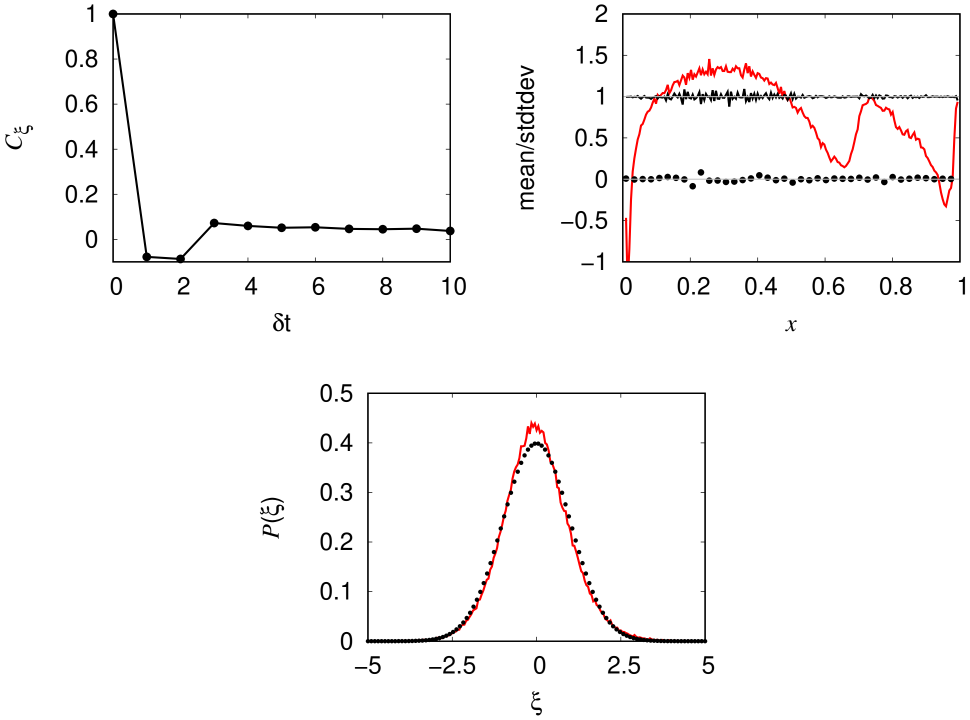

Figure A.13: **Noise test for dLE model of hard sphere nucleation.** Shown are the noise autocorrelation (top left), the estimated averages (black dots) and standard deviations (black lines) of the noise (top right) compared to the expected values given by grey lines and the noise distribution estimated by the dLE model in red compared to the expectation in black (bottom).

# Bibliography

[1] C. Tanford and J. Reynolds. *Natures Robots: A history of Proteins.* Oxford University Press, 2001.

[2] J. C. Whisstock and A. M. Lesk. “Prediction of protein function from protein sequence and structure”. In: *Q. Rev. Biophys.* 36, 307 (2003).

[3] A. Platt, H. C. Ross, S. Hankin and R. J. Reece. “The insertion of two amino acids into a transcriptional inducer converts it into a galactokinase”. In: *Proc. Natl. Acad. Sci. USA* 97, 3154 (2000).

[4] H. Frauenfelder, S. Sligar, and P. Wolynes. “The energy landscapes and motions of proteins”. In: *Science* 254, 1598 (1991).

[5] J. N. Onuchic, Z. Luthey-Schulten and P. G. Wolynes. “Theory of protein folding: The energy landscape perspective”. In: *Ann. Rev. Phys. Chem.* 48, 545 (1997).

[6] K. A. Dill and H. S. Chan. “From levinthal to pathways to funnels: The ”New View” of protein folding kinetics”. In: *Nat. Struct. Biol.* 4, 10 (1997).

[7] D. J. Wales. *Energy Landscapes.* Cambridge University Press, 2003.

[8] K. Henzler-Wildman and D. Kern. “Dynamic personalities of proteins”. In: *Nature* 450, 964 (2007).

[9] J. C. Kendrew, G. Bodo, H. M. Dintzis, R. G. Parrish, H. Wyckoff and D. C. Phillips. “A three-dimensional model of the myoglobin molecule obtained by X-ray analysis”. In: *Nature* 181, 662 (1958).

[10] K. Wuthrich. “The way to NMR structures of proteins”. In: *Nat. Struct. Mol. Biol.* 8, 923 (2001).

[11] H. Dietz and M. Rief. “Exploring the energy landscape of gfp by single-molecule mechanical experiments”. In: *Proc. Natl. Acad. Sci. USA* 101, 16192 (2004).

[12] D. Frenkel and B. Smit. *Understanding Molecular Simulations.* Academic, San Diego, 2002.

[13] D. C. Rapaport. *The Art of Molecular Dynamics Simulation.* Cambridge university press, 2004.

[14] H. J. C. Berendsen. *Simulating the Physical World.* Cambridge University Press, 2007.

[15] R. Salomon-Ferrer, D.A. Case and R.C. Walker. “An overview of the Amber biomolecular simulation package”. In: *WIREs Comput. Mol. Sci.* 3, 198 (2013).

[16] W. F. van Gunsteren, S. R. Billeter, A. A. Eising, P. H. Hünenberger, P. Krüger, A. E. Mark, W. R. P. Scott and I. G. Tironi. *Biomolecular Simulation: The GROMOS96 Manual and User Guide.* Vdf Hochschulverlag AG an der ETH Zürich, Zürich, 1996.

[17] M. J. Abraham, D. van der Spoel, E. Lindahl, B. Hess and the GROMACS development team. *GROMACS User Manual version 2018*. www.gromacs.org. 2018.

[18] Y. Sugita and Y. Okamoto. “Replica-exchange molecular dynamics method for protein folding”. In: *Chem. Phy. Lett.* 314, 141 (1999).

[19] A. Laio and M. Parrinello. “Escaping free-energy minima”. In: *Proc. Natl. Acad. Sci. USA* 99, 12562 (2002).

[20] C. Chipot and A. Pohorille. *Free Energy Calculations*. Springer, 2007.

[21] J. Kästner. “Umbrella sampling”. In: *Wiley Interdiscip. Rev. Comput. Mol. Sci.* 1, 932 (2011).

[22] R. B. Best and G. Hummer. “Reaction coordinates and rates from transition paths”. In: *Proc. Natl. Acad. Sci. USA* 102, 6732 (2005).

[23] O. F. Lange and H. Grubmüller. “Generalized correlation for biomolecular dynamics”. In: *Proteins* 62, 1053 (2006).

[24] J. S. Hub and B. L. de Groot. “Detection of functional modes in protein dynamics”. In: *PLoS Comput. Biol.* 5, e1000480 (2009).

[25] M. A. Rohrdanz, W. Zheng and C. Clementi. “Discovering mountain passes via torchlight: Methods for the definition of reaction coordinates and pathways in complex macromolecular reactions”. In: *Annu. Rev. Phys. Chem* 64, 295 (2013).

[26] B. Peters. “Reaction coordinates and mechanistic hypothesis tests”. In: *Annu. Rev. Phys. Chem.* 67, 669 (2016).

[27] F. Sittel and G. Stock. “Perspective: Identification of collective coordinates and metastable states of protein dynamics”. In: *J. Chem. Phys.* 149, 150901 (2018).

[28] P. Hänggi, P. Talkner and M. Borkovec. “Reaction-rate theory: Fifty years after Kramers”. In: *Rev. Mod. Phys.* 62, 251 (1990).

[29] H. Grabert, P. Hänggi and P. P. Talkner. “Microdynamics and nonlinear stochastic processes of gross variables”. In: *J. Stat. Phys.* 22, 537 (1980).

[30] R. Kubo, M. Toda and N. Hashitsume. *Statistical Physics II. Nonequilibrium Statistical Mechanics*. Springer, 1985.

[31] R. Zwanzig. *Nonequilibrium Statistical Mechanics*. Oxford University Press, 2001.

[32] T. Schilling. “Coarse-grained modelling out of equilibrium”. In: *Phys. Rep.* (to be published) (2021).

[33] N. Pottier. *Nonequilibrium Statistical Physics*. Oxford University Press, 2010.

[34] R. B. Best and G. Hummer. “Coordinate-dependent diffusion in protein folding”. In: *Proc. Natl. Acad. Sci. USA* 107, 1088 (2010).

[35] J. C. F. Schulz, L. Schmidt, R. B. Best, J. Dzubiella and R. R. Netz. “Peptide chain dynamics in light and heavy water: Zooming in on internal friction”. In: *J. Am. Chem. Soc.* 134, 6273 (2012).

[36] J. E. Straub, M. Borkovec and B. J. Berne. “Calculation of dynamic friction on intramolecular degrees of freedom”. In: *J. Phys. Chem.* 91, 4995 (1987).

[37] R. B. Best and G. Hummer. “Diffusive model of protein folding dynamics with Kramers turnover in rate”. In: *Phys. Rev. Lett.* 96, 228104 (2006).

[38] O. F. Lange and H. Grubmüller. “Collective Langevin dynamics of conformational motions in proteins”. In: *J. Chem. Phys.* 124, 214903 (2006).

[39] I. Horenko, C. Hartmann, C. Schütte and F. Noè. “Data-based parameter estimation of generalized multidimensional Langevin processes”. In: *Phys. Rev. E* 76, 016706 (2007).

[40] C. Micheletti, G. Bussi and A. Laio. “Optimal Langevin modeling of out-of-equilibrium molecular dynamics simulations”. In: *J. Chem. Phys.* 129, 074105 (2008).

[41] J. A. Morrone, J. Li and B. J. Berne. “Interplay between hydrodynamics and the free energy surface in the assembly of nanoscale hydrophobes”. In: *J. Phys. Chem. B* 116, 378 (2012).

[42] S. Wolf and G. Stock. “Targeted molecular dynamics calculations of free energy profiles using a nonequilibrium friction correction”. In: *J. Chem. Theory Comput.* 14, 6175 (2018).

[43] N. Schaudinnus, A. J. Rzepiela, R. Hegger and G. Stock. “Data driven Langevin modeling of biomolecular dynamics”. In: *J. Chem. Phys.* 138, 204106 (2013).

[44] N. Schaudinnus, B. Bastian, R. Hegger and G. Stock. “Multidimensional Langevin modeling of nonoverdamped dynamics”. In: *Phys. Rev. Lett.* 115, 050602 (2015).

[45] N. Schaudinnus, B. Lickert, M. Biswas and G. Stock. “Global Langevin model of multidimensional biomolecular dynamics”. In: *J. Chem. Phys.* 145, 184114 (2016).

[46] J. D. Chodera and W. C. Swope. “Obtaining longtime protein folding dynamics from short-time molecular dynamics simulations”. In: *Multiscale Model. Simul.* 5, 1214 (2006).

[47] N.-V. Buchete and G. Hummer. “Coarse master equations for peptide folding dynamics”. In: *J. Phys. Chem. B* 112, 6057 (2008).

[48] G. R. Bowman, K. A. Beauchamp, G. Boxer and V. S. Pande. “Progress and challenges in the automated construction of Markov state models for full protein systems”. In: *J. Chem. Phys.* 131, 124101 (2009).

[49] V. S. Pande, K. Beauchamp and G. R. Bowman. “Everything you wanted to know about Markov state models but were afraid to ask”. In: *Methods* 52, 99 (2010).

[50] J.-H. Prinz, H. Wu, M. Sarich, B. Keller, M. Senne, B. Held, J. D. Chodera, C. Schütte and F. Noè. “Markov models of molecular kinetics: Generation and validation”. In: *J. Chem. Phys.* 134, 174105 (2011).

[51] K. A. Beauchamp, G. R. Bowman, T. J. Lane, L. Maibaum, I. S. Haque and V. S. Pande. “Msmbuilder2: Modeling conformational dynamics on the picosecond to millisecond scale”. In: *J. Chem. Theory Comput.* 7, 3412 (2011).

[52] G. R. Bowman, V. S. Pande and F. Noè. *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation.* Springer Science & Business Media, 2013.

[53] W. Wei, C. Siqin, Z. Lizhe and H. Xuhui. “Constructing Markov state models to elucidate the functional conformational changes of complex biomolecules”. In: *Wiley Interdiscip. Rev. Comput. Mol. Sci.* 8, e1343 (2017).

[54] B. E. Husic and V. S. Pande. “Markov state models: From an art to a science”. In: *J. Am. Chem. Soc.* 140, 2386 (2018).

[55] F. Noè and E. Rosta. “Markov models of molecular kinetics”. In: *J. Chem. Phys.* 151, 190401 (2019).

[56] M. Ester, H.-P. Kriegel, J. Sander and X. Xu. *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise.* AAAI Press, 1996.

[57] B. Keller, X. Daura, and W. F. van Gunsteren. “Comparing geometric and kinetic cluster algorithms for molecular simulation data”. In: *J. Chem. Phys.* 132, 074110 (2010).

[58] F. K. Sheong, D.-A. Silva, L. Meng, Y. Zhao and X. Huang. “Automatic state partitioning for multibody systems (APM): An efficient algorithm for constructing Markov state models to elucidate conformational dynamics of multibody systems”. In: *J. Chem. Theory Comput.* 11, 17 (2015).

[59] A. Rodriguez and A. Laio. “Clustering by fast search and find of density peaks”. In: *Science* 344, 1492 (2014).

[60] F. Sittel and G. Stock. “Robust density-based clustering to identify metastable conformational states of proteins”. In: *J. Chem. Theory Comput.* 12, 2426 (2016).

[61] S. Liu, L. Zhu, F. K. Sheong, W. Wang and X. Huang. “Adaptive partitioning by local density-peaks: An efficient density-based clustering algorithm for analyzing molecular dynamics trajectories”. In: *J. Comput. Chem.* 38, 152 (2017).

[62] S. V. Krivov, S. Muff, A. Caflisch and M. Karplus. “One-dimensional barrier-preserving free-energy projections of a $\beta$-sheet miniprotein: New insights into the folding process”. In: *J. Phys. Chem. B* 112, 8701 (2008).

[63] F. Rao and M. Karplus. “Protein dynamics investigated by inherent structure analysis”. In: *Proc. Natl. Acad. Sci. USA* 107, 9152 (2010).

[64] A. Jain and G. Stock. “Identifying metastable states of folding proteins”. In: *J. Chem. Theory Comput.* 8, 3810 (2012).

[65] A. Jain and G. Stock. “Hierarchical folding free energy landscape of HP35 revealed by most probable path clustering”. In: *J. Phys. Chem. B* 118, 7750 (2014).

[66] D. Nagel, A. Weber, B. Lickert and G. Stock. “Dynamical coring of Markov state models”. In: *J. Chem. Phys* 150, 094111 (2019).

[67] S. J. Marrink, A. H. de Vries and A. E. Mark. “Coarse grained model for semiquantitative lipid simulations”. In: *J. Phys. Chem. B* 108, 750 (2004).

[68] P. K. Depa and J. K. Maranas. “Dynamical evolution in coarse-grained molecular dynamics simulations of polyethylene melts”. In: *J. Chem. Phys* 126, 054903 (2007).

[69] D. Fritz, K. Koschke, V. A. Harmandaris, N. F. A. van der Vegt and K. Kremer. “Multiscale modeling of soft matter: Scaling of dynamics”. In: *Phys. Chem. Chem. Phys.* 13, 10412 (2011).

[70] S. Buchenberg, N. Schaudinnus and G. Stock. “Hierarchical biomolecular dynamics: Picosecond hydrogen bonding regulates microsecond conformational transitions”. In: *J. Chem. Theory Comput.* 11, 1330 (2015).

[71] M. Ernst, S. Wolf and G. Stock. “Identification and validation of reaction coordinates describing protein functional motion: Hierarchical dynamics of T4 Lysozyme”. In: *J. Chem. Theory Comput.* 13, 5076 (2017).

[72] S. Brandt, F. Sittel, M. Ernst and G. Stock. “Machine learning of biomolecular reaction coordinates”. In: *J. Phys. Chem. Lett.* 9, 2144 (2018).

[73] M. Biswas, B. Lickert and G. Stock. “Metadynamics enhanced Markov modeling of protein dynamics”. In: *J. Phys. Chem. B* 122, 5508 (2018).

[74] H. Meyer, T. Voigtmann and T. Schilling. “On the non-stationary generalized Langevin equation”. In: *J. Chem. Phys.* 147, 214110 (2017).

[75] H. Meyer, T. Voigtmann and T. Schilling. “On the dynamics of reaction coordinates in classical, time-dependent, many-body processes”. In: *J. Chem. Phys.* 150, 174118 (2019).

[76] G. Bussi, D. Donadio and M. Parrinello. “Canonical sampling through velocity rescaling”. In: *J. Chem. Phys* 126, 014101 (2007).

[77] H. J. Berendsen, J. P. M. Postma, W. F. van Gusteren and J. R. Haak. “Molecular dynamics with coupling to an external bath”. In: *J. Chem. Phys* 81, 3684 (1984).

[78] L. Sawle and K. Ghosh. “Convergence of molecular dynamics simulation of protein native states: Feasibility vs self-consistency dilemma”. In: *J. Chem. Theory Comput.* 12, 861 (2016).

[79] R. Hegger, A. Altis, P. H. Nguyen and G. Stock. “How complex is the dynamics of peptide folding?” In: *Phys. Rev. Lett.* 98, 028102 (2007).

[80] S. Piana and A. Laio. “Advillin folding takes place on a hypersurface of small dimensionality”. In: *Phys. Rev. Lett.* 101, 208101 (2008).

[81] E. Facco and M. d’Errico. “Estimating the intrinsic dimension of datasets by a minimal neighborhood information”. In: *Sci. Rep.* 7, 12140 (2017).

[82] P. Das, M. Moll, H. Stamati, L. E. Kavraki and C. Clementi. “Low-dimensional, free-energy landscapes of protein-folding reactions by nonlinear dimensionality reduction”. In: *Proc. Natl. Acad. Sci. USA.* 103, 9885 (2006).

[83] W. M. Brown, S. Martin, S. N. Pollock, E. A. Coutsias and J.-P. Watson. “Algorithmic dimensionality reduction for molecular structure analysis”. In: *J. Chem. Phys.* 129, 064118 (2008).

[84] M. Ceriotti, G. A. Tribello and M. Parrinello. “Simplifying the representation of complex free-energy landscapes using sketch-map”. In: *Proc. Natl. Acad. Sci. USA* 108, 13023 (2011).

[85] M. Duan, J. Fan, M. Li, L. Han and S. Huo. “Evaluation of dimensionalityreduction methods from peptide folding-unfolding simulations”. In: *J. Chem. Theory Comput.* 9, 2490 (2013).

[86] A. Ma and A. R. Dinner. “Automatic method for identifying reaction coordinates in complex systems”. In: *J. Phys. Chem. B* 109, 6769 (2005).

[87] E. Chiavazzo, R. Covino, R. R. Coifman, C. W. Gear, A. S. Georgiou, G. Hummer and I. G. Kevrekidis. “Intrinsic map dynamics exploration for uncharted effective free-energy landscapes”. In: *Proc. Natl. Acad. Sci. USA.* 114, E5494 (2017).

[88] W. Chen, A. R. Tan and A. L. Ferguson. “Collective variable discovery and enhanced sampling using autoencoders: Innovations in network architecture and error function design”. In: *J. Chem. Phys.* 149, 072312 (2018).

[89] C. Wehmeyer and F. Noè. “Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics”. In: *J. Chem. Phys.* 148, 241703 (2018).

[90] A. Amadei, A. B. M. Linssen and H. J. C. Berendsen. “Essential dynamics of proteins”. In: *Proteins* 17, 412 (1993).

[91] Y. Mu, P. H. Nguyen and G. Stock. “Energy landscape of a small peptide revealed by dihedral angle principal component analysis”. In: *Proteins* 58, 45 (2005).

[92] L. Molgedey and H. G. Schuster. “Separation of a mixture of independent signals using time delayed correlations”. In: *Phys. Rev. Lett.* 72, 3634 (1994).

[93] G. Perez-Hernandez, F. Paul, T. Giorgino, G. De Fabritiis and F. Noè. “Identification of slow molecular order parameters for Markov model construction”. In: *J. Chem. Phys.* 139, 015102 (2013).

[94] C. R. Schwantes and V. S. Pande. “Improvements in Markov state model construction reveal many non-native interactions in the folding of NTL9”. In: *J. Chem. Theory Comput.* 9, 2000 (2013).

[95] F. Sittel, A. Jain and G. Stock. “Principal component analysis of molecular dynamics: On the use of Cartesian vs. internal coordinates”. In: *J. Chem. Phys.* 141, 014111 (2014).

[96] A. Altis, P. H. Nguyen, R. Hegger and G. Stock. “Dihedral angle principal component analysis of molecular dynamics simulations”. In: *J. Chem. Phys.* 126, 244111 (2007).

[97] F. Sittel, T. Filk and G. Stock. “Principal component analysis on a torus: Theory and application to protein dynamics”. In: *J. Chem. Phys.* 147, 244101 (2017).

[98] K. D. Ball, R. S. Berry, R. E. Kunz, F. Y. Li, A. Proykova and D. J. Wales. “From topographies to dynamics on multidimensional potential energy surfaces of atomic clusters”. In: *Science* 271, 966 (1996).

[99] L. J. Curtis, H. G. Berry and J. Bromander. “Analysis of multi-exponential decay curves”. In: *Physica Scripta* 2, 216 (1970).

[100] A. K. Jain. “Data clustering: 50 years beyond k-means”. In: *Pattern Recognit. Lett.* 31, 651 (2010).

[101] A. Rodriguez, M. d’Errico, E. Facco and A. Laio. “Computing the free energy without collective variables”. In: *J. Chem. Theory Comput.* 14, 1206 (2018).

[102] P. Langevin. “On the theory of Brownian Motion”. In: *C. R. Acad. Sci* 146, 530 (1908).

[103] S. Röblitz and M. Weber. “Fuzzy spectral clustering by PCCA+: Application to Markov state models and data classification”. In: *Adv. Data Anal. Classi.* 7, 147 (2013).

[104] G. R. Bowman, L. Meng and X. Huang. “Quantitative comparison of alternative methods for coarse-graining biological networks”. In: *J. Chem. Phys* 139, 121905 (2013).

[105] G. Hummer and A. Szabo. “Optimal dimensionality reduction of multistate kinetic and Markov-state models”. In: *J. Phys. Chem. B* 119, 9029 (2015).

[106] D. Nerukh, C. H. Jensen, and R. C. Glen. “Identifying and correcting non-Markov states in peptide conformational dynamics”. In: *J. Chem. Phys* 132, 084104 (2010).

[107] D. Nerukh. “Non-Markov state model of peptide dynamics”. In: *J. Mol. Liq.* 176, 65 (2012).

[108] P. Koltai, H. Wu, F. Noè and C. Schütte. “Optimal data-driven estimation of generalized Markov state models for non-equilibrium dynamics”. In: *Computation* 6(1), 22 (2018).

[109] H. Wu and F. Noè. “Variational approach for learning Markov processes from time series data”. In: *J. Nonlinear Sci.* 30, 23 (2020).

[110] A. Einstein. “Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen [On the movement of small particles suspended in stationary liquids required by the molecular-kinetic theory of heat]”. In: *Ann. Phys.* 17, 549 (1905).

[111] A. Einstein. “Zur Theorie der Brownschen Bewegung [On the theorie of Brownian motion]”. In: *Ann. Phys.* 19, 371 (1906).

[112] M. von Smoluchowski. “Sur le chemin moyen parcouru par les molécules d’un gaz et surson rapport avec le théorie de la diffusion [On the average path taken by gas molecules and its relation with the theory of diffusion]”. In: *Bulletin International de l’Académie des Sciences de Cracovie* 3, 202 (1906).

[113] G. G. Stokes. “On the effect of internal friction of fluids on the motion of pendulums”. In: *Transactions of the Cambridge Philosophical Society* 9, 8 (1851).

[114] H. Risken. *The Fokker-Plank Equation.* Springer, 1989.

[115] W. Greiner, L. Neise and H. Stöcker. *Thermodynamics and Statistical Mechanics.* 2nd edition. Springer, 1997.

[116] D. T. Gillespie. “The mathematics of Brownian motion and Johnson noise”. In: *Am. J. Phys.* 64, 225 (1995).

[117] A. O. Caldeira and A. J. Leggett. “Path integral approach to quantum Brownian motion”. In: *Physica* 121A, 587 (1983).

[118] D. Chandler. *Introduction to Modern Statistical Mechanics.* Oxford University Press, 1987.

[119] M. H. Cohen. “Classical Langevin dynamics for model Hamiltonians”. In: *phys. stat. sol. (b)* 1, 252 (2003).

[120] C. Hijon, P. Espanol, E. Vanden-Eijnden and R. Delgado-Buscalioni. “Mori-Zwanzig formalism as a practical computational tool”. In: *Faraday Discuss.* 144, 301 (2010).

[121] G. Bussi and M. Parrinello. “Accurate sampling using Langevin dynamics”. In: *Phys. Rev. E* 75, 2289 (2007).

[122] P. Hänggi, F. Marchesoni, and P. Grigolini. “Bistable flow driven by coloured Gaussian noise-a critical study”. In: *Z.Phys.B* 56, 333 (1984).

[123] D. A. Sivak, J. D. Chodera and G. E. Crooks. “Time step rescaling recovers continuous-time dynamical properties for discrete-time Langevin integration of nonequilibrium systems”. In: *J. Phys. Chem. B* 118, 6466 (2014).

[124] J. Fass, D. A. Sivak, G. E. Crooks, K. A. Beauchamp, B. Leimkuhler and J. D. Chodera. “Quantifying configuration-sampling error in Langevin simulations of complex molecular systems”. In: *Entropy* 20, 318 (2018).

[125] S. Wolf, B. Lickert, S. Bray and G. Stock. “Multisecond ligand dissociation dynamics from atomistic simulations”. In: *Nat. Commun.* 11, 2918 (2020).

[126] J. Schlitter, M. Engels and P. Krüger. “Targeted molecular dynamics - A new approach for searching pathways of conformational transitions”. In: *J. Mol. Graph.* 12, 84 (1994).

[127] M. R. Sørensen and A. F. Voter. “Temperature-accelerated dynamics for simulation of infrequent events”. In: *J. Chem. Phys* 112, 9599 (2000).

[128] S. Siegert, R. Friedrich and J. Peinke. “Analysis of data sets of stochastic systems”. In: *Phys. Lett. A* 243, 275 (1998).

[129] J. Timmer. “Parameter estimation in nonlinear stochastic differential equations”. In: *Chaos, Solitons and Fractals* 11, 2571 (2000).

[130] J. Gradisek, S. Siegert, R. Friedrich and I. Grabec. “Analysis of time series from stochastic processes”. In: *Phys. Rev. E* 62, 3146 (2000).

[131] H. Kantz and T. Schreiber. *Nonlinear Time Series Analysis.* Cambridge University Press, 1997.

[132] R. Hegger and G. Stock. “Multidimensional Langevin modeling of biomolecular dynamics”. In: *J. Chem. Phys.* 130, 034106 (2009).

[133] P. Grassberger. “An optimized box-assisted algorithm for fractal dimensions”. In: *Phys. Lett. A* 148, 63 (1990).

[134] R. H. Landau, M. J. Paez and C. C. Bordeianu. *Computational Physics: Problem Solving with Python.* 3rd edition. Wiley-VCH, 2015.

[135] C. Jarzynski. “Nonequilibrium equality for free energy differences”. In: *Phys. Rev. Lett.* 78, 2690 (1997).

[136] S. Vaikuntanathan and C. Jarzynski. “Escorted free energy simulations: Improving convergence by reducing dissipation”. In: *Phys. Rev. Lett.* 100, 190601 (2008).

[137] J. Servantie and P. Gaspard. “Methods of calculation of a friction coefficient: Application to nanotubes”. In: *Phys. Rev. Lett.* 91, 185503 (2003).

[138] M. Post, S. Wolf and G. Stock. “Principal component analysis of nonequilibrium molecular dynamics simulations”. In: *J. Chem. Phys.* 150, 204110 (2019).

[139] P. Tiwary, V. Limongelli, M. Salvalaglio and M. Parrinello. “Kinetics of protein–ligand unbinding: Predicting pathways, rates, and rate-limiting steps”. In: *Proc. Natl. Acad. Sci. USA* 112, E386 (2015).

[140] A. Perez, J. L. MacCallum and K. Dill. “Accelerating molecular simulations of proteins using Bayesian inference on weak information”. In: *Proc. Natl. Acad. Sci. USA* 112, 11846 (2015).

[141] R. G. Mullen, J.-E. Shea and B. Peters. “Transmission coefficients, committors, and solvent coordinates in ion-pair dissociation”. In: *J. Chem. Theory Comput.* 10, 659 (2014).

[142] J. M. Porrà, K.-G. Wang and J. Masoliver. “Generalized Langevin equations: Anomalous diffusion and probability distributions”. In: *Phys. Rev. E* 53, 5872 (1996).

[143] N. Plattner and F. Noè. “Protein conformational plasticity and complex ligand-binding kinetics explored by atomistic simulations and Markov models”. In: *Nat. Commun.* 6, 7653 (2015).

[144] A. Altis, M. Otten, P. H. Nguyen, R. Hegger and G. Stock. “Construction of the free energy landscape of biomolecules via dihedral angle principle component analysis”. In: *J. Chem. Phys.* 128, 245102 (2008).

[145] A. Perez, F. Sittel, G. Stock and K. Dill. “MELD-path efficiently computes conformational transitions, including multiple and diverse paths”. In: *J. Chem. Theory Comput.* 14, 2109 (2018).

[146] D. Nagel, A. Weber and G. Stock. “MSMPathfinder: Identification of pathways in Markov state models”. In: *J. Chem. Theory Comput.* 16, 7874 (2020).

[147] M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess and E. Lindahl. “GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers”. In: *SoftwareX* 12, 19 (2015).

[148] G. A. Tribello, M. Bonomi, D. Branduardi, C. Camilloni and G. Bussi. “PLUMED 2: New feathers for an old bird”. In: *Comp. Phys. Comm.* 185, 604 (2014).

[149] B. Lickert. “Global Langevin model of multidimensional biomolecular dynamics”. Master thesis. University of Freiburg, 2016.

[150] M. Ernst. “Finding reaction coordinates for protein folding and functional motion”. PhD thesis. University of Freiburg, 2018.

[151] B. L. de Groot, S. Hayward, D. M. F. van Aalten, A. Amadei and H. J. C. Berendsen. “Domain motions in Bacteriophage T4 Lysozyme: A comparison between molecular dynamics and crystallographic data”. In: *Proteins* 31, 116 (1998).

[152] P. Jolles and J. Jolles. “What’s new in lysozyme research? Always a model system, today as yesterday.” In: *Molecular and Cellular Biochemistry* 63, 165 (1984).

[153] O. Krichevsky. “T4 lysozyme as a Pac-Man: How fast can it chew?” In: *Biophys. J.* 103, 1414 (2012).

[154] T. S. van Erp. “Efficiency analysis of reaction rate calculation methods using analytical models I: The two-dimensional sharp barrier”. In: *J. Chem. Phys* 125, 174106 (2006).

[155] F. Guillain and D. Thusius. "Use of proflavine as an indicator in temperature-jump studies of the binding of a competitive inhibitor to trypsin". In: *J. Am. Chem. Soc.* 92, 5534 (1970).

[156] M. Marquart, J. Walter, J. Deisenhofer, W. Bode and R. Huber. "The geometry of the reactive site and of the peptide groups in trypsin, trypsinogen and its complexes with inhibitors". In: *Acta Crystallogr. B* 39, 480 (1983).

[157] J. Schiebel, R. Gaspari, T. Wulsdorf, K. Ngo, C. Sohn, T. E. Schrader, A. Cavalli, A. Ostermann, A. Heine and G. Klebe. "Intriguing role of water in protein-ligand binding studied by neutron crystallography on trypsin complexes". In: *Nat. Commun.* 9, 166 (2018).

[158] I. Buch, T. Giorgino and G. De Fabritiis. "Complete reconstruction of an enzyme-inhibitor binding process by molecular dynamics simulations". In: *Proc. Natl. Acad. Sci. USA* 108, 10184 (2011).

[159] I. Teo, C. G. Mayne, K. Schulten and T. Lelievre. "Adaptive multilevel splitting method for molecular dynamics calculation of benzamidine-trypsin dissociation time". In: *J. Chem. Theory Comput.* 12, 2983 (2016).

[160] L. W. Votapka, B. R. Jagger, A. Heyneman and R. E. Amaro. "SEEKR: Simulation enabled estimation of kinetic rates, a computational tool to estimate molecular kinetics and its application to trypsin–benzamidine binding". In: *J. Phys. Chem. B* 121, 3597 (2017).

[161] R. M. Betz and R. O. Dror. "How effectively can adaptive sampling methods capture spontaneous ligand binding?" In: *J. Chem. Theory Comput.* 15, 2053 (2019).

[162] S. Wolf, M. Amaral, M. Lowinski, F. Vallée, D. Musil, J. Güldenhaupt, M. K. Dreyer, J. Bomke, M. Frech, J. Schlitter and K. Gerwert. "Estimation of protein-ligand unbinding kinetics using non-equilibrium targeted molecular dynamics simulations". In: *J. Chem. Inf. Model.* 59, 5135 (2019).

[163] M. Amaral, D. B. Kokh, J. Bomke, A. Wegener, H.-P. Buchstaller, H. M. Eggenweiler, P. Matias, C. Sirrenberg, R. C. Wade and M. Frech. "Protein conformational flexibility modulates kinetics and thermodynamics of drug binding". In: *Nat. Commun.* 8, 2276 (2017).

[164] D. B. Kokh, M. Amaral, J. Bomke, U. Grädler, D. Musil, H.-P. Buchstaller, M. K. Dreyer, M. Frech, M. Lowinski, F. Vallée, M. Bianciotto, A. Rak and R. C. Wade. "Estimation of drug-target residence times by $\tau$-random acceleration molecular dynamics simulations". In: *J. Chem. Theory Comput.* 14, 3859 (2018).

[165] D. A. Schuetz, M. Bernetti, M. Bertazzo, D. Musil, H.-M. Eggenweiler, M. Recanatini, M. Masetti, G. F. Ecker and A. Cavalli. "Predicting residence time and drug unbinding pathway through scaled molecular dynamics". In: *J. Chem. Inf. Model.* 59, 535 (2019).

[166] S. Bray. "Approaches to analyzing protein-ligand dissociation with targeted molecular dynamics". Master thesis. University of Freiburg, 2018.

[167] N. J. Bruce, G. K. Ganotra, D. B. Kokh, S. K. Sadiq and R. C. Wade. “New approaches for computing ligand-receptor binding kinetics”. In: *Curr. Opin. Struct. Biol.* 49, 1 (2018).

[168] R. Hernandez. “The projection of a mechanical system onto the irreversible generalized Langevin equation”. In: *J. Chem. Phys* 111, 7701 (1999).

[169] M. G. McPhie, P. J. Daivis, I. K. Snook, J. Ennis and D. J. Evans. “Generalized Langevin equation for nonequilibrium systems”. In: *Physica A* 299, 412 (2001).

[170] S. Kawai and T. Komatsuzaki. “Derivation of the generalized Langevin equation in nonstationary environments”. In: *J. Chem. Phys.* 134, 114523 (2011).

[171] B. Cui and A. Zaccone. “Generalized Langevin equation and fluctuation-dissipation theorem for particle-bath systems in external oscillating fields”. In: *Phys. Rev. E* 97, 060102 (2018).

[172] R. Zwanzig. “Nonlinear Generalized Langevin Equations”. In: *J. Stat. Phys.* 9, 215 (1973).

[173] J. O. Daldrop, B. G. Kowalik and R. R. Netz. “External potential modifies friction of molecular solutes in water”. In: *Phys. Rev. X* 7, 041065 (2017).

[174] H. Grubmüller, B. Heymann and P. Tavan. “Ligand binding: Molecular mechanics calculation of the streptavidin-biotin rupture force”. In: *Science* 271, 997 (1996).

[175] B. Isralewitz, M. Gao and K. Schulten. “Steered molecular dynamics and mechanical functions of proteins”. In: *Curr. Opin. Struct. Biol.* 11, 224 (2001).

[176] S. Park and K. Schulten. “Calculating potentials of mean force from steered molecular dynamics simulations”. In: *J. Chem. Phys.* 120, 5946 (2004).

[177] H. Meyer, S. Wolf, G. Stock and T. Schilling. “A numerical procedure to evaluate memory effects in non-equilibrium coarse-grained models”. In: *Adv. Theory Simul.* 111, 2000197 (2020).

[178] S. Buchenberg, F. Sittel and G. Stock. “Time-resolved observation of protein allosteric communication”. In: *Proc. Natl. Acad. Sci. USA* 114, E6804 (2017).

[179] O. Bozovic, C. Zanobini, A. Gulzar, B. Jankovic, D. Buhrke, M. Post, S. Wolf, G. Stock and P. Hamm. “Real-time observation of ligand-induced allosteric transitions in a PDZ domain”. In: *Proc. Natl. Acad. Sci. USA* 117, 26031 (2020).

[180] A. Kuhnhold, H. Meyer, G. Amati, P. Pelagejcev and T. Schilling. “Derivation of an exact, nonequilibrium framework for nucleation: Nucleation is a priori neither diffusive nor Markovian”. In: *Phys. Rev. E* 100, 052140 (2019).

[181] H. Meyer, F. Glatzel, W. Wöhler and T. Schilling. “Evaluation of memory effects at phase transitions and during relaxation processes”. In: *Phys. Rev. E* 103, 022102 (2021).

[182] P. R. ten Wolde, M. J. Ruiz-Montero and D. Frenkel. “Numerical evidence for bcc ordering at the surface of a critical fcc nucleus”. In: *Phys. Rev. Lett.* 75, 2714 (1995).

[183] P. N. Pusey, E. Zaccarelli, C. Valeriani, E. Sanz, W. C. K. Poon and M. E. Cates. “Hard spheres: Crystallization and glass formation”. In: *Phil. Trans. R. Soc. A* 367, 4993 (2009).

[184] S. Pronk, S. Pall, R. Schulz, P. Larsson, P. Bjelkmar, R. Apostolov, M. R. Shirts, J. C. Smith, P. M. Kasson, D. van der Spoel, B. Hess and E. Lindahl. "GROMACS 4.5: A high-throughput and highly parallel open source molecular simulation toolkit". In: *Bioinformatics* 29, 845 (2013).

[185] W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey and M. Klein. "Comparison of simple potential functions for simulating liquid water". In: *J. Chem. Phys.* 79, 926 (1983).

[186] V. Hornak, R. Abel, A. Okur, B. Strockbine, A. Roitberg and C. Simmerling. "Comparison of multiple Amber force fields and development of improved protein backbone parameters". In: *Proteins* 65, 712 (2006).

[187] S. Miyamoto and P. A. Kollman. "SETTLE: An analytical version of the SHAKE and RATTLE algorithm for rigid water models". In: *J. Comput. Chem.* 13, 952 (1992).

[188] T. Darden, D. York and L. Petersen. "Particle mesh Ewald: An N log(N) method for Ewald sums in large systems". In: *J. Chem. Phys.* 98, 10089 (1993).

[189] M. Parrinello and A. Rahman. "Polymorphic transitions in single crystals: A new molecular dynamics method". In: *J. App. Phys.* 52, 7182 (1981).

[190] I. G. Tironi and W. F. van Gunsteren. "A molecular dynamics simulation study of chloroform". In: *Mol. Phys.* 83, 381 (1994).

[191] B. Hess, H. Bekker, H. J. C. Berendsen and J. G. E. M. Fraaije. "LINCS: A linear constraint solver for molecular simulations". In: *J. Comp. Chem.* 18, 1463 (1997).

[192] B. Hess, C. Kutzner, D. van der Spoel and E. Lindahl. "GROMACS 4: Algorithms for highly efficient, load-balanced, and scalable molecular simulation". In: *J. Chem. Theory Comput.* 4, 435 (2008).

[193] R. B. Best and G. Hummer. "Optimized molecular dynamics force fields applied to the helix-coil transition of polypeptides". In: *J. Phys. Chem. B* 113, 9004 (2009).

[194] K. Lindorff-Larsen, S. Piana, K. Palmo, P. Maragakis, J. L. Klepeis, R. O. Dror and D. E. Shaw. "Improved side-chain torsion potentials for the Amber ff99SB protein force field". In: *Proteins* 78, 1950 (2010).

[195] X.-J. Zhang and B. Matthews. "Conservation of solvent-binding sites in 10 crystal forms of T4 Lysozyme". In: *Proteins Sci.* 3, 1031 (1994).

[196] B. Hess. "P-LINCS: A parallel linear constraint solver for molecular simulation". In: *J. Chem. Theory Comput.* 4, 116 (2008).

[197] U. Essmann, L. Perera, M. L. Berkowitz, T. Darden and H. Lee. "A smooth particle mesh Ewald method". In: *J. Chem. Phys* 103, 8577 (1995).

[198] M. H. M. Olsson, C. R. Søndergaard, M. Rostkowski and J. H. Jensen. "PROPKA3: Consistent treatment of internal and surface residues in empirical p$K_a$ predictions". In: *J. Chem. Theory Comput.* 7, 525 (2011).

[199] J. P. Ryckaert, G. Ciccotti and H. J. C. Berendsen. "Numerical-integration of cartesian equations of motions of a system with constraints-molecular dynamics of N-alkanes". In: *J. Comput. Phys.* 23, 327 ().

[200] J. Wang and R. Brüschweiler. “2D entropy of discrete molecular ensembles”. In: *J. Chem. Theory Comput.* 2, 18 (2006).

[201] A. W. Sousa da Silva and W. F. Vranken. “ACPYPE - AnteChamber PYthon Parser interfacE”. In: *BMC Res. Notes* 5, 367 (2012).

[202] J. M. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman and D. A. Case. “Development and testing of a general Amber force field”. In: *J. Comput. Chem.* 25, 1157 (2004).

[203] C. I. Bayly, P. Cieplak, W. D. Cornell and P. A. Kollman. “A well-behaved electrostatic potential based method using charge restraints for deriving atomic charges: The RESP model”. In: *J . Phys. Chem.* 97, 10269 (1993).

[204] F. Neese. “The ORCA program system”. In: *WIREs Comput. Mol. Sci.* 2, 73 (2012).

[205] T. Lu and F. Chen. “Multiwfn: A multifunctional wavefunction analyzer”. In: *J. Comput. Chem.* 33, 580 (2012).

[206] A. Jakalian, B. L. Bush, D. B. Jack and C. I. Bayly. “Fast, efficient generation of high-quality atomic charges. AM1-BCC model: I. Method”. In: *J. Comput. Chem.* 21, 132 (2000).

[207] A. Jakalian, D. B. Jack and C. I. Bayly. “Fast, efficient generation of high-quality atomic charges. AM1-BCC model: II. Parameterization and validation.” In: *J. Comput. Chem.* 23, 1623 (2002).

[208] J. Güldenhaupt, M. Amaral, C. Kötting, J. Schartner, D. Musil, M. Frech and K. Gerwert. “Ligand-induced conformational changes in Hsp90 monitored time resolved and label free - Towards a conformational activity screening for drug discovery”. In: *Angew. Chem. Int. Ed.* 57, 9955 (2018).

[209] I. Hughes and T. Hase. *Measurements and Their Uncertainties – A Practical Guide to Modern Error Analysis.* Oxford University Press, 2010.

# Danksagung

Zuallererst gilt mein Dank meinem Betreuer Prof. Gerhard Stock, der mir nicht nur die Möglichkeit zu dieser Doktorarbeit gegeben hat, sondern mich auch immer aktiv dabei unterstützt hat. Egal um welches Thema es ging, er war auf dem Laufenden und hatte immer Ideen was man noch untersuchen oder wie man etwas interpretieren könnte. Betrachtet man die Anzahl an verschiedenen Gruppenprojekten und den sonstigen Professorenalltag, ist dies schwerlich überzubewerten.

Des Weiteren gilt mein Dank PD Dr. Steffen Wolf, dessen erstaunlich detailliertes Wissen zu MD Simulationen und deren Interpretation (und die Bereitschaft dieses Wissen geduldig weiterzugeben) viele Untersuchungen und Überlegungen erst möglich gemacht haben. Außerdem möchte ich mich bei Matthias Post bedanken, der mit vielen überaus hilfreichen Diskussionen und seinem mathematischen Verständnis zu dieser Arbeit beigetragen hat.

Weiterhin möchte ich mich bei den übrigen Gruppenmitgliedern für die freundliche und unaufgeregt Gruppenatmosphäre bedanken, die sie immer verbreitet haben. Ich möchte an dieser Stelle insbesondere Marion Furtwängler-Fritz hervorheben, die immer dafür gesorgt hat, dass die Verwaltung der Gruppe läuft und es so ermöglicht hat, dass ich bis heute (dankenswerterweise) nur einen groben Eindruck von den Windungen der Universitätsbürokratie habe.

Des Weiteren möchte ich an dieser Stelle Prof. Tanja Schilling und ihrer Gruppe für viele interessante und hilfreiche Diskussionen danken. Von ihnen habe ich auch die Daten zur Kristallbildung von harten Kugeln erhalten, wie ich nochmals hervorheben möchte.

Auch möchte ich an dieser Stelle Dr. Norbert Schaudinnus nicht unerwähnt lassen, der mich zu Zeiten von Bachelor- und Masterarbeit betreut hat und mich für die Langevinmodellierung rekrutiert hat. Hätte er mir nicht von den Freuden der dLE-Modellierung berichtet, würde diese Arbeit sehr anders aussehen oder vielleicht gar nicht existieren.

Außerdem möchte ich meinen Eltern danken, ohne die ich nicht der wäre, der ich bin und die mich immer auf meinem Weg unterstützt haben. Dies ist schwerlich zu unterschätzen. Auch meine Patentante will ich an dieser Stellen nicht unerwähnt lassen, die mich ebenfalls immer freundlich begleitet hat.

Und last but not least möchte ich meinem Freundeskreis danken, der mich nun schon einige Jahre begleitet und dabei dafür gesorgt hat, dass ich auch noch an andere Dinge als nur die Universität und die Doktorarbeit denke.