Advances in Experimental Medicine and Biology 797

Gregory R. Bowman
Vijay S. Pande
Frank Noé *Editors*

An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation

Springer

Advances in Experimental Medicine and Biology

Volume 797

For further volumes:
www.springer.com/series/5584

Gregory R. Bowman • Vijay S. Pande •
Frank Noé
Editors

An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation

Springer

Editors

Gregory R. Bowman
University of California
Berkeley, CA, USA

Frank Noé
Freie Universität Berlin
Berlin, Germany

Vijay S. Pande
Department of Chemistry
Stanford University
Stanford, CA, USA

ISSN 0065-2598 ISSN 2214-8019 (electronic)
Advances in Experimental Medicine and Biology
ISBN 978-94-007-7605-0 ISBN 978-94-007-7606-7 (eBook)
DOI 10.1007/978-94-007-7606-7
Springer Dordrecht Heidelberg New York London

Library of Congress Control Number: 2013956358

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Contents

Contributors

Gregory R. Bowman Departments of Molecular & Cell Biology and Chemistry, University of California, Berkeley, CA, USA; University of California, Berkeley, USA

John D. Chodera Memorial Sloan-Kettering Cancer Center, New York, NY, USA

Gianni De Fabritiis Computational Biophysics Laboratory (GRIB-IMIM), Universitat Pompeu Fabra, Barcelona Biomedical Research Park (PRBB), Barcelona, Spain

Xuhui Huang Department of Chemistry, Division of Biomedical Engineering, Center of Systems Biology and Human Health, Institute for Advance Study, The Hong Kong University of Science and Technology, Kowloon, Hong Kong

Frank Noé Freie Universität Berlin, Berlin, Germany

Vijay S. Pande Department of Chemistry, Stanford University, Stanford, CA, USA; Stanford University, Stanford, CA, USA

Jan-Hendrik Prinz Freie Universität Berlin, Berlin, Germany

Benoît Roux Department of Biochemistry and Molecular Biology, The University of Chicago, Chicago, USA

Marco Sarich Freie Universität Berlin, Berlin, Germany

Christof Schütte Freie Universität Berlin, Berlin, Germany

Deniz Sezer Faculty of Engineering and Natural Sciences, Sabancı University, Istanbul, Turkey

Eric Vanden-Eijnden Courant Institute, New York University, New York, NY, USA

Acronyms

ESR	Electron spin resonance
MD	Molecular dynamics (simulation)
MSM	Markov state model
PCCA	Perron cluster cluster analysis
TPT	Transition path theory
TPS	Transition path sampling

Mathematical Symbols

$\mathbf{T}(\tau)$ A transition probability matrix (row-stochastic) in $\mathbb{R}^{n\times n}$ describing the probabilities of hopping amongst a discrete set of states. The elements $T_{ij}(\tau)$ give the probability of an $i \to j$ transition during a time interval τ.

$\hat{\mathbf{T}}(\tau)$ An estimate of $\mathbf{T}(\tau)$ from trajectory data.

$\mathbf{C}(\tau)$ A transition count matrix (row-dominant) in $\mathbb{R}^{n\times n}$ describing the number of transitions observed amongst a discrete set of states. The elements $c_{ij}(\tau)$ count the number of $i \to j$ transitions observed, each of which occurred during a time interval τ.

τ The time resolution (or lag time) of a model.

n The number of discrete states.

$\mathbf{p}(t)$ A (column) vector in $\mathbb{R}^n$ where the entry $\mathbf{p}_i(t)$ specifies the probability of being in state i at time t.

$\boldsymbol{\pi}$ A (column) vector in $\mathbb{R}^n$ where the entry $\boldsymbol{\pi}_i$ specifies the equilibrium probability of being in state i.

λ_i The i'th largest eigenvalue of a transition probability matrix T. The largest eigenvalue is λ_1 and eigenvalues are ordered such that $1 = \lambda_1 > \lambda_2 > \lambda_3$.

$\boldsymbol{\psi}_i$ The i'th right eigenvector of a transition probability matrix T in $\mathbb{R}^n$. The first right eigenvector is $\boldsymbol{\psi}_1$.

$\boldsymbol{\phi}_i$ The i'th left eigenvector of a transition probability matrix T in $\mathbb{R}^n$. The first left eigenvector is $\boldsymbol{\phi}_1$.

χ_i An indicator function for state i that is 1 within state i and 0 elsewhere. It may also refer to the degree of membership in state i.

$\boldsymbol{\theta}_i$ An experimental observable characteristic of sate i.

q_i The commitor probability for state i. That is, the probability of reaching some predefined set of final states from state i before reaching some predefined set of initial states.

Ω A continuous state space (including positions and momenta).

$\mathbf{x}(t)$ A state in Ω (including positions and momenta) at time t.

$\mu(\mathbf{x})$ The stationary density of $\mathbf{x}$.

$p(\mathbf{x}, \mathbf{y}; \tau)$ The transition probability density to $\mathbf{y} \in \Omega$ after time τ given the system is in $\mathbf{x} \in \Omega$.

$\mathcal{T}(\tau)$	A transfer operator that propagates the continuous dynamics for a time τ.
m	The number of dominant eigenfunctions/eigenvalues considered.
$S_1, \ldots, S_n$	Discrete sets which partition the state space Ω.
$\mu_i(\mathbf{x})$	The local stationary density restricted to a discrete state i.
$\langle f, g \rangle$	The scalar product $\langle f, g \rangle = \int f(\mathbf{x}) g(\mathbf{x})\, d\mathbf{x}$.
$\langle f, g \rangle_\mu$	The weighted scalar product $\langle f, g \rangle_\mu = \int \mu(\mathbf{x}) f(\mathbf{x}) g(\mathbf{x})\, d\mathbf{x}$.

1 Introduction and Overview of This Book

Gregory R. Bowman, Vijay S. Pande, and Frank Noé

Computer simulations are a powerful way of understanding molecular systems, especially those that are difficult to probe experimentally. However, to fully realize their potential, we need methods that can provide understanding, make a quantitative connection with experiment, and drive efficient simulations.

The main purpose of this book is to introduce Markov state models (MSMs) and demonstrate that they meet all three of these requirements. In short, MSMs are network models that provide a map of the free energy landscape that ultimately determines a molecule's structure and dynamics. These maps can be used to understand a system, predict experiments, or decide where to run new simulations to refine the map. Protein folding and function will often be used to illustrate the principles in this book as these problems have largely driven the development of MSMs; however, the methods are equally applicable to other molecular systems and possibly entirely different problems. Whether you are an experimentalist interested in understanding a bit of theory and how it could complement your work or a theorist seeking to understand the details of these methods, we hope this book will be useful to you.

This introduction provides a brief overview of the background leading to the development of MSMs, what MSMs are, and the contents of this book.

1.1 Background

Molecular systems are exquisitely sensitive to atomistic details—for example, a single point mutation can have dramatic effects on protein folding or function—a complete understanding would require atomically detailed models that capture both the thermodynamics and kinetics of the system of interest. There are many powerful experimental methods for probing the structure and dynamics of molecular systems but, currently, none can provide a complete understanding of a system.

Structural biologists have developed a range of methods for building atomically detailed models of proteins and other molecules; however, we are far more limited when it comes to dynamics. For example, when monitoring the relaxation of an ensemble of unfolded proteins back to the native state, one typically sees simple behavior that can be fit well by a single or double exponential. By Occam's razor, it is difficult to justify explaining such data with anything more complicated than a two- or three-state model. To push beyond these extremely coarse models, one has to begin making perturbations like mutations or trying

G.R. Bowman (✉)
Departments of Molecular & Cell Biology and Chemistry, University of California, Berkeley, CA 94720, USA
e-mail: gregoryrbowman@gmail.com

V.S. Pande
Department of Chemistry, Stanford University, Stanford, CA 94305, USA

F. Noé
Institut für Mathematik II, Freie Universität Berlin, Arnimallee 2-6, 14195 Berlin, Germany

G.R. Bowman et al. (eds.), *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology 797,
DOI 10.1007/978-94-007-7606-7_1, © Springer Science+Business Media Dordrecht 2014

to incorporate other experimental data. However, the sensitivity of many molecular processes to atomistic changes makes interpreting the effects of perturbations difficult and combining different types of experimental data is also nontrivial—for example, how does one weight the relative contributions of two different types of data to a model? As a result, while there are certainly many opportunities in these directions, there is currently no clear path to building atomically detailed models for the entirety of a system from experimental data alone.

An alternative is to develop computer models that can complement experiment by providing an unambiguous description of a system's atomic motions. Ideally, these models could be validated by comparison to existing experimental data. One could then delve into the rich structural and kinetic information the model would provide to explain the origins of experimental results and generate hypotheses to guide the design of new experiments.

Atomistic molecular dynamics simulations are one powerful tool for achieving this vision. In these simulations, one iteratively evaluates the force each atom experiences due to the other atoms in the system, calculates where each atom will be some small timestep in the future, and finally updates their positions.

Unfortunately, it is extremely challenging to reach biologically relevant timescales in a molecular dynamics simulation, much less to obtain sufficient statistics to accurately characterize a system's behavior. The large forces and small length scales involved in such simulations necessitate a very small timestep—typically on the order of a femtosecond, or 10^{-15} seconds. One must then build up, about one femtosecond at a time, to the microseconds, milliseconds, and seconds timescales where many of the molecular processes of interest typically occur. Simulating a single millisecond on a typical desktop computer could easily take hundreds of years and is still essentially intractable with large computer clusters, though some progress has been made with distributed computing and specialized hardware.

Many advanced methods have been developed to overcome this gap between biological and simulation timescales but none is a magic bullet. For example, generalized ensemble methods—like replica exchange—allow a simulation to perform a random walk in temperature space. The hope is that at low temperatures the simulation will slowly explore the landscape of interest but that at high temperatures the system can easily jump to new regions of conformational space. Such methods are extremely powerful for small systems where energetic barriers dominate but can actually perform worse than conventional molecular dynamics for more complicated systems where entropic barriers dominate because these will become even more insurmountable at high temperatures. Coarse-graining can also provide reasonable speedups by reducing the number of pairwise interactions that must be calculated. However, there is always the danger that the degrees of freedom one coarse-grains out are actually important, in which case the coarse-grained simulation is of no value.

Even if these advanced methods could access arbitrarily long timescales, the issue of how to extract understanding from them would still remain. One cannot simply report what happened in a simulation because molecular processes like protein folding are inherently stochastic, so the exact sequence of events in one simulation is extremely unlikely to appear in a second trajectory.

One common analysis method is to project the free energy landscape onto order parameters but, once again, this is not a general solution. Projections of the free energy surface are really only valid if the order parameters chosen are truly reaction coordinates for the process of interest—i.e. they accurately reflect progression from reactants to products. In a very few cases, it is clear what the reaction coordinates are. For example, the alanine dipeptide only has two degrees of freedom, so it is perfectly legitimate to project the system's free energy landscape onto these order parameters. However, for processes like protein folding that occur in extremely high-dimensional spaces, finding a reaction coordinate is not so simple. Researchers often project free energy surfaces for proteins onto popular order parameters, like the number of native contacts or the RMSD to a known crystal structure, but one can find drastically different landscapes by choosing different order parameters Therefore, these methods often do not provide clear and consistent models of molecular processes.

Clustering the conformations sampled with a set of simulations based on some geometric criterion—like the RMSD between conformations—is a less biased approach but is still not completely satisfactory. One major advantage of clustering is that it is less biased than projections since no reaction coordinate has to be assumed a priori. Furthermore, once the data has been clustered, many analyses can be performed easily. For example, comparison of the relative amounts of time spent in different clusters gives information about their relative free energies. One can also attempt to estimate the transition rates between clusters from the number of transitions observed between them and then begin looking at the most probable pathways between arbitrary start and end points. However, many important questions remain. For example, how many clusters are necessary and where, exactly, should the boundaries between them lie? Given two different clusterings, which one is better? Does a given clustering contain useful information? As will be discussed in more detail later, many problems can also arise when trying to estimate kinetic parameters from these models.

1.2 Markov State Models

A Markov model consists of a network of conformational states and a transition probability matrix describing the chances of jumping from one state to another in some small time interval. Many readers will recognize them as discrete time master equation models. Importantly, the states in an MSM are defined based on kinetic criteria rather than geometric criteria. Therefore, it is possible to accurately identify the boundaries between free energy basins and model dynamic processes like the relaxation to equilibrium.

A Markov model is a coarse-graining of a system's dynamics that reflects the underlying free energy landscape that determine's the system's structure and dynamics. Intuitively, it is often useful to think of the states in a Markov model as corresponding to free energy minima. However, as discussed in the next few chapters, this is not always necessarily true. Nonetheless, Markov models can provide important insights into a molecule because we have a much better intuition for states and rates (or, equivalently, transition probabilities) then we do for the large numbers of three dimensional structures generated by MD simulations.

The states and rates picture also provides a natural means to make a quantitative connection with experiments. For example, it is often possible to calculate an experimental observable (like the distance between two probes) for each state. A set of initial conditions can then be prepared by populating a subset of states and the relaxation to equilibrium can be modeled using the transition probabilities between states. This dynamics can be projected onto the experimental observable and the resulting signal can be compared to experiment.

Finally, adaptive sampling methods leverage Markov models to direct efficient simulations. In adaptive sampling, one iteratively run simulations, builds a Markov model, and then uses the current model to decide where to spawn new simulations to improve the model. Such methods can lead to tremendous improvements in computational efficiency compared to simply running one long simulation and waiting for it to gather statistics on the entirety of conformational space.

1.3 Outline of This Book

The remainder of this book can be divided into two sections. The first section, which includes Chaps. 2 through 7, presents the theoretical foundations of Markov state models. The second section, which includes Chaps. 8 through 10, focuses on a number of exciting applications of Markov models that serve to demonstrate the value of this approach. Below, we briefly review the contents of each chapter.

Chapter 2 provides a more thorough overview of Markov state models and how they are constructed. This discussion includes a description of the key steps for building an MSM and some of the options available for each stage of the model building process. An important theme is that there is no single right way to perform many of these steps. Therefore, it is valuable to have some understanding of the tradeoffs between the available options.

Chapter 3 lays the theoretical foundation of MSMs. As indicated by the name, Markov models assume that the current discrete state of the system is sufficient to know the probabilities of jumping to any other state in the next time interval, without having to know the previous history. While the Markov assumption may be correct for the dynamics in the full-dimensional phase space, it cannot be *exactly* correct for the discrete partition of state space used for the MSM. The associated error, i.e. the difference of the MSM kinetics from the exact kinetics is a *discretization error*. Fortunately, we do not depend on a leap of faith when constructing MSMs. As a result of thorough mathematical work, especially during the last couple of years, the MSM *discretization error* is now well understood and can even be quantitatively bounded. Chapter 3 describes the nature of this error in the absence of additional statistical error, derives properties that a "good" partition of state space must fulfill, and suggests advanced approaches for MSM construction that go beyond simple state decomposition by clustering.

In practice, constructing MSMs progresses by defining a partitioning of conformational space into states and subsequently testing and possibly refining it. In order to do so, the MD trajectory data must be mapped on the discrete state space partitioning, and the MSM transition matrix must be estimated. Chapter 4 describes this step in detail and derives statistically optimal estimators for the transition matrix given a dataset and a state space partitioning. Subsequently, practical tests are described to assess the quality of the estimated MSM. It is these tests that will report on success or failure of the MSM to be a consistent kinetic model, and appropriate steps can be taken, e.g. by refining the state space partitioning used.

Since an MSM is estimated from a finite amount of MD trajectory data, the associated transition matrix and all properties computed from it will involve statistical uncertainty. Clearly, this is an issue for any model of the equilibrium or kinetic properties that is built from computer simulations—not just MSMs. Fortunately, for MSMs we now have a very complete theory that allows us to quantify these statistical errors from the number of transitions observed between the discrete sub-states. Chapter 5 attempts to give an overview of these methods and then goes into detail with Bayesian methods to sample the statistical uncertainties of transition matrices, and any quantity computed from them. Importantly, one can use estimates of uncertainties from an existing MSM to decide where to run new simulations in order to refine the model as efficiently as possible. Such methods are called adaptive sampling.

Chapter 6 gives an overview of some of the most useful analyses that can be performed with a valid Markov model. Three aspects are discussed and illustrated using a toy model of protein folding. First, we describe the significance of eigenvalues and eigenvectors of MSM transition matrices. Eigenvalues are related to the relaxation timescales of kinetic processes, and eigenvectors indicate the associated structural changes. Consequently, the eigenvectors associated with the slowest relaxation timescales can be used to find the metastable states of the molecular system studied. Secondly, the ability to associate relaxation timescales with structural changes via the eigenvalue-eigenvector duality is arguably one of the main advantages of MSMs over many other approaches to analyze MD simulation data. It permits one to uniquely assign structural transition events to experimentally measurable timescales, which makes MSMs a very valuable tool for quantitatively comparing simulation and experiment. We can go further and quantitatively predict the relaxation or correlation functions measured by kinetic experiments using three ingredients: the MSM eigenvalues, eigenvectors and the mean value of the spectroscopic observable for each discrete state. Chapter 6 describes the associated theory.

Finally, MSMs allow us to compute complex kinetic quantities that may not be directly experimentally accessible. One example is the ensemble of transition pathways and the transition state

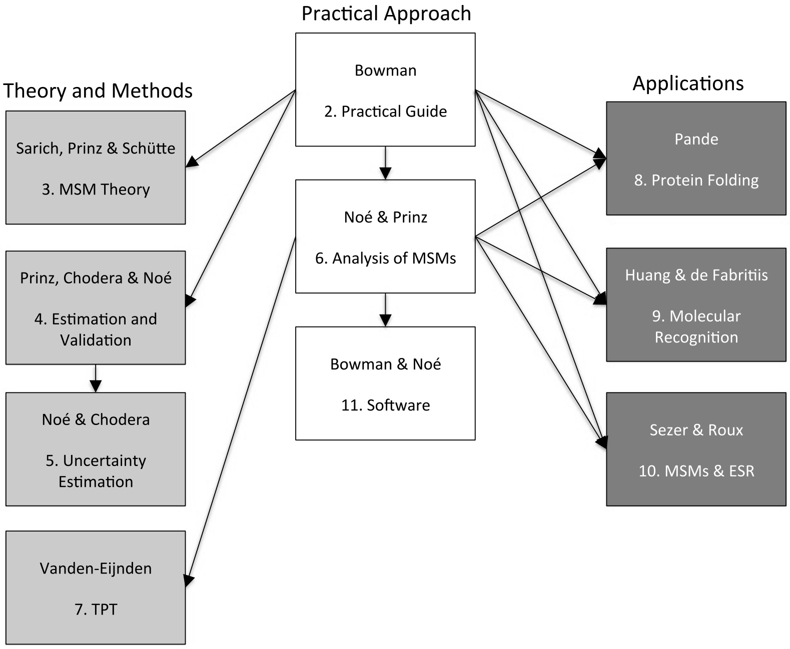


Fig. 1.1 Overview of the chapters in this book

ensemble. Given an MSM, both can be easily computed with transition path theory. Chapter 6 gives an introduction to transition path theory and illustrates it on the folding of the Pin WW peptide.

Chapter 7 is a theoretical chapter that goes into more detail on transition path theory. While transition path theory was originally derived for continuous Markov processes, this chapter focuses on its use in conjunction with MSMs and illustrates it using a simple example—a random walk in a two-dimensional maze. The basic mathematical quantities needed for computing transition pathways are defined and the equations for computing them from transition matrices are given. Furthermore, an approach to efficiently generate samples of reactive trajectories is introduced.

MSMs meet all three of the requirements laid out at the beginning of this chapter: providing understanding, making a quantitative connection with experiment, and driving efficient simulations. The subsequent application chapters will show that MSMs have already proven consistent with existing experimental data for a variety of molecular processes, allowed researchers to better understand—and sometimes even reinterpret—existing data, and led to new hypotheses that have been borne out in subsequent experiments.

Chapter 8 describes the application of Markov models to the protein folding problem and the new insights this has provided. This problem has two major components. First, how can we predict the structure of a protein from its sequence? And, second what is the sequence of events that allows a protein to fold? Besides showing how Markov models address both of these issues, this chapter will discuss how MSMs have allowed researchers to study much larger and slower systems than would otherwise be possible.

Chapter 9 summarizes recent work on using Markov models to understand how proteins bind small molecules. This application has important implications for drug design and our understanding of signaling within cells. It also presents an interesting methodological challenge because it is non-trivial to move from studying single-body problems (like protein folding, where all the atoms in the system of interest are covalently linked together) to multi-body problems.

Chapter 10 discusses how Markov models can be used to connect with new experimental techniques, like electron spin resonance. An important emphasis is the impressive degree of agreement between simulation and experiment that one can achieve.

Since the construction, validation and analysis of MSMs is a nontrivial task, the existence of software that can support the user in these tasks is crucial. Chapter 11 provides an overview of existing MSM software packages and their current capabilities. Clearly, these packages are rapidly evolving and thus this chapter is just meant as a starting point. Therefore, links are provided to the manuals and tutorials of the software packages described.

Figure 1.1 provides an overview of the chapters of this book. For readers who decide not to follow the sequence of chapters in the book, we indicate the dependencies between chapters from the viewpoint of a practically oriented reader that is unfamiliar with MSMs. Theoretically inclined readers may start with the theory sections. Readers familiar with MSMs may read the book chapters in any sequence.

2 An Overview and Practical Guide to Building Markov State Models

Gregory R. Bowman

The main purpose of this chapter is to provide a practical guide to building Markov models, with an emphasis on partitioning a molecule's conformational space into a valid set of states. This process is often referred to as constructing a state decomposition.

2.1 The Big Picture

The ideal state decomposition method would perform a truly kinetic clustering of a data set to accurately resolve the barriers between metastable states. Unfortunately, there is no simple means to calculate the average transition time between two arbitrary conformations.

One alternative would be to build states based purely on geometric criteria. However, this approach turns out to be inadequate because there is no physical reason that the locations of free energy barriers should correlate with geometric criteria. For example, two conformations with a 5 Å RMSD could fall within the same free energy basin if they only differ by pivoting of a hinge motion while another pair of conformations separated by the same distance could fall in different basins if they differ by strand pairings in a beta sheet. Mistakenly grouping together conformations that are not within the same free energy basin can create states with large internal free energy barriers. Such states will violate the Markov property because a system that enters the state on one side of the barrier will behave differently than one that enters on the other side, thereby introducing history dependence. Higher order Markov chains could be used to capture this history dependence, however, doing so greatly increases the number of parameters that must be determined, making it harder to obtain sufficient statistics. Moreover, people generally do not have nearly as well developed an intuition for processes with history dependence as we do for Markov models, so higher order models would provide less understanding.

At present, the most common approach for building MSMs is a two-stage process exploiting both geometry and kinetics [1–6]. In this two-stage approach, one uses a kinetically-relevant geometric clustering to create a starting point for a more purely kinetic clustering. By kinetically-relevant, I simply mean a clustering that only groups conformations together if the system can transition between them quickly relative to transitions between clusters.

The objective of the first stage is to create small volume elements in conformational space—called microstates—that are essentially the same structure using a geometric clustering. The motivation for starting with such a clustering follows that employed in the study of probability distribution functions, where one recognizes that the probability of a single point is vanishingly small and, therefore, works with small volume

G.R. Bowman (✉)
University of California, Berkeley 94720, USA
e-mail: gregoryrbowman@gmail.com

G.R. Bowman et al. (eds.), *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology 797,
DOI 10.1007/978-94-007-7606-7_2, © Springer Science+Business Media Dordrecht 2014

elements instead. At this stage, one would like to go out of their way to divide phase space as finely as possible to ensure that no microstate contains large free energy barriers. However, this objective is counterbalanced by the need to maintain sufficient statistics for each state such that transition probabilities between each pair of states can be estimated accurately. Given a set of microstates that meets these requirements, the transition probability between a pair of states can be calculated numerically by counting the number of times a simulation started in one of them and ended in the other because now two simulations have a finite probability of entering the same volume element. As discussed shortly, there are a number of ways to create and validate microstate models. Such models are excellent for making a quantitative connection with experiments because of their high resolution. However, they are often difficult to understand because they typically have tens of thousands of states.

To make a more understandable model, one can perform a kinetic clustering of a kinetically-relevant set of microstates to form larger aggregates—called macrostates—that correspond to free energy basins. One objective of this type of coarse-graining is to create mesoscale models that are still quantitative but are much more compact than the initial microstate model. These models may still be too complex to understand, however, the reduced state space makes them much easier to work with. A second objective is to coarse-grain the model so much that one can actually understand it. Often, these models will only be qualitatively correct—no longer able to make a quantitative connection with experiment. However, such extreme coarse-grainings are excellent for gaining an intuition for a system and generating new hypotheses to be tested with higher resolution models and, ultimately, with experiments.

To summarize, the key steps for building an MSM are

1. Choose an appropriate distance metric and cluster your simulation data into microstates.
2. Test the kinetic relevance of this clustering and choose an appropriate lag time (or observation interval) based on the Markov time of the model (smallest lag time that gives Markovian behavior).
3. Estimate the microstate model's transition probability matrix.
4. Coarse-grain the model to create either quantitative mesoscale models or qualitative models for guiding one's intuition.
5. Use the qualitative models to understand your system and the microstate or mesoscale models to model experiments.

Following is an explanation of the various alternatives for each of these steps. Throughout this discussion, a model refers to a transition probability matrix and one or more representative conformations from each state.

2.2 Clustering to Generate Microstates

The major objective of this step is to construct a kinetically-relevant clustering using geometric criteria. Such a clustering should only group together conformations the system can jump between rapidly. Many clusterings may satisfy this requirement, so there is not necessarily a single right answer. The resulting microstate model can then be used for making a quantitative connection with experiment or as a starting point for kinetic clustering.

Some of the key choices for this step are which distance metric to use, which clustering algorithm to use, how many clusters to generate, and which data to cluster.

2.2.1 Choosing a Distance Metric

It should come as no surprise that creating a kinetically-relevant clustering is best achieved with a kinetically-relevant distance metric. In particular, it is necessary for conformations separated by small distances to interconvert rapidly. Any distance metric that satisfies this requirement is sufficient given infinite data—i.e. the ability to create an infinitude of infinitely small states. However, one can typically make far better use of finite data by employing distance metrics that best capture the relevant dynamics. For example, the opening angle may be a good choice for studying the opening and closing of a hinged-protein [7].

In leu of an obvious problem specific metric, the root-mean-square deviation (RMSD) between atoms is often a reasonable choice for protein dynamics [1, 2, 6]. Large RMSDs are hard to interpret, but two conformations separated by only a few Å are likely to interconvert rapidly. For extremely detailed models, the all-atom RMSD may be useful—though one must be careful about symmetry issues, like the invariance of Phe to a 180 degree flip. Basing the RMSD on α-carbons or all backbone heavy atoms is often sufficient though.

2.2.2 Choosing a Clustering Algorithm

Here, we briefly review a number of the clustering algorithms currently in use and their relative merits. There are many other options and there is great value in assessing their relative merits [5, 8]. For the purposes of this review, however, I hope only to describe a few of the most common options with the intent that this analysis will serve as a guide for evaluating other options.

2.2.2.1 *k*-Centers Clustering

In k-centers clustering, one tries to create a set of clusters with approximately equal radii by optimizing the objective function

$$\min_{\sigma} \max_{i} d\big(x_i, \sigma(x_i)\big) \tag{2.1}$$

where $\sigma(x)$ is a function that maps a conformation (x) to the nearest cluster center and $d(x, y)$ is the distance between two conformations x and y. The minimization occurs over all clusterings (σ) with k states and the max is taken over all conformations in the dataset. The radius of a cluster is just the maximum distance between any data point in that cluster and the cluster's center.

One advantage of k-centers is that it divides up conformational space more evenly than other algorithms by ensuring that states have similar radii [9, 10]. Intuitively, one can think of this algorithm as creating clusters with approximately equal volumes. However, one must take care not to take this too literally as very small variations in the radii of clusters in high-dimensional spaces can give rise to huge variations in their volumes. Having a more or less even division of conformational space into microstates is of value because it helps avoid situations where some regions are unnecessarily divided into an excess of states while other regions are not sufficiently broken-up to avoid large internal free energy barriers. Properties of the model, like the slowest relaxation time, should also be insensitive to the exact clustering as long as one purposefully over-divides conformational space into a large number of states.

Another advantage of k-centers is that there is an extremely fast, deterministic approximation to this algorithm [9, 10]. This approximate algorithm has an $O(kN)$ runtime, where k is the number of clusters and N is the number of conformations being sampled, so it is applicable to even extremely large data sets. This algorithm works as follows (Fig. 2.1):

1. Choose an arbitrary point as the initial cluster center and assume all other data points are initially in that cluster (Fig. 2.1A).
2. Calculate the distance from every other data point to the current cluster center.
3. Select the furthest point from any existing cluster center as the next cluster center (Fig. 2.1B).
4. Calculate the distance from every data point to this new cluster center and reassign any of them that are closer to the new center than their previous cluster center to the new cluster (Fig. 2.1B).
5. Repeat steps 3 and 4 until some cutoff criterion—like the number of clusters or maximum size of any cluster—is reached (Fig. 2.1C).

The most appropriate cutoff criterion depends on the process of interest. For example, creating clusters until each state has a radius of less than 3 Å RMSD is often an appropriate starting point for protein folding, where the relevant conformational space is huge and a rather coarse partitioning will do. For more subtle conformational changes where there is a small space and more

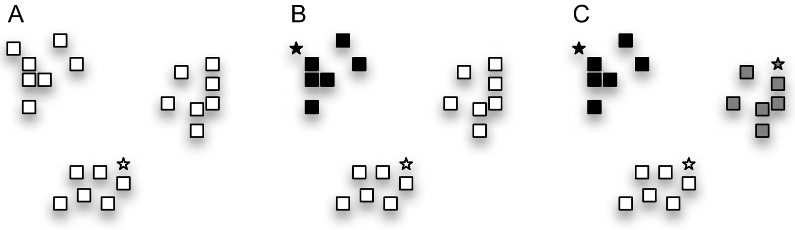


Fig. 2.1 An example of k-centers clustering of a set of data points (*squares*). (**A**) First, a random data point (*white star*) is chosen as the initial cluster center and all data points are assigned to it (*white squares*). (**B**) Next, the data point furthest from the previous cluster center is chosen as the next cluster center (*black star*). All the data points that are closer to the new cluster center than any existing center are assigned to the new center (*black squares*). (**C**) The algorithm continues choosing the data point that is furthest from any existing center (in this case the *gray star*) and assigning data points that are closer to it to the new center (*gray squares*)

detail is required, a cutoff of 1 Å may be a better starting place.

One disadvantage of k-centers is that building a viable set of microstates with this algorithm often requires creating a large number of clusters. For example, modeling the folding of a 35 residue variant of the villin headpiece—one of the smallest known proteins—still required 10,000 states [11]. As a result, one needs a great deal of sampling to ensure adequate statistics for each state. The large number of states also leads to large microstate transition matrices that can be computationally demanding to work with. A more ideal algorithm would use kinetic criteria to have larger or smaller states as needed to accurately capture the underlying landscape. Finally, the “centers” created by k-centers are not necessarily anywhere near the geometric center of the cluster. Instead, they are often on the periphery of the cluster because the approximate algorithm presented here is always choosing the data point furthest from all the existing cluster centers as the next one and, therefore, is biased towards choosing data points at the very edge of the space sampled (Fig. 2.1). Thus, the cluster centers are not necessarily representative of the data assigned to them. One must use other strategies to identify representative conformations for a cluster created with k-centers, like drawing a few random conformations from it.

2.2.2.2 k-Medoids Clustering

The k- medoids algorithm minimizes the average distance between data points and the center they are assigned to by optimizing

$$\frac{1}{N}\sum_i d\big(x_i,\sigma(x_i)\big)^2 \qquad (2.2)$$

where N is the number of data points, $\sigma(x)$ is a function that maps a conformation (x) to the nearest cluster center, and $d(x,y)$ is the distance between two conformations x and y.

The k-medoids algorithm is very similar to k-means but with the important difference that only data points can be cluster centers. In k-means, the cluster center is the average of all the data points belonging to that cluster. However, taking the average of a number of protein conformations does not make physical sense as it can easily lead to unphysical behavior like steric clashes and extremely unlikely bond lengths/angles. Thus, k-medoids is preferable.

One advantage of k-medoids over k-centers is that k-medoids tends to create clusters with a more equal number of samples. For example, if a data set has a densely sampled region and a sparsely sampled region, then k-medoids will tend to place more clusters in the densely sampled region. This feature is useful in that it helps avoid states with too few counts to make statistically reliable estimates of the transition probabilities to other states. However, k-medoids may also over-divide some regions of conformational space and under-divide others. For in-

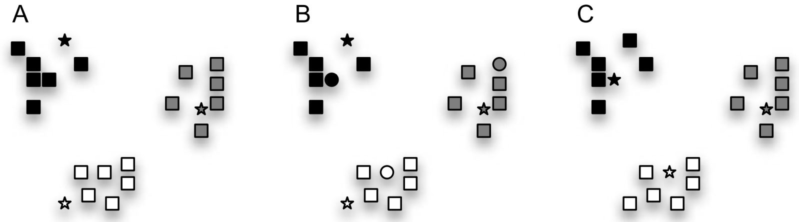


Fig. 2.2 An example of k-medoids clustering of a set of data points (*squares*). (**A**) First, the user decides how many clusters to construct (in this case $k = 3$). Then k random data points are chosen as initial cluster centers (the *three stars*). All data points are then assigned to the closest center (as indicated by the color coding). (**B**) Next, a random data point in each cluster is proposed as a new cluster center (*circles*). (**C**) If the newly proposed center is closer on average to all the data points in the cluster than the previous center, then it is chosen as the new center for that cluster (as in the *black* and *white clusters*). Otherwise, the newly proposed center is rejected and the previous center is kept (as in the *gray cluster*). Finally, all the data points are reassigned to their closest center. Note that this is an extremely contrived example. Usually there will not be such a clear distinction between clusters, the number of clusters will be unclear, and the initial cluster centers may end up being very close to one another, requiring a number of iterations of updating before convergence to a reasonable set of centers

stance, in protein folding, k-medoids is likely to create many clusters in the folded ensemble and very few clusters in more sparsely populated unfolded regions. Therefore, one will get essentially redundant clusters in the folded ensemble while mistakenly grouping together kinetically distinct unfolded conformations into states that violate the Markov assumption. However, this problem may not arise for other processes, like conformational changes, where the relevant regions of conformational space may be more evenly sampled.

The k-medoids algorithm works as follows (Fig. 2.2):

1. Randomly choose k conformations as the initial cluster centers (Fig. 2.2A).
2. Assign each data point to the closest center.
3. For each cluster C, propose a random data point $z \in C$ as the new center (Fig. 2.2B) and evaluate the change using

$$\sum_{x_i \in C} d(x_i, z)^2 \qquad (2.3)$$

   If the newly proposed center reduces the objective function compared to the previous center, then replace the current cluster center with z (Fig. 2.2C).
4. Repeat steps 2 and 3 for a specified number of iterations or until the algorithm converges to a stable result.

To speedup the algorithm further, it is common to propose a number of possible new centers for each cluster during step 2.

One advantage of k-medoids is that the resulting centers are actually representative of the data assigned to them because they lie at the center of the cluster. A disadvantage is that the number of clusters must be chosen *a priori*, compared to k-centers where it is possible to choose a physically meaningful criterion for determining the number of states.

2.2.2.3 Hybrid k-Centers/k-Medoids Clustering

A hybrid approach has been developed to strike a balance between the strengths and weaknesses of the k-centers and k-medoids algorithms [4]. This algorithm simultaneously optimizes the objective functions for both k-centers (Eq. (2.1)) and k-medoids (Eq. (2.2)) as follows:

1. Perform an approximate k-centers clustering, as in Sect. 2.2.2.1.
2. Update the centers with a number of iterations of the k-medoids update step (Steps 2 and 3 of the k-medoids algorithm in Sect. 2.2.2.2), rejecting any proposed moves that increase the k-centers objective function in Eq. (2.1).

This hybrid approach appears to be a good, general purpose method for building microstate models. For example, like k-centers, this method still gives a more even discretization of conformational space than a pure k-medoids clustering and one can specify a physically meaningful criterion for determining the number of states to create. The k-medoids update step also results in cluster centers that are representative of the data assigned to them. More importantly, using this update step shifts the centers to the densest regions of conformational space within a state, leading to better resolution of the boundaries between states. As a result, this algorithm yields shorter Markov times with fewer states. Having fewer states means each one has better statistics and less data is required to parameterize the model.

There is still room for improving upon this hybrid approach though. For instance, this algorithm still tries to avoid states with large internal barriers by creating a large number of clusters. As discussed previously, parameterizing models with more states requires more data to obtain sufficient statistics and large transition matrices can be challenging to work with.

2.2.3 Subsampling

One final question that deserves some consideration is which data to cluster. In an ideal case, where one could perform a purely kinetic clustering of simulation data, the answer to this question would be simple: cluster all the data. However, using all the data is not always optimal when starting off with a geometric clustering. For example, an RMSD-based k-centers clustering will select every structural outlier as a cluster center before starting to subdivide the well-sampled regions of conformational space. There are also practical limitations, like the number of conformations that can be stored in memory on typical computers.

Using a subsample of ones data to define a set of microstates can lead to better models because this strategy reduces the impact of outliers [11]. Put another way, clustering a subsample of one's data focuses the cluster centers on the better sampled regions of conformational space. After defining a state space based on a subsample of the data, one can then assign all the data to these microstates, thereby obtaining more statistics. Outliers will then be absorbed into the closest cluster, where they will have little impact on the quality of the model. For protein folding—which typically occurs on time scales of a microsecond or longer—a fruitful procedure has been to store conformations every 1 ns, cluster conformations sampled at a 10 ns interval, and then assign all the data to the resulting microstates.

2.3 Estimating Transition Matrices

In theory, estimating a transition matrix should just be a matter of counting. The first step is to assign data to clusters, which we will number from 0 to $n-1$. Now each trajectory can be thought of as a series of microstate assignments rather than as a series of conformations. The number of transitions between each pair of states can then be counted and stored as a transition count matrix (C), where C_{ij} is the number of transitions observed from state i to state j. With infinite data, one could just use the maximum likelihood estimate for the transition probability between each pair of states to convert the transition count matrix into a transition probability matrix (T). That is,

$$T_{ij}(\tau) = \frac{C_{ij}}{\sum_k C_{ik}} \tag{2.4}$$

where τ is the lag time of the model. However, in practice, estimating transition matrices is complicated by a number of issues, like finite sampling and imperfections in microstate definitions.

2.3.1 Counting Transitions

Counting transitions sounds like a simple task, but there are actually a variety of options that must be considered. First of all, one must choose a lag time at which the model satisfies the Markov assumption—as discussed in the next section. Choosing an appropriate lag time actually requires estimating transition matrices at a variety

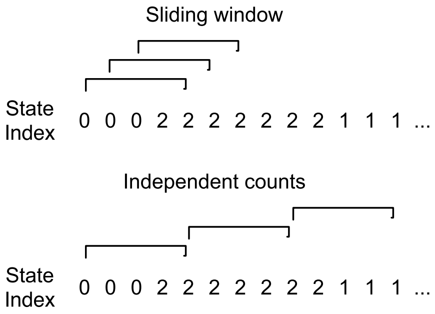


Fig. 2.3 An example of the two methods of counting transitions, assuming a 4-step lag time. Each *panel* shows a trajectory as a series of state indices (i.e. assuming the states have been numbered from 0 to $n-1$). The *top panel* shows how the first three transitions would be counted using the sliding window approach (*brackets*). All three transitions are from state 0 to 2. The *bottom panel* shows how to ensure independent counts. The three transitions (indicated with *brackets*) are from state 0 to 2, from state 2 to 2, and from state 2 to 1

of lag times, so we will cover the process of estimating these matrices first.

In an ideal case, where one has an excess of data relative to the slowest relaxation time in the system, one could simply look at independent transitions at the lag time (τ). That is, one could look at the state indices at an interval of τ. As shown in Fig. 2.3, one could then count transitions as $\sigma(0) \to \sigma(\tau)$, $\sigma(\tau) \to \sigma(2\tau)$, $\sigma(2\tau) \to \sigma(3\tau)\ldots$ where $\sigma(t)$ is the state index of the simulation at time t. However, with finite data, this can lead to imprecise estimates of transition probabilities due to model uncertainty.

Practically, it is often useful to use a sliding window approach. In this approach, one assumes conformations were sampled at a regular interval Δ, where $\Delta < \tau$. For example, one could store conformations every 100 ps and have a lag time of 10 ns. As shown in Fig. 2.3, one could then count transitions as $\sigma(0) \to \sigma(\tau)$, $\sigma(\Delta) \to \sigma(\Delta+\tau)$, $\sigma(2\Delta) \to \sigma(2\Delta+\tau)\ldots$ where $\sigma(t)$ is the state index of the simulation at time t. The sliding window approach will give a more precise estimate of transition probabilities but will lead to underestimates of model uncertainty (see Sects. 4.1 and 5.1).

The sliding window approach is recommended for estimating maximum likelihood transition matrices as precisely as possible. Counting independent transitions should be used when estimating model uncertainty.

2.3.2 Detailed Balance

Another major issue is satisfying detailed balance—also called microscopic reversibility. That is, every time there is a transition from state i to j, there should also be a compensating transition from state j to i. Without this property, one would get source and sink states that would prevent the model from accurately describing long time scale behavior.

Poorly sampled microstates are one issue that can break detailed balance. In particular, some states may have a single transition into or out of them. A simple maximum likelihood estimate of transition probabilities would then turn these states into sources or sinks. Therefore, it is often useful to trim off these states [4, 12, 13].

One must also satisfy detailed balance between every pair of states. One simple way of enforcing detailed balance is to assume that every time there is a transition from state i to j, there must be a corresponding transition from state j to i. The maximum likelihood estimate of the number of transitions from state i to j is then

$$\hat{C}_{ij}(\tau) = \frac{C_{ij} + C_{ji}}{2} \tag{2.5}$$

where $\hat{C}_{ij}$ is an estimate of the reversible counts from state i to j and C_{ij} are the number of transitions actually observed. This method is perfectly valid if one has true equilibrium sampling. Furthermore, it is extremely robust in the sense that this algorithm will always give an estimate of the transition probability matrix that is consistent with the original data. However, if one has limited data, as is often the case, then one's estimate of the transition probability matrix will be extremely biased towards the starting conditions of their simulations. For example, the equilibrium probability of each state will just be proportional to the number of data points in it.

One alternative is to use maximum likelihood methods that try to estimate the reversible transition probability matrix that is most likely to have given rise to the observed data [4, 5, 11]. These methods will be described in more detail in Sect. 4.6. Here, I will just note that these methods are extremely powerful when they work. However, at present, they often fail to converge or converge on transition probability matrices that over-emphasize the importance of poorly sampled states.

2.3.3 Ergodicity

One final point is that a valid MSM must be ergodic. That is, the network of states must be fully connected. Physically, this means that it is possible to reach any state from an arbitrarily chosen starting state. Disconnected components can arise when different initial conformations are employed and sufficient sampling is not obtained to observe mixing (or overlap) between simulations started from different structures. When this happens, it is impossible to determine the relative equilibrium probabilities of disconnected components or the probabilities of transitions between them. Two possible solutions are (1) to discard all but one of the components (typically the largest one) [4, 12, 13] or (2) to collect more data until the network of states becomes completely connected.

2.4 Model Validation and Lag Time Selection

Before drawing any conclusions from a model, it is crucial to test whether or not it is kinetically-relevant and to choose an appropriate lag time. The dynamics of a perfectly specified system, including solvent degrees of freedom and every atom's velocity, is certainly Markovian because the next conformation is simply a deterministic function of the system's current state. However, even microstate models effectively coarse-grain the system. For example, conformations are grouped together and water degrees of freedom are often ignored. Therefore, both microstate models and coarse-grainings thereof may only be Markovian at longer time scales, if at all. As discussed previously, large internal barriers can lead to models that violate the Markov assumption.

2.4.1 Tests Based on the Chapman-Kolmogorov Equation

Many tests of model validity make use of the Chapman-Kolmogorov equation

$$T(n\tau) = T(\tau)^n \tag{2.6}$$

where n is an integer number of steps, each one lag time τ in length. This equation captures the fact that taking n steps with an MSM with a lag time of τ should be equivalent to an MSM with a lag time of $n\tau$.

Plotting the relaxation time scales of a model—also called its implied time scales—as a function of the lag time is one use of the Chapman-Kolmogorov equation that provides some model validation and a means of choosing an appropriate lag time [14]. As will be discussed in more detail in Sect. 3.2, the relaxation times of a model are a function of the eigenvalues of its transition probability matrix

$$t_i = -\frac{\tau}{\ln \lambda_i} \tag{2.7}$$

where t_i is a relaxation time, τ is the lag time, and λ_i is an eigenvalue. Based on the Chapman-Kolmogorov equation, the relaxation times for a Markov model with a lag time of $n\tau$ should be the same as those for a Markov model with a lag time of τ

$$\begin{aligned} t_i &= -\frac{n\tau}{\ln \lambda_{i,T(n\tau)}} = -\frac{n\tau}{\ln \lambda^n_{i,T(\tau)}} \\ &= -\frac{n\tau}{n \ln \lambda_{i,T(\tau)}} = -\frac{\tau}{\ln \lambda_{i,T(\tau)}} \end{aligned} \tag{2.8}$$

where $\lambda_{i,T(\tau)}$ is an eigenvalue of $T(\tau)$. Therefore, examining a plot of the relaxation timescales as a function of the lag time should give an indication of when a model starts to satisfy the Markov assumption, if at all. Beyond the Markov time (the smallest lag time that gives Markovian behavior), the relaxation time scales should be level,

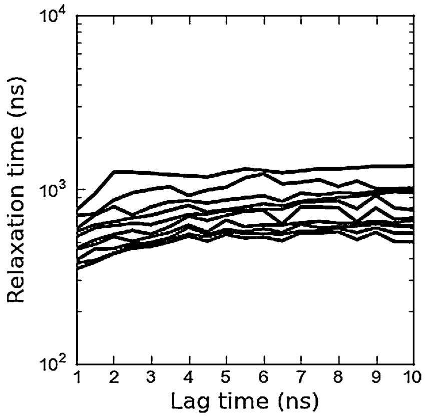


Fig. 2.4 An example relaxation timescale (or implied timescale) plot with a Markov time of ∼2 ns. Data comes from Ref. [15]

as shown in Fig. 2.4. If the relaxation time scales never level-off, then it is likely that one or more states have large internal barriers and a new state decomposition is necessary, and possibly more data as well. Unfortunately, this is a rather subjective test, particularly when dealing with finite statistics.

One can also go beyond plots of the relaxation timescales and test the Chapman-Kolmogorov equation on a state by state basis [5]. More details on this approach are given in Sect. 4.8.

2.4.2 Correlation Function Tests

Comparing correlation functions from the raw data and an MSM is one alternative to Chapman-Kolmogorov-based tests when one has sufficiently long simulations. To calculate correlation functions for a single long trajectory, one first calculates some property of interest for each snapshot—like the RMSD to a crystal structure—and then calculates

$$c(t) = \langle \theta(0)\theta(t) \rangle \tag{2.9}$$

where $\theta(t)$ is the observable at time t.

One can also calculate a (normalized) correlation function for an MSM using

$$c(n) = \frac{\sum_{i=1}^{N} \lambda_i^n (\boldsymbol{\theta} \cdot \boldsymbol{\phi}_i)^2}{\sum_{i=1}^{N} (\boldsymbol{\theta} \cdot \boldsymbol{\phi}_i)^2} \tag{2.10}$$

where n is the number of steps (t/τ), N is the number of states, $\boldsymbol{\theta}$ is a vector of observables for each state, and $\boldsymbol{\phi}_i$ is the ith left eigenvector [16]. Unfortunately, this test cannot be used if you only have short simulations. One needs at least one long simulation compared to the relaxation time of the model to calculate the reference correlation function.

2.5 Coarse-Graining to Generate Macrostates

As discussed in Sect. 2.1, there are a number of advantages to coarse-graining microstate models by merging rapidly mixing microstates into larger macrostates. First of all, one can sometimes build mesoscale models that are just as quantitatively predictive as the original microstate model but are far more compact. Secondly, one can build models with few enough states that they are comprehendible and can be used to gain an intuition for a system and generate hypotheses, though they may no longer be quantitatively predictive.

Two major questions have to be addressed to build these coarse-grained models. First, how should one determine which microstates to merge together? Secondly, how many macrostates should one build?

Here, we review a number of methods that have been developed to answer these questions.

2.5.1 PCCA

Perron Cluster Cluster Analysis (PCCA) uses the eigenspectrum of a transition probability matrix to construct coarse-grained models [17, 18]. This method derives its name from the Perron-Frobenius theorem, which states that a real square matrix with positive entries (e.g. a transition probability matrix) has a unique largest real eigenvalue and that the corresponding eigenvector has strictly positive components. The term Perron Cluster refers to a set of eigenvalues clustered near the largest eigenvalue and separated from the rest of the eigenspectrum by a reasonable gap. As discussed shortly, the eigenvectors

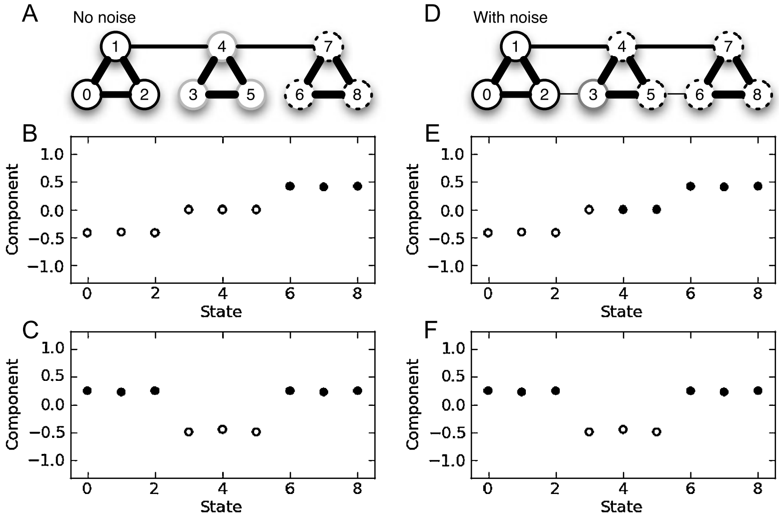


Fig. 2.5 Two simple models demonstrating the power and pitfalls of PCCA. (**A**) A simple model with well-sampled transitions. Each of the nine microstates has 1,000 self-transitions. Each *thick line* corresponds to 100 transitions and the *medium weight lines* correspond to 10 transitions. This model can be coarse-grained into three macrostates consisting of microstates 0–2 (*black outline*), 3–5 (*gray outline*), and 6–8 (*dotted outline*). Panels (**B**) and (**C**) show the second and third eigenvectors of this simple model, respectively. *Open circles* are used for eigenvector components that are less than or equal to zero and filled circles are used for components that are greater than zero. (**D**) The same simple model with two poorly sampled transitions (noise) added between states 2–3 and states 5–6. These transitions have only a single count. Their presence barely changes the model's eigenvectors. However, they alter the sign structure of the second eigenvector (panel (**E**)) and, therefore, PCCA finds the wrong coarse-graining into three macrostates: microstates 0–2 (*black outline*), 3 (*gray outline*) and 4–8 (*dotted outline*). Panels (**B**) and (**C**) show the second and third eigenvectors of the simple model from panel (**D**), respectively. If you try this model on your own computer, note that your results may vary slightly due to the symmetry of the system

corresponding to the Perron Cluster can be used to coarse-grain an MSM.

We begin with a discussion of PCCA because it highlights many of the issues that must be considered when coarse-graining MSMs. PCCA was also one of the first methods for coarse-graining MSMs and, therefore, provided at least some of the intellectual inspiration for many of the other methods. However, the other methods presented here are likely to perform better than PCCA, which does not provide a numerically robust clustering.

As discussed previously, the eigenvalues of a transition probability matrix can be converted into time scales. The corresponding eigenvectors describe what transitions are occurring on each of these time scales [17]. The largest eigenvalue (λ_1) is always 1 for a model that is connected and obeys detailed balance. The components of the corresponding eigenvector are proportional to the equilibrium populations of each state. The remaining eigenvalues ($\lambda_n < \lambda_{n-1} < \cdots < \lambda_2 < 1$) are real-valued and can be converted into time scales using Eq. (2.7). The corresponding right eigenvector describes what is happening on this time scale. That is, states with negative eigenvector components are interconverting with states with positive components and the magnitude of these components is proportional to the state's degree of participation. The left eigenvectors contain the same information but weighted by the equilibrium population of each state.

In PCCA, one starts off with all microstates merged into a single macrostate and then iteratively breaks the most kinetically diverse macrostate into two smaller states based on the next slowest right eigenvector [17, 18]. As an example, let's consider the model shown in Fig. 2.5A. By eye, it is clear this model can be divided into 3 macrostates: one containing microstates 0–2, one containing microstates 3–5, and one containing microstates 6–8. To find these states with PCCA, one would begin by using the second eigenvector (ψ_2, Fig. 2.5B) to split the microstates into one group with components less than or equal to zero (open circles in Fig. 2.5B) and one group with components greater than zero (filled circles in Fig. 2.5B). This splitting would give rise to two macrostates, one containing microstates 0–5 and another containing microstates 6–8. Next, PCCA chooses the group with the greatest spread in eigenvector components and uses the next slowest eigenvector (ψ_3) to divide this group in two. In this example, PCCA would select the group containing microstates 0–5 because it has states with components of ψ_2 ranging from about -0.4 to 0, whereas the other group only has states with components of about 0.4. Using ψ_3, PCCA would then split this group into two smaller groups containing microstates 0–2 and 3–5, respectively. This second split gives us the natural grouping into three macrostates we can see by eye, demonstrating the utility of this automated procedure. Further iterations of this algorithm could then be used to create more macrostates.

One attractive feature of this algorithm is that it provides a natural way to choose how many states to construct. If the system of interest has a well defined set of free energy basins—i.e. with large barriers between them and significantly smaller barriers within them—then the system will exhibit a separation of time scales. That is, there should be a Perron Cluster of eigenvalues near 1 that are separated from the rest of the eigenvalue spectrum by a reasonable gap. For example, the eigenvalues of our simple model from Fig. 2.5A are 1.0, 0.997, 0.992, 0.752, 0.750, 0.750, 0.750, 0.746, 0.735. There is a clear gap between the third and fourth eigenvalues of this model, and this gap would become even clearer after converting the eigenvalues into time scales. This gap indicates a separation of time scales that permits a three state macrostate model to capture the slowest relaxation time scales of the system. In general, if there is a gap after the nth eigenvalue (counting the eigenvalue of 1), then one should be able to construct a reasonable macrostate model with n states. Unfortunately, many real world systems do not have a clear separation of time scales. Instead, they have a continuum of eigenvalues. In such cases, the number of macrostates is best seen as an adjustable parameter one can vary depending on the properties of the model they are interested in.

One major limitation of PCCA is that it can suffer from propagation of error when not all microstates participate strongly in each eigenmode [19]. For example, in the simple model from Fig. 2.5A, microstates 3–5 have zero components in the second eigenvector, indicating they do not participate in the slowest relaxation process. This simple model is purposefully very clean, so all three of these microstates were placed in the same group during the first splitting PCCA performed based on the second eigenvector. However, in real world scenarios, one rarely gets states with eigenvector components of exactly zero because of factors like poorly sampled transitions. States that do not participate strongly in a given eigenmode (i.e. have eigenvector components that are nearly zero) will be assigned to macrostates rather arbitrarily, leading to compounding error as more eigenvectors are considered. For example, the simple model in Fig. 2.5D is the same as the one we've used so far but with the addition of two noisy transitions with only a single count between states 2–3 and states 5–6. In the second eigenvector of this model, microstates 3–5 have very small magnitude eigenvector components with different signs (Fig. 2.5E). Therefore, splitting this model into two groups based on the second eigenvector gives one set containing microstates 0–3 and another containing microstates 4–8. Despite the fact that microstates 3–5 should form a single macrostate, they have already been split apart at this early stage. As PCCA considers more eigenvectors, it will propagate this error. In more complicated models, new errors can also be introduced at each stage due to weakly participating states. Unfortunately, there is often a continuum of eigenvector components, so there is currently no clear way to separate weakly participating states from strongly participating ones and deal with the weakly participating ones separately.

A number of heuristic methods have been introduced to fix this problem but none are entirely satisfactory [1]. For example, intuitively, a partitioning into metastable states should maximize the self-transition probability of each state. Doing so is equivalent to minimizing the transition rates between states, or placing the boundaries between states along the largest free energy barriers. Therefore, one can try to correct for the errors introduced during PCCA by doing a simulated annealing procedure to maximize the total metastability (Q) of the model

$$Q = \sum_{n} T_{ii} \tag{2.11}$$

where n is the number of macrostates and T_{ii} is the self-transition probability for state i. In such a procedure, one tries randomly assigning microstates to new macrostates and each proposed move is accepted or rejected according to a Monte Carlo criterion. That is, moves that increase the metastability are always accepted and moves that reduce it are only accepted with some small probability. This procedure often works for simple models but becomes intractable for real world models because it can converge on non-sensical results or even completely fail to converge to a stable solution.

PCCA is also prone to handle poorly sampled states and transitions improperly because it does not account for statistical uncertainty in a model [21]. For example, it is common for the clustering algorithms described in Sect. 2.2 to make conformations that are geometric outliers into their own microstates [11]. These states will have very few transitions to other states and, therefore, will appear to be separated from them by large free energy barriers. As a result, PCCA will often make these poorly sampled microstates into singleton macrostates—i.e. macrostates containing a single microstate. Human examination of these states, however, often reveals that they are unlikely to be physically meaningful.

2.5.2 PCCA+

PCCA+ is a more robust version of PCCA that avoids the pitfall of propagation of error [19, 20]. This improvement is accomplished by considering the relevant eigenvectors simultaneously instead of sequentially. More specifically, PCCA+ tries to find a set of indicator functions that best reproduces the n slowest dynamical eigenvectors. For example, to construct a three-state macrostate model for the simple model in Fig. 2.5D, PCCA+ would consider the second and third eigenvectors simultaneously. PCCA+ would then fit these eigenvectors with three step functions: one that is 1 in states 0–2 and 0 elsewhere, a second that is 1 in states 3–5 and 0 elsewhere, and a third that is 1 in states 6–8 and 0 elsewhere. The details of how this optimization is achieved are very similar to spectral clustering and are described in Ref. [19].

While PCCA+ does not suffer from the propagation of error that occurs in PCCA, this method still relies on a maximum likelihood estimate of the transition probability matrix. Therefore, PCCA+ still tends to create singleton macrostates. Furthermore, PCCA+ can require quite a bit of memory, so creating mesoscale models is often computationally intractable.

2.5.3 SHC

Super-level-set hierarchical clustering (SHC) tries to deal with model uncertainty by treating low population states differently from high population ones [22]. Inspired by developments in topological data analysis, SHC first divides all the microstates into sets with similar populations (called level-sets) [23]. PCCA or PCCA+ is then used to divide each set into macrostates. Finally, overlap between the macrostates at each level is used to stitch these models together. Typically, PCCA(+) is not applied to the least populated states. Instead, these are just lumped into the macrostate they transition to most quickly, thereby avoiding creating singleton macrostates.

One added benefit of this approach is that the hierarchy of models SHC creates can give insight into the hierarchy of free energy basins that actually exist in the underlying free energy landscape.

For example, macrostates from the most populated level correspond to the deepest free energy basins. Some of the macrostates at less populated levels simply correspond to the same macrostates that are present at more populated levels. However, some reflect less populated intermediates between these deeper minima and can provide insight into how the system transitions between the most populated free energy basins.

SHC could benefit greatly from a more formal justification. One philosophical short coming is that SHC was inspired by methods that make use of density level sets (sets of data with approximately equal densities). However, as discussed earlier, estimating densities in high-dimensional spaces is extremely difficult. A more formal justification for breaking the microstates into levels would be preferable. One practical implication of this short-coming is that there is no clear way to define the level sets *a priori*. Instead, one must make a rather arbitrary choice and then try varying the density level sets to check for robustness. Therefore, SHC requires more computation than a single application of PCCA or PCCA+. A more formal justification for this method could provide insight into how to choose the density level sets and make this an extremely powerful method though. New work on framing SHC in terms of a Nystrom expansion of the transition probability matrix may provide such a formal justification.

2.5.4 BACE

More recently, a Bayesian agglomerative clustering engine (BACE) has been developed for dealing with uncertainty in a more automated fashion [21]. BACE exploits the observation that rapidly mixing states should also be kinetically similar—that is, they should have similar transition probabilities—to determine which states to lump together. The algorithm works by iteratively merging the most kinetically similar states, as judged by a Bayes factor for determining how likely the transitions observed for each state are to have come from the same underlying distribution

$$\ln \frac{P(\text{different}|C)}{P(\text{same}|C)} \approx \hat{C}_i \mathscr{D}(p_i \| q) + \hat{C}_j \mathscr{D}(p_j \| q) \tag{2.12}$$

where $P(\text{different}|C)$ is the probability the counts (C) from states i and j came from different underlying probability distributions and $P(\text{same}|C)$ is the probability they came from the same distribution, $\hat{C}_i$ is the number of counts originating in state i, $\mathscr{D}(p_i \| q) = \sum_k p_{ik} \ln \frac{p_{ik}}{q_k}$ is the relative entropy between probability distribution p_i and q, p_i is a vector of maximum likelihood transition probabilities from state i, and $q = \frac{\hat{C}_i p_i + \hat{C}_j p_j}{\hat{C}_i + \hat{C}_j}$ is the vector of expected transition probabilities from combining states i and j. Deriving this expression involves integrating over all possible transition probability distributions out of each state, so the method naturally takes into account uncertainty in the microstate model's transition probability matrix.

In addition to outperforming many other methods, BACE has an appealing information theoretic interpretation and provides a way to determine which levels of the hierarchy of models it creates are most deserving of further analysis. Specifically, Eq. (2.12) is identical to the Jensen-Shannon divergence, a popular measure from information theory [24]. Therefore, BACE can be interpreted as creating the coarse-graining that retains the maximum information about the original model's kinetics. The BACE Bayes factor also provides a means to determine how many states to create. One can simply monitor the Bayes factor as one merges states and watch for dramatic jumps. Models preceding these jumps are particularly deserving of further analysis because further coarse-graining greatly reduces the model quality.

Fully characterizing the strengths and weaknesses of BACE compared to other methods will require further application of this method. BACE is probably extremely useful for building mesoscale models because it can build them quickly and accurately. However, it may be less useful for building extremely coarse-grained models. The fewer the states one requests from BACE, the more iterations it must run and the longer the algorithm takes to complete. BACE could also suffer from propagation of error, as seen in PCCA, as a mistake early on will never be corrected later.

2.6 Recommended Protocol

At present, one of the most robust and widely used—but not necessarily optimal!—protocols for modeling proteins and other biomolecules is:

1. Cluster a simulation data set with the hybrid k-centers/k-medoids algorithm based on the RMSD between backbone heavy atoms, ensuring that every cluster has a radius on the order of a few Å.
2. Validate the kinetic relevance of this clustering and choose an appropriate lag time based on the model's relaxation time scales (or implied time scales) as a function of the lag time. For each lag time, estimate the transition matrices by:
   a. Removing states that only have one transition with any other state.
   b. Counting transitions using a sliding window and assuming that for every transition from state i to j, there is a corresponding transition from j to i.
3. Use the transition probability matrix at the desired lag time and representative conformations from each state to model experiments.
4. Coarse-grain the model with PCCA+ to gain an intuition for the system. Make sure to test how quantitative the coarse-grained model is by examining how closely the macrostate model's relaxation times agree with the microstate model.

2.7 Advanced Topics and Future Directions

Before moving on to the next chapter, I would like to briefly review a number of advanced topics and list some of the future directions that could lead to more effective methods for building MSMs.

2.7.1 Seeding

In seeding, one uses an inexpensive method to choose a variety of starting conformations for simulations that will later be used to build an MSM [25]. Intuitively, seeding allows one to explore a wider swath of conformational space more quickly than would be possible by starting every simulation from a single conformation, like a crystal structure. Ideally, one would like to find the greatest possible variety of relevant conformations. One way of doing this is to use generalized ensemble simulations like replica exchange to quickly explore conformational space and then use random conformations from these simulations as starting points for constant temperature simulations. This procedure helps focus the starting conformations on thermodynamically relevant regions of phase space. Less thermodynamically relevant conformations should quickly relax to more populated regions of conformational space.

When using seeding, one must be careful to ensure that the resulting model is connected. Simulations started from conformations that are too kinetically distant from any of the other starting points may never overlap with the rest of the trajectories, making it impossible to determine transition rates between them or their relative equilibrium populations.

2.7.2 Cores

Fluctuations at the tops of barriers between states can lead to recrossing events where the system appears to rapidly jump back and forth between the start and end state [26]. These recrossing events can lead to over-estimates of the transition rates between states if they are not dealt with properly.

Cores are one way of reducing the affect of recrossing [16, 27]. The basic idea is to define a core region within each state, leaving a no-man's land between each pair of states. Transitions are only counted when a trajectory leaves the core of one state and enters the core of another. A trajectory that makes an excursion into no-man's land but then returns to the core it started in before entering the core of any other state is said never to have left its initial state.

2.7.3 Comparing Multiple Sequences

In addition to providing insight into a single system, MSMs are also a powerful way of comparing different systems. For instance, in protein folding, there is great interest in comparing slight variations of a single protein and how the mutations that distinguish them change properties like the folding rate or stability of the native state. For this to be possible, it is essential that a common state space be used. For example, one can construct a common microstate space by clustering two protein sequences using a set of atoms that they share in common [28]. Properties like the equilibrium populations of a set of states or the transition rates between them can then be compared between the two systems.

2.7.4 Open Challenges

MSM methods are sufficiently well developed to pursue many exciting applications. However, there is still a great deal of room for further methodological improvements. Here, I list just a few of them.

1. As discussed previously, one would ideally like to build MSMs using a truly kinetic distance metric from the beginning. New clustering methods or distance metrics that better reflect a system's kinetics would be of tremendous value.
2. Many of the methods for validating MSMs and choosing important parameters, like the lag time, are very subjective. Quantitative approaches to model validation would allow for more automatic model construction.
3. More robust methods for estimating transition matrices that satisfy detailed balance would also be useful. Current methods are either too biased or too unreliable.
4. There is still a need for more efficient and accurate coarse-graining methods.

References

1. Chodera JD et al (2007) Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics. J Chem Phys 126:155101
2. Bowman GR, Huang X, Pande VS (2009) Using generalized ensemble simulations and Markov state models to identify conformational states. Methods 49:97–201
3. Noé F et al (2009) Constructing the equilibrium ensemble of folding pathways from short off-equilibrium simulations. Proc Natl Acad Sci USA 106:19011–19016
4. Beauchamp KA et al (2011) MSMBuilder2: modeling conformational dynamics on the picosecond to millisecond scale. J Chem Theory Comput 7:3412–3419
5. Prinz JH et al (2011) Markov models of molecular kinetics: generation and validation. J Chem Phys 134:174105
6. Senne M, Trendelkamp-Schroer B, Mey ASJ, Schütte C, Noé F (2012) EMMA—a software package for Markov model building and analysis. J Chem Theory Comput 8:2223
7. Silva DA, Bowman GR, Sosa-Peinado A, Huang X (2011) A role for both conformational selection and induced fit in ligand binding by the LAO protein. PLoS Comput Biol 7:e1002054
8. Keller B, Daura X, van Gunsteren WF (2010) Comparing geometric and kinetic cluster algorithms for molecular simulation data. J Chem Phys 132:074110
9. Gonzalez T (1985) Clustering to minimize the maximum intercluster distance. Theor Comput Sci 38:293
10. Dasgupta S, Long PM (2005) Performance guarantees for hierarchical clustering. J Comput Syst Sci 70:555
11. Bowman GR, Beauchamp KA, Boxer G, Pande VS (2009) Progress and challenges in the automated construction of Markov state models for full protein systems. J Chem Phys 131:124101
12. Noe F et al (2009) Constructing the equilibrium ensemble of folding pathways from short off-equilibrium simulations. Proc Natl Acad Sci USA 106:19011
13. Tarjan R (1972) Depth-first search and linear graph algorithms. SIAM J Comput 1:146
14. Swope WC, Pitera JW, Suits F (2004) Describing protein folding kinetics by molecular dynamics simulations, 1: theory. J Phys Chem B 108:6571
15. Bowman GR, Geissler PL (2012) Equilibrium fluctuations of a single folded protein reveal a multitude of potential cryptic allosteric sites. Proc Natl Acad Sci USA 109:11681
16. Buchete NV, Hummer G (2008) Coarse master equations for peptide folding dynamics. J Phys Chem B 112:6057

17. Schütte C, Fischer A, Huisinga W, Deuflhard P (1999) A direct approach to conformational dynamics based on hybrid Monte Carlo. J Comput Phys 151:146
18. Deuflhard P, Huisinga W, Fischer A, Schütte C (2000) Identification of almost invariant aggregates in reversible nearly uncoupled Markov chains. Linear Algebra Appl 315:39
19. Deuflhard P, Weber M (2005) Robust Perron cluster analysis in conformation dynamics. Linear Algebra Appl 398:161
20. Noé F, Horenko I, Schütte C, Smith JC (2007) Hierarchical analysis of conformational dynamics in biomolecules: transition networks of metastable states. J Chem Phys 126:155102
21. Bowman GR (2012) Improved coarse-graining of Markov state models via explicit consideration of statistical uncertainty. J Chem Phys 137:134111
22. Yao Y et al (2009) Topological methods for exploring low-density states in biomolecular folding pathways. J Chem Phys 130:144115
23. Singh G, Memoli F, Carlsson G. Mapper: a topological mapping tool for point cloud data. In: Eurographics symposium on point-based graphics
24. Lin J (1991) Divergence measures based on the Shannon entropy. IEEE Trans Inf Theory 37:145
25. Huang X, Bowman GR, Bacallado S, Pande VS (2009) Rapid equilibrium sampling initiated from nonequilibrium data. Proc Natl Acad Sci USA 106:19765
26. Bolhuis PG, Chandler D, Dellago C, Geissler PL (2002) Transition path sampling: throwing ropes over rough mountain passes, in the dark. Annu Rev Phys Chem 53:291
27. Schütte C et al (2011) Markov state models based on milestoning. J Chem Phys 134:204105
28. Levin AM et al (2012) Exploiting a natural conformational switch to engineer an interleukin-2 superkine. Nature 484:529

3 Markov Model Theory

Marco Sarich, Jan-Hendrik Prinz, and Christof Schütte

3.1 Continuous Molecular Dynamics

A variety of simulation models that all yield the same stationary properties, but have different dynamical behaviors, are available to study a given molecular model. The choice of the dynamical model must therefore be guided by both a desire to mimic the relevant physics for the system of interest (such as whether the system is allowed to exchange energy with an external heat bath during the course of dynamical evolution), balanced with computational convenience (e.g. the use of a stochastic thermostat in place of explicitly simulating a large external reservoir) [8]. Going into the details of these models is beyond the scope of the present study, and therefore we will simply state the minimal physical properties that we expect the dynamical model to obey. In the following we pursue the theoretical outline from Ref. [31] (Sects. 3.1–3.7) and Ref. [37] (Sects. 3.1–3.8) which should both be used for reference purposes.

Consider a state space Ω which contains all dynamical variables needed to describe the instantaneous state of the system. Ω may be discrete or continuous, and we treat the more general continuous case here. For molecular systems, Ω usually contains both positions and velocities of the species of interest and surrounding bath particles. $\mathbf{x}(t) \in \Omega$ will denote the state of the system at time t. The dynamical process considered is $(\mathbf{x}(t))_{t \in T}$, $T \subset \mathbb{R}_{0+}$, which is continuous in space, and may be either time-continuous (for theoretical investigations) or time-discrete (when considering time-stepping schemes for computational purposes). For the rest of the article, the dynamical process will also be denoted by $\mathbf{x}(t)$ for the sake of simplicity; we assume that $\mathbf{x}(t)$ has the following properties:

1. $\mathbf{x}(t)$ is a Markov process in the full state space Ω, i.e. the instantaneous change of the system ($d\mathbf{x}(t)/dt$ in time-continuous dynamics and $\mathbf{x}(t + \Delta t)$ in time-discrete dynamics with time step Δt), is calculated based on $\mathbf{x}(t)$ alone and does not require the previous history. In addition, we assume that the process is time-homogeneous, such that the transition probability density $p(\mathbf{x}, \mathbf{y}; \tau)$ for $\mathbf{x}, \mathbf{y} \in \Omega$ and $\tau \in \mathbb{R}_{0+}$ is well-defined:

$$p(\mathbf{x}, A; \tau) = \mathbb{P}\big[\mathbf{x}(t + \tau) \in A \,\big|\, \mathbf{x}(t) = \mathbf{x}\big] \tag{3.1}$$

i.e. the probability that a trajectory started at time t from the point $\mathbf{x} \in \Omega$ will be in set A at time $t + \tau$. Such a transition probability density for the diffusion process in a one-dimensional potential is depicted in Fig. 3.1b. Whenever $p(\mathbf{x}, A; \tau)$ has an absolutely continuous probability density $p(\mathbf{x}, \mathbf{y}; \tau)$ it is given by integrating the transition probability density over region A:

M. Sarich · J.-H. Prinz · C. Schütte (✉)
Freie Universität Berlin, Arnimallee 6, 14195 Berlin, Germany
e-mail: christof.schuette@fu-berlin.de

G.R. Bowman et al. (eds.), *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology 797,
DOI 10.1007/978-94-007-7606-7_3, © Springer Science+Business Media Dordrecht 2014

$$p(\mathbf{x}, A; \tau) = \mathbb{P}\big[\mathbf{x}(t+\tau) \in A \,\big|\, \mathbf{x}(t) = \mathbf{x}\big] \tag{3.2}$$

$$= \int_A d\mathbf{y}\, p(\mathbf{x}, \mathbf{y}; \tau). \tag{3.3}$$

2. $\mathbf{x}(t)$ is ergodic, i.e., the process $\mathbf{x}(t)$ is aperiodic, the space Ω does not have two or more subsets that are dynamically disconnected, and for $t \to \infty$ each state $\mathbf{x}$ will be visited infinitely often. The running average of a function $f : \Omega \to \mathbb{R}^d$ then is given by a unique stationary density $\mu(\mathbf{x})$ in the sense that for almost every initial state $\mathbf{x}$ we have

$$\lim_{T\to\infty} \frac{1}{T} \int_0^T dt\, f\big(\mathbf{x}(t)\big) = \int_\Omega d\mathbf{x}\, f(\mathbf{x})\mu(\mathbf{x}),$$

that is, the fraction of time that the system spends in any of its states during an infinitely long trajectory is given by the stationary density (invariant measure) $\mu(\mathbf{x}) : \Omega \to \mathbb{R}_{0+}$ with $\int_\Omega d\mathbf{x}\, \mu(\mathbf{x}) = 1$, where the stationarity of the density means that

$$\int_A d\mathbf{x}\, \mu(\mathbf{x}) = \int_\Omega d\mathbf{x}\, p(\mathbf{x}, A; \tau)\mu(\mathbf{x}),$$

which takes the simpler form

$$\mu(\mathbf{y}) = \int_\Omega d\mathbf{x}\, p(\mathbf{x}, \mathbf{y}; \tau)\mu(\mathbf{x}),$$

whenever the transition probability has a density. We assume that this stationary density μ is unique. In cases relevant for molecular dynamics the stationary density always corresponds to the equilibrium probability density for some associated thermodynamic ensemble (e.g. NVT, NpT). For molecular dynamics at constant temperature T, the dynamics above yield a stationary density $\mu(\mathbf{x})$ that is a function of T, namely the Boltzmann distribution

$$\mu(\mathbf{x}) = Z(\beta)^{-1} \exp\big(-\beta H(\mathbf{x})\big) \tag{3.4}$$

with Hamiltonian $H(\mathbf{x})$ and $\beta = 1/k_B T$ where k_B is the Boltzmann constant and $k_B T$ is the thermal energy. $Z(\beta) = \int d\mathbf{x}\, \exp(-\beta H(\mathbf{x}))$ is the partition function. By means of illustration, Fig. 3.1a shows the stationary density $\mu(\mathbf{x})$ for a diffusion process on a potential with high barriers.

3. $\mathbf{x}(t)$ is reversible, i.e., $p(\mathbf{x}, \mathbf{y}; \tau)$ fulfills the condition of *detailed balance*:

$$\mu(\mathbf{x}) p(\mathbf{x}, \mathbf{y}; \tau) = \mu(\mathbf{y}) p(\mathbf{y}, \mathbf{x}; \tau), \tag{3.5}$$

i.e., in equilibrium, the fraction of systems transitioning from $\mathbf{x}$ to $\mathbf{y}$ per time is the same as the fraction of systems transitioning from $\mathbf{y}$ to $\mathbf{x}$. Note that this "reversibility" is a more general concept than the time-reversibility of the dynamical equations e.g. encountered in Hamiltonian dynamics. For example, Brownian dynamics on some potential are reversible as they fulfill Eq. (3.5), but are not time-reversible in the same sense as Hamiltonian dynamics are, due to the stochasticity of individual realizations. Although detailed balance is not essential for the construction of Markov models, we will subsequently assume detailed balance as this allows much more profound analytical statements to be made, and just comment on generalizations here and there. The rationale is that one typically expects detailed balance to be fulfilled in equilibrium molecular dynamics based on a simple physical argument: For dynamics that have no detailed balance, there exists a set of states which form a loop in state space which is traversed in one direction with higher probability than in the reverse direction. This means that one could design a machine which uses this preference of direction in order to produce work. However, a system in equilibrium is driven only by thermal energy, and conversion of pure thermal energy into work contradicts the second law of thermodynamics. Thus, this argument concludes that equilibrium molecular dynamics must be reversible and fulfill detailed balance. Despite the popularity of this argument there are dynamical processes used in molecular dynamics that do *not* satisfy detailed balance in the above sense. Langevin molecular dynamics may be the most prominent example. However, the Langevin process exhibits an *extended detailed balance* [18]

$$\mu(\mathbf{x}) p(\mathbf{x}, \mathbf{y}; \tau) = \mu(A\mathbf{y}) p(A\mathbf{y}, A\mathbf{x}; \tau),$$

where A is the linear operation that flips the sign of the momenta in the state $\mathbf{x}$. This prop-

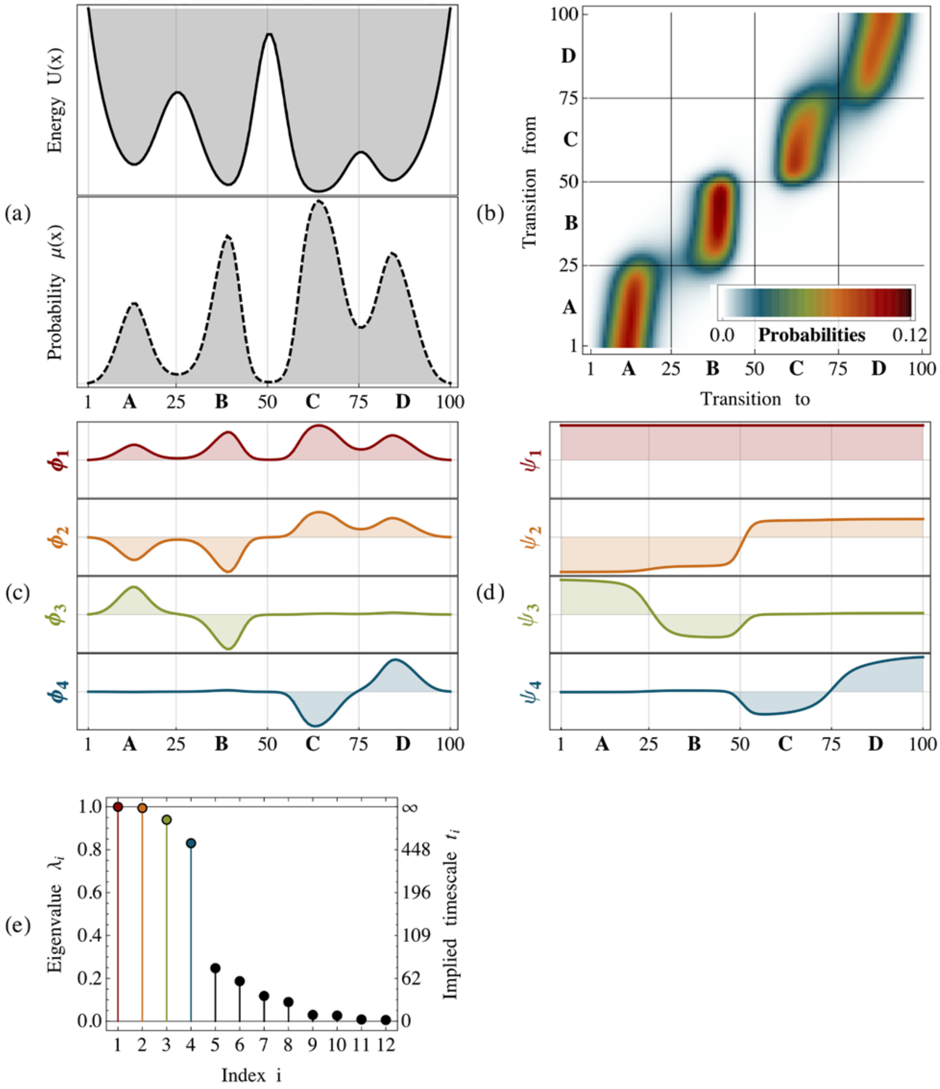


Fig. 3.1 (**a**) Potential energy function with four metastable states and corresponding stationary density $\mu(\mathbf{x})$. (**b**) Density plot of the transfer operator for a simple diffusion-in-potential dynamics defined on the range $\Omega = [0, 100]$, *black* and *red* indicates high transition probability, *white* zero transition probability. Of particular interest is the nearly block-diagonal structure, where the transition density is large within blocks allowing rapid transitions within metastable basins, and small or nearly zero for jumps between different metastable basins. (**c**) The four dominant eigenfunctions of the transfer operator, $\psi_1, \dots, \psi_4$, which indicate the associated dynamical processes. The first eigenfunction is associated to the stationary process, the second to a transition between $A + B \leftrightarrow C + D$ and the third and fourth eigenfunction to transitions between $A \leftrightarrow B$ and $C \leftrightarrow D$, respectively. (**d**) The four dominant eigenfunctions of the transfer operator weighted with the stationary density, $\phi_1, \dots, \phi_4$. (**e**) Eigenvalues of the transfer operator, The gap between the four metastable processes ($\lambda_i \approx 1$) and the fast processes is clearly visible

erty allows to reproduce the results for processes with detailed balance to this case, see remarks below.

The above conditions do not place overly burdensome restrictions on the choices of dynamical models used to describe equilibrium dynamics. Many stochastic thermostats are consistent with the above assumptions, e.g. Hybrid Monte Carlo [14, 37], overdamped Langevin (also called Brownian or Smoluchowski) dynamics [15, 16], and stepwise-thermalized Hamiltonian dynamics [42]. When simulating solvated systems, a weak friction or collision rate can be used; this can often be selected in a manner that is physically motivated by the heat conductivity of the material of interest and the system size [1].

We note that the use of finite-timestep integrators for these models of dynamics can sometimes be problematic, as the phase space density sampled can differ from the density desired. Generally, integrators based on symplectic Hamiltonian integrators (such as velocity Verlet [42]) offer greater stability for our purposes.

While technically, a Markov model analysis can be constructed for any choice of dynamical model, it must be noted that several popular dynamical schemes violate the assumptions above, and using them means that one is (currently) doing so without a solid theoretical basis, e.g. regarding the boundedness of the discretization error analyzed in Sect. 3.3 below. For example, Nosé-Hoover and Berendsen are either not ergodic or do not generate the correct stationary distribution for the desired ensemble [10]. Energy-conserving Hamiltonian dynamics on one hand may well be ergodic regarding the projected volume measure on the energy surface but this invariant measure is *not* unique, and on the other hand it is *not* ergodic wrt. the equilibrium probability density for some associated thermodynamic ensemble of interest.

3.2 Transfer Operator Approach and the Dominant Spectrum

At this point we shift from focusing on the evolution of individual trajectories to the time evolution of an ensemble density. Consider an ensemble of molecular systems at a point in time t, distributed in state space Ω according to a probability density $p_t(\mathbf{x})$ that is different from the stationary density $\mu(\mathbf{x})$. If we now wait for some time τ, the probability distribution of the ensemble will have changed because each system copy undergoes transitions in state space according to the transition probability density $p(\mathbf{x},\mathbf{y};\tau)$. The change of the probability density $p_t(\mathbf{x})$ to $p_{t+\tau}(\mathbf{x})$ can be described with the action of a continuous operator. From a physical point of view, it seems straightforward to define the *propagator* $\mathcal{Q}(\tau)$ as follows:

$$\begin{aligned} p_{t+\tau}(\mathbf{y}) &= \mathcal{Q}(\tau)\circ p_t(\mathbf{y}) && (3.6)\\ &= \int_\Omega d\mathbf{x}\, p(\mathbf{x},\mathbf{y};\tau)p_t(\mathbf{x}). && (3.7)\end{aligned}$$

Applying $\mathcal{Q}(\tau)$ to a probability density $p_t(\mathbf{x})$ will result in a modified probability density $p_{t+\tau}(\mathbf{x})$ that is more similar to the stationary density $\mu(\mathbf{x})$, to which the ensemble must relax after infinite time. An equivalent description is provided by the *transfer operator* $\mathcal{T}(\tau)$ [36, 37], which has nicer properties from a mathematical point of view. $\mathcal{T}(\tau)$ is defined as [35, 36, 39]:

$$\begin{aligned} u_{t+\tau}(\mathbf{y}) &= \mathcal{T}(\tau)\circ u_t(\mathbf{y}) && (3.8)\\ &= \frac{1}{\mu(\mathbf{y})}\int_\Omega d\mathbf{x}\, p(\mathbf{x},\mathbf{y};\tau)\mu(\mathbf{x})u_t(\mathbf{x}). && (3.9)\end{aligned}$$

$\mathcal{T}(\tau)$ does not propagate probability densities, but instead functions $u_t(\mathbf{x})$ that differ from probability densities by a factor of the stationary density $\mu(\mathbf{x})$, i.e.:

$$p_t(\mathbf{x}) = \mu(\mathbf{x})u_t(\mathbf{x}). \tag{3.10}$$

The relationship between the two densities and operators is shown in the scheme below:

$$\begin{tikzcd} p_t \arrow[r, "\mathcal{Q}(\tau)"] \arrow[d, "\cdot\mu^{-1}"'] & p_{t+\tau} & \text{probability densities} \\ u_t \arrow[r, "\mathcal{T}(\tau)"] & u_{t+\tau} \arrow[u, "\cdot\mu"'] & \text{densities in } \mu\text{-weighted space} \end{tikzcd}$$

It is important to note that $\mathcal{Q}$ and $\mathcal{T}$ in fact do *not* only propagate probability densities but *general* functions $f:\Omega\to\mathbb{R}$. Since both operators have the property to conserve positivity and mass, a probability density is always transported into a probability density.

Alternatively to $\mathcal{Q}$ and $\mathcal{T}$ which describe the transport of densities exactly by a chosen time-discretization τ, one could investigate the density transport with a time-continuous operator $\mathcal{L}$ called *generator* which is the continuous basis of rate matrices that are frequently used in physical chemistry [5, 40, 41] and is related to the Fokker-Planck equation [22, 36]. Here, we do not investigate $\mathcal{L}$ in detail, but only point out that the existence of a generator implies that we have

$$\mathcal{T}(\tau) = \exp(\tau\mathcal{L}), \tag{3.11}$$

with $\mathcal{L}$ acting on the same μ-weighted space as $\mathcal{T}$, while $\mathcal{Q}(\tau) = \exp(\tau L)$ for L acting on the unweighted densities/functions. The so-called semigroup-property (3.11) implies that $\mathcal{T}(\tau)$ and $\mathcal{L}$ have the same eigenvectors, while the eigenvalues λ of $\mathcal{T}(\tau)$ and the eigenvalues η of $\mathcal{L}$ are related via $\lambda = \exp(\tau\eta)$. This is of importance since most of the following considerations using $\mathcal{T}(\tau)$ can be generalized to $\mathcal{L}$.

Equation (3.9) is a formal definition. When the particular kind of dynamics is known it can be written in a more specific form [37]. However, the general form (3.9) is sufficient for the present analysis. The continuous operators have the following general properties:

- Both $\mathcal{Q}(\tau)$ and $\mathcal{T}(\tau)$ fulfill the Chapman–Kolmogorov Equation

$$p_{t+k\tau}(\mathbf{x}) = \left[\mathcal{Q}(\tau)\right]^k \circ p_t(\mathbf{x}), \tag{3.12}$$

$$u_{t+k\tau}(\mathbf{x}) = \left[\mathcal{T}(\tau)\right]^k \circ u_t(\mathbf{x}) \tag{3.13}$$

  where $[\mathcal{T}(\tau)]^k$ refers to the k-fold application of the operator, i.e. $\mathcal{Q}(\tau)$ and $\mathcal{T}(\tau)$ can be used to propagate the evolution of the dynamics to arbitrarily long times $t + k\tau$.
- We consider the two operators on the Hilbert space of square integrable functions. More specifically, we work with two Hilbert spaces, one with unweighted functions,

$$L^2 = \left\{ u : \Omega \to \mathbb{C} : \right.$$
$$\left. \|u\|_2^2 = \int_\Omega d\mathbf{x}\, \left|u(\mathbf{x})\right|^2 < \infty \right\},$$

  in which we consider $\mathcal{Q}$, the other with μ-weighted functions

$$L_\mu^2 = \left\{ u : \Omega \to \mathbb{C} : \right.$$
$$\left. \|u\|_{2,\mu}^2 = \int_\Omega d\mathbf{x}\, \left|u(\mathbf{x})\right|^2 \mu(\mathbf{x}) < \infty \right\},$$

  where we consider $\mathcal{T}$. These spaces come with the following two scalar products

$$\langle u, v\rangle = \int_\Omega d\mathbf{x}\, u(\mathbf{x})^* v(\mathbf{x}),$$
$$\langle u, v\rangle_\mu = \int_\Omega d\mathbf{x}\, u(\mathbf{x})^* v(\mathbf{x})\mu(\mathbf{x}),$$

  where the star indicates complex conjugation.
- $\mathcal{Q}(\tau)$ has eigenfunctions $\phi_i(\mathbf{x})$ and associated eigenvalues λ_i (see Figs. 3.1c and e):

$$\mathcal{Q}(\tau) \circ \phi_i(\mathbf{x}) = \lambda_i \phi_i(\mathbf{x}), \tag{3.14}$$

  while $\mathcal{T}(\tau)$ has eigenfunctions $\psi_i(\mathbf{x})$ with the same corresponding eigenvalues:

$$\mathcal{T}(\tau) \circ \psi_i(\mathbf{x}) = \lambda_i \psi_i(\mathbf{x}). \tag{3.15}$$

  When the dynamics are reversible, all eigenvalues λ_i are real-valued and lie in the interval $-1 < \lambda_i \leq 1$ [36, 37] (this is only true in L_μ^2 and not in the other function spaces). Moreover, the two types of eigenfunctions are related by a factor of the stationary density $\mu(\mathbf{x})$:

$$\phi_i(\mathbf{x}) = \mu(\mathbf{x})\psi_i(\mathbf{x}), \tag{3.16}$$

  and their lengths are defined by the normalization condition that the scalar product is unity for all corresponding eigenfunctions:

$$\langle \phi_i, \psi_i\rangle = \langle \psi_i, \psi_i\rangle_\mu = 1$$

  for all $i = 1 \ldots m$. Due to reversibility, non-corresponding eigenfunctions are orthogonal:

$$\langle \phi_i, \psi_j\rangle = 0$$

  for all $i \neq j$. When $\mathcal{T}(\tau)$ is approximated by a reversible transition matrix on a discrete state space, $\phi_i(\mathbf{x})$ and $\psi_i(\mathbf{x})$ are approximated by the left and right eigenvectors of that transition matrix, respectively (compare Figs. 3.1c and d).

- In general the spectrum of the two operators contains a continuous part, called the essential spectrum, and a discrete part, called the discrete spectrum that contains only isolated eigenvalues [19]. The essential spectral radius $0 \leq r \leq 1$ is the minimal value such that for all elements λ of the essential spectrum we have $|\lambda| \leq r$. In all of the following we assume that the essential spectral radius is bounded away from 1 in L^2_μ, that is, $0 \leq r < 1$. Then, every element λ of the spectrum with $|\lambda| > r$ is in the discrete spectrum, i.e., is an isolated eigenvalue for which an eigenvector exists. Our assumption is not always satisfied but is a condition on the dynamics: For example, for deterministic Hamiltonian systems it is $r = 1$, while for Langevin dynamics with periodic boundary conditions or with fast enough growing potential at infinity, we have $r = 0$. In the following, we assume $r < 1$ and ignore the essential spectrum; we only consider a finite number of m isolated, *dominant* eigenvalue/eigenfunction pairs and sort them by non-ascending eigenvalue, i.e. $\lambda_1 = 1 > \lambda_2 \geq \lambda_3 \geq \cdots \geq \lambda_m$, with $r < |\lambda_m|$. In addition we assume that the largest eigenvalue $\lambda = 1$ is simple so that μ is the only invariant measure.
- The eigenfunction associated with the largest eigenvalue $\lambda = 1$ corresponds to the stationary distribution $\mu(\mathbf{x})$ (see Fig. 3.1d, top):
$$\mathscr{Q}(\tau) \circ \mu(\mathbf{x}) = \mu(\mathbf{x}) = \phi_1(\mathbf{x}), \tag{3.17}$$
and the corresponding eigenfunction of $\mathscr{T}(\tau)$) is a constant function on all state space Ω (see Fig. 3.1c, top):
$$\mathscr{T}(\tau) \circ \mathbf{1} = \mathbf{1} = \psi_1(\mathbf{x}), \tag{3.18}$$

due to the relationship $\phi_1(\mathbf{x}) = \mu(\mathbf{x})\psi_1(\mathbf{x}) = \mu(\mathbf{x})$.

To see the significance of the other eigenvalue/eigenfunction pairs, we exploit that the dynamics can be decomposed exactly into a superposition of m individual slow dynamical processes and the remaining fast processes. For $\mathscr{T}(\tau)$, this yields:

$$\begin{aligned} u_{t+k\tau}(\mathbf{x}) &= \mathscr{T}_{\text{slow}}(k\tau) \circ u_t(\mathbf{x}) \\ &\quad + \mathscr{T}_{\text{fast}}(k\tau) \circ u_t(\mathbf{x}) && (3.19) \\ &= \sum_{i=1}^{m} \lambda_i^k \langle u_t, \phi_i \rangle \psi_i(\mathbf{x}) \\ &\quad + \mathscr{T}_{\text{fast}}(k\tau) \circ u_t(\mathbf{x}) && (3.20) \\ &= \sum_{i=1}^{m} \lambda_i^k \langle u_t, \psi_i \rangle_\mu \psi_i(\mathbf{x}) \\ &\quad + \mathscr{T}_{\text{fast}}(k\tau) \circ u_t(\mathbf{x}). && (3.21) \end{aligned}$$

Here, $\mathscr{T}_{\text{slow}}$ is the *dominant*, or slowly-decaying part consisting of the m slowest processes with $\lambda_i \geq \lambda_m$, while $\mathscr{T}_{\text{fast}}$ contains all (infinitely many) fast processes that are usually not of interest and which decay with geometric rate at least as fast as $|\lambda_{m+1}|^k$:

$$\frac{\|\mathscr{T}_{\text{fast}}(k\tau) \circ u_t\|^2_{2,\mu}}{\|u_t\|^2_{2,\mu}} \leq |\lambda_{m+1}|^k.$$

This decomposition requires that subspaces $\mathscr{T}_{\text{slow}}$ and $\mathscr{T}_{\text{fast}}$ are orthogonal, which is a consequence of detailed balance. This decomposition permits a compelling physical interpretation: The slow dynamics are a superposition of dynamical processes, each of which can be associated to one eigenfunction ψ_i (or ϕ_i) and a corresponding eigenvalue λ_i (see Figs. 3.1c–e). These processes decay with increasing time index k. In the long-time limit where $k \to \infty$, only the first term with $\lambda_1 = 1$ remains, recovering to the stationary distribution $\phi_1(\mathbf{x}) = \mu(\mathbf{x})$. All other terms correspond to processes with eigenvalues $\lambda_i < 1$ and decay over time, thus the associated eigenfunctions correspond to processes that decay under the action of the dynamics and represent the dynamical rearrangements taking place while the ensemble relaxes towards the equilibrium distribution. The closer λ_i is to 1, the slower the corresponding process decays; conversely, the closer it is to 0, the faster.

Thus the λ_i for $i = 2, \ldots, m$ each correspond to a physical timescale, indicating how quickly the process decays or transports density toward equilibrium (see Fig. 3.1e):

$$t_i = -\frac{\tau}{\ln \lambda_i}, \tag{3.22}$$

which is often called the ith implied timescale [8, 42]. Thus, Eq. (3.19) can be rewritten in terms of implied timescales as:

$$u_{t+k\tau}(\mathbf{x}) = 1 + \sum_{i=2}^{m} \exp\left(-\frac{k\tau}{t_i}\right) \langle u_t, \psi_i \rangle_\mu \psi_i(\mathbf{x}) \\ + \mathscr{T}_{\text{fast}}(k\tau) \circ u_t(\mathbf{x}). \tag{3.23}$$

This implies that when there are gaps amongst the first m eigenvalues, the system has dynamical processes acting simultaneously on different timescales. For example, a system with two-state kinetics would have $\lambda_1 = 1$, $\lambda_2 \approx 1$ and $\lambda_3 \ll \lambda_2$ ($t_3 \ll t_2$), while a system with a clear involvement of an additional kinetic intermediate would have $\lambda_3 \sim \lambda_2$ ($t_3 \sim t_2$).

In Fig. 3.1, the second process, ψ_2, corresponds to the slow ($\lambda_2 = 0.9944$) exchange between basins $A + B$ and basins $C + D$, as reflected by the opposite signs of the elements of ψ_2 in these regions (Fig. 3.1c). The next-slowest processes are the $A \leftrightarrow B$ transition and then the $C \leftrightarrow D$ transition, while the subsequent eigenvalues are clearly separated from the dominant spectrum and correspond to much faster local diffusion processes. The three slowest processes effectively partition the dynamics into four metastable states corresponding to basins A, B, C and D, which are indicated by the different sign structures of the eigenfunctions (Fig. 3.1c). The metastable states can be calculated from the eigenfunction structure, e.g. using the PCCA method [11, 12, 32].

Of special interest is the slowest relaxation time, t_2. This timescale identifies the *worst case* global equilibration or decorrelation time of the system; no structural observable can relax more slowly than this timescale. Thus, if one desires to calculate an expectation value $\mathbb{E}(a)$ of an observable $a(\mathbf{x})$ which has a non-negligible overlap with the second eigenfunction, $\langle a, \psi_2 \rangle > 0$, a straightforward single-run MD trajectory would need to be many times t_2 in length in order to compute an unbiased estimate of $\mathbb{E}(a)$.

3.3 Discretization of State Space

While molecular dynamics in full continuous state space Ω is Markovian by construction, the term *Markov State Model* (MSM) or shortly *Markov model* is due to the fact that in practice, state space must be somehow discretized in order to obtain a computationally tractable description of the dynamics as it has first been introduced in [35]. The Markov model then consists of the partitioning of state space used together with the transition matrix modeling the jump process of the observed trajectory projected onto these discrete states. However, this jump process (Fig. 3.2) is no longer Markovian, as the information where the continuous process would be within the local discrete state is lost in the course of discretization. The jump statistics generated by the projection, however, defines a Markov process on the discrete state space associated with the partition. Modeling the long-time statistics of the original process with this discrete state space Markov process is an approximation, i.e., it involves a *discretization error*. In the current section, this discretization error is analyzed and it is shown what needs to be done in order to keep it small.

The discretization error is a *systematic error* of a Markov model since it causes a deterministic deviation of the Markov model dynamics from the true dynamics that persists even when the statistical error is excluded by excessive sampling. In order to focus on this effect alone, it is assumed in this section that the statistical estimation error is zero, i.e., transition probabilities between discrete states can be calculated exactly. The results suggest that the discretization error of a Markov model can be made small enough for the MSM to be useful in accurately describing the relaxation kinetics, even for very large and complex molecular systems. This approach is illustrated in Fig. 3.3.

In practical use, the Markov model is not obtained by actually discretizing the continuous propagator. Rather, one defines a discretization of state space and then estimates the corresponding discretized transfer operator from a finite quantity of simulation data, such as several long or many short MD trajectories that transition between the

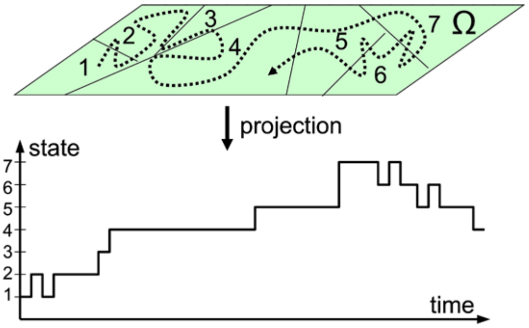


Fig. 3.2 Scheme: The true continuous dynamics (*dashed line*) is projected onto the discrete state space. MSMs approximate the resulting jump process by a Markov jump process

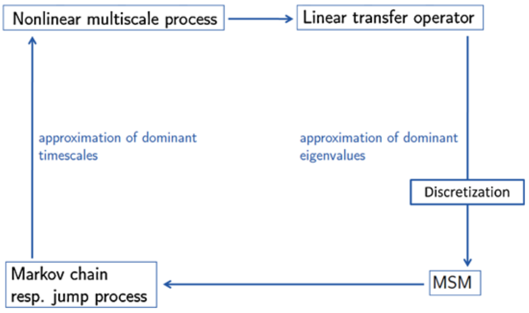

Nonlinear multiscale process
Linear transfer operator
approximation of dominant timescales
approximation of dominant eigenvalues
Discretization
Markov chain resp. jump process
MSM

Fig. 3.3 Illustration of our approach: The continuous dynamics is highly nonlinear and has many scales. It is represented by the linear propagator $\mathscr{T}$, whose discretization yields a finite-dimensional transition matrix that represents the Markov State Model (MSM). If the discretization error is small enough, the Markov chain or jump process induced by the MSM is a good approximation of the dominant timescales of the original continuous dynamics

discrete states. The statistical estimation error involved in this estimation will be discussed in the subsequent chapters; the rest of the current chapter focuses only on the approximation error due to discretization of the transfer operator.

Here we consider a discretization of state space Ω into n sets. In practice, this discretization is often a simple partition with sharp boundaries, but in some cases it may be desirable to discretize Ω into fuzzy sets [46]. We can describe both cases by defining membership functions $\chi_i(\mathbf{x})$ that quantify the probability of point $\mathbf{x}$ to belong to set i [47] which have the property $\sum_{i=1}^{n} \chi_i(\mathbf{x}) = 1$. We will concentrate on a crisp partitioning with step functions:

$$\chi_i(\mathbf{x}) = \chi_i^{\text{crisp}}(\mathbf{x}) = \begin{cases} 1 & \mathbf{x} \in S_i, \\ 0 & \mathbf{x} \notin S_i. \end{cases} \tag{3.24}$$

Here we have used n sets $S = \{S_1, \ldots, S_n\}$ which entirely partition state space ($\bigcup_{i=1}^{n} S_i = \Omega$) and have no overlap ($S_i \cap S_j = \emptyset$ for all $i \neq j$). A typical example of such a crisp partitioning is a Voronoi tessellation [45], where one defines n centers $\bar{\mathbf{x}}_i$, $i = 1 \ldots n$, and set S_i is the union of all points $\mathbf{x} \in \Omega$ which are closer to $\bar{\mathbf{x}}_i$ than to any other center using some distance metric (illustrated in Figs. 3.4b and c). Note that such a discretization may be restricted to some subset of the degrees of freedom, e.g. in MD one often ignores velocities and solvent coordinates when discretizing.

The stationary probability π_i to be in set i is then given by the full stationary density as:

$$\pi_i = \int_{\mathbf{x} \in S_i} d\mathbf{x}\, \mu(\mathbf{x}),$$

and the local stationary density $\mu_i(\mathbf{x})$ restricted to set i (see Fig. 3.4b) is given by

$$\mu_i(\mathbf{x}) = \begin{cases} \frac{\mu(\mathbf{x})}{\pi_i} & \mathbf{x} \in S_i, \\ 0 & \mathbf{x} \notin S_i. \end{cases} \tag{3.25}$$

These properties are local, i.e. they do not require information about the full state space.

3.4 Transition Matrix

Together with the discretization, the Markov model is defined by the row-stochastic transition probability matrix, $\mathbf{T}(\tau) \in \mathbb{R}^{n \times n}$, which is the discrete approximation of the transfer operator described in Sect. 3.2 *via*:

$$T_{ij}(\tau) = \frac{\langle \chi_j, (\mathscr{T}(\tau) \circ \chi_i) \rangle_\mu}{\langle \chi_i, \chi_i \rangle_\mu} \tag{3.26}$$

Physically, each element $T_{ij}(\tau)$ represents the time-stationary probability to find the system in state j at time $t + \tau$ given that it was in state i at time t. By definition of the conditional probability, this is equal to:

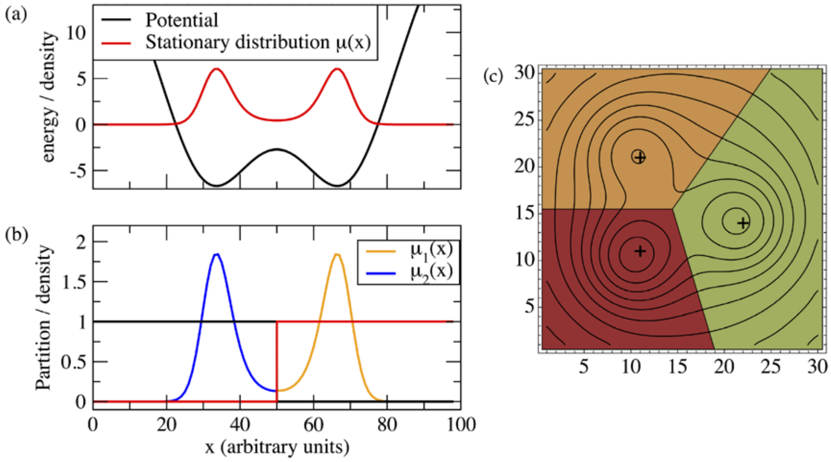


Fig. 3.4 Crisp state space discretization illustrated on a one-dimensional two-well and a two-dimensional three-well potential. (**a**) Two-well potential (*black*) and stationary distribution $\mu(\mathbf{x})$ (*red*). (**b**) Characteristic functions $v_1(\mathbf{x}) = \chi_1(\mathbf{x})$, $v_2(\mathbf{x}) = \chi_2(\mathbf{x})$ (*black* and *red*). This discretization has the corresponding local densities $\mu_1(\mathbf{x})$, $\mu_2(\mathbf{x})$ (*blue* and *yellow*), see Eq. (3.25). (**c**) Three-well potential (black contours indicate the isopotential lines) with a crisp partitioning into three states using a Voronoi partition with the centers denoted (+)

$$T_{ij}(\tau) = \mathbb{P}\big[\mathbf{x}(t+\tau) \in S_j \,\big|\, \mathbf{x}(t) \in S_i\big] \tag{3.27}$$

$$= \frac{\mathbb{P}[\mathbf{x}(t+\tau) \in S_j \cap \mathbf{x}(t) \in S_i]}{\mathbb{P}[\mathbf{x}(t) \in S_i]} \tag{3.28}$$

$$= \frac{\int_{S_i} d\mathbf{x}\, \mu_i(\mathbf{x}) p(\mathbf{x}, S_j; \tau)}{\int_{\mathbf{x}} d\mathbf{x}\, \mu_i(\mathbf{x})}, \tag{3.29}$$

where we have used Eq. (3.2). Note that in this case the integrals run over individual sets and only need the local equilibrium distributions $\mu_i(\mathbf{x})$ as weights. This is a very powerful feature: In order to estimate transition probabilities, we do not need any information about the global equilibrium distribution of the system, and the dynamical information needed extends only over time τ. In principle, the full dynamical information of the discretized system can be obtained by initiating trajectories of length τ out of each state i as long as we draw the starting points of these simulations from a local equilibrium density $\mu_i(\mathbf{x})$ [24, 37, 47].

The transition matrix can also be written in terms of correlation functions [42]:

$$T_{ij}(\tau) = \frac{\mathbb{E}[\chi_i(\mathbf{x}(t))\chi_j(\mathbf{x}(t+\tau))]}{\mathbb{E}[\chi_i(\mathbf{x}(t))]} = \frac{c_{ij}^{\mathrm{corr}}(\tau)}{\pi_i}, \tag{3.30}$$

where the unconditional transition probability $c_{ij}^{\mathrm{corr}}(\tau) = \pi_i T_{ij}(\tau)$ is an equilibrium time correlation function which is normalized such that $\sum_{i,j} c_{ij}^{\mathrm{corr}}(\tau) = 1$. For dynamics fulfilling detailed balance, the correlation matrix is symmetric ($c_{ij}^{\mathrm{corr}}(\tau) = c_{ji}^{\mathrm{corr}}(\tau)$).

Since the transition matrix $\mathbf{T}(\tau)$ is a discretization of the transfer operator $\mathscr{T}$ [35, 36, 36, 37] (Sect. 3.2), we can relate the functions that are transported by $\mathscr{T}$ (functions u_t in Eq. (3.8)) to column vectors that are multiplied to the matrix from the right while the probability densities p_t (Eq. (3.10)) correspond to row vectors that are multiplied to the matrix from the left. Suppose that $\mathbf{p}(t) \in \mathbb{R}^n$ is a column vector whose elements denote the probability, or population, to be within any set $j \in \{1, \dots, n\}$ at time t. After time τ, the probabilities will have changed according to:

$$p_j(t+\tau) = \sum_{i=1}^{n} p_i(t) T_{ij}(\tau), \tag{3.31}$$

or in matrix form:

$$\mathbf{p}^T(t+\tau) = \mathbf{p}^T(t)\mathbf{T}(\tau) \tag{3.32}$$

Note that an alternative convention often used in the literature is to write $\mathbf{T}(\tau)$ as a column-stochastic matrix, obtained by taking the trans-pose of the row-stochastic transition matrix defined here.

The stationary probabilities of discrete states, π_i, yield the unique discrete stationary distribution of $\mathbf{T}$:

$$\boldsymbol{\pi}^T = \boldsymbol{\pi}^T \mathbf{T}(\tau). \tag{3.33}$$

All equations encountered so far are concerned with the discrete state space given by the partition sets, i.e., $\mathbf{p}^T(t)$ and $\mathbf{p}^T(t+\tau)$ are probability distribution on the discrete state space. The probability distribution on the continuous state space related to $\mathbf{p}^T(t)$ is

$$u_t(\mathbf{x}) = \sum_i p_i(t)\chi_i(\mathbf{x}).$$

If we propagate u_t with the true dynamics for time τ, we get $u_{t+\tau} = \mathscr{T}(\tau) \circ u_t$. However, $u_{t+\tau}$ and $\mathbf{p}^T(t+\tau)$ will no longer be perfectly related as above, i.e., we will only have

$$u_{t+\tau}(\mathbf{x}) \approx \sum_i p_i(t+\tau)\chi_i(\mathbf{x}).$$

We wish now to understand the error involved with this approximation. Moreover, we wish to model the system kinetics on long timescales by *approximating* the true dynamics with a Markov chain on the discrete state space of n states. Using $\mathbf{T}(\tau)$ as a Markov model *predicts* that for later times, $t + k\tau$, the probability distribution will evolve as:

$$\mathbf{p}^T(t+k\tau) = \mathbf{p}^T(t)\mathbf{T}^k(\tau), \tag{3.34}$$

on the discrete state space which can only approximate the true distribution,

$$u_{t+k\tau} = \big(\mathscr{T}(\tau)\big)^k \circ u_t,$$

that would have been produced by the continuous transfer operator, as Eq. (3.34) is the result of a state space discretization. The discretization error involved in this approximation thus depends on how this discretization is chosen and is analyzed in detail below. A description alternative to that of transition matrices quite common in chemical physics is using rate matrices and Master equations [5, 24, 26, 40, 41, 48].

3.5 Discretization Error and Non-Markovianity

The Markov model $\mathbf{T}(\tau)$ is indeed a model in the sense that it only approximates the long-time dynamics based on a discretization of state space, making the dynamical equation (3.34) approximate. Here we analyze the approximation quality of Markov models in detail and obtain a description that reveals which properties the state space discretization and the lag time τ must fulfill in order to obtain a good model.

Previous works have mainly discussed the quality of a Markov model in terms of its “Markovianity” and introduced tests of Markovianity of the process $\mathbf{x}(t)$ projected onto the discrete state space. The space-continuous dynamics $\mathbf{x}(t)$ described in Sect. 3.1 is, by definition, Markovian in full state space Ω and it can thus be exactly described by a linear operator, such as the transfer operator $\mathscr{T}(\tau)$ defined in Eq. (3.8). The problem lies in the fact that by performing a state space discretization, continuous states $\mathbf{x} \in \Omega$ are grouped into discrete states s_i (Sect. 3.3), thus “erasing” information of the exact location within these states and “projecting” a continuous trajectory $\mathbf{x}(t)$ onto a discrete trajectory $s(t) = s(\mathbf{x}(t))$. This jump process, $s(t)$, is *not Markovian*, but Markov models attempt to approximate $s(t)$ with a Markov chain.

Thus, non-Markovianity occurs when the full state space resolution is reduced by mapping the continuous dynamics onto a reduced space. In Markov models of molecular dynamics, this reduction consists usually of both neglect of degrees of freedom and discretization of the resolved degrees of freedom. Markov models typically only use atom positions, thus the velocities are projected out [9, 32]. So far, Markov models have also neglected solvent degrees of freedom and have only used the solute coordinates [9, 33], and the effect of this was studied in detail in [23]. Indeed, it may be necessary to incorporate solvent coordinates in situations where the solvent molecules are involved in slow processes that are not easily detected in the solute coordinates [25]. Often, Markov models are also based on distance metrics that only involve a subset of the solute

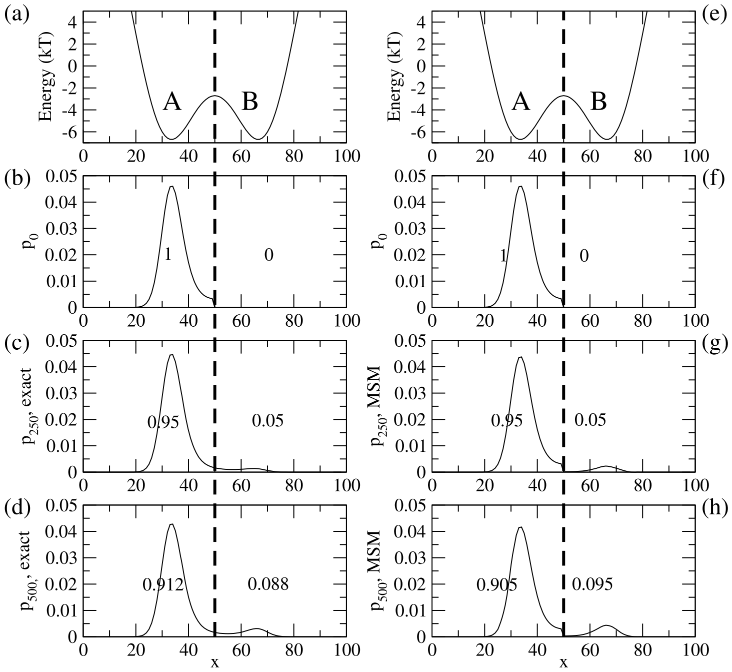


Fig. 3.5 Illustration of the discretization error by comparing the dynamics of the diffusion in a double well potential (**a**, **e**) (see Supplementary Information for setup) at time steps 0 (**b**), 250 (**c**), 500 (**d**) with the predictions of a Markov model parametrized with a too short lag time $\tau = 250$ at the corresponding times 0 (**f**), 250 (**g**), 500 (**h**). (**b**, **c**, **d**) show the true density $p_t(\mathbf{x})$ (*solid black line*) and the probabilities associated with the two discrete states left and right of the *dashed line*. The numbers in (**f**, **g**, **h**) are the discrete state probabilities $p_i(t + k\tau)$ predicted by the Markov model while the *solid black line* shows the hypothetical density $p_i(t + k\tau)\mu_i(\mathbf{x})$ that inherently assumed by the Markov model by using the discrete state probabilities to correspondingly weight the local stationary densities

atoms, such as RMSD between heavy atom or alpha carbon coordinates [4, 9, 33], or backbone dihedral angles [5, 32]. Possibly the strongest approximation is caused by clustering or lumping sets of coordinates in the selected coordinate subspace into discrete states [4, 5, 9, 26, 33]. Formally, all of these operations aggregate sets of points in continuous state space Ω into discrete states, and the question to be addressed is what is the magnitude of the discretization error caused by treating the non-Markovian jump process between these sets as a Markov chain.

Consider the diffusive dynamics model depicted in Fig. 3.5a and let us follow the evolution of the dynamics given that we start from a local equilibrium in basin A (Fig. 3.5b), either with the exact dynamics, or with the Markov model dynamics on the discrete state space A and B. After a step τ, both dynamics have transported a fraction of 0.1 of the ensemble to B. The true dynamics resolves the fact that much of this is still located near the saddle point (Fig. 3.5c). The Markov model cannot resolve local densities within its discrete states, which is equivalent to assuming that for the next step the ensemble has already equilibrated within the discrete state (Fig. 3.5g). This difference affects the discrete state (basin) probabilities at time 2τ: In the true dynamics, a significant part of the 0.1 fraction can cross back to A as it is still near the saddle point, while this is not the case in the Markov model where the 0.1 fraction is assumed to be relaxed to states mostly around the minimum (Compare Figs. 3.5d and h). As a result, the probability to be in state B is higher in the Markov model than in the true dynamics. The difference between the

Markov model dynamics and the true dynamics is thus a result of discretization, because the discretized model can no longer resolve deviations from local equilibrium density $\mu_i(\mathbf{x})$ within the discrete state.

This picture suggests the discretization error to have two properties: (a) the finer the discretization, the smaller the discretization error is, and (b) when increasing the coarse-graining time, or time resolution, of our model, τ, the errors for any fixed point in time t should diminish, because we have less often made the approximation of imposing local equilibrium.

3.6 Quantifying the Discretization Error

In order to quantify the discretization error, we will exploit the fact that the construction of a Markov State Model can be related to a projection of the transfer operator $\mathscr{T}(\tau)$. This projection, denoted Q, is the orthogonal projection with respect to the scalar product $\langle\cdot,\cdot\rangle_\mu$ onto a finite dimensional space D spanned by a given basis $q_1,\dots,q_n$, e.g., for $q_i=\chi_i$ being the characteristic functions from (3.24) that are associated with a crisp partitioning of the state space. For a general function u, Qu is the best possible representation of u in the space D. In general, it can be calculated [32, 37, 40] that the *projected propagation operator*, that is, the best representation of the propagator $\mathscr{T}$ in our space D, has the form $Q\mathscr{T}(\tau)Q$. It can be represented by the matrix $\mathbf{T}(\tau)=T(\tau)M^{-1}$ with

$$\begin{aligned} T_{ij}(\tau) &= \frac{\langle q_j,(\mathscr{T}(\tau)\circ q_i)\rangle_\mu}{\langle q_i,q_i\rangle_\mu}, \\ M_{ij} &= \frac{\langle q_j,q_i\rangle_\mu}{\langle q_i,q_i\rangle_\mu}. \end{aligned} \tag{3.35}$$

If we choose $q_i=\chi_i$ being the characteristic functions from (3.24) that are associated with a crisp partitioning of the state space into sets $S_1,\dots,S_n$, we find $M=Id$ because of orthogonality of the characteristic functions. Moreover, in this case, as calculated in (3.29)

$$\mathbf{T}(\tau)_{ij}=T_{ij}(\tau)=\mathbb{P}\big[\mathbf{x}(t+\tau)\in S_j\,\big|\,\mathbf{x}(t)\in S_i\big].$$

This means that this MSM transition matrix can be interpreted as the projection of the transfer operator with Q being the projection onto the discretization basis. Together with Fig. 3.5 this suggests a practical measure to quantify the discretization error. Densities, eigenfunctions or any other function $f(\mathbf{x})$ of the continuous state $\mathbf{x}$, are approximated by its best-approximations $\hat{f}(\mathbf{x})=Qf(\mathbf{x})$ within the space spanned by the discretization basis $q_1,\dots,q_n$. In the case of a crisp partitioning of state space, functions $f(\mathbf{x})$ are approximated through the discretization $S_1,\dots,S_n$ by step functions $\hat{f}(\mathbf{x})$ that are constant within the discrete states:

$$\hat{f}(\mathbf{x})=Qf(\mathbf{x})=\sum_{i=1}^{n}a_i\chi_i(\mathbf{x}) \tag{3.36}$$

where the coefficients are given by the projection weighted by the probability of each state:

$$a_i=\frac{\langle f,\chi_i\rangle_\mu}{\langle \mathbf{1},\chi_i\rangle_\mu}=\frac{\int_{S_i}d\mathbf{x}\,\mu(\mathbf{x})f(\mathbf{x})}{\int_{S_i}d\mathbf{x}\,\mu(\mathbf{x})}. \tag{3.37}$$

The approximation error that is caused by the discretization is then defined as the μ-weighted Euclidean norm $\|\cdot\|_{\mu,2}$ of the difference between discretized and original function:

$$\delta_f \equiv \big\|f(\mathbf{x})-\hat{f}(\mathbf{x})\big\|_{\mu,2} \tag{3.38}$$

$$=\left(\int_\Omega d\mathbf{x}\,\mu(\mathbf{x})\big(f(\mathbf{x})-\hat{f}(\mathbf{x})\big)^2\right)^{1/2}. \tag{3.39}$$

The projection allows the comparison between true and Markov model dynamics to be made exactly as suggested by Fig. 3.5: In both cases we start with an arbitrary initial density projected onto discrete states, $Qp_0(\mathbf{x})$, then transport this density either with the true or with the Markov model dynamics for some time $k\tau$. Subsequently, the densities is again projected onto discrete states by Q and then compared:

- The true dynamics transports the initial density $Qp_0(\mathbf{x})$ to $[\mathscr{T}(\tau)]^kQp_0(\mathbf{x})$
- The Markov model dynamics transports the initial density $Qp_0(\mathbf{x})$ to $Q\mathscr{T}(\tau)Qp_0(\mathbf{x})$ in one step and to $Q[\mathscr{T}(\tau)Q]^kp_0(\mathbf{x})$ in k steps using the identity for projections $Q\circ Q=Q$.

- Projecting both densities to local densities and comparing yields the difference

$$\varepsilon(k) = \left\| Q\left[\mathscr{T}(\tau)\right]^k Q p_0(\mathbf{x}) - Q\left[\mathscr{T}(\tau)Q\right]^k p_0(\mathbf{x}) \right\|_{\mu,2} \tag{3.40}$$

$$= \left\| \left[Q\left[\mathscr{T}(\tau)\right]^k Q - Q\left[\mathscr{T}(\tau)Q\right]^k\right] p_0(\mathbf{x}) \right\|_{\mu,2}. \tag{3.41}$$

In order to remove the dependency on the initial distribution $p_0(\mathbf{x})$, we assume the worst case: the maximal possible value of $\varepsilon(k)$ amongst all possible $p_0(\mathbf{x})$ is given by the operator-2-norm of the error matrix $[Q[\mathscr{T}(\tau)]^k Q - Q[\mathscr{T}(\tau)Q]^k]$, where $\|A\|_{\mu,2} \equiv \max_{\|x\|=1} \|Ax\|_{\mu,2}$ [17], thus the Markov model error is defined as:

$$E(k) := \left\| Q\left[\mathscr{T}(\tau)\right]^k Q - Q\left[\mathscr{T}(\tau)Q\right]^k \right\|_{\mu,2}, \tag{3.42}$$

which measures the maximum possible difference between the true probability density at time $k\tau$ and the probability density predicted by the Markov model at the same time.

In order to quantify this error, we declare our explicit interest in the m slowest processes with eigenvalues $1 = \lambda_1 > \lambda_2 \geq \lambda_3 \geq \cdots \geq \lambda_m$. Generally, $m \leq n$, i.e. we are interested in less processes than the number of n eigenvectors that a Markov model with n states has. We define the following two quantities:

- $\delta_i := \|\psi_i(\mathbf{x}) - Q\psi_i(\mathbf{x})\|_{\mu,2}$ is the *eigenfunction approximation error*, quantifying the error of approximating the true continuous eigenfunctions of the transfer operator, ψ_i (see Fig. 3.6 for illustration), for all $i \in \{1, \ldots, m\}$. $\delta := \max_i \delta_i$ is the largest approximation error amongst these first m eigenfunctions
- $\eta(\tau) := \frac{\lambda_{m+1}(\tau)}{\lambda_2(\tau)}$ is the *spectral error*, the error due to neglecting the fast subspace of the transfer operator, which decays to 0 with increasing lag time: $\lim_{\tau\to\infty} \eta(\tau) = 0$.

The general statement is that the Markov model error $E(k)$ can be proven [36] to be bounded from above by the following expression:

$$E(k) \leq \min\left\{2, \left[m\delta + \eta(\tau)\right]\left[a(\delta) + b(\tau)\right]\right\}\lambda_2^k \tag{3.43}$$

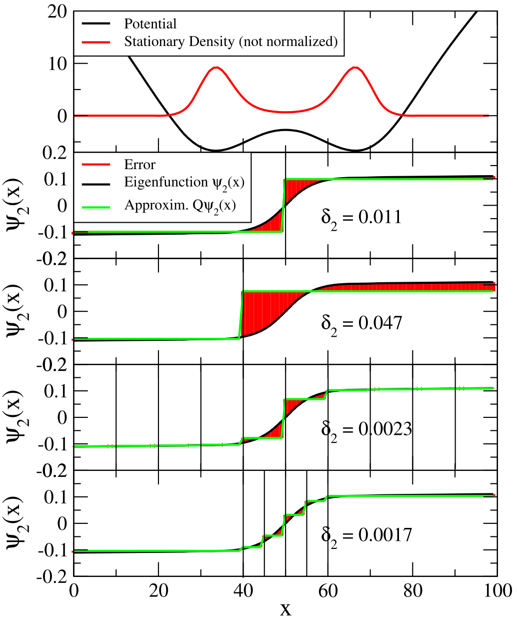


Fig. 3.6 Illustration of the eigenfunction approximation error δ_2 on the slow transition in the diffusion in a double well (*top*, *black line*). The slowest eigenfunction is shown in the lower four panels (*black*), along with the step approximations (*green*) of the partitions (*vertical black lines*) at $x = 50$; $x = 40$; $x = 10, 20, \ldots, 80, 90$; and $x = 40, 45, 50, 55, 60$. The eigenfunction approximation error δ_2 is shown as *red area* and its norm is printed

with

$$a(\delta) = \sqrt{m}(k-1)\delta, \tag{3.44}$$

$$b(\tau) = \frac{\eta(\tau)}{1-\eta(\tau)}\left(1 - \eta(\tau)^{k-1}\right). \tag{3.45}$$

This implies two observations:

1. For long times, the overall error decays to zero with λ_2^k, where $0 < \lambda_2 < 1$, thus the stationary distribution (recovered as $k \to \infty$) is always correctly modeled, even if the kinetics are badly approximated.
2. The error during the kinetically interesting timescales consists of a product whose terms contain separately the discretization error and spectral error. Thus, the overall error can be diminished by choosing a discretization basis $q_1, \ldots, q_n$ that approximates the slow eigenfunctions well, and using a large lag time τ. For a crisp partitioning this implies that the

discretization has to be fine enough to trace the slow eigenfunctions well.

Depending on the ratio $\lambda_{m+1}(\tau)/\lambda_2(\tau)$, the decay of the spectral error $\eta(\tau)$ with τ might be slow. It is thus interesting to consider a special case of discretization that yields $n = m$ and $\delta = 0$. This would be achieved by a Markov model that uses a fuzzy partition with membership functions $q_1, \dots, q_n$ derived from the first m eigenfunctions $\psi_1, \dots, \psi_m$ [24]. In this case, the space spanned by $q_1, \dots, q_n$ would equal the dominant eigenspace and hence the projection error would be $\delta = 0$. From a more practical point of view, this situation can be approached by using a Markov model with $n > m$ states located such that they discretize the first m eigenfunctions with a vanishing discretization error, and then declaring that we are *only* interested in these m slowest relaxation processes. In this case we do not need to rely on the upper bound of the error from Eq. (3.43), but directly get the important result $E(k) = 0$.

In other words, a Markov model can approximate the kinetics of slow processes *arbitrarily well*, provided the discretization can be made sufficiently fine or improved in a way that continues to minimize the eigenfunction approximation error. This observation can be rationalized by Eq. (3.19) which shows that the dynamics of the transfer operator can be exactly decomposed into a superposition of slow and fast processes.

An important consequence of the δ-dependence of the error is that the best partition is not necessarily metastable. Previous work [9, 9, 20, 32, 36, 42] has focused on the construction of partitions with high metastability (defined as the trace of the transition matrix $\mathbf{T}(\tau)$), e.g. the two-state partition shown in Fig. 3.6b). This approach was based on the idea that the discretized dynamics must be approximately Markovian if the system remained in each partition sufficiently long to approximately lose memory [9]. While it can be shown that if a system has m metastable sets with $\lambda_m \gg \lambda_{m+1}$, then the most metastable partition into $n = m$ sets also minimizes the discretization error [36, 36], the expression for the discretization error given here has two further profound ramifications: First, even in the case where there exists a strong separation of timescales so the system has clearly m metastable sets, the discretization error can be reduced *even further* by splitting the metastable partition into a total of $n > m$ sets which are not metastable. And second, even in the *absence* of a strong separation of timescales, the discretization error can be made arbitrarily small by making the partition finer, especially in transition regions, where the eigenfunctions change most rapidly (see Fig. 3.6b).

Figure 3.7 illustrates the Markov model discretization error on a two-dimensional three-well example where two slow processes are of interest. The left panels show a metastable partition with 3 sets. As seen in Fig. 3.7d, the discretization errors $|\psi_2 - Q\psi_2|(\mathbf{x})$ and $|\psi_3 - Q\psi_3|(\mathbf{x})$ are large near the transition regions, where the eigenfunctions $\psi_2(\mathbf{x})$ and $\psi_3(\mathbf{x})$ change rapidly, leading to a large discretization error. Using a random partition (Fig. 3.7, third column) makes the situation worse, but increasing the number of states reduces the discretization error (Fig. 3.7, fourth column), thereby increasing the quality of the Markov model. When states are chosen such as to well approximate the eigenfunctions, a very small error can be obtained with few sets (Fig. 3.7, second column).

These results suggest that an adaptive discretization algorithm may be constructed which minimizes the $E(k)$ error. Such an algorithm could iteratively modify the definitions of discretization sets as suggested previously [9], but instead of maximizing metastability it would minimize the $E(k)$ error which can be evaluated by comparing eigenvector approximations on a coarse discretization compared to a reference evaluated on a finer discretization [36].

One of the most intriguing insights from both Eq. (3.19) and the results of the discretization error is that if, for a given system, only the slowest dynamical processes are of interest, it is sufficient to discretize the state space such that the first few eigenvectors are well represented (in terms of small approximation errors δ_i). For example, if one is interested in processes on timescales t^* or slower, then the number m of eigenfunctions that need to be resolved is equal to the number of

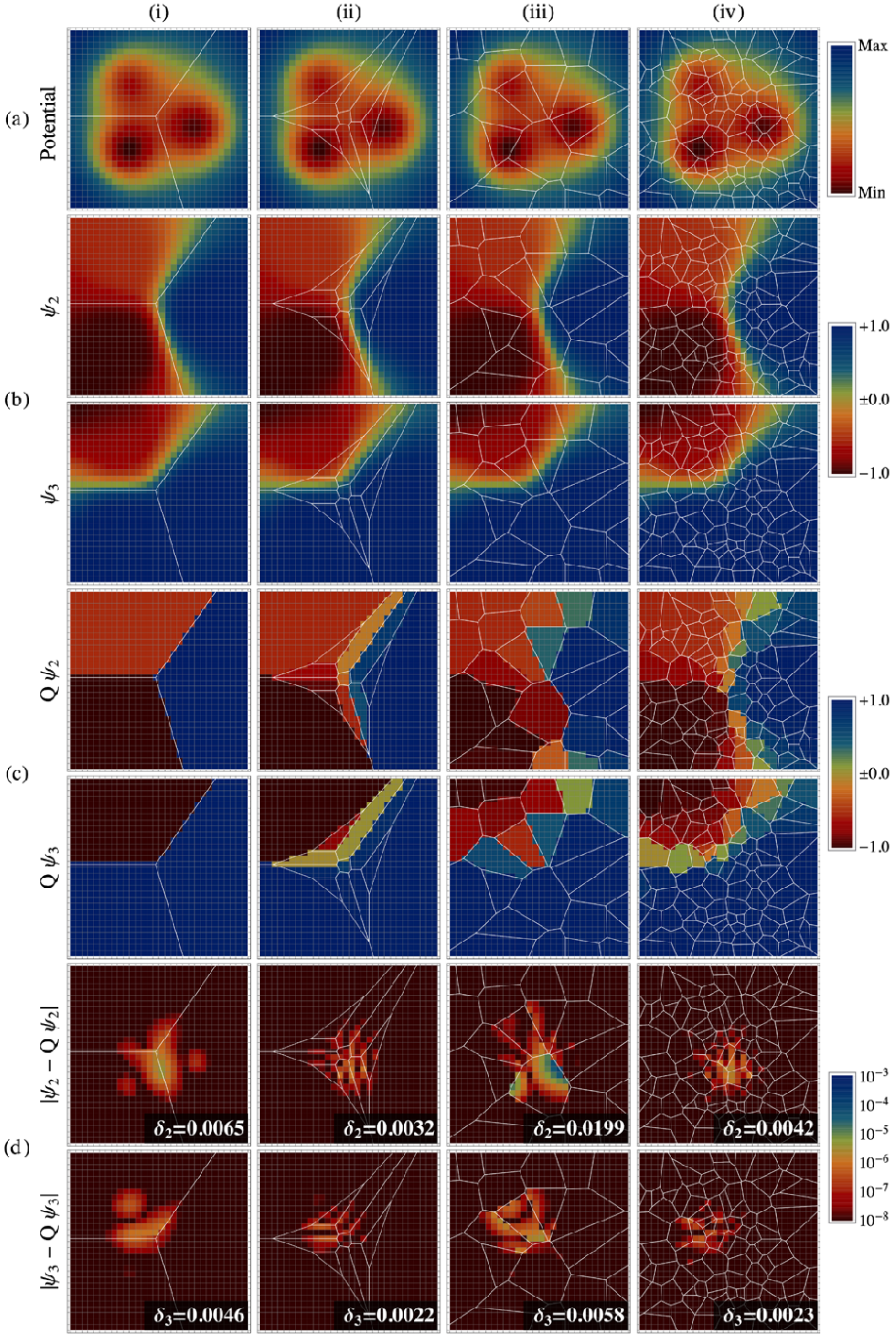


Fig. 3.7 Illustration of the eigenfunction approximation errors δ_2 and δ_3 on the two slowest processes in a two-dimensional three-well diffusion model (see Supplementary Information for model details). The *columns* from left to right show different state space discretizations with white lines as state boundaries: (**i**) 3 states with maximum metastability, (**ii**) the metastable states were further subdivided manually into 13 states to better resolve the transition region, resulting in a partition where no individual state is metastable, (**iii**)/(**iv**) Voronoi partition using 25/100 randomly chosen centers, respectively. (**a**) Potential, (**b**) The exact eigenfunctions of the slow processes, $\psi_2(\mathbf{x})$ and $\psi_3(\mathbf{x})$, (**c**) The approximation of eigenfunctions with discrete states, $Q\psi_2(\mathbf{x})$ and $Q\psi_3(\mathbf{x})$, (**d**) Approximation errors $|\psi_2 - Q\psi_2|(\mathbf{x})$ and $|\psi_3 - Q\psi_3|(\mathbf{x})$. The error norms δ_2 and δ_3 are given

implied timescales with $t_i \geq t^*$. Due to the perfect decoupling of processes for reversible dynamics in the eigenfunctions (see Eqs. (3.20)–(3.21)), no gap after these first m timescales of interest is needed. Note that the quality of the Markov model does not depend on the dimensionality of the simulated system, i.e. the number of atoms. Thus, if only the slowest process of the system is of interest (such as the folding process in a two-state folder), only a one-dimensional parameter, namely the level of the dominant eigenfunction, needs to be approximated with the clustering, even if the system is huge. This opens a way to discretize state spaces of very large molecular systems.

3.7 Approximation of Eigenvalues and Timescales by Markov Models

One of the most interesting kinetic properties of molecular systems are the intrinsic timescales of the system. They can be both experimentally accessed *via* relaxation or correlation functions that are measurable with various spectroscopic techniques [2, 5, 20, 28], but can also be directly calculated from the Markov model eigenvalues as implied timescales, Eq. (3.22). Thus, we investigate the question how well the dominant eigenvalues λ_i are approximated by the Markov model, which immediately results in estimates for how accurately a Markov model may reproduce the implied timescales of the original dynamics. Consider the first m eigenvalues of $\mathcal{T}(\tau)$, $1 = \lambda_1(\tau) > \lambda_2(\tau) \geq \cdots \geq \lambda_m(\tau)$, and let $1 = \hat{\lambda}_1(\tau) > \hat{\lambda}_2(\tau) \geq \cdots \geq \hat{\lambda}_m(\tau)$ denote the associated eigenvalues of the Markov model $\mathbf{T}(\tau)$. The rigorous mathematical estimate from [13] states that

$$\max_{j=1,\dots,m} \left|\lambda_j(\tau) - \hat{\lambda}_j(\tau)\right| \leq (m-1)\lambda_2(\tau)\delta^2, \tag{3.46}$$

where δ is again the maximum discretization error of the first m eigenfunctions, showing that the eigenvalues are well reproduced when the discretization well traces these eigenfunctions. In particular if we are only interested in the eigenvalue of the slowest process, $\lambda_2(\tau)$, which is often experimentally reported *via* the slowest relaxation time of the system, t_2, the following estimate of the approximation error can be given:

$$\frac{|\lambda_2(\tau) - \hat{\lambda}_2(\tau)|}{|\lambda_2(\tau)|} \leq \delta_2^2. \tag{3.47}$$

As $\lambda_2(\tau)$ corresponds to a slow process, we can make the restriction $\lambda_2(\tau) > 0$. Moreover, the discretization error of Markov models based on full partitions of state space is such that the eigenvalues are always underestimated [13], thus $\lambda_2(\tau) - \hat{\lambda}_2(\tau) > 0$. Using Eq. (3.22), we thus obtain the estimate for the discretization error of the largest implied timescale and the corresponding smallest implied rate, $k_2 = t_2^{-1}$:

$$\hat{t}_2^{-1} - t_2^{-1} = \hat{k}_2 - k_2 \leq -\tau^{-1} \ln\bigl(1 - \delta_2^2\bigr), \tag{3.48}$$

which implies that for either $\delta_2 \to 0_+$ or $\tau \to \infty$, the error in the largest implied timescale or smallest implied rate tends to zero. Moreover, since $\lambda_2(\tau) \to 0$ for $\tau \to \infty$, this is also true for the other processes:

$$\lim_{\tau\to\infty} \frac{|\lambda_j(\tau) - \hat{\lambda}_j(\tau)|}{|\lambda_j(\tau)|} = 0, \tag{3.49}$$

and also

$$\lim_{\delta\to 0} \frac{|\lambda_j(\tau) - \hat{\lambda}_j(\tau)|}{|\lambda_j(\tau)|} = 0, \tag{3.50}$$

which means that the error of the implied timescales also vanishes for either sufficiently long lag times τ or for sufficiently fine discretization. This fact has been empirically observed in many previous studies [2, 5, 9, 26, 32, 33, 42], but can now be understood in detail in terms of the discretization error. It is worth noting that observing convergence of the slowest implied timescales in τ is not a test of Markovianity. While Markovian dynamics implies constancy of implied timescales in τ [32, 42], the reverse is not true and would require the eigenvectors to be constant as well. However, observing the lag time-dependence of the implied timescales is a useful approach to choose a lag time τ at which $\mathbf{T}(\tau)$

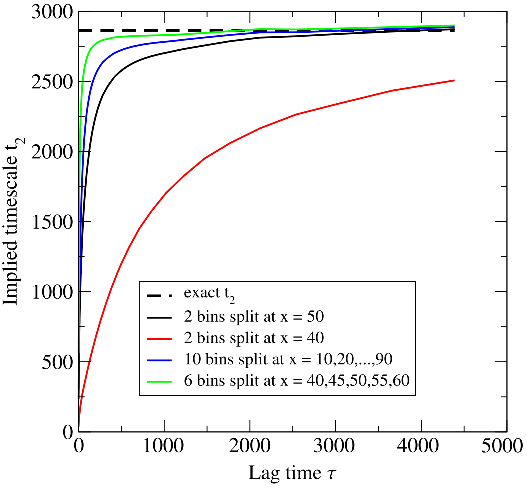


Fig. 3.8 Convergence of the slowest implied timescale $t_2 = -\tau / \ln \lambda_2(\tau)$ of the diffusion in a double-well potential depending on the MSM discretization. The metastable partition (*black*, *solid*) has greater error than non-metastable partitions (*blue*, *green*) with more states that better trace the change of the slow eigenfunction near the transition state

shall be calculated, but this model needs to be validated subsequently (see Chapter Estimation and Validation).

Figure 3.8 shows the slowest implied timescale t_2 for the diffusion in a two-well potential (see Fig. 3.6) with discretizations shown in Fig. 3.6. The two-state partition at $x = 50$ requires a lag time of ≈ 2000 steps in order to reach an error of <3 % with respect to the true implied timescale, which is somewhat slower than t_2 itself. When the two-state partition is distorted by shifting the discretization border to $x = 40$, this quality is not reached before the process itself has relaxed. Thus, in this system two states are not sufficient to build a Markov model that is at the same time precise and has a time resolution good enough to trace the decay of the slowest process. By using more states and particularly a finer discretization of the transition region, the same approximation quality is obtained with only $\tau \approx 1500$ (blue) and $\tau \approx 500$ steps (green).

Figure 3.9 shows the two slowest implied timescales t_2, t_3 for the diffusion in a two-dimensional three-well potential with discretizations shown in Fig. 3.7a. The metastable 3-state partition requires a lag time of ≈ 1000 steps in order to reach an error of <3 % with respect to the true implied timescale, which is somewhat shorter than the slow but longer than the fast timescale, while refining the discretization near the transition states achieves the same precision with $\tau \approx 200$ using only 12 states. A k-means clustering with $k = 25$ is worse than the metastable partition, as some clusters cross over the transition region and fail to resolve the slow eigenfunctions. Increasing the number of clusters to $k = 100$ improves the result significantly, but is still worse than the 12 states that have been manually chosen so as to well resolve the transition region. This suggests that excellent MSMs could be built with rather few states when an adaptive algorithm that more finely partitions the transition region is employed.

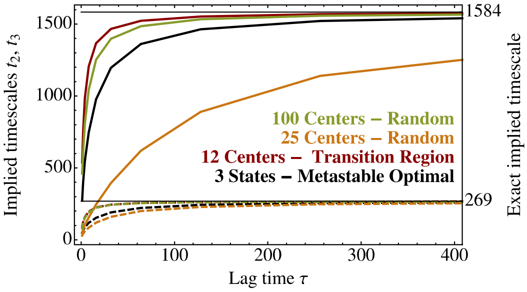


Fig. 3.9 Implied timescales for the two slowest processes in the two-dimensional three-well diffusion model (see Fig. 3.7a for potential and Supplementary Information for details). The *colors black*, *red*, *yellow*, *green* correspond to the four choices of discrete states shown in columns 1 to 4 of Fig. 3.7. A fine discretization of the transition region clearly gives the best approximation to the timescales at small lag times

Note that the estimate (3.46) bounds the maximal eigenvalue error for the dominant eigenvalues by the maximal projection error of the dominant eigenfunction. In [34], it is also shown that if u is a selected, maybe *non*-dominant eigenfunctions for an eigenvalue $\lambda(\tau)$ and $\delta = \|u - Qu\|$ is its discretization error, the associated MSM will inherit an eigenvalue $\hat{\lambda}(\tau)$ with

$$\left|\lambda(\tau) - \lambda(\hat{\tau})\right| \leq 2\lambda_2(\tau)\delta \qquad (3.51)$$

if $\delta = \|u - Qu\| \leq 3/4$. That is, it is also possible to approximate selected timescales well by choosing the discretization such that it traces the associated eigenfunctions well without having to take all slower eigenfunctions into account [34].

3.8 An Alternative Set-Oriented Projection

In the last sections, we have derived an interpretation of Markov models as projections of the transfer operator $\mathscr{T}(\tau)$ and connected their discretization error in terms of density propagation and in terms of eigenvalue and timescale approximation to projection errors. Estimates for these quality measures show that the discretization basis should be chosen such that it traces the slow eigenfunctions well. For a crisp partitioning this means that the eigenfunctions should be well approximated by step-functions induced by the sets $S_1, \dots, S_n$. On the other hand, the results are not restricted to projections onto step-functions. Theoretically, one could choose an arbitrary discretization basis $q_1, \dots, q_n$ for constructing the Markov model, see [40], and the corresponding MSM matrix would formally be given by $\mathbf{T}(\tau) = T(\tau)M^{-1}$ (3.35). In praxis, the basis $q_1, \dots, q_n$ has be chosen such that it leads to interpretable matrices $T(\tau)$ and M in terms of transition probabilities between sets. Otherwise, one will not be able compute estimates for these matrices and thus for the resulting Markov model. For a crisp partitioning $S_1, \dots, S_n$ and the associated characteristic functions $\chi_1, \dots, \chi_n$ we have this property

$$T_{ij}(\tau) = \mathbb{P}\big[\mathbf{x}(t+\tau) \in S_j \,\big|\, \mathbf{x}(t) \in S_i\big], \qquad M = Id.$$

The drawback of this method is that coarse partitionings always lead to coarse step-functions that might not approximate the eigenfunctions well. Therefore, a refinement might be necessary in regions where the slow eigenfunctions are varying strongly.

In this section, we will show how to derive another set-oriented discretization basis where a rather coarse partitioning does not lead to a coarse discretization basis. The main idea goes back to [32, 37, 40]: simply decrease the size of the sets $S_1, \dots, S_n$ on which the constancy of the discretization basis in enforced. Of course, the resulting sets do not cover the whole state space and hence do not form a crisp partitioning anymore. We will call such sets *core sets* in the following and denote them by $C_1, \dots, C_n$ in order to distinguish from a crisp partitioning $S_1, \dots, S_n$.

The two questions that we have to answer are (a) how these core sets induce a discretization basis that approximates the slow eigenfunctions well, and (b) how to interpret the transition probabilities between the core sets to calculate the transition matrices $T(\tau)$ and M with respect to this basis. The idea is to attach a fuzzy partitioning to the core sets that is connected to the dynamics of the process itself. For every core set C_i we define the so called *committor* function

$$q_i(x) = \mathbb{P}\big[x(\sigma) \in C_i \,\big|\, x(0) = x\big], \tag{3.52}$$

where σ is the first time one of the core sets is entered by the process.

That is, $q_i(x)$ is the probability that the process will hit the set C_i next rather than the other core sets when being started in point x. From the definition it follows that [27, 40]

$$\begin{aligned} q_i(x) &= 1 \quad \text{for all } x \in C_i, \\ q_i(x) &= 0 \quad \text{for all } x \in C_j,\, j \neq i, \\ \sum_i q_i(x) &= 1 \quad \text{for all } x. \end{aligned}$$

The advantage of taking committor functions as discretization basis is that the core sets, on which the committor functions equal to characteristic functions, do not have to cover the whole state space. It is allowed to have a region $C = \Omega \setminus \bigcup_j C_j$ that is not partitioned and where the values of the committor functions can continuously vary between 0 and 1. This means that the part of state space can be shrinked, where the slow eigenfunctions need to be similar to step-functions. Moreover, it has been shown in [34] that the approximation of the slow eigenfunctions by the committors inside of the fuzzy region C is accurate if the region C is left by the process quickly enough. Being more precise, this approximation error is dominated by the ratio of the expected time the process needs to leave the region C and the implied timescale that is associated with the target eigenfunction. We will see that it is computationally very suitable that this is the main constraint on C.

Beside the good approximation of the slow eigenfunctions, the committor discretization has another advantage. We can calculate the Markov model from transition probabilities between the core sets without having to compute the committor functions explicitly. It is shown in [32, 37, 40] that for the committor basis we can interpret the matrices $T(\tau)$ and M as follows:

Define $x^+(t)$ as the index of the core set that is hit next after time t, and $x^-(t)$ as the index of the core set that the process has visited last before time t, then

$$M_{ij} = \mathbb{P}\big[x^+(t) = j \,\big|\, x^-(t) = i\big],$$

and

$$T_{ij}(\tau) = \mathbb{P}\big[x^+(t+\tau) = j \,\big|\, x^-(t) = i\big].$$

Note that this can be interpreted in terms of a transition behavior. If we interpret $T_{ij}(\tau)$ as the probability that a transition occurs from index i at time t to index j at time $t+\tau$, then we say a transition has occurred if the process was in core set i at time t *or at least came last from this set* and after a time-step τ it was in core set j *or at least went next to this set afterwards*. Figure 3.10 illustrates this interpretation. As mentioned above, these transition probabilities can be directly estimated from realizations without having to compute the committor functions.

The effect on the approximation of the slowest eigenfunction can be seen in Fig. 3.11.

Computationally, there is an important insight. Assume a crisp partitioning of state space into n sets $S_1, \ldots, S_n$ is given. Now, a committor discretization would allow to avoid a part of state space from being discretized, as long as the process leaves this part typically much faster than the interesting timescales. On the one hand, these parts of state space exactly correspond to regions where the slow eigenfunctions are varying strongly. So starting from the crisp partitioning we can benefit most by simply ignoring this part of state space and treating the remaining sets as core sets. On the other hand, removing such part of state space, where the process does not spend a lot of time in, does not effect the computational effort in order to generate transitions with respect to the crisp partitioning or the resulting core sets as in Fig. 3.10. Summarizing, one can always enhance a model based on a crisp partitioning by simply declaring a part of state space as not be-

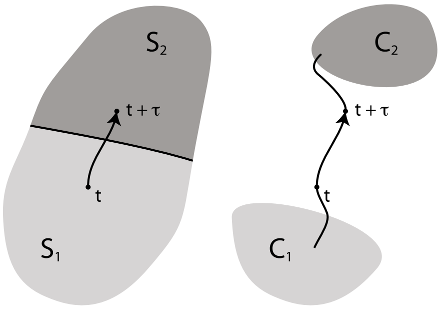


Fig. 3.10 Counting a transition in the sense of $T_{12}(\tau)$ for a crisp partitioning (*left hand side*) and for core sets (*right hand side*) that do not cover the whole state space

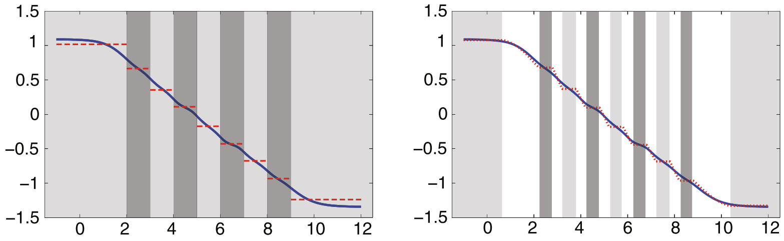


Fig. 3.11 The benefit of removing a part of state space that is typically left quickly by the process. On the *left hand side*: step-function approximation of the first non-trivial eigenfunction. *Right hand side*: committor approximation of the same eigenfunction

longing to the discretization anymore, as long as this part of state space is usually left quickly by the process, and computationally one can get this enhancement for free.

Let us illustrate this feature by an example. We consider again a diffusion in the potential that is illustrated in Fig. 3.12. Now, Fig. 3.13 shows an optimal crisp partitioning into 9 sets, where every well of the potential falls into another partitioning set. Moreover, it shows the core set discretization where only a part of the transition region between the wells was excluded from the crisp discretization.

The approximation by a step-function is too coarse while removing the transition region and shrinking the size of the sets leads to a smoother and better interpolation of the eigenfunction. As we discussed in the previous sections, this has a direct impact on the approximation quality of the associated Markov model. For example, the following table shows the implied timescales t_i of the original Markov process and the approximations by the crisp partitioning and the enhanced core set MSM.

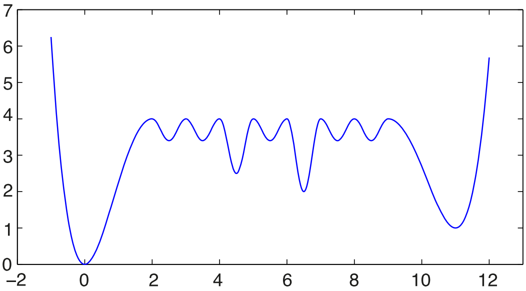


Fig. 3.12 Example: diffusion in this multi-well potential

	t_2	t_3	t_4	t_5
original	17.5267	3.1701	0.9804	0.4524
core sets	17.3298	3.1332	0.9690	0.4430
crisp partition	16.5478	2.9073	0.8941	0.4006

As expected, the approximation quality in terms of timescale approximation could be increased by simply removing a small part of state space from the discretization. The same enhancement is achieved with respect to the density propagation error $E(k)$ (3.42). Figure 3.14 compares the resulting error for the crisp partitioning (black) and the core set discretization (blue) and increasing k.

$$E(k) := \left\| Q\left[\mathcal{T}(\tau)\right]^k Q - Q\left[\mathcal{T}(\tau) Q\right]^k \right\|_{\mu,2}.$$

Fig. 3.13 A metastable crisp partitioning into 9 sets (*left*), and the derived core set discretization by removing the transition region between the wells (*right*)

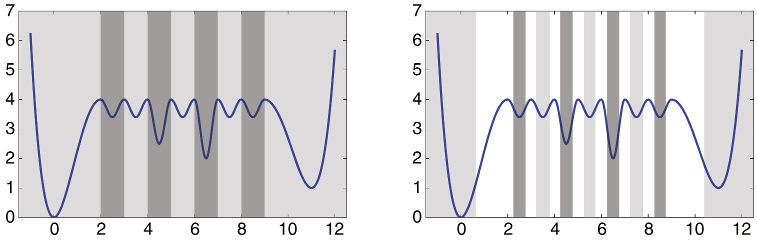
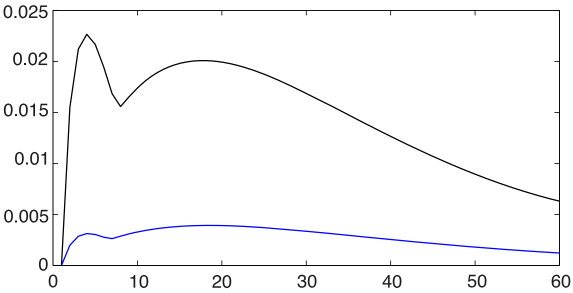


Fig. 3.14 Density propagation error $E(k)$ over k. *Black*: crisp partitioning. *Blue*: core sets



References

1. Andersen HC (1980) Molecular dynamics simulations at constant pressure and/or temperature. J Chem Phys 72(4):2384–2393. doi:10.1063/1.439486
2. Bieri O, Wirz J, Hellrung B, Schutkowski M, Drewello M, Kiefhaber T (1999) The speed limit for protein folding measured by triplet-triplet energy transfer. Proc Natl Acad Sci USA 96(17):9597–9601. http://www.pnas.org/content/96/17/9597.abstract
3. Bowman GR, Beauchamp KA, Boxer G, Pande VS (2009) Progress and challenges in the automated construction of Markov state models for full protein systems. J Chem Phys 131(12):124,101+. doi:10.1063/1.3216567
4. Buchete NV, Hummer G (2008) Coarse Master Equations for Peptide Folding Dynamics. J Phys Chem B 112:6057–6069
5. Chan CK, Hu Y, Takahashi S, Rousseau DL, Eaton WA, Hofrichter J (1997) Submillisecond protein folding kinetics studied by ultrarapid mixing. Proc Natl Acad Sci USA 94(5):1779–1784. http://www.pnas.org/content/94/5/1779.abstract
6. Chodera JD, Dill KA, Singhal N, Pande VS, Swope WC, Pitera JW (2007) Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics. J Chem Phys 126:155,101
7. Chodera JD, Noé F (2010) Probability distributions of molecular observables computed from Markov models, II: uncertainties in observables and their time-evolution. J Chem Phys 133:105,102
8. Chodera JD, Swope WC, Noé F, Prinz JH, Pande VS (2010) Dynamical reweighting: improved estimates of dynamical properties from simulations at multiple temperatures. J Phys Chem. doi:10.1063/1.3592152
9. Chodera JD, Swope WC, Pitera JW, Dill KA (2006) Long-time protein folding dynamics from short-time molecular dynamics simulations. Multiscale Model Simul 5:1214–1226
10. Cooke B, Schmidler SC (2008) Preserving the Boltzmann ensemble in replica-exchange molecular dynamics. J Chem Phys 129:164,112
11. Deuflhard P, Huisinga W, Fischer A, Schütte C (2000) Identification of almost invariant aggregates in reversible nearly uncoupled Markov chains. Linear Algebra Appl 315:39–59
12. Deuflhard P, Weber M (2005) Robust Perron cluster analysis in conformation dynamics. Linear Algebra Appl 398:161–184
13. Djurdjevac N, Sarich M, Schütte C (2012) Estimating the eigenvalue error of Markov state models. Multiscale Model Simul 10(1):61–81
14. Duane S (1987) Hybrid Monte Carlo. Phys Lett B 195(2):216–222. doi:10.1016/0370-2693(87)91197-X
15. Ermak DL (1975) A computer simulation of charged particles in solution, I: technique and equilibrium properties. J Chem Phys 62(10):4189–4196
16. Ermak DL, Yeh Y (1974) Equilibrium electrostatic effects on the behavior of polyions in solution: polyion-mobile ion interaction. Chem Phys Lett 24(2):243–248
17. Golub GH, van Loan CF (1996) Matrix computations, 3rd edn. John Hopkins University Press, Baltimore
18. Herau R, Hitrik M, Sjoestrand J (2010) Tunnel effect and symmetries for Kramers Fokker-Planck type operators. arXiv:1007.0838v1
19. Huisinga W (2001) Metastability of Markovian systems: a transfer operator based approach in application to molecular dynamics. PhD thesis, Fachbereich Mathematik und Informatik, FU Berlin
20. Huisinga W, Meyn S, Schütte C (2004) Phase transitions and metastability for Markovian and molecular systems. Ann Appl Probab 14:419–458
21. Jäger M, Nguyen H, Crane JC, Kelly JW, Gruebele M (2001) The folding mechanism of a beta-sheet: the WW domain. J Mol Biol 311(2):373–393. doi:10.1006/jmbi.2001.4873
22. van Kampen NG (2006) Stochastic processes in physics and chemistry, 4th edn. Elsevier, Amsterdam
23. Keller B, Hünenberger P, van Gunsteren W (2010) An analysis of the validity of Markov state models for emulating the dynamics of classical molecular systems and ensembles. J Chem Theor Comput. doi:10.1021/ct200069c
24. Kube S, Weber M (2007) A coarse graining method for the identification of transition rates between molecular conformations. J Chem Phys 126:024,103–024,113
25. Meerbach E, Schütte C, Horenko I, Schmidt B (2007) Metastable conformational structure and dynamics: peptides between gas phase and aqueous solution. Series in chemical physics, vol 87. Springer, Berlin, pp 796–806
26. Metzner P, Horenko I, Schütte C (2007) Generator estimation of Markov jump processes based on incomplete observations nonequidistant in time. Phys Rev E 76(6):066,702+. doi:10.1103/PhysRevE.76.066702
27. Metzner P, Schütte C, Vanden-Eijnden E (2009) Transition path theory for Markov jump processes. Multiscale Model Simul 7(3):1192–1219
28. Neuweiler H, Doose S, Sauer M (2005) A microscopic view of miniprotein folding: enhanced folding efficiency through formation of an intermediate. Proc Natl Acad Sci USA 102(46):16,650–16,655. doi:10.1073/pnas.0507351102
29. Noé F, Horenko I, Schütte C, Smith JC (2007) Hierarchical analysis of conformational dynamics in biomolecules: transition networks of metastable states. J Chem Phys 126:155,102
30. Noé F, Schütte C, Vanden-Eijnden E, Reich L, Weikl TR (2009) Constructing the full ensemble of folding pathways from short off-equilibrium simulations. Proc Natl Acad Sci USA 106:19,011–19,016
31. Prinz JH et al (2011) Markov models of molecular kinetics: generation and validation. J Chem Phys 134:174,105

32. Sarich M (2011) Projected transfer operators. PhD thesis, Freie Universität Berlin
33. Sarich M, Noé F, Schütte C (2010) On the approximation error of Markov state models. Multiscale Model Simul 8:1154–1177
34. Sarich M, Schütte C (2012) Approximating selected non-dominant timescales by Markov state models. Comm Math Sci (in press)
35. Schütte C (1998) Conformational dynamics: modelling, theory, algorithm, and applications to biomolecules. Habilitation thesis, Fachbereich Mathematik und Informatik, FU Berlin
36. Schütte C, Huisinga W (2003) Biomolecular conformations can be identified as metastable sets of molecular dynamics. In: Handbook of numerical analysis. Elsevier, Amsterdam, pp 699–744
37. Schütte C, Sarich M (2013) Metastability and Markov state models in molecular dynamics: modeling, analysis, algorithmic approaches. Courant lecture notes, vol 24. American Mathematical Society, Providence
38. Schütte C, Fischer A, Huisinga W, Deuflhard P (1999) A direct approach to conformational dynamics based on hybrid Monte Carlo. J Comput Phys 151:146–168
39. Schütte C, Noé F, Meerbach E, Metzner P, Hartmann C (2009) Conformation dynamics. In: Jeltsch RGW (ed) Proceedings of the international congress on industrial and applied mathematics (ICIAM). EMS Publishing House, New York, pp 297–336
40. Schütte C, Noé F, Lu J, Sarich M, Vanden-Eijnden E (2011) Markov state models based on milestoning. J Chem Phys 134(19)
41. Sriraman S, Kevrekidis IG, Hummer G (2005) Coarse master equation from Bayesian analysis of replica molecular dynamics simulations. J Phys Chem B 109:6479–6484
42. Swope WC, Andersen HC, Berens PH, Wilson KR (1982) A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: application to small water clusters. J Chem Phys 76(1):637–649
43. Swope WC, Pitera JW, Suits F (2004) Describing protein folding kinetics by molecular dynamics simulations, 1: theory. J Phys Chem B 108:6571–6581
44. Swope WC, Pitera JW, Suits F, Pitman M, Eleftheriou M (2004) Describing protein folding kinetics by molecular dynamics simulations, 2: example applications to alanine dipeptide and beta-hairpin peptide. J Phys Chem B 108:6582–6594
45. Voronoi MG (1908) Nouvelles applications des parametres continus a la theorie des formes quadratiques. J Reine Angew Math 134:198–287
46. Weber M (2006) Meshless methods in conformation dynamics. PhD thesis, Freie Universität Berlin
47. Weber M (2010) A subspace approach to molecular Markov state models via an infinitesimal generator. ZIB report 09-27-rev
48. Weber M (2011) A subspace approach to molecular Markov state models via a new infinitesimal generator. Habilitation thesis, Fachbereich Mathematik und Informatik, Freie Universität Berlin

4 Estimation and Validation of Markov Models

Jan-Hendrik Prinz, John D. Chodera, and Frank Noé

In this chapter, we discuss the problem of estimating the state-to-state *transition matrix* of a Markov model given a set of trajectory data and a discretization of configuration space, the selection of an appropriate lag time (or observation interval) τ, and validation of the resulting model to ensure it is consistent with the data used to construct it. We presume the trajectory data has been generated by one or more molecular dynamics simulations initiated from configurations sampled from either global equilibrium or a local equilibrium within one or more of the discretized conformational states. These states can be generated according to methods discussed in previous chapters. This chapter follows Ref. [20] which should be used for citation purposes.

4.1 Preliminaries: The Transition Count Matrix

For simplicity, we first consider the case of a single equilibrium simulation trajectory X consisting of N configurations sampled at a fixed time interval Δt,

$$X = \bigl(\mathbf{x}_1 = \mathbf{x}(t=0), \mathbf{x}_2 = \mathbf{x}(t=\Delta t), \dots, \mathbf{x}_N = \mathbf{x}\bigl(t=(N-1)\Delta t\bigr)\bigr). \tag{4.1}$$

The generalization to multiple trajectories will be discussed subsequently.

We suppose we have already defined a crisp state space discretization $\{S_1, \dots, S_K\}$ where each structure can be assigned uniquely to a discrete state,

$$\mathbf{x}_k \in S_i \Rightarrow s_k = i, \quad k \in \{1, \dots, N\},\ i \in \{1, \dots, K\}$$

allowing the trajectory to be encoded as the sequence $(s_1, \dots, s_N)$ of discrete states visited at times $n\Delta t$ along the trajectory.

We assume that the initial configuration $\mathbf{x}_1$ was drawn from $\mu_{s_1}(\mathbf{x})$, the equilibrium density within the initial state s_1. There are numerous strategies that can be used to sample from the initial distribution $\mu_{s_1}(\mathbf{x})$ for the purposes of initiating a simulation from s_1, such as sampling from a reweighted ensemble (e.g. generated by replica-exchange [24] or well-converged meta-dynamics [13] simulations) or utilizing a potential energy bias $U_{\text{bias}}(\mathbf{x}) = -k_B T \ln \mu_i(\mathbf{x})$ to rapidly equilibrate a simulation within the state before removing the biasing potential to generate unbiased dynamical trajectories [21]. Note that in the limit of very small discrete states, this problem vanishes as $\mu_i(\mathbf{x})$ can then be well approximated by a step function (see the Supplementary Material for [17]).

We can now define a state-to-state transition count matrix $\mathbf{C}^{\text{obs}}(\tau) = [c_{ij}^{\text{obs}}(\tau)]$ at lag time τ,

J.-H. Prinz · F. Noé (✉)
Freie Universität Berlin, Arnimallee 6, 14195 Berlin, Germany
e-mail: frank.noe@fu-berlin.de

J.D. Chodera
Memorial Sloan-Kettering Cancer Center, New York, NY 10065, USA

G.R. Bowman et al. (eds.), *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology 797,
DOI 10.1007/978-94-007-7606-7_4, © Springer Science+Business Media Dordrecht 2014

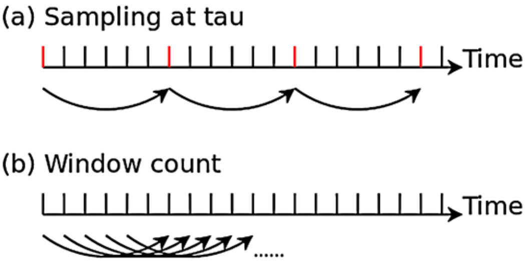


Fig. 4.1 Transition counting schemes for estimating $c_{ij}^{\text{obs}}(\tau)$: (**a**) Sampling the trajectory at discrete multiples of the lag time τ, versus (**b**) utilizing a moving overlapping window count

where $\tau = l\Delta t$ now denotes an integer multiple $l \in \mathbb{Z}$ of the available time resolution Δt,

$$c_{ij}^{\text{obs}}(\tau) = c_{ij}^{\text{obs}}(l\Delta t). \tag{4.2}$$

This allows us to construct an estimator for the correlation matrix defined in Eq. (3.30),

$$\hat{c}_{ij}^{\text{corr}}(\tau) = \frac{c_{ij}^{\text{obs}}(\tau)}{N-l}. \tag{4.3}$$

When the state space is discretized by a crisp partitioning, $\hat{c}_{ij}^{\text{corr}}(\tau)$ simply counts the number of times we observe the trajectory initially in state i and in state j a time τ later. If multiple trajectories are available, then the count matrices of these trajectories can be computed independently and summed.

As a shorthand notation, we define the row sums of $\mathbf{C}^{\text{obs}}(\tau)$,

$$c_{i}^{\text{obs}}(\tau) \equiv \sum_{k=1}^{n} c_{ik}^{\text{obs}}(\tau). \tag{4.4}$$

For a crisp partitioning, $c_i^{\text{obs}}(\tau)$ is simply the number of times the trajectory visited state i (excluding the last $l-1$ snapshots of the trajectory).

4.2 Counting Schemes

We distinguish between two approaches to counting (see Fig. 4.1 for an illustration):

1. *Sampling the trajectory at time lag* τ: Here, the trajectory is sampled at time lag τ, and only sample points at $n\tau$, $n \in \mathbb{Z}$, are used for counting:

$$\begin{aligned} c_{ij}^{\text{obs}}(\tau) &= c_{ij}^{\text{obs}}(l\Delta t) \\ &= \sum_{k=0}^{\lfloor (N-1)/l \rfloor -1} \chi_i(\mathbf{x}_{(l\cdot k)+1})\chi_j(\mathbf{x}_{(l\cdot k)+l+1}). \end{aligned} \tag{4.5}$$

This corresponds to the original meaning of counting transitions in the concrete case of a given trajectory and when the observed jump process is Markovian at lag time τ, this generates statistically independent transition counts. It is therefore straightforward to use the resulting count matrix in order to derive expressions for the likelihood and posterior of a transition matrix (see Sect. 4.3 below), which is important in order to obtain statistical models that do not underestimate the uncertainties [7, 8, 15]. A disadvantage of this approach is that intermediate data is ignored, which can lead to numerical problems when states that have been briefly visited only at intermediate times might be missed. This can sometimes cause numerical or algorithmic problems.

2. *Window count:* In this method we use a count window of width τ that is shifted along the time line:

$$\begin{aligned} c_{ij}^{\text{obs}}(\tau) &= c_{ij}^{\text{obs}}(l\Delta t) \\ &= \frac{\lfloor (N-1)/l \rfloor}{N-l} \sum_{k=1}^{N-l} \chi_i(\mathbf{x}_k)\chi_j(\mathbf{x}_{k+l}). \end{aligned} \tag{4.6}$$

This method uses all observed samples, although nearby transitions such as $t \to t+\tau$ and $(t+\Delta t) \to (t+\Delta t)+\tau$ cannot be assumed to be statistically independent. Doing so would generate a posterior distribution that is too narrow in the Bayesian uncertainty analysis approaches below, and should not be used with these methods. For this reason a correction factor is introduced and the final counting then corresponds to an estimation of the expected average number of transitions seen in trajectories of length N[1] Even without the correction factor, the expectation value of the estimated transition matrix $\mathbf{T}(\tau)$ is unbiased, and thus maximum posterior estimators (Sect. 4.5) are asymptotically correct, such that the window count method is preferred for the case of transition matrix estimation.

4.3 Likelihood, Prior and Posterior Distribution

It is intuitively clear that in the limit of an infinitely long trajectory, the elements of the true transition matrix are uniquely given by the trivial estimator,

$$\hat{T}_{ij}(\tau) \equiv \frac{c_{ij}^{\mathrm{obs}}(\tau)}{c_i^{\mathrm{obs}}(\tau)}, \tag{4.7}$$

i.e., the fraction of time the system is found to be in state j a time τ after it was initially observed to be in state i. For a trajectory of finite length, the underlying or generating transition matrix $\mathbf{T}(\tau)$ no more be determined uniquely since several transition matrices allow for generation of the same finite trajectory but with different probabilities.

Assuming that the matrix $\mathbf{C}^{\mathrm{obs}}(\tau)$ contains statistically independent transition counts (see discussion in Sect. 4.2 above), the probability that a particular $\mathbf{T}(\tau)$ would generate an observed sequence $s_1, \ldots, s_N$ is given by the product of the individual jump probabilities, $T_{s_k,s_{k+1}}(\tau)$ along the sequence $\{s_k\}$, $k \in \{1, 1+l, 1+2l, \ldots\}$ [1]. In terms of our notation, this probability can be rewritten using the observed count matrix $\mathbf{C}^{\mathrm{obs}}(\tau)$, hereafter suppressing the τ argument as convenient,

$$p\big(\mathbf{C}^{\mathrm{obs}}\big|\mathbf{T}\big) = \prod_{i,j=1}^{n} T_{ij}^{c_{ij}^{\mathrm{obs}}}. \tag{4.8}$$

The probability that the true transition matrix which generated these observed counts has the value $\mathbf{T}(\tau)$ can then be computed by Bayes' rule,

$$p\big(\mathbf{T}\big|\mathbf{C}^{\mathrm{obs}}\big) \propto p(\mathbf{T})\,p\big(\mathbf{C}^{\mathrm{obs}}\big|\mathbf{T}\big) = p(\mathbf{T}) \prod_{i,j=1}^{n} T_{ij}^{c_{ij}^{\mathrm{obs}}}. \tag{4.9}$$

We term $p(\mathbf{T}|\mathbf{C}^{\mathrm{obs}})$ the *posterior* probability of the transition matrix $\mathbf{T}$, $p(\mathbf{T})$ the *prior* probability of transition matrices before observing any data and $p(\mathbf{C}^{\mathrm{obs}}|\mathbf{T})$ is called the *likelihood*.

In transition matrix estimation, one is interested in the most probable matrices $\mathbf{T}$, i.e., the $\mathbf{T}$ with a large density in the posterior probability. The prior probability should be chosen such that it restricts the posterior to solutions that are physically meaningful in the situation where little observation data is available, but otherwise should be "weak", i.e., impose little bias (see Sect. 4.4 for a discussion on the choice of priors). For computational convenience, it is common to select a prior that is conjugate to the likelihood, i.e., shares the same functional form as the likelihood. This leads to a posterior of the form

$$p\big(\mathbf{T}\big|\mathbf{C}^{\mathrm{obs}}\big) \propto \prod_{i,j=1}^{n} T_{ij}^{c_{ij}^{\mathrm{prior}}+c_{ij}^{\mathrm{obs}}} = \prod_{i,j=1}^{n} T_{ij}^{c_{ij}}, \tag{4.10}$$

with the prior (pseudo-)count matrix $\mathbf{C}^{\mathrm{prior}} \equiv (c_{ij}^{\mathrm{prior}})$ and we have defined the effective total number of counts, or posterior counts $\mathbf{C} \equiv \mathbf{C}^{\mathrm{prior}} + \mathbf{C}^{\mathrm{obs}}$. When a uniform distribution is used as a prior ($\mathbf{C}^{\mathrm{prior}} = 0$, $p(\mathbf{T}) \propto 1$), likelihood and posterior distribution are identical. Since any estimator will be based only on the posterior counts, it is reasonably to split the discussion into priors and estimation algorithms.

[1] At the moment it is an open question how to best make use of all observed data while at the same time using statistically independent, or at least uncorrelated counts. It appears straightforward to use the window method, obtaining non-integer effective counts, but an estimation of the real uncorrelated counts is still an open question. A safe approach is to use the window count method for estimating the transition matrix and sampling the trajectory at lag τ when computing count matrices for error estimators.

4.4 Choices of Prior for Transition Matrix T

The choice of the prior distribution $p(\mathbf{T})$ allows the estimation of the Markov model to be controlled when only a limited quantity of observation data is available. A prior serves two purposes: (1) it may guarantee numerical stability of the estimator in the case where this would be otherwise difficult due to few observed transitions, such as early on in the data collection process, and (2) it can help to enforce physically reasonable constraints on the solution, such as detailed balance in physical systems at equilibrium.

In principle, the choice of prior is irrelevant in the data-rich regime; once sufficient data has been collected for the likelihood function to overwhelm the prior, the posterior distribution will very closely resemble the data-driven likelihood function. However, since one often does not always operate in a data-rich regime, especially during early stages of data collection and model building, it is important to understand the effects of the choice of prior on any model results in order to avoid overinterpreting features that may depend on the choice of prior rather than the data.

Several choices of prior $p(\mathbf{T})$ are in common use that can be defined by their choice of $\mathbf{C}^{\text{prior}}$ prior (pseudo-)count matrix:

- **The uniform prior** $c_{ij}^{\text{prior}} = 0$. In this case the posterior distribution is identical to the likelihood and all maximum probability estimators introduced subsequently become maximum likelihood estimators. When using maximum probability estimators, the uniform prior is preferable whenever sufficient observation data is available, as it gives full weight to the data. However, other features of the distribution (4.10), such as the expectation and the variances, are strongly affected by a uniform prior (see Sect. 4.4).
- **The flat/symmetric prior** $c_{ij}^{\text{prior}} = \gamma \in [-1, \infty]$. This prior is symmetric like the uniform prior since *a priori* all transitions are considered equally likely. The factor γ determines the *concentration* level of the symmetric prior in the sense that large values of γ prefer very similar values (unconcentrated probabilities) in the absence of observations while small values ($\gamma \to -1$) prefers all probability to be concentrated in a single transition. This prior can be relatively influential especially in the low-data regime as their influence to expectation values and variances depends the relation between the number of observations and the number of pseudo-counts, $n^2(\gamma + 1)/N$ (see Sect. 4.4). If the number of states n in the model is large then even small γ can cause the prior to overwhelm the available observed statistics and dominate the estimation of the transition matrix. To avoid this issue, Refs. [11, 23] suggested the choice $\gamma = n^{-1} - 1$, as this is identical to adding a single transition pseudo count to each row of the transition matrix, thus making the amount of information that the prior contributes to the outgoing transition probability distribution of each state independent of the number of states n in the system. However, even this choice may make the prior difficult to overcome by observation data, especially in large transition matrices as these are typically very sparse; a great deal of data needs to be collected in order to ensure that the nonzero transition probabilities dominate over transitions that have not yet been observed.
- **The neighbor prior** is a so-called *mixed prior*, as it takes information from the observation into account. It is defined as

$$c_{ij}^{prior} = \begin{cases} \gamma & \text{if } c_{ij}^{\text{obs}} + c_{ji}^{\text{obs}} > 0, \\ -1 & \text{else.} \end{cases} \tag{4.11}$$

This prior is more subtle than the flat prior as it only concentrates probability to transition pairs between which a transition has actually been observed, thus ensuring connectivity and that the detailed balance constraint $\pi_i T_{ij} = \pi_j T_{ji}$ can be satisfied with non-zero equilibrium probabilities. This choice has been found to be useful in a number of recent studies [2, 19].

4.5 Maximum Probability Estimators

We will now derive the *maximum probability estimator* for $\mathbf{T}$ by finding the transition matrix that maximizes $p(\mathbf{T}|\mathbf{C})$. Numerically, the posterior probability is difficult to work with due to the product over many small terms in (4.10). For optimization purposes, a common trick is to work instead with the log-likelihood,

$$Q(\mathbf{T}) \equiv \log p(\mathbf{T}|\mathbf{C}) = \sum_{i,j=1}^{n} c_{ij} \log T_{ij} \tag{4.12}$$

which is useful since the logarithm is a monotonic function—the maximum of $\log f(x)$ is also the maximum of $f(x)$. However, this function is not bounded from below, since for $T_{ij} \to 0$, $Q \to -\infty$. Of course, we need to further restrict ourselves to sets of variables which actually form valid row-stochastic transition matrices,

$$\begin{aligned} T_{ij} &\geq 0 \quad \forall i, j = 1, \ldots, n, \\ \sum_j T_{ij} &= 1 \quad \forall i = 1, \ldots, n. \end{aligned} \tag{4.13}$$

When imposing equality constraints, it is convenient to employ the method of Lagrange multipliers. The Lagrangian for $Q(\mathbf{T})$ is given by,

$$\begin{aligned} F(\mathbf{T}) = Q(\mathbf{T}) + \lambda_1\left(\sum_j T_{1j} - 1\right) + \cdots \\ + \lambda_m\left(\sum_j T_{mj} - 1\right). \end{aligned} \tag{4.14}$$

This function is maximized by the maximum probability transition matrix subject to the imposed constraints (4.13). When all $c_i > 0$, it turns out that $F(\mathbf{T})$ has a single stationary point—a maximum—which can be easily found by setting the partial derivatives to zero. These partial derivatives are given by

$$\frac{\partial \log F}{\partial T_{ij}} = \frac{c_{ij}}{T_{ij}} + \lambda_i. \tag{4.15}$$

Equating these with zero to find the maximum leads to

$$\frac{c_{ij}}{\hat{T}_{ij}} + \lambda_i = 0 \quad \Leftrightarrow \quad \lambda_i \hat{T}_{ij} = -c_{ij}.$$

We now make use of the row-stochastic nature of the transition matrix $\mathbf{T}$,

$$\lambda_i \sum_j \hat{T}_{ij} = \lambda_i = -\sum_j c_{ij} = -c_i$$

and thus obtain,

$$\begin{aligned} \frac{c_{ij}}{\hat{T}_{ij}} - c_i &= 0, \\ \hat{T}_{ij} &= \frac{c_{ij}}{c_i}. \end{aligned} \tag{4.16}$$

It turns out that $\hat{\mathbf{T}}(\tau)$, as provided by Eq. (4.16), is the maximum of the posterior $p(\mathbf{T}|\mathbf{C}^{\text{obs}})$ and thus also of the likelihood $p(\mathbf{C}^{\text{obs}}|\mathbf{T})$ when transition matrices are assumed to be uniformly distributed *a priori* (corresponding to the choice $c_{ij}^{\text{prior}} = 0$). In the limit of infinite sampling, i.e., trajectory length $N \to \infty$, $p(\mathbf{T}|\mathbf{C}^{\text{obs}})$ converges towards a Dirac delta distribution with its peak at $\hat{\mathbf{T}}(\tau)$. In this case the prior contribution vanishes:

$$\begin{aligned} \lim_{N\to\infty} \hat{T}_{ij} &= \lim_{N\to\infty} \frac{c_{ij}^{\text{prior}} + c_{ij}^{\text{obs}}}{c_i^{\text{prior}} + c_i^{\text{obs}}} \\ &= \lim_{N\to\infty} \frac{c_{ij}^{\text{obs}}}{c_i^{\text{obs}}} = T_{ij}, \end{aligned} \tag{4.17}$$

i.e., the estimator is “asymptotically unbiased”.

4.6 Reversible Transition Matrix Estimation

Note that the estimator $\hat{\mathbf{T}}(\tau)$ computed in Sect. 4.5 above will not necessarily fulfill the detailed balance constraint $\pi_i \hat{T}_{ij} = \pi_j \hat{T}_{ji}$ even if the underlying dynamics used to generate the dataset is in equilibrium and the true transition matrix represents a physical system that satisfies detailed balance $\pi_i T_{ij} = \pi_j T_{ji}$. It is often advantageous to enforce the detailed balance constraint during the estimation process; this guarantees, for example, that all eigenvalues of the transition matrix will be real (rather than complex). As there is no known closed form solution for the maximum probability estimator with the detailed balance constraint, we present two iterative methods subsequently.

Let $x_{ij} \equiv \pi_i T_{ij}$ be the unconditional transition probability to observe the system originally in state i and then in state j a time τ later—that is, the probability of observing a specific transition $i \to j$ out of all possible transitions. Because the transition matrix $\mathbf{T}$ is row-stochastic, the x_{ij} fulfill the constraint $\sum_{i,j} x_{ij} = 1$, and we can express the detailed balance condition as a symmetry constraint on the nonnegative matrix $\mathbf{X} \equiv [x_{ij}]$, namely $x_{ij} = x_{ji}$ for all pairs of states i, j. It is hence sufficient to store the x_{ij} with $i \le j$ in order to construct a reversible transition matrix.

The elements of the reversible transition matrix $\mathbf{T} = [T_{ij}]$ corresponding to a given symmetric $\mathbf{X}$ are given by,

$$T_{ij} = \frac{x_{ij}}{x_i} \tag{4.18}$$

where $x_i \equiv \sum_j x_{ij}$ is the ith row or column sum of $\mathbf{X}$.

The log-likelihood of $\mathbf{X}$ given a count matrix $\mathbf{C} = [c_{ij}]$ is given by,

$$\begin{aligned} Q(\mathbf{X}) &= \log p(\mathbf{C}|\mathbf{X}) \\ &= \sum_i c_{ii} \log \frac{x_{ii}}{x_i} \\ &\quad + \sum_{i<j} \left(c_{ij} \log \frac{x_{ij}}{x_i} + c_{ji} \log \frac{x_{ji}}{x_j} \right). \end{aligned} \tag{4.19}$$

We next consider several approaches for determining the estimator $\hat{\mathbf{X}}$ that maximizes the log-likelihood $Q(\mathbf{X})$,

$$\hat{\mathbf{X}} = \arg\max_{\mathbf{X}} Q(\mathbf{X}). \tag{4.20}$$

4.6.1 Optimization by Self-Consistent Iteration

An iterative approach can be constructed by making use of the convexity of $Q(\mathbf{X})$, as suggested in Ref. [4]. We first note that the partial derivatives of Q are given by,

$$\begin{aligned} \frac{\partial Q}{\partial x_{ij}} &= \frac{c_{ij}}{x_{ij}} + \frac{c_{ji}}{x_{ji}} - \sum_{i'=1}^{n} \frac{c_{ii'}}{\sum_{k=1}^{n} x_{ik}} \\ &\quad - \sum_{j'=1}^{n} \frac{c_{jj'}}{\sum_{k=1}^{n} x_{jk}}. \end{aligned} \tag{4.21}$$

Writing $c_i = \sum_{k=1}^{n} c_{ik}$ and $x_i = \sum_{k=1}^{n} x_{ik}$ we have,

$$\frac{\partial Q}{\partial x_{ij}} = \frac{c_{ji} + c_{ij}}{x_{ij}} - \frac{c_i}{x_i} - \frac{c_j}{x_j}. \tag{4.22}$$

When $Q(\mathbf{X})$ is maximized, we have $\frac{\partial Q}{\partial x_{ji}} = 0$, which yields the self-consistency condition,

$$\hat{x}_{ij} = \frac{c_{ij} + c_{ji}}{\frac{c_i}{\hat{x}_i} + \frac{c_j}{\hat{x}_j}}. \tag{4.23}$$

This approach can then be used in an iterative scheme where the previous iterate $\mathbf{X}^{(n)}$ is used to produce the next iterate $\mathbf{X}^{(n+1)}$, according to the update rule,

$$x_{ij}^{(n+1)} = \frac{c_{ij} + c_{ji}}{\frac{c_i}{x_i^{(n)}} + \frac{c_j}{x_j^{(n)}}}, \tag{4.24}$$

which can be iterated until the log-likelihood converges. The transition matrix $\hat{\mathbf{T}}$ can then be calculated from $\hat{\mathbf{X}}$ via Eq. (4.18).

4.6.2 Optimization by Iterative Conditional Maximization

An alternative optimization approach utilizes the fact that the conditional probability for a single x_{ij} has a closed-form solution for the optimum if all the other elements of $\mathbf{X}$ are held fixed, allowing an iterative element-wise optimization scheme to be constructed [20]. Such a conditional optimization can be done by solving a one-dimensional quadratic optimization problem for each element of the $\mathbf{X}$-matrix. We distinguish between diagonal and off-diagonal entries:

1. $i = j$

$$\begin{aligned}
&\arg\max_{x_{ii}} \log p(\mathbf{C}|\mathbf{X}) \\
&\quad = \arg\max_{x_{ii}} c_{ii} \log \frac{x_{ii}}{x_{ii} + x_{i,-i}} \\
&\qquad + \sum_{k\neq i} c_{ik} \log \frac{x_{ik}}{x_{ii} + x_{i,-i}} \\
&\quad = \arg\max_{x_{ii}} c_{ii} \log x_{ii} - c_i \log(x_{ii} + x_{i,-i})
\end{aligned} \tag{4.25}$$

where $x_{i,-j} = \sum_{k\neq j} x_{ik} = x_i - x_{ij}$. The maximum has a simple closed form solution,

$$x_{ii} = \frac{c_{ii} x_{i,-i}}{c_i - c_{ii}}. \tag{4.26}$$

2. $i < j$

$$\begin{aligned}
&\arg\max_{x_{ij}} \log p(\mathbf{C}|\mathbf{X}) \\
&\quad = \arg\max_{x_{ij}} \biggl(c_{ij} \log \frac{x_{ij}}{x_{ij} + x_{i,-j}} \\
&\qquad + \sum_{k\neq j} c_{ik} \log \frac{x_{ik}}{x_{ij} + x_{i,-j}} \\
&\qquad + c_{ji} \log \frac{x_{ji}}{x_{ji} + x_{j,-i}} \\
&\qquad + \sum_{k\neq i} c_{jk} \log \frac{x_{jk}}{x_{ji} + x_{j,-i}} \biggr) \\
&\quad = \arg\max_{x_{ij}} (c_{ij} + c_{ji}) \log x_{ij} \\
&\qquad - c_i \log(x_{ij} + x_{i,-j}) \\
&\qquad - c_j \log(x_{ij} + x_{j,-i}).
\end{aligned} \tag{4.27}$$

Then the optimal x_{ij} satisfies

$$\frac{c_{ij} + c_{ji}}{x_{ij}} = \frac{x_i}{x_{ij} + x_{i,-j}} + \frac{c_j}{x_{ij} + x_{j,-i}}, \tag{4.28}$$

$$\begin{aligned}
&(c_{i,-j} + c_{j,-i}) x_{ij}^2 + \bigl(c_i x_{j,-i} + c_j x_{i,-j} \\
&\quad - (c_{ij} + c_{ji})(x_{i,-j} + x_{j,-i})\bigr) x_{ij} \\
&\quad - (c_{ij} + c_{ji}) x_{i,-j} x_{j,-i} = 0.
\end{aligned} \tag{4.29}$$

Therefore

$$x_{ij} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{4.30}$$

where

$$\begin{aligned}
a &= c_{i,-j} + c_{j,-i}, \\
b &= c_i x_{j,-i} + c_j x_{i,-j} \\
&\quad - (c_{ij} + c_{ji})(x_{i,-j} + x_{j,-i}), \\
c &= -(c_{ij} + c_{ji}) x_{i,-j} x_{j,-i}.
\end{aligned}$$

Note that $x_{ij} > 0$ and

$$\frac{-b - \sqrt{b^2 - 4ac}}{2a} < 0 \tag{4.31}$$

for $a > 0$ and $ac < 0$, such that

$$x_{ij} = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \tag{4.32}$$

is the desired maximum.

The maximum probability estimator is then obtained by the following iterative Algorithm 1 (proven to converge to the optimum in Ref. [20], Supplementary Information), which is iterated until some stopping criterion is met (e.g. change of $\max_{i,j}\{x_{ij}\}$ in one iteration is smaller than a given constant or the number of iterations exceeds a pre-defined threshold).

While both algorithms should converge to the same estimator for convex $Q(\mathbf{X})$, the rate at which the algorithms converge, as well as the computational effort required to reach the same error, differs. Figure 4.2 compares the convergence rates of the two schemes using a small and a large transition matrix as examples. It is seen that the iterative elementwise optimizer converges asymptotically in approximately fivefold fewer steps compared to the direct method, while using slightly more CPU time per iteration.

Note that the optimum sought by Eq. (4.20) exhibits $\hat{T}_{ij} = 0$ if $c_{ij} + c_{ji} = 0$ for both estimators. Thus, in both optimization algorithms, the sparsity structure of the matrix $\mathbf{C} + \mathbf{C}^T$ can be used in order to restrict all iterations to the elements that will result in a nonzero element $\hat{T}_{ij} > 0$.

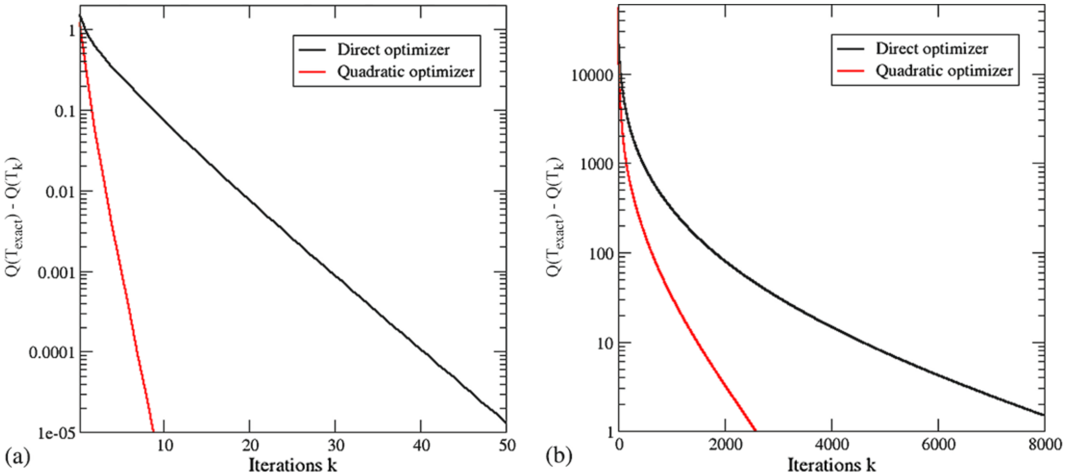


Fig. 4.2 Performance comparison of the direct and quadratic optimizer for reversible transition matrix estimation. Shown is the difference of the current likelihood to the optimal likelihood. (**a**) Results for the 3×3 count matrix $C = \begin{bmatrix} 5 & 2 & 0 \\ 1 & 1 & 1 \\ 2 & 5 & 20 \end{bmatrix}$. (**b**) Results for the 1734×1734 count matrix from Pin WW folding simulations used in Ref. [17]

4.7 Validation 1: Implied Timescales

One of the most interesting kinetic properties of molecular systems are the *relaxation timescales* of the system. Various experimental spectroscopic techniques measure relaxation or correlation functions that represent linear combinations of exponentially-decaying components, with each component representing a different relaxation timescale of the overall system [3, 6, 12, 14]. These relaxation timescales can also be directly computed for the simulated system from the eigenvalues of the transition matrix $\mathbf{T}(\tau)$ of the corresponding Markov model, where they are known as *implied timescales* t_i,

$$t_i(\tau) = -\frac{\tau}{\ln \lambda_i(\tau)},$$

where here $t_i(\tau)$ denotes the ith slowest implied timescale determined from the ith largest eigenvalue $\lambda_i(\tau)$, which is computed from the transition matrix $\mathbf{T}(\tau)$.

One popular test of the quality or self-consistency of a Markov state model was suggested in Ref. [26]. If the MSM constructed for a lag time τ_0 approximates the dynamics well for all $\tau \geq \tau_0$, then we should find that the eigenvalues $\lambda_i(k\tau_0)$ are well approximated by $[\lambda_i(\tau_0)]^k$ for $k = 1, 2, \ldots$, and hence that the implied timescales are relatively constant over this range of $\tau = k\tau_0$,

$$t_i(k\tau_0) = -\frac{\tau_0}{\ln \lambda_i(\tau_0)} \approx t_i \quad \text{for } k = 1, 2, \ldots \tag{4.43}$$

Ref. [26] has therefore suggested that one test whether the slowest implied timescales, $t_i(k\tau)$, computed from different lag times $\tau = k\tau_0$ are approximately constant for $k = 1, 2, \ldots$ to assess the quality of the MSM constructed at lag time τ_0.

Two notes of caution must be made at this point: (1) Observing convergence of the slowest implied timescales in τ is not a strict test of Markovianity. While Markovian dynamics, $\mathbf{T}(k\tau_0) = [\mathbf{T}(\tau_0)]^k$, implies constancy of implied timescales in τ [16, 25], the reverse is not true and would require the eigenvectors to be constant as well. (2) The argument above does not include the effect of statistical error and is thus strictly only valid for the limit of good sampling. In many practical cases, statistics are insufficient and the implied timescales do not show the expected behavior that permits the quality of the discretization to be assessed. In this case, additional sampling is required.

Algorithm 1 Maximum probability estimator of reversible transition matrices

(1) For all $i, j = 1, \ldots, n$: initialize

$$x_{ij} = x_{ji} := c_{ij} + c_{ji} \tag{4.33}$$

$$x_i := \sum_j x_{ij} \tag{4.34}$$

(2) Repeat until stopping criterion is met (1.1.) For all $i = 1, \ldots, n$:

$$\text{update } x_{ii} := \frac{c_{ii}(x_i - x_{ii})}{c_i - c_{ii}} \tag{4.35}$$

$$\text{update } x_i := \sum_j x_{ij} \tag{4.36}$$

(1.2) For all $i = 1, \ldots, n-1$, $j = i+1, \ldots, n$:

$$a = c_i - c_{ij} + c_j - c_{ji} \tag{4.37}$$

$$\begin{aligned} b = {} & c_i(x_j - x_{ij}) + c_j(x_i - x_{ij}) \\ & - (c_{ij} + c_{ji})(x_i + x_j - 2x_{ij}) \end{aligned} \tag{4.38}$$

$$\begin{aligned} c = {} & -(c_{ij} + c_{ji})(x_i - x_{ij}) \\ & \times (x_j - x_{ij}) \end{aligned} \tag{4.39}$$

$$\text{update } x_{ij} = x_{ji} := \frac{-b + \sqrt{b^2 - 4ac}}{2a} \tag{4.40}$$

$$\text{update } x_i := \sum_j x_{ij} \tag{4.41}$$

(2) Update T_{ij}, $i, j = 1, \ldots, n$:

$$T_{ij} := \frac{x_{ij}}{x_i} \tag{4.42}$$

However, it has been empirically observed that constant implied timescales are a very strong indicator that the MSM constructed for lag time τ_0 approximates the underlying dynamics well [9].

If the discretized dynamics are not exactly Markovian, how do the implied timescales $t_i(\tau)$ estimated from the transition matrix $\mathbf{T}(\tau)$ behave with lag time τ? In Ref. [18], we derived the following tight bound for the discrepancy between the implied timescale $t_2(\tau)$ computed from the transition matrix $\mathbf{T}(\tau)$ of the MSM and the true dominant relaxation timescale t_2^* of the system (ignoring statistical error),

$$\frac{t_2^* - t_2(\tau)}{t_2(\tau)} \leq \frac{t_2^*}{\tau} \ln \frac{1}{\alpha}$$

where $\alpha = \langle \psi_2, \hat{\psi}_2 \rangle_\mu$ is the discretization quality with respect to the second propagator eigenfunction. In simple words, if $\alpha = 1$, the state space discretization resolves the slowest process perfectly, while if $\alpha = 0$, the slowest process is completely concealed by the discretization.

We observe two things: (1) the true implied timescale t_2^* is well approximated if the state space discretization is very good ($\alpha \approx 1$ such that $\ln \alpha^{-1} \approx 0$), and (2) the implied timescale $t_2(\tau)$ converges towards the true implied timescale t_2^* as the lag time τ is increased. Unfortunately, this convergence with τ is slow, with systematic errors decaying only as τ^{-1}.

Following Ref. [10], we can make similar statements for the other relaxation processes of the system. The implied timescales $t_j(\tau)$ converge to their true relaxation timescales t_j as either the discretization quality $\alpha_j \equiv \langle \psi_j, \hat{\psi}_j \rangle_\mu$ increases or the lag time τ increases,

$$\lim_{\tau \to \infty} \left| t_j(\tau) - t_j^* \right| = 0, \tag{4.44}$$

and also,

$$\lim_{\delta_j \to 0} \left| t_j(\tau) - t_j^* \right| = 0, \tag{4.45}$$

where $\delta_j = 1 - \alpha_j$ is the projection error of the state space discretization with respect to the jth dynamical process. This fact has been empirically observed in numerous previous studies [5, 7, 8, 16, 17, 25, 26].

From the mathematical results above, the following rationale to assess the quality of the state space discretization can be used:

1. For a given state space discretization, estimate a series of transition matrices $\mathbf{T}(\tau_k)$ for $\tau_k \equiv k\Delta t$, where Δt is the time step between saved trajectory frames and k is a variable integer, using the methods described in Sects. 4.1–4.6.
2. Compute the m largest eigenvalues of $\mathbf{T}(\tau_k)$, and from these the m slowest implied timescales $t_i(\tau_k)$ depending on lag time τ_k.

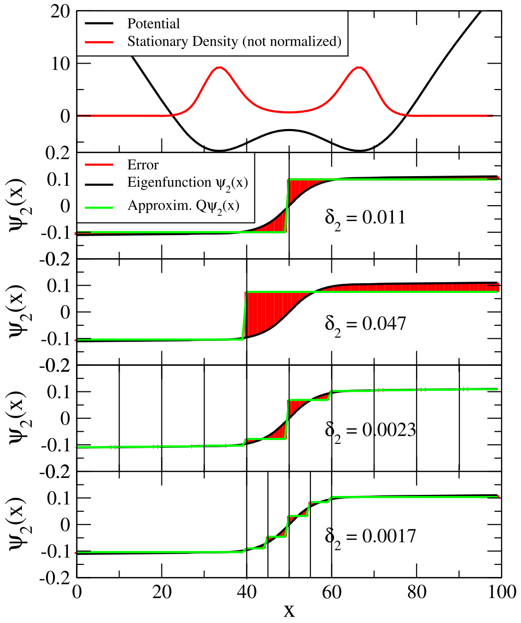


Fig. 4.3 Illustration of the eigenfunction approximation error δ_2 on the slow transition in the diffusion in a double well (*top*, *black line*). The slowest eigenfunction is shown in the lower four panels (*black*), along with the step approximations (*green*) of the partitions (*vertical black lines*) at $x = 50$; $x = 40$; $x = 10, 20, \ldots, 80, 90$; and $x = 40, 45, 50, 55, 60$. The eigenfunction approximation error δ_2 is shown as red area and its norm is printed

3. When the implied timescales $t_i(\tau_k)$ reach an approximately constant value for increasing lag time τ_k, the state space discretization is sufficiently good to resolve the dynamics in these slowest processes. Usually, it is also expected that the lag times for which this approximate constant value is reached are significantly smaller than the true relaxation timescales t_i^* of interest.
4. Select the minimal lagtime τ at which $t_i(\tau)$ are approximately constant, and use $\mathbf{T}(\tau)$ as Markov model.

We conclude that, given *sufficient* statistics, observing the lag time-dependence of the implied timescales is a useful approach to asses the quality of the discretization, and to choose a lag time τ at which $\mathbf{T}(\tau)$ shall be estimated, but this model needs to be subsequently validated (see Sect. 4.8).

As illustrative examples, consider a one-dimensional (see Figs. 4.4 and 4.3) and a two-dimensional (see Figs. 4.5 and 4.6) comparison of diffusion processes under various state space discretizations. Figure 4.4 shows the slowest implied timescale t_2 for the diffusion in a two-well potential (see Fig. 4.3) with discretizations shown in Fig. 4.3. The two-state partition at $x = 50$ requires a lag time of ~2000 steps in order to reach an error of <3 % with respect to the true implied timescale, which is somewhat slower than t_2 itself. When the two-state partition is distorted by shifting the discretization border to $x = 40$, this quality is not reached before the process itself has relaxed. Thus, in this system two states are not sufficient to build a Markov model that is at the same time precise and has a time resolution good enough to trace the decay of the slowest process. By using more states and particularly a finer discretization of the transition region, the same approximation quality is obtained with only $\tau \approx 1500$ (blue) and $\tau \approx 500$ steps (green).

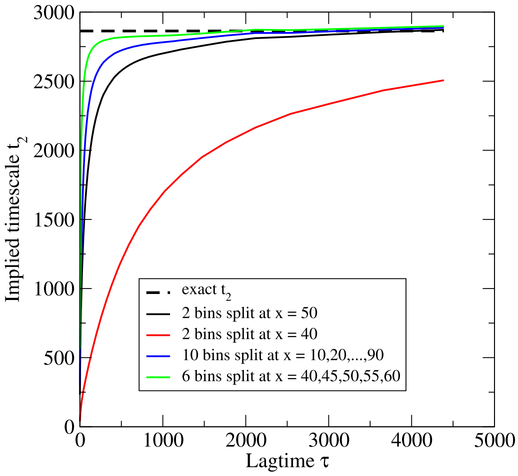


Fig. 4.4 Convergence of the slowest implied timescale $t_2 = -\tau / \ln \lambda_2(\tau)$ for the diffusion in a double-well potential for various MSM discretizations (see Fig. 4.3). The metastable partition (*black*, *solid*) has greater error than non-metastable partitions (*blue*, *green*) with more states that better trace the change of the slow eigenfunction near the transition state

Figure 4.6 shows the two slowest implied timescales t_2, t_3 for the diffusion in a two-dimensional three-well potential with discretizations shown in Fig. 4.5a. The metastable 3-state partition requires a lag time of ≈1000 steps in order to reach an error of <3 % with respect to the true implied timescale, which is

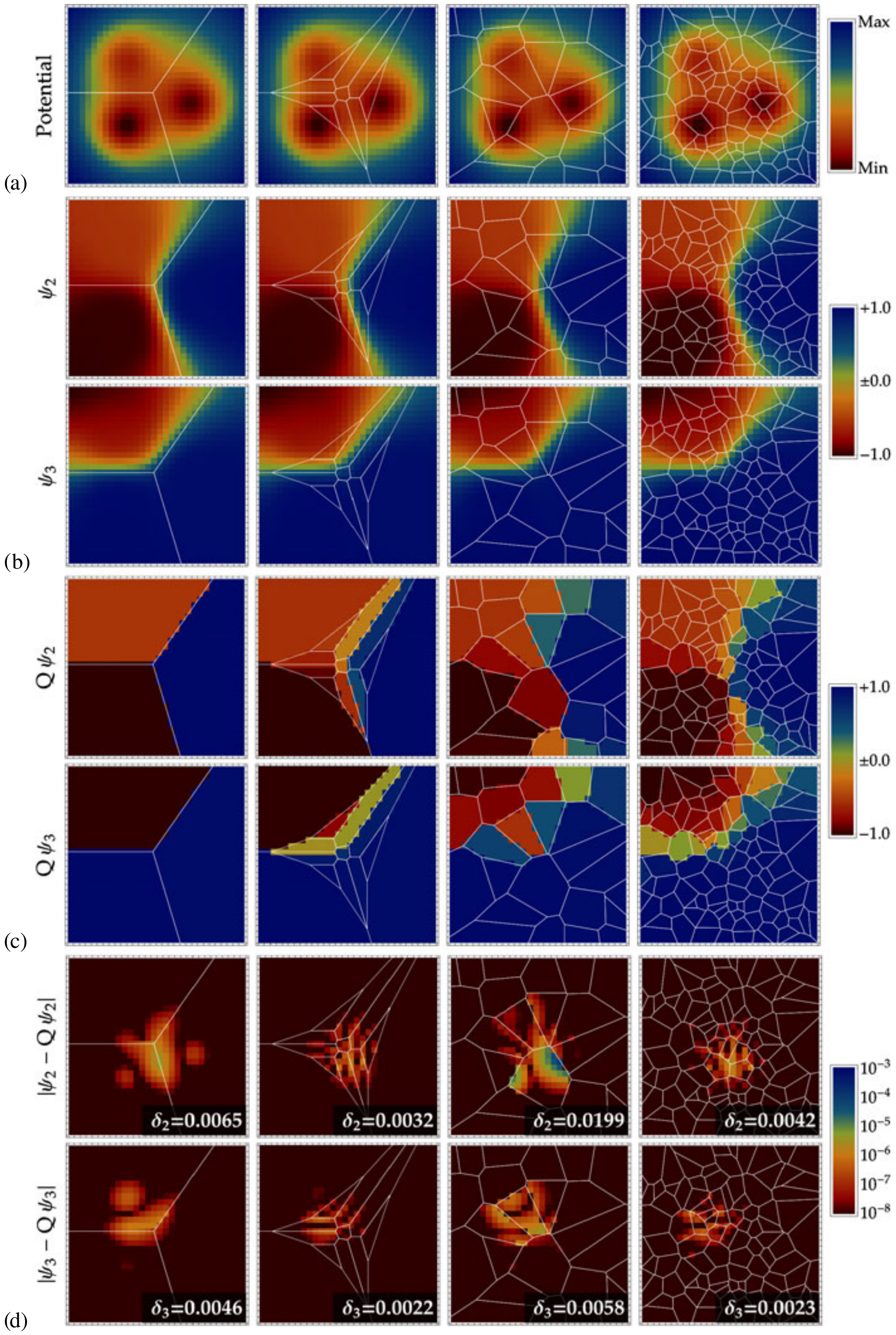


Fig. 4.5 Illustration of the eigenfunction approximation errors δ_2 and δ_3 on the two slowest processes in a two-dimensional three-well diffusion model. The *columns* (**i**, **ii**, **iii**, **iv**) from left to right show different state space discretizations with white lines as state boundaries: (**i**) 3 states with maximum metastability, (**ii**) the metastable states were further subdivided manually to better resolve the transition region, resulting in a partition where no individual state is metastable, (**iii**)/(**iv**) Voronoi partition using 25/100 randomly chosen centers, respectively. (**a**) Potential, (**b**) The exact eigenfunctions of the slow processes, $\psi_2(\mathbf{x})$ and $\psi_3(\mathbf{x})$, (**c**) The approximation of eigenfunctions with discrete states, $Q\psi_2(\mathbf{x})$ and $Q\psi_3(\mathbf{x})$, (**d**) Approximation errors $|\psi_2 - Q\psi_2|(\mathbf{x})$ and $|\psi_3 - Q\psi_3|(\mathbf{x})$. The error norms δ_2 and δ_3 are given

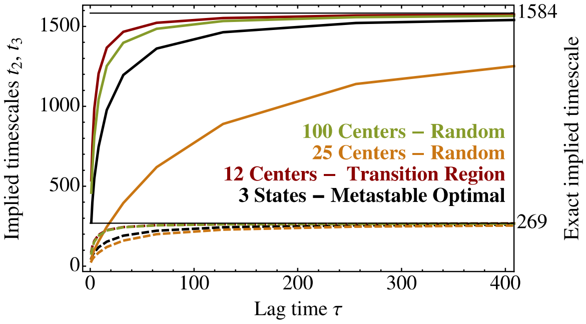


Fig. 4.6 Implied timescales for the two slowest processes in the two-dimensional three-well diffusion model (see Fig. 4.5a for an illustration of the potential energy surface, and Supplementary Information for details of the potential). The colors black, red, yellow, green correspond to the four choices of discrete states shown in columns 1 to 4 of Fig. 4.5. A fine discretization of the transition region clearly gives the best approximation to the timescales at small lag times

somewhat shorter than the slow but longer than the fast timescale, while refining the discretization near the transition states achieves the same precision with $\tau \approx 200$ using only 12 states. A k-means clustering with $k = 25$ is worse than the metastable partition, as some clusters cross over the transition region and fail to resolve the slow eigenfunctions. Increasing the number of clusters to $k = 100$ improves the result significantly, but is still worse than the 12 states that have been manually chosen so as to well resolve the transition region. This suggests that excellent MSMs could be built with rather few states when an adaptive algorithm that more finely partitions the transition region is employed.

4.8 Validation 2: Chapman-Kolmogorov Test

Above, we formulated criteria for selecting a state space discretization and a lag time τ that minimize the discretization error of a MSM. In practice, however, it is essential to test whether the resulting Markov model is at least consistent with the data used to parametrize it to within statistical error. While theoretical studies can compare the difference between Markov model propagation and true propagation in the continuous space [20], in practical situations, one is limited to measuring the propagation error using the available state space discretization. In particular, we are interested in checking whether the approximation,

$$\left[\hat{\mathbf{T}}(\tau)\right]^k \approx \hat{\mathbf{T}}(k\tau), \qquad (4.46)$$

holds to within statistical uncertainty. Here, $\hat{\mathbf{T}}(\tau)$ is the transition matrix estimated from the data at lag time τ (the Markov model), and $\hat{\mathbf{T}}(k\tau)$ is the transition matrix estimated from the same data at longer lag times $k\tau$. Note that when the nonreversible maximum likelihood estimator, Eq. (4.16), is used, this approximation is trivially exact for $k = 1$ since the Markov model was parametrized at lag time τ. For all $k \gg t_2/\tau$, the approximation should always be good, as Markov models correctly model the stationary distribution reached after a few global relaxation times t_2, even for bad choices of τ and discretization. Thus, this test is only sensitive in ranges of k greater one and smaller than the global relaxation time t_2 of the system.

Although there are various ways of how a test for Eq. (4.46), any implementation should consider the following points:

1. For large transition matrices, individual elements of $\hat{\mathbf{T}}(k\tau)$ or $[\hat{\mathbf{T}}(\tau)]^k$ can have large statistical uncertainty (even if this uncertainty does not have much impact on the overall dynamics), and comparing n^2 elements may be cumbersome. Therefore, we suggest to compare the probability of being in a given set of states, A, when starting from a well-defined

starting distribution. This simplifies the test to few observables and allows to check the kinetics of states that are of special interest, such as folded/unfolded states or metastable states.

2. The test should be done for all times $k\tau$ for which trajectory data is available. Tests that compare Markov models that differ only one lag step (τ and 2τ) are likely to be unreliable as small approximation errors at short times may amplify at long times.
3. The quality of the approximation (4.46) should be judged within the statistical uncertainties induced by the data.

Here, we present an implementation that takes these properties into account. Let $\boldsymbol{\pi}$ be the stationary probability of the Markov model $\hat{\mathbf{T}}(\tau)$. The corresponding stationary distribution restricted to a set A is then given by,

$$w_i^A = \begin{cases} \frac{\pi_i}{\sum_{j\in A}\pi_j} & i \in A, \\ 0 & i \notin A, \end{cases} \tag{4.47}$$

where π_i is the probability of state i.

As a model test, the following "relaxation experiment" may be carried out for each set: Using $\mathbf{w}^A$ as an initial probability vector for each of the sets under consideration, the probability of finding the system in that set at times $k\tau$ is then computed according to (i) the observed trajectory data and (ii) the Markov model, and these probabilities are subsequently compared. The trajectory-based time-dependence of the probability to be at set A after time $k\tau$ with starting distribution $\mathbf{w}^A$ is given by:

$$p_{\mathrm{MD}}(A, A; k\tau) = \sum_{i\in A} w_i^A \, p_{\mathrm{MD}}(i, A; k\tau), \tag{4.48}$$

where $p_{\mathrm{MD}}(i, A; k\tau)$ is the trajectory-based estimate of the probability to be in set A at time $k\tau$ when starting from state i at time 0:

$$p_{\mathrm{MD}}(i, A; k\tau) = \frac{\sum_{j\in A} c_{ij}^{\mathrm{obs}}(k\tau)}{\sum_{j=1}^{n} c_{ij}^{\mathrm{obs}}(k\tau)}. \tag{4.49}$$

Likewise, the probability to be at A according to the Markov model is given by:

$$p_{\mathrm{MSM}}(A, A; k\tau) = \sum_{i\in A} \left[\left(\mathbf{w}^A\right)^T \mathbf{T}^k(\tau) \right]_i. \tag{4.50}$$

Testing the validity of the Markov model then amounts to testing how well the presumed equality,

$$p_{\mathrm{MD}}(A, A; k\tau) \approx p_{\mathrm{MSM}}(A, A; k\tau), \tag{4.51}$$

holds, which is essentially a test of the Chapman-Kolmogorov property. Note that the initial distribution $\mathbf{w}^A$ is simply a chosen reference distribution with respect to which the comparison is made, here chosen as in Eq. (4.47).

Equation (4.51) is not expected to be an exact equality even for perfectly Markovian systems due to statistical uncertainties; only a finite number of transitions are available to estimate the true transition probabilities, meaning there will be residual statistical error in the estimated transition matrices. To account for this, the uncertainties (one-sigma standard error) of the transition probabilities estimated from MD trajectories are computed as,

$$\begin{aligned} &\epsilon_{\mathrm{MD}}(A, A; k\tau) \\ &\quad = \sqrt{k \frac{p_{\mathrm{MD}}(A, A; k\tau) - [p_{\mathrm{MD}}(A, A; k\tau)]^2}{\sum_{i\in A}\sum_{j=1}^{n} c_{ij}^{\mathrm{obs}}(k\tau)}}. \end{aligned} \tag{4.52}$$

A practical test then consists of assessing whether Eq. (4.51) holds within these uncertainties. The uncertainty of $p_{\mathrm{MSM}}(A, A; k\tau)$ can be calculated using the methods described in Sect. 4.4. However, this is not necessary if the test already succeeds while using only the uncertainties $\epsilon_{\mathrm{MD}}(A, A; k\tau)$.

For illustration, we show results of this test using a 10^6 step trajectory of a diffusion in a double-well potential (see Fig. 4.3 for details). Figure 4.7 shows the relaxation out of the left well using a two-state discretization splitting at $x = 50$ (see Fig. 4.3b for state definition and Fig. 4.7a for results) and using a six-state discretization splitting at $x = \{40, 45, 50, 55, 60\}$ (see Fig. 4.3c for state definition and Fig. 4.7b for results). The two-state discretization provides spurious results for $\tau = 100$, good results for $\tau = 500$ and for $\tau = 2000$ the results are excellent within the statistical uncertainty of the trajectory. For the six-state discretization even $\tau = 100$ is

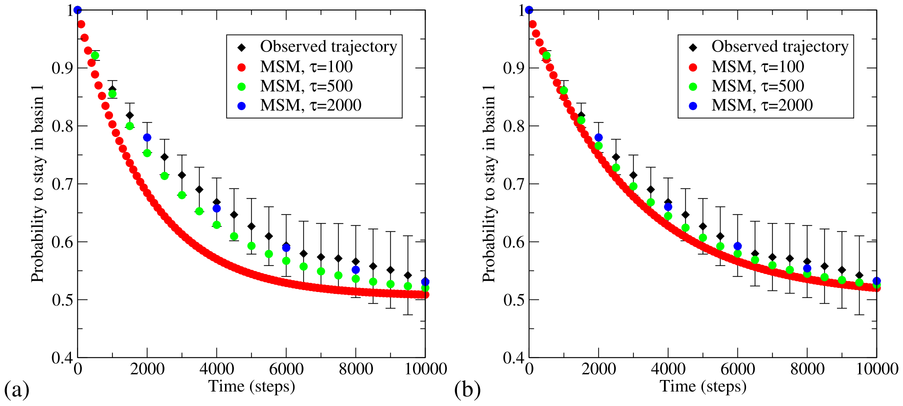


Fig. 4.7 Chapman-Kolmogorov test applied to the two-well diffusion system (see Fig. 4.3 for details) using a trajectory of length 10^6 steps. Tested are Markov models that use lag times $\tau = 100, 500, 2000$ and (**a**) 2-state discretization (split at $x = 50$), (**b**) 6-state discretization (split at $x = 40, 45, 50, 55, 60$)

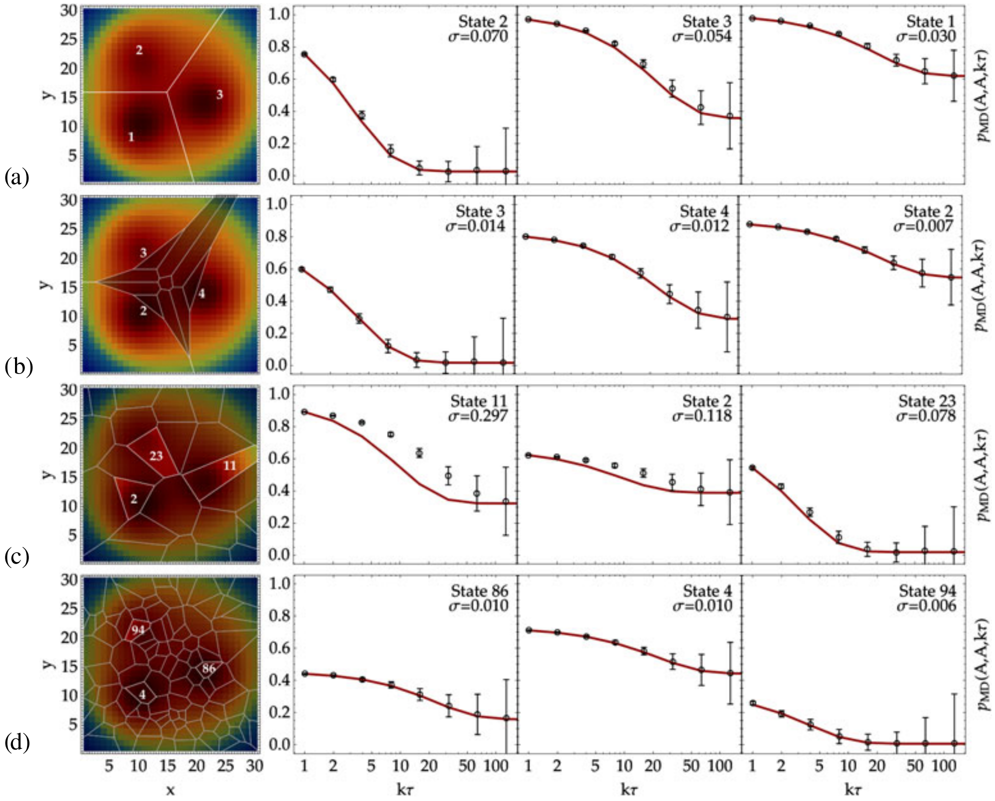


Fig. 4.8 Chapman-Kolmogorov test for the three well diffusion model (see also Fig. 4.5). For each of four discretizations (*first column*, **a**, **b**, **c**, **d**), the Chapman-Kolmogorov test is shown for the three states with the greatest error (labeled with white figures in the *first column*). Relaxation curves from a 250,000 step trajectory, $p_{\mathrm{MD}}(A, A; k\tau)$ (*black*) along with the uncertainties $\epsilon_{\mathrm{MD}}(A, A, k\tau)$ are compared to the model prediction, $p_{\mathrm{MSM}}(A, A; k\tau)$ (*red*). The total error σ given in the top right corners is measured as the 2-norm of the vector containing the differences $p_{\mathrm{MD}}(A, A; k\tau) - p_{\mathrm{MSM}}(A, A; k\tau)$ for time points in the range $k\tau \in [1, 128]$

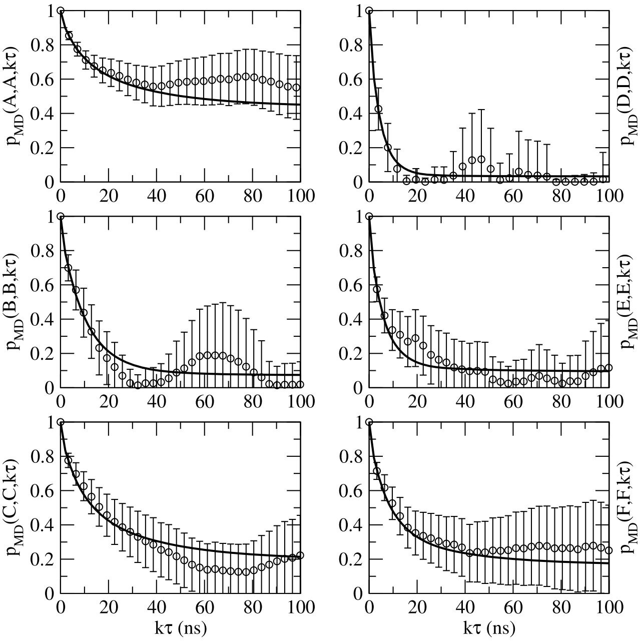


Fig. 4.9 Chapman-Kolmogorov test for six metastable sets A to F in MR121-GSGS-W. *Solid curve*: $p_{\mathrm{MSM}}(A, A; k\tau)$ to $p_{\mathrm{MSM}}(F, F; k\tau)$ predicted by the MSM parameterized at lag time $\tau = 2$ ns. *Bullets with error bars*: expectation values and uncertainties of $p_{\mathrm{MD}}(A, A; k\tau)$ to $p_{\mathrm{MD}}(F, F; k\tau)$ directly calculated from the simulation data up to 100 ns

within the error bars while $\tau = 500$ and $\tau = 2000$ both yield excellent approximations.

Figure 4.8 shows the corresponding results for the three-well diffusion model (see also Fig. 4.5). A single 250,000 step trajectory started from the energy minimum at $\mathbf{x} = (10, 10)$ was simulated. For each of the four different discretizations shown in the first column of Fig. 4.8 the probability to stay in a state is shown for the three states with the largest Markov model error (highlighted in Fig. 4.8, left column). It is apparent that the metastable three-state discretization (Fig. 4.8a) performs well, however sacrificing metastability in order to more finely discretize the transition region generates a superior discretization (Fig. 4.8b). The "uninformed" random 25-state clustering (Fig. 4.8c) performs worst but can be improved significantly by using more states (Fig. 4.8d). This further supports the theoretical finding that the quality of the Markov model depends on the approximation quality of the dominant eigenvectors [22] which can be achieved by either a clustering adapted to the eigenfunctions or using more states.

Figure 4.9 shows Chapman-Kolmogorov test results for the six most metastable sets of the MR121-GSGS-W peptide using a Markov model based on a Voronoi discretization using least-squares RMSD to 1000 peptide configurations equally spaced in time selected from the trajectory. The lag time was set to $\tau = 2$ ns. The metastable states are determined by dominant eigenvectors and have been calculated with the PCCA+ method [16, 27]. The Markov model agrees with the observed trajectory within statistical uncertainty for all metastable states.

References

1. Anderson TW, Goodman LA (1957) Statistical inference about Markov chains. Ann Math Stat 28:89–110
2. Beauchamp KA, Bowman GR, Lane TJ, Maibaum L, Haque IS, Pande VS (2011) MSMBuilder2: modeling conformational dynamics at the picosecond to millisecond scale. J Chem Theory Comput 7(10):3412–3419. doi:10.1021/ct200463m
3. Bieri O, Wirz J, Hellrung B, Schutkowski M, Drewello M, Kiefhaber T (1999) The speed limit for protein folding measured by triplet-triplet energy transfer. Proc Natl Acad Sci USA 96(17):9597–9601. http://www.pnas.org/content/96/17/9597.abstract
4. Bowman GR, Beauchamp KA, Boxer G, Pande VS (2009) Progress and challenges in the automated construction of Markov state models for full protein systems. J Chem Phys 131(12):124,101+. doi:10.1063/1.3216567

5. Buchete NV, Hummer G (2008) Coarse master equations for peptide folding dynamics. J Phys Chem B 112:6057–6069
6. Chan CK, Hu Y, Takahashi S, Rousseau DL, Eaton WA, Hofrichter J (1997) Submillisecond protein folding kinetics studied by ultrarapid mixing. Proc Natl Acad Sci USA 94(5):1779–1784. http://www.pnas.org/content/94/5/1779.abstract
7. Chodera JD, Dill KA, Singhal N, Pande VS, Swope WC, Pitera JW (2007) Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics. J Chem Phys 126:155,101
8. Chodera JD, Noé F (2010) Probability distributions of molecular observables computed from Markov models, II: uncertainties in observables and their time-evolution. J Chem Phys 133:105,102
9. Chodera JD, Swope WC, Pitera JW, Dill KA (2006) Long-time protein folding dynamics from short-time molecular dynamics simulations. Multiscale Model Simul 5:1214–1226
10. Djurdjevac N, Sarich M, Schütte C (2010) Estimating the eigenvalue error of Markov state models. Multiscale Model Simul. doi:10.1137/10079891
11. Hinrichs NS, Pande VS (2007) Calculation of the distribution of eigenvalues and eigenvectors in Markovian state models for molecular dynamics. J Chem Phys 126:244,101
12. Jäger M, Nguyen H, Crane JC, Kelly JW, Gruebele M (2001) The folding mechanism of a beta-sheet: the WW domain. J Mol Biol 311(2):373–393. doi:10.1006/jmbi.2001.4873
13. Laio A, Parrinello M (2002) Escaping free energy minima. Proc Natl Acad Sci USA 99:12,562
14. Neuweiler H, Doose S, Sauer M (2005) A microscopic view of miniprotein folding: enhanced folding efficiency through formation of an intermediate. Proc Natl Acad Sci USA 102(46):16,650–16,655. doi:10.1073/pnas.0507351102
15. Noé F (2008) Probability distributions of molecular observables computed from Markov models. J Chem Phys 128:244,103
16. Noé F, Horenko I, Schütte C, Smith JC (2007) Hierarchical analysis of conformational dynamics in biomolecules: transition networks of metastable states. J Chem Phys 126:155,102
17. Noé F, Schütte C, Vanden-Eijnden E, Reich L, Weikl TR (2009) Constructing the full ensemble of folding pathways from short off-equilibrium simulations. Proc Natl Acad Sci USA 106:19,011–19,016
18. Prinz JH, Chodera JD, Noé F (2013) Spectral rate theory for two-state kinetics. Phys Rev X (accepted)
19. Prinz JH, Held M, Smith JC, Noé F (2011) Efficient computation of committor probabilities and transition state ensembles. Multiscale Model Simul 9:545
20. Prinz JH, Wu H, Sarich M, Keller B, Fischbach M, Held M, Chodera JD, Schütte C, Noé F (2011) Markov models of molecular kinetics: generation and validation. J Chem Phys 134:174,105
21. Röblitz S (2009) Statistical error estimation and grid-free hierarchical refinement in conformation dynamics. PhD thesis
22. Sarich M, Noé F, Schütte C (2010) On the approximation error of Markov state models. Multiscale Model Simul 8:1154–1177
23. Singhal N, Pande VS (2005) Error analysis and efficient sampling in Markovian state models for molecular dynamics. J Chem Phys 123:204,909
24. Sugita Y, Okamoto Y (1999) Replica-exchange molecular dynamics method for protein folding. Chem Phys Lett 314:141–151
25. Swope WC, Pitera JW, Suits F (2004) Describing protein folding kinetics by molecular dynamics simulations, 1: theory. J Phys Chem B 108:6571–6581
26. Swope WC, Pitera JW, Suits F, Pitman M, Eleftheriou M (2004) Describing protein folding kinetics by molecular dynamics simulations, 2: example applications to alanine dipeptide and beta-hairpin peptide. J Phys Chem B 108:6582–6594
27. Weber M (2003) Improved Perron cluster analysis. ZIB report 03-04

Uncertainty Estimation 5

Frank Noé and John D. Chodera

As only a finite quantity of data can be collected for the construction of Markov state models, the parameters characterizing the model and any properties computed from it will always be statistically uncertain. This chapter is concerned with the quantification of this statistical uncertainty, and its use in validation of model quality and prediction of properties using the model. In the following sections we proceed along Refs. [2, 7, 11] which should be used for reference purposes.

5.1 Uncertainties in Transition Matrix Elements

We first consider the uncertainty in the transition matrix $\mathbf{T}(\tau)$ itself estimated from a finite quantity of data. It may be the case that the uncertainty in individual elements $T_{ij}(\tau)$ may be of interest, in which case standard errors or confidence intervals of these estimates may be sufficient tools to quantify the uncertainty.

For a transition matrix estimated without the detailed balance constraint, the expectation and variance of individual elements follow from well-known properties of the distribution of stochastic matrices [1]. These uncertainties do, however, depend on the choice of prior used in modeling the full posterior for the transition matrix (Sect. 4.4). Under a uniform prior, the expectation and variance of an individual element T_{ij} is given by,

$$\mathbb{E}[T_{ij}] = \frac{c_{ij}+1}{c_i+n} \equiv \bar{T}_{ij}, \tag{5.1}$$

$$\begin{aligned} \operatorname{Var}[T_{ij}] &= \frac{(c_{ij}+1)((c_i+n)-(c_{ij}+1))}{(c_i+n)^2((c_i+n)+1)} \\ &= \frac{\bar{T}_{ij}(1-\bar{T}_{ij})}{c_i+n+1}, \end{aligned} \tag{5.2}$$

where c_{ij} and c_i are the elements and row sums, respectively, of the observed count matrix $\mathbf{C}^{\mathrm{obs}}$ (Sect. 4.2).

To see the effect that the choice of prior has on the computed uncertainties, consider a trajectory of a given molecular system which is analyzed with two different state space discretizations. Assume one discretization uses $n = 10$ states, and the other $n = 1000$. Assume that a lag time τ has been chosen which is identical and long enough to provide Markov models with small discretization error for both n (as suggested in Sect. 4.7). With a uniform prior ($c_{ij} = c_{ij}^{\mathrm{obs}}$), the posterior expectation $\bar{T}_{ij}$ would be different for the two discretizations: While in the $n = 10$ case we can get a distinct transition matrix estimation, in the $n = 1000$ case, most c_{ij} are probably zero and $c_i \ll n$, such that the expectation value would be biased towards the uninformative $T_{ij} \approx 1/n \pm 1/n$ matrix, and many observed transitions would be needed to overcome this bias. This behavior is

F. Noé (✉)
Freie Universität Berlin, Arnimallee 6, 14195 Berlin, Germany
e-mail: frank.noe@fu-berlin.de

J.D. Chodera
Memorial Sloan-Kettering Cancer Center, New York, NY 10065, USA
e-mail: choderaj@mskcc.org

G.R. Bowman et al. (eds.), *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology 797,
DOI 10.1007/978-94-007-7606-7_5, © Springer Science+Business Media Dordrecht 2014

undesirable. Thus, for uncertainty estimation it is suggested to use a prior which allows the observation data to have more impact also in the low-data regime.

On the other hand, the "null prior" [10] defined by

$$c_{ij}^{\mathrm{prior}} \to -1 \quad \forall i, j \in \{1, \ldots, n\}, \tag{5.3}$$

leans to the other extreme. Under the null prior, the expectation and the variance of the marginalized posterior for a single T_{ij} become,

$$\bar{T}_{ij} = \mathbb{E}[T_{ij}] = \frac{c_{ij}^{\mathrm{obs}}}{c_i^{\mathrm{obs}}} = \hat{T}_{ij}, \tag{5.4}$$

$$\begin{aligned}\mathrm{Var}(T_{ij}) &= \frac{c_{ij}^{\mathrm{obs}}(c_i^{\mathrm{obs}} - c_{ij}^{\mathrm{obs}})}{(c_i^{\mathrm{obs}})^2(c_i^{\mathrm{obs}} + 1)} \\ &= \frac{\hat{T}_{ij}(1 - \hat{T}_{ij})}{c_i^{\mathrm{obs}} + 1}.\end{aligned} \tag{5.5}$$

Thus, with a null prior, the expectation value is located at the likelihood maximum. Both expectation value and variance are independent of the number of discretization bins used. The variance of any T_{ij} asymptotically decays with the number of transitions out of the state i, which is expected for sampling expectations from the central limit theorem.

5.2 Uncertainties in Computed Properties

In practice, one is often not primarily interested in the uncertainties of the transition matrix elements themselves, but rather in the uncertainties in properties computed *from* the transition matrix. Here, we review two different approaches for this purpose.

- **Linear error perturbation** [4, 12, 13]. Here, the transition matrix posterior distribution is approximated by a multivariate Gaussian, and the property of interest—taken to be a function of the transition matrix or its eigenvalues and eigenvectors—is approximated by a first-order Taylor expansion about the center of this Gaussian. This results in a Gaussian distribution of the property of interest, with a mean and a covariance matrix that can be computed in terms of the count matrix $\mathbf{C}$. This approach has the advantage that error estimates and their rates of reduction for different sampling strategies can be computed through a direct procedure. As a result, it is convenient for situations where uncertainty estimates are used as part of an adaptive sampling procedure [4, 8, 9, 13]. The disadvantage of this approach is that the Gaussian approximation of the transition matrix posterior in only asymptotically correct, and can easily break down when few counts have been observed. In the low-data regime, the resulting Gaussian distribution for the property of interest often gives substantial probability to unphysical or meaningless values, such as when transition matrix elements T_{ij} are allowed to assume values outside the range [0, 1]). Moreover, the property of interest is approximated linearly which can introduce a significant error when this property is nonlinear.
- **Markov chain Monte Carlo (MCMC) sampling of transition matrices** [2, 6, 7]. Here, transition matrices are sampled from the posterior distribution, and the property of interest is computed for each of these and stored as samples from the posterior distribution of the property. This approach requires that the sampling procedure be run sufficiently long that good estimates of standard deviations or confidence intervals of the posterior distribution of the property of interest can be computed, which may be time-consuming. The advantage of this approach is that no assumptions are made concerning the functional form of the distribution or the property being computed. Furthermore, this approach can be straightforwardly applied to any function or property of transition matrices, including complex properties such as transition path distributions [10] without deriving the expressions necessary for the linear error perturbation analysis—often a cumbersome task. However, for large state spaces, the transition matrix $\mathbf{T}$ may grow so large as to make this procedure impractical.

5.3 Linear Error Propagation

We start again with the posterior distribution of row-stochastic transition matrices without the detailed balance constraint, given by Eq. (4.10). Defining a new matrix $\mathbf{U}$,

$$\mathbf{U} = [u_{ij}] = [c_{ij} + 1], \tag{5.6}$$

and using that the posterior probability $p(\mathbf{T} \mid \mathbf{C}^{\text{obs}})$ implicitly contains the prior probabilities Eq. (4.10) can be rewritten as:

$$p(\mathbf{T} \mid \mathbf{C}) = p\big(\mathbf{T} \bigm| \mathbf{C}^{\text{obs}}\big) \propto \prod_i \prod_j T_{ij}^{u_{ij}-1} \tag{5.7}$$

such that

$$\mathbf{T}_{i*} \sim \prod_i \text{Dir}(\mathbf{u}_{i*}) \tag{5.8}$$

where $\text{Dir}(\boldsymbol{\alpha})$ denotes the Dirichlet distribution, and $\boldsymbol{\theta} \sim \text{Dir}(\boldsymbol{\alpha})$ implies that $\boldsymbol{\theta}$ is drawn from the distribution

$$p(\boldsymbol{\theta}) \propto \prod_i \theta_i^{\alpha_i - 1}. \tag{5.9}$$

Based on well-established properties of this distribution, and using the abbreviation $u_i = \sum_j u_{ij}$, the moments of $p(\mathbf{T} \mid \mathbf{C})$ can be directly computed,

$$\begin{aligned}
\big[\mathbb{E}(\mathbf{T})\big]_{ij} &= \frac{u_{ij}}{u_i} = \frac{c_{ij}+1}{c_i+n} = \bar{T}_{ij}, \\
\big(\arg\max p(\mathbf{T}|\mathbf{C})\big)_{ij} &= \frac{u_{ij}-1}{u_{ij}-n} = \frac{c_{ij}}{c_i} = \hat{T}_{ij}, \\
\text{Var}(T_{ij}) &= \frac{u_{ij}(u_i - u_{ij})}{u_i^2(u_i+1)} \\
&= \frac{\bar{T}_{ij}(1-\bar{T}_{ij})}{(u_i+1)} \\
&= \frac{\bar{T}_{ij}(1-\bar{T}_{ij})}{c_i+n-1}, \\
\text{Cov}(T_{ij}, T_{ik}) &= \frac{-u_{ij}u_{ik}}{u_i^2(u_i+1)} \qquad \forall j \neq k.
\end{aligned}$$

Next, we determine how the uncertainties given by the variances and covariances of the transition matrix elements propagate onto uncertainties of functions derived from transition matrices, such as eigenvalues. If we do not have constraints between different rows, such as are imposed by detailed balance, the rows can be treated as independent random vectors, and thus,

$$\text{Cov}(T_{ij}, T_{lk}) = 0, \quad i \neq l. \tag{5.10}$$

We can thus define a covariance matrix $\boldsymbol{\Sigma}^{(i)}$ separately for each row i as,

$$\begin{aligned}
\Sigma_{jk}^{(i)} &:= \text{Cov}(T_{ij}, T_{ik}) \\
&= \frac{1}{u_i^2(u_i+1)}[u_i\delta_{jk}u_{ij} - u_{ij}u_{ik}] \\
&= \frac{1}{c_i}\big[\delta_{jk}\bar{T}_{ij} - \bar{T}_{ij}\bar{T}_{ik}^T\big],
\end{aligned}$$

where δ is the Kronecker delta. Alternatively, we can write the covariance matrix $\boldsymbol{\Sigma}^{(i)}$ in vector notation,

$$\begin{aligned}
\boldsymbol{\Sigma}^{(i)} &= \frac{1}{u_i^2(u_i+1)}\big[u_i \,\text{diag}(\mathbf{u}_{i*}) - \mathbf{u}_{i*}(\mathbf{u}_{i*})^T\big] \\
&= \frac{1}{c_i}\big[\text{diag}(\bar{\mathbf{T}}_{i*}) - \bar{\mathbf{T}}_{i*}(\bar{\mathbf{T}}_{i*})^T\big].
\end{aligned}$$

In the limit of many observed transition counts, the covariance for the Dirichlet processes scales approximately with the inverse of the total number of counts in a row, c_i.

With a sufficient number of counts c_i in each row i, the Dirichlet process resembles a multivariate Gaussian distribution, and we can approximate it as such using the mean and variance computed above,

$$\mathbf{T}_{i*} \sim \text{Normal}\big(\hat{\mathbf{T}}_{i*}, \boldsymbol{\Sigma}^{(i)}\big). \tag{5.11}$$

This approximate distribution is used in a Gaussian error propagation for linear functions of the transition matrix. Let us assume that we are interested in computing the statistical error of a scalar functions $f(\mathbf{T}) : \mathbb{R}^{n \times n} \to \mathbb{R}$. The first order Taylor approximation is given by:

$$f(\mathbf{T}) = f(\hat{\mathbf{T}}) + \sum_{i,j} \frac{\partial f}{\partial T_{ij}}\bigg|_{\hat{\mathbf{T}}} (T_{ij} - \hat{T}_{ij}).$$

Since the uncertainty in the rows of $\mathbf{T}$ contribute independently to the uncertainty in f, we define a sensitivity vector $\mathbf{s}^{(i)}$ for each row separately

$$s_j^{(i)} = \frac{\partial f}{\partial T_{ij}}(\hat{\mathbf{T}})$$

that measures the sensitivity of the scalar function with respect to changes in the transition matrix elements. Then, with the function for the error propagation, we get

$$\hat{f} = f(\hat{\mathbf{T}})$$

obtaining an approximation for the variance in f,

$$\mathrm{Var}(f) = \mathrm{Cov}(f, f) = \sum_i \left(\mathbf{s}^{(i)}\right)^T \boldsymbol{\Sigma}^{(i)} \mathbf{s}^{(i)}.$$

or, more general, for the covariances between different scalar functions f, and g

$$\mathrm{Cov}(f, g) = \sum_i \left(\mathbf{s}[f]^{(i)}\right)^T \boldsymbol{\Sigma}^{(i)} \mathbf{s}[g]^{(i)}.$$

where $\mathbf{s}[f]^{(i)}$ and $\mathbf{s}[g]^{(i)}$ refer to the sensitivities of f and g respectively. The limitation of this approach is that it does not work well in situations where the Transition matrix distribution is far from Gaussian (especially in the situation of little data). Furthermore, the more nonlinear a given function of interest is in terms of T_{ij}, the more the estimated uncertainty on this function might be wrong.

5.3.1 Example: Eigenvalues

As an example, we consider the computation of statistical error in a particular eigenvalue λ_k of the transition matrix $\mathbf{T}$ using the linear error propagation scheme, closely following the approach described in Refs. [4, 13].

We start from the eigenvalue decomposition of the transition matrix $\mathbf{T}$, omitting the dependence on the lag time τ,

$$\boldsymbol{\Lambda} = \boldsymbol{\Phi} \mathbf{T} \boldsymbol{\Psi} \tag{5.12}$$

where $\boldsymbol{\Psi} = [\boldsymbol{\psi}_1, \dots, \boldsymbol{\psi}_n]$ is the right eigenvector matrix, $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \dots, \boldsymbol{\phi}_n]^T = \boldsymbol{\Psi}^{-1}$ is the left eigenvector matrix, and $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_i)$ is the diagonal matrix of eigenvalues. For the kth eigenvalue-eigenvector pair, we have,

$$\lambda^{(k)} = \left(\boldsymbol{\phi}^{(k)}\right)^T \mathbf{T} \boldsymbol{\psi}^{(k)} = \sum_{i,j} \phi_i^{(k)} T_{ij} \psi_j^{(k)}.$$

We wish to compute the statistical error of the eigenvalues $\lambda^{(k)}$ via linear error perturbation. In general, both the eigenvalues and eigenvectors simultaneously depend on perturbations in the elements of $\mathbf{T}$ in a complex way. To first order, the partial derivatives of the eigenvalues with respect to the transition matrix elements is given by the inner product of left and right eigenvectors,

$$\frac{\partial \lambda^{(k)}}{\partial T_{ij}} = \phi_i^{(k)} \psi_j^{(k)}. \tag{5.13}$$

This expression for the eigenvalue sensitivity may be combined with Eq. (5.11) in order to yield the linear perturbation result,

$$\begin{aligned}\mathrm{Var}\big(\lambda^{(k)}\big) &= \sum_{i=1}^{n} \sum_{a,b} \frac{\partial \lambda^{(k)}}{\partial T_{ia}} \mathrm{Cov}(T_{ab}) \frac{\partial \lambda^{(k)}}{\partial T_{ib}} \\ &= \sum_{i=1}^{n} \sum_{a,b} \phi_i^{(k)} \psi_a^{(k)} \Bigg(\sum_a \frac{u_{ia}(u_i - u_{ia})}{u_i^2(u_i+1)} \\ &\quad + \sum_{a,b\neq a} \frac{-u_{ia}u_{ib}}{u_i^2(u_i+1)} \Bigg) \phi_i^{(k)} \psi_b^{(k)}.\end{aligned}$$

5.4 Sampling Transition Matrices Without Detailed Balance Constraint

In a full Bayesian approach, we sample the posterior distribution,

$$p(\mathbf{T} \mid \mathbf{C}) \propto p(\mathbf{T}) p(\mathbf{C} \mid \mathbf{T}) = \prod_{i,j} T_{ij}^{c_{ij}} \tag{5.14}$$

where we recall that the total count matrix $\mathbf{C} = \mathbf{C}^{\mathrm{obs}} + \mathbf{C}^{\mathrm{prior}}$, as discussed in Chap. 4, makes the use of different priors straightforward. If the only constraint of $\mathbf{T}$ is that it is a stochastic matrix, but we do not expect that $\mathbf{T}$ fulfills detailed balance, we can view Eq. (5.14) as a product of Dirichlet distributions, one for each row (see Eq. (5.7)).

We are then faced with the problem of sampling random variables from the distribution,

$$\mathbf{T}_{i*} \sim \mathrm{Dir}(\mathbf{u}_{i*}). \tag{5.15}$$

A fast way to generate Dirichlet-distributed random variables is to draw n independent samples $y_1, \dots, y_n$ from univariate Gamma distributions, each with density,

$$y_j \sim \mathrm{Gamma}(c_{ij}+1, 1) = \frac{y_j^{c_{ij}} e^{-y_j}}{\Gamma(c_{ij}+1)}, \quad j = 1, \dots, n, \tag{5.16}$$

and then obtain the T_{ij} by normalization of each row,

$$T_{ij} = \frac{y_j}{\sum_{m=1}^{n} y_k}. \tag{5.17}$$

Repeating this procedure independently for every row $i = 1, \dots, n$ will generate a statistically independent sample of $\mathbf{T}$ from distribution (5.14).

5.5 Sampling the Reversible Transition Matrix Distribution

No similarly simple approach to direct generation of statistically independent samples of the distribution (5.14) exists when the transition matrix $\mathbf{T}$ is further constrained to satisfy that the transition matrices fulfill detailed balance. To include the detailed balance constraints, we consider sampling Eq. (5.14) using the Metropolis-Hastings algorithm, where we propose a change to the transition matrix, $\mathbf{T} \to \mathbf{T}'$. This proposal is accepted with probability given by the Metropolis-Hastings criterion,

$$\begin{aligned} p_{\mathrm{acc}} &= \frac{p(\mathbf{T}' \to \mathbf{T})}{p(\mathbf{T} \to \mathbf{T}')} \frac{p(\mathbf{T}'|\mathbf{C})}{p(\mathbf{T}|\mathbf{C})} \\ &= \frac{p(\mathbf{T}' \to \mathbf{T})}{p(\mathbf{T} \to \mathbf{T}')} \frac{p(\mathbf{C}|\mathbf{T}')}{p(\mathbf{C}|\mathbf{T})} \\ &= \frac{p(\mathbf{T}' \to \mathbf{T})}{p(\mathbf{T} \to \mathbf{T}')} \frac{\prod_{i,j} T'^{c_{ij}}_{ij}}{\prod_{i,j} T^{c_{ij}}_{ij}}. \end{aligned} \tag{5.18}$$

This scheme requires efficient schemes to generate proposals $\mathbf{T} \to \mathbf{T}'$ that maintain the detailed balance constraint and are likely to be accepted, as well as a method of efficiently computing the ratio of transition probabilities $p(\mathbf{T}' \to \mathbf{T})/p(\mathbf{T} \to \mathbf{T}')$ for each proposal. Such a scheme was worked out in detail in Ref. [7], and we summarize the resulting method as Algorithm 2.

Example 1 Every 2 × 2 transition matrix is reversible. To see this, we can compute the stationary distribution from the dominant eigenvector,

$$\boldsymbol{\pi} = \left(\frac{T_{21}}{T_{12}+T_{21}}, \frac{T_{12}}{T_{12}+T_{21}} \right), \tag{5.19}$$

from which we can see that detailed balance is always fulfilled,

$$\pi_1 T_{12} = \frac{T_{21}}{T_{12}+T_{21}} T_{12} = \frac{T_{12}}{T_{12}+T_{21}} T_{21} = \pi_2 T_{21}. \tag{5.20}$$

Indeed, for 2 × 2 matrices the nonreversible transition matrix sampling scheme (Sect. 5.4) generates the same distribution as the reversible transition matrix sampling scheme in Algorithm 2. See Fig. 5.1B for an illustration of this sampling scheme applied to a 2 × 2 matrix.

Example 2 Figure 5.2 illustrates how the distribution of a 3 × 3 transition matrix differs between the nonreversible (panels B, E, H) and reversible (panels C, F, I) cases. For the matrix studied here, the distribution of reversible matrices is slightly narrower.

5.5.1 Sampling with Fixed Stationary Distribution

In some cases, the stationary distribution, $\boldsymbol{\pi}$, may be known exactly or to very small statistical error. For example, an efficient equilibrium simulation scheme (such as parallel tempering or metadynamics) or a Monte Carlo method may have generated a very precise estimate of $\boldsymbol{\pi}$ by simulating a perturbed system or one with unphysical dynamics. It may be useful to incorporate this information about $\boldsymbol{\pi}$ when inferring the posterior distribution of transition matrices, since it may significantly reduce the uncertainty.

Algorithm 2 Metropolis Monte Carlo sampling of reversible stochastic matrices

```
Input: Transition count matrix $\mathbf{C} \in \mathbb{N}_0^{n \times n}$. Number of samples $N$.
Output: Ensemble of reversible transition matrices, $\mathbf{T}_1, \ldots, \mathbf{T}_N$.
1. Initialize $T_{ij}^{(0)} = (c_{ij} + c_{ji}) / (\sum_{m=1}^{m} c_{ik} + c_{ki}) \forall i, j \in (1, \ldots, m)$.
2. Compute $\boldsymbol{\pi}$ as stationary distribution of $\mathbf{T}^{(0)}$ by solving $\boldsymbol{\pi}^{(0)} = \boldsymbol{\pi}^{(0)} \mathbf{T}^{(0)}$.
3. For $k = 1 \ldots N$
    3.1. Generate uniform random variables: $r_1, r_2 \sim \mathrm{Uniform}[0, 1]$
    3.2. $\mathbf{T}^{(k)} := \mathbf{T}^{(k-1)}$
    3.3. If $(r_1 < 0.5)$ *Reversible Element Shift:*
        3.3.1. Generate uniform random variables:
                $i, j \in \{1, \ldots, n\}, \Delta \in \left[\max\left\{-T_{ii}^{(k)}, -\frac{\pi_j^{(k)}}{\pi_i^{(k)}} T_{jj}^{(k)}\right\}, T_{ij}^{(k)}\right]$.
        3.3.2. $p_{acc} := \left(\frac{(T_{ij}^{(k)} - \Delta)^2 + T_{ji}^{(k)} - \frac{\pi_i^{(k)}}{\pi_j^{(k)}} \Delta)^2}{(T_{ij}^{(k)})^2 + (T_{ji}^{(k)})^2}\right) \left(\frac{T_{ii}^{(k)} + \Delta}{T_{ii}^{(k)}}\right)^{c_{ii}} \left(\frac{T_{ij}^{(k)} - \Delta}{T_{ij}^{(k)}}\right)^{c_{ij}} \left(\frac{T_{jj}^{(k)} + \frac{\pi_i^{(k)}}{\pi_j^{(k)}} \Delta}{T_{jj}^{(k)}}\right)^{c_{jj}}$
                $\times \left(\frac{T_{ji}^{(k)} - \frac{\pi_i^{(k)}}{\pi_j^{(k)}} \Delta}{T_{ji}^{(k)}}\right)^{c_{ji}}$
        3.3.3. If $(r_2 \leq p_{acc})$:
                Set $T_{ii}^{(k)} := T_{ii}^{(k-1)} + \Delta$ ; $T_{ij}^{(k)} := T_{ij}^{(k-1)} - \Delta$
                and $T_{jj}^{(k)} := T_{jj}^{(k-1)} + \Delta \pi_i^{(k)} / \pi_j^{(k)}$ ; $T_{ji}^{(k)} := T_{ji}^{(k-1)} - \Delta \pi_i^{(k)} / \pi_j^{(k)}$
    else *Node Shift*:
        3.3.4. Generate uniform random variables: $i \in (1, \ldots, n), \alpha \in \left[0, \frac{1}{1 - T_{ii}^{(k)}}\right]$.
        3.3.5. $p_{acc} := \alpha^{(n-2+c_i-c_{ii})} \left(\frac{1 - \alpha(1 - T_{ii}^{(k)})}{T_{ii}^{(k)}}\right)^{c_{ii}}$
        3.3.6. If $r_2 \leq p_{acc}$:
                For all $j \neq i$, set $T_{ij}^{(k)} := \alpha T_{ij}^{(k-1)}$.
                Set $T_{ii}^{(k)} = 1 - \sum_{j \neq i} T_{ij}^{(k)}$
        3.3.7. Update stationary distribution:
                For all $j \neq i$:, set $\pi_j^{(k)} := \frac{\alpha \pi_j^{(k-1)}}{\pi_i^{(k-1)} + \alpha(1 - \pi_i^{(k-1)})}$.
                Set $\pi_i^{(k)} := 1 - \sum_{j \neq i} \pi_j^{(k)}$.
```

To do this, we first note that the two types of Monte Carlo proposals utilized in Algorithm 2 above for sampling reversible transition matrices. One type of proposal (reversible element shifts) changes $\boldsymbol{\pi}$, while the other preserves $\boldsymbol{\pi}$ (node shift). We can suggest a straightforward modification of the **T**-sampling algorithm that will ensure $\boldsymbol{\pi}$ is constrained to some specified value during the sampling procedure.

We first give an algorithm to construct an initial transition matrix $\mathbf{T}^{(0)}$ with a specified stationary distribution $\boldsymbol{\pi}$ from a given count matrix **C** (Algorithm 3), and then use this to initialize a Monte Carlo transition matrix sampling algorithm that preserves the stationary distribution (Algorithm 4).

5.6 Full Bayesian Approach with Uncertainty in the Observables

Suppose we are interested in some experimentally-measurable function of state $A(\mathbf{x})$. An experiment may be able to measure an expecta-

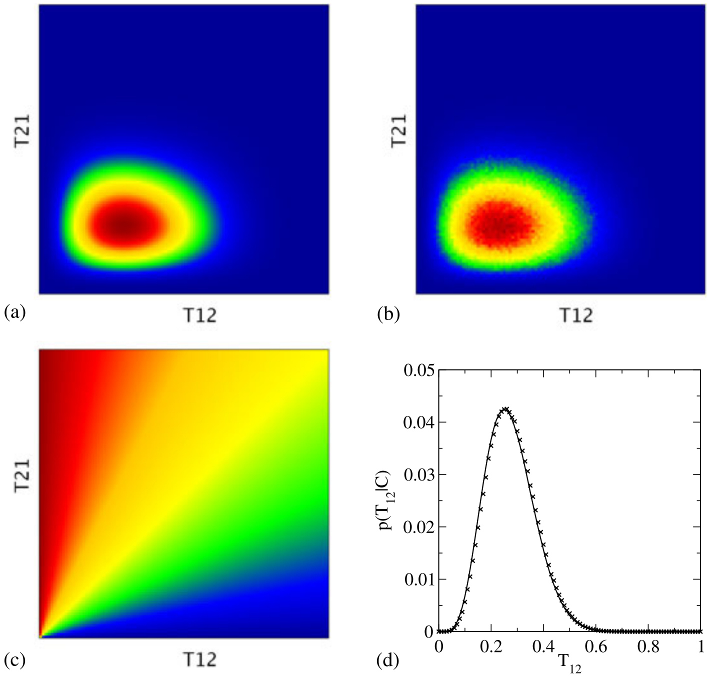


Fig. 5.1 Illustration of sampling of transition probability matrices for the observation $C = \begin{pmatrix} 5 & 2 \\ 3 & 10 \end{pmatrix}$ and a uniform prior. Panels (**a**), (**b**), and (**c**) show the probability distribution on the off-diagonal matrix elements. The color encodes the probability density, with *blue* = 0 and *red* = 1. Each density was scaled such that its maximum is equal to 1. (**a**) Analytic density of stochastic matrices. (**b**) Sampled density of stochastic matrices (these matrices automatically fulfill detailed balance). (**c**) Stationary probability of the first state π_1. When sampling with respect to a fixed stationary probability distribution π^*, the ensemble is fixed to the line $T_{21} = T_{12}\pi_1^*/(1-\pi_1^*)$. (**d**) Sampled and exact density of T_{12} of reversible matrices with fixed stationary distribution $\pi^* = (0.5, 0.5)$

tion $\langle A \rangle$ or correlation functions $\langle A(0)A(t) \rangle$, and we would like to compute the corresponding properties from the Markov model constructed from a molecular simulation and decide whether they agree with experiment to within statistical uncertainty, or if a prediction from the model is sufficiently precise to be useful. The previous framework for sampling transition matrices can be used in the following manner: (i) Assign the state-averaged value of the observable, $a_i = \int_{S_i} d\mathbf{x}\mu(\mathbf{x})A(\mathbf{x})$, to each discrete state. (ii) Generate an ensemble of **T**-matrices according to the sampling scheme described above. (iii) Calculate the desired expectation or correlation function for each **T**-matrix using the discrete vector $\mathbf{a} = [a_i]$. This approach involves several approximations that each deserve discussion. Here, we want to generalize the approach by eliminating one important approximation—that the values a_i are known exactly without statistical error themselves.

In a typical simulation scenario, the average a_i is itself calculated by a statistical sample. When a simulation trajectory $\mathbf{x}_t$ is available, then typically the time average

$$\hat{a}_i = \frac{\sum_t \chi_i(\mathbf{x}_t)A(\mathbf{x}_t)}{\sum_t \chi_i(\mathbf{x}_t)} \tag{5.21}$$

is employed, where χ_i is the indicator function of state i. The estimate $\hat{a}_i$ may in fact have significant statistical error because the number of uncorrelated samples of $\mathbf{x}_t$ inside any state i is finite, and possibly rather small. In order to estimate the distribution of expectation or correlation functions of A due to both, the statistical uncertainty of **T** and the statistical uncertainty of $\hat{a}_i$, we propose a full Bayesian approach using a Gibbs

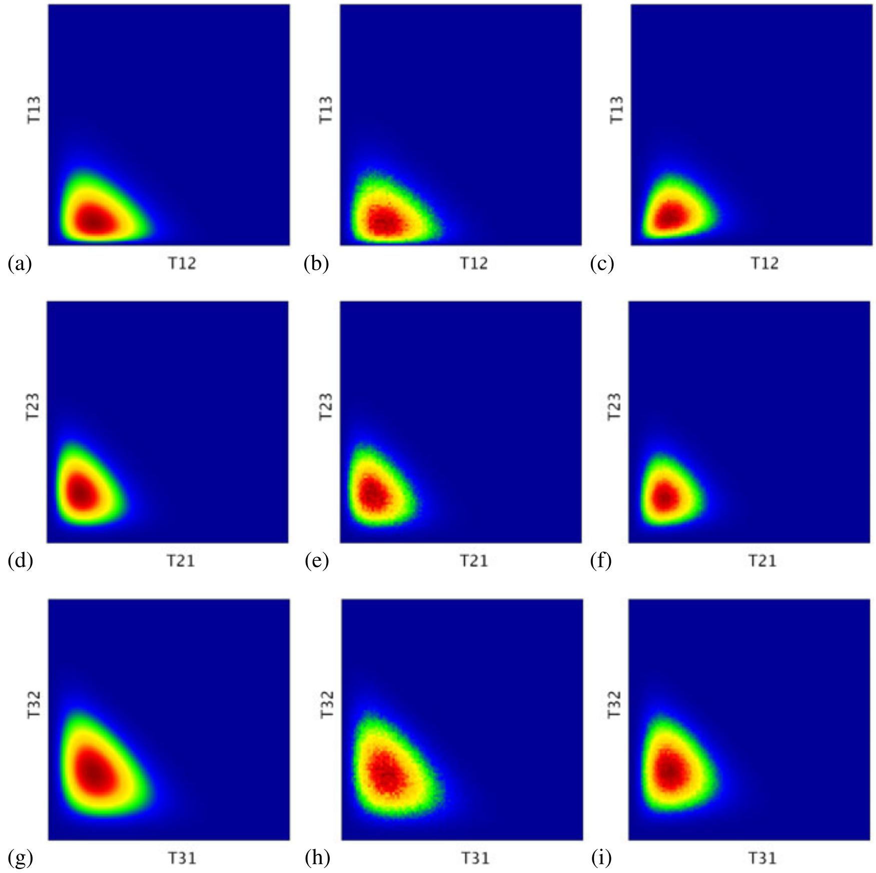


Fig. 5.2 Visualization of the probability density of transition matrices for the count matrix $\mathbf{C}^{\text{obs}} = \begin{pmatrix} 8 & 2 & 1 \\ 2 & 10 & 3 \\ 2 & 3 & 6 \end{pmatrix}$ and a uniform prior. Different two-dimensional joint marginal distributions are shown in the rows. The analytic and sampled distributions for stochastic matrices are shown in *columns 1* and *2*, respectively. *Column 3* shows the sampled distribution for stochastic matrices fulfilling detailed balance. Note how the peaks are more sharply peaked when the detailed balance constraint is imposed (*column 3*) compared to the corresponding transition matrices without detailed balance constraint (*column 2*)

sampling scheme, here illustrated for the expectation $\mathbb{E}[A]$ (Algorithm 5).

While the transition matrix $\mathbf{T}^{(k)}$ can be sampled using the framework described in the previous sections, an approach to sample $\mathbf{a}^{(k)}$ introduced in Ref. [2] is described subsequently.

5.6.1 Sampling State Expectations $\mathbf{a}^{(k)}$

Consider the expectation of some molecular observable $A(\mathbf{x})$ computed from Eq. (5.21). Temporally sequential samples $A_t \equiv A(\mathbf{x}_t)$ collected with a temporal resolution of the Markov time τ are subsequently presumed to be uncorrelated. We also assume that the set of samples $A(\mathbf{x}_t)$ for those configurations $\mathbf{x}_t$ appearing in state i are collected in the set $\{A_m\}_{m=1}^{N}$ in the remainder of this section, generally abbreviated as $\{A_m\}$.

Because only a finite number of samples N are collected for each state, there will be a degree of uncertainty in this estimate. Unlike the problem of inferring the transition matrix elements, however, we cannot write an exact expression for the probability of observing a single sample A_m in terms of a simple parametric form, since its probability distribution may be arbitrarily complex,

$$p_i(A_m) = \frac{1}{\pi_i} \int_{S_i} d\mathbf{x}\, \delta\big(A_m - A(\mathbf{x})\big) \mu(\mathbf{x}). \quad (5.22)$$

Algorithm 3 Generation of an initial transition matrix $\mathbf{T}^{(0)}$ given count matrix $\mathbf{C}$ and a specified stationary distribution $\boldsymbol{\pi}$

```
Input: Stationary distribution $\boldsymbol{\pi}$ and transition count matrix $\mathbf{C}$.
Output: Transition matrix $\mathbf{T}$ that has stationary distribution $\boldsymbol{\pi}$.
1. Define $\mathbf{Y} \in \mathbb{R}^{n \times n}$ as:
    $$y_{ij} = \begin{cases} \frac{\pi_i c_{ij}}{2\sum_k c_{ik}} + \frac{\pi_j c_{ji}}{2\sum_k c_{jk}} & i \neq j, \\ 0 & i = j. \end{cases}$$
2. Define $\mathbf{X} \in \mathbb{R}^{n \times n}$ as:
    $$o = \max_i \left\{ \sum_k x_{ik} \right\},$$
    $$x_{ij} = \begin{cases} \frac{y_{ij}}{o} & i \neq j, \\ \pi_i - \sum_k \frac{y_{ik}}{o} & i = j. \end{cases}$$
3. Define $\mathbf{T}^{(0)} \in \mathbb{R}^{n \times n}$ as
    $$T_{ij}^{(0)} = \frac{x_{ij}}{\sum_k x_{ik}}.$$
```

Algorithm 4 Metropolis-Hastings Monte Carlo sampling of reversible stochastic matrices with probability distribution of stationary distributions $p(\boldsymbol{\pi})$

```
Input: Transition count matrix $\mathbf{C} \in \mathbb{N}_0^{n \times n}$. Number of samples $n_1, n_2$. Stationary distribution $p(\boldsymbol{\pi})$
Output: Ensemble of reversible transition matrices, $\mathbf{T}_1, \ldots, \mathbf{T}_N$.
1. For $k = 1 \ldots n_1$
    1.1. Draw $\boldsymbol{\pi}^{(k)}$ from $p(\boldsymbol{\pi})$
    1.2. Initialize $\mathbf{T}^{(0)}$ using Algorithm 3.
    1.3. For $l = 1 \ldots n_2$
        1.3.1. Use *reversible element shift* from Algorithm 2 to update the transition matrix.
```

Algorithm 5 Gibbs sampler for the joint estimation of $p(\mathbb{E}[A])$

```
1. For $k = 1 \ldots N$
    1.1. Sample observables
            $\mathbf{a}^{(k)} \sim p(\mathbf{a} \mid \mathbf{x}_t)$.
    1.2. Sample transition matrix
            $\mathbf{T}^{(k)} \sim p(\mathbf{T} \mid \mathbf{x}_t) = p(\mathbf{T} \mid \mathbf{C}^{\text{obs}})$.
    1.3. Compute $\pi^{(k)}$ as the stationary distribution of $\mathbf{T}^{(k)}$ such that
            $[\boldsymbol{\pi}^{(k)}]^{\mathbb{T}} = [\mathbf{T}^{(k)}][\boldsymbol{\pi}^{(k)}]^{\mathbb{T}}$.
    1.3. Generate a sample of the expectation value:
            $$A^{(k)} = \sum_{i=1}^{n} a_i^{(k)} \pi_i^{(k)}.$$
```

Despite this, the central limit theorem states that the behavior of $\hat{a}_i$ approaches a normal distribution (generally very rapidly) as the number of samples N increases. We will therefore make the assumption that $p_i(A_m)$ is *normal*—that is, we assume the distribution can be characterized by mean μ_i and variance σ_i^2,

$$A_m \sim \text{Normal}\big(\mu_i, \sigma_i^2\big) \qquad (5.23)$$

where the normal distribution implies the probability density for A_m is approximated by

$$\begin{aligned}&\tilde{p}_i\big(A_m; \mu_i, \sigma_i^2\big)\\ &\quad = (2\pi)^{-1/2}\sigma_i^{-1}\exp\left[-\frac{1}{2\sigma_i^2}(A_m - \mu_i)^2\right].\end{aligned} \qquad (5.24)$$

While this may seem like a drastic assumption, it turns out this approximation allows us to do a surprisingly good job of inferring the distribution of the error in $\delta\hat{a}_i \equiv \hat{a}_i - \langle A\rangle_i$ even for a small number of samples from each state, and generally gives an overestimate of the error (which is arguably less dangerous than an underestimate) for smaller sample sizes. While the validity of this approximation is illustrated in a subsequent example, we continue below to develop the ramifications of this approximation.

Consider the sample mean estimator for $\langle A\rangle_i$,

$$\hat{\mu} = \frac{1}{N}\sum_{m=1}^{N} A_m. \qquad (5.25)$$

The asymptotic variance of $\hat{\mu}$, which provides a good estimate of the statistical uncertainty in $\hat{\mu}$ in the large-sample limit, is given as a simple consequence of the central limit theorem,

$$\begin{aligned}\delta^2\hat{\mu} &\equiv \mathbb{E}\big[\big(\hat{\mu} - \mathbb{E}[\hat{\mu}]\big)^2\big]\\ &= \frac{\operatorname{Var} A_m}{N} \approx \frac{\hat{\sigma}^2}{N}\end{aligned} \qquad (5.26)$$

where the unbiased estimator for the variance $\sigma^2 \equiv \operatorname{Var} A_m$ is given by

$$\hat{\sigma}^2 \equiv \frac{1}{N-1}\sum_{m=1}^{N}(A_m - \hat{\mu})^2 \qquad (5.27)$$

Suppose we now *assume* the distribution of A from state i is normal (Eq. (5.24)),

$$A|\mu, \sigma^2 \sim \text{Normal}\big(\mu, \sigma^2\big). \qquad (5.28)$$

Were this to be a reasonable model, we could model the timeseries of the observable $A_t \equiv A(x_t)$ by the hierarchical process:

$$\begin{aligned}&s_t|s_{t-1}, \mathbf{T} \sim \text{Bernoulli}(T_{s_{t-1}1}, \ldots, T_{s_{t-1}N}),\\ &A_t|\mu_{s_t}, \sigma_{s_t}^2 \sim \text{Normal}\big(\mu_{s_t}, \sigma_{s_t}^2\big).\end{aligned} \qquad (5.29)$$

Here, the notation $\text{Bernoulli}(\pi_1, \ldots, \pi_N)$ denotes a Bernoulli scheme where discrete outcome n has associated probability π_n of being selected. We will demonstrate below how this model does in fact recapitulate the expected behavior in the limit where there are sufficient samples from each state.

We choose the (improper) Jeffreys prior [5],

$$p\big(\mu, \sigma^2\big) \propto \sigma^{-2} \qquad (5.30)$$

because it satisfies intuitively reasonable reparameterization [5] and information-theoretic [3] invariance principles. Note that this prior is uniform in $(\mu, \log\sigma)$.

The posterior is then given by

$$\begin{aligned}&p\big(\mu, \sigma^2 \,\big|\, \{A_m\}\big)\\ &\quad \propto \left[\prod_{n=1}^{N} p\big(A_m \,\big|\, \mu, \sigma^2\big)\right] p\big(\mu, \sigma^2\big)\\ &\quad \propto \sigma^{-(N+2)}\exp\left[-\frac{1}{2\sigma^2}\sum_{m=1}^{N}(A_m - \mu)^2\right].\end{aligned} \qquad (5.31)$$

Rewriting in terms of the sample statistics $\hat{\mu}$ and $\hat{\sigma}^2$, we obtain

$$\begin{aligned}&p\big(\mu, \sigma^2 \,\big|\, \{A_m\}\big)\\ &\quad \propto \sigma^{-(N+2)}\exp\left\{-\frac{1}{2\sigma^2}\left[\sum_{m=1}^{N}(A_m - \hat{\mu})^2\right.\right.\\ &\qquad \left.\left. + N(\hat{\mu} - \mu)^2\right]\right\}\end{aligned}$$

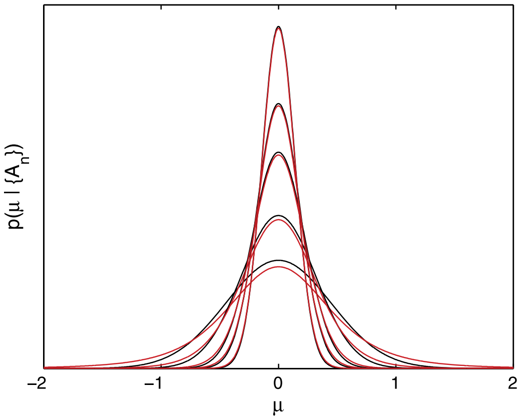


Fig. 5.3 Approach to normality for marginal distribution of the mean $p(\mu|\{A_m\})$. For fixed $\hat{\mu}$ and $\hat{\sigma}^2$, the marginal posterior distribution of μ (*red*), a scaled and shifted Student t-distribution, rapidly approaches the normal distribution (*black*) expected from asymptotic statistics. The PDF is shown for sample sizes of $N = 5$ (the broadest), 10, 20, and 30

$$\propto \sigma^{-(N+2)} \exp\left\{-\frac{1}{2\sigma^2}\big[(N-1)\hat{\sigma}^2 + N(\hat{\mu}-\mu)^2\big]\right\}. \tag{5.32}$$

The posterior has marginal distributions

$$\begin{aligned} \sigma^2|\{A_m\} &\sim \text{Inv--}\chi^2\big(N-1, \hat{\sigma}^2\big), \\ \mu|\{A_m\} &\sim \text{t}_{N-1}\big(\hat{\mu}, \hat{\sigma}^2/N\big) \end{aligned} \tag{5.33}$$

where σ^2 is distributed according to scaled inverse chi-square distribution with $N-1$ degrees of freedom, and μ according to Student's t-distribution with $N-1$ degrees of freedom that has been shifted to be centered about $\hat{\mu}$ and whose width has been scaled by $\hat{\sigma}^2/N$.

As can be seen in Fig. 5.3, as the number of degrees of freedom increases, the marginal posterior for μ approaches the normal distribution with the asymptotic behavior expected from standard frequentest analysis for the standard error of the mean, namely

$$\mu \to \text{N}\big(\hat{\mu}, \hat{\sigma}^2/N\big). \tag{5.34}$$

At low sample counts, the t-distribution is lower and wider than the normal distribution, meaning that confidence intervals computed from this distribution will be somewhat larger than those of the corresponding normal estimate for small samples. In some sense, this partly compensates for $\hat{\sigma}^2$ being a poor estimate of the true variance for small sample sizes, which would naturally lead to underestimates of the statistical uncertainty. In any case, this is also far from the asymptotic limit where the normal distribution with variance $\hat{\sigma}^2/N$ is expected to model the uncertainty well.

The posterior can also be decomposed as

$$\begin{aligned} &p\big(\mu, \sigma^2 \mid \{A_m\}\big) \\ &\quad = p\big(\mu \mid \sigma^2, \{A_m\}\big) p\big(\sigma^2 \mid \{A_m\}\big). \end{aligned} \tag{5.35}$$

This readily suggests a two-step sampling scheme for generating uncorrelated samples of (μ, σ^2), in which we first sample σ^2 from its marginal distribution, and then μ from its distribution conditional on σ^2

$$\begin{aligned} \sigma^2|\{A_m\} &\sim \text{Inv--}\chi^2\big(N-1, \hat{\sigma}^2\big), \\ \mu|\sigma^2, \{A_m\} &\sim \text{N}\big(\hat{\mu}, \sigma^2/N\big). \end{aligned} \tag{5.36}$$

Alternatively, if the scaled inverse-chi-square distribution is not available, the χ^2-distribution (among others) can be used to sample σ^2:

$$(N-1)\big(\hat{\sigma}^2/\sigma^2\big) \mid \{A_m\} \sim \chi^2(N-1) \tag{5.37}$$

where the first argument is the shape parameter and the second argument is the scale parameter.

5.6.2 Illustration of Fully Bayesian Sampling Scheme

Using the sampling procedures described previously, we are now equipped with a scheme to sample from the joint posterior describing our confidence in that a Markov model characterized by a transition matrix $\mathbf{T}$ and state expectations $\boldsymbol{\mu}_i$, $i = 1, \ldots, M$, produced the observed trajectory data. Using a set of models sampled from this posterior, we can characterize the statistical component of the uncertainty as it propagates into equilibrium averages, non-equilibrium relaxations, and (non-)equilibrium correlation measurements computed from the Markov model. To ensure the correctness of this procedure, however, we first test its ability to correctly characterize the posterior distribution for a finite-size sample from a true Markovian model system.

How can we test a Bayesian posterior distribution? One of the more powerful features of a Bayesian model is its ability to provide confidence intervals that correctly reflect the level of certainty that the true value will lie within it. For example, if the experiment were to be repeated many times, the true value of the parameter being estimated should fall within the confidence interval for a 95 % confidence level 95 % of the time. As an illustrative example, consider a biased coin where the probability of turning heads is θ. From an observed sample of N coin flips, we can estimate θ using a Binomial model for the number of coin flips that turn up heads and a conjugate Beta Jeffreys prior [3, 5]. Each time we run an experiment and generate a new independent collection of N samples, we get a different posterior estimate for θ, and a different confidence interval (Fig. 5.4, top). If we run many trials and record what fraction of the time the true (unknown) value of θ falls within the confidence interval estimated from that trial, we can see if our model is correct. If correct, the observed confidence level should match the desired confidence level (Fig. 5.4, bottom right). Deviation from parity means that the posterior is either two broad or too narrow, and that the statistical uncertainty is being either over- or underestimated (Fig. 5.4, bottom left).

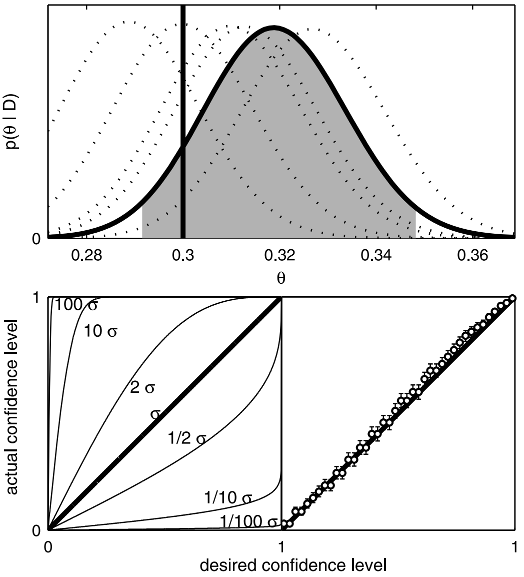


Fig. 5.4 Testing the posterior for inference of a biased coin flip experiment. *Top:* Posterior distribution for inferring the probability of heads, θ, for a biased coin from a sequence of $N = 1000$ coin flips (*dark line*) with 95 % symmetric confidence interval about the mean (*shaded area*). The true probability of heads is 0.3 (*vertical thick line*). Posteriors from five different experiments are shown as *dotted lines*. *Bottom left:* Desired and actual confidence levels for an idealized normal posterior distribution that either overestimates (*upper left curves*) or underestimates (*bottom right curves*) the true posterior variance by different degrees. *Bottom right:* Desired and actual confidence levels for the Binomial-Beta posterior for the coin flip problem depicted in *upper panel*. *Error bars* show 95 % confidence intervals estimates from 1000 independent experimental trials. For inference, we use a likelihood function such that the observed number of heads is $N_H|\theta \sim \text{Binomial}(N_H, N, \theta)$ and conjugate Jeffreys prior [3, 5] $\theta \sim \text{Beta}(1/2, 1/2)$ which produces posterior $\theta|N_H \sim \text{Beta}(N_H + 1/2, N_T + 1/2)$ along with constraint $N_H + N_T = N$

We performed a similar test on a three-state model system, using a model (reversible, row-stochastic) transition matrix for one Markov time is given by

$$\mathbf{T}(1) = \begin{bmatrix} 0.86207 & 0.12931 & 0.00862 \\ 0.15625 & 0.83333 & 0.01041 \\ 0.00199 & 0.00199 & 0.99602 \end{bmatrix}. \tag{5.38}$$

Each state is characterized by a mean value of the observable $A(x)$, fixed to 3, 2, and 1 for the first, second, and third states, respectively. The equilibrium populations are $\boldsymbol{\pi} \approx [0.16250.13450.7031]$. Simulation from this model involves a stochastic transition according to the transition element T_{ij} followed by observation of the value of $A(x)$ sampled i.i.d. from the current state's probability distribution $p_i(A)$. Multiple independent realizations of this process were carried out, and subjected to the Bayesian inference procedure for transition matrices and observables described above. The nonequilibrium relaxation $\langle A\rangle_{\rho_0}$ from the initial condition $\rho_0 = [100]$ in which all density is concentrated in state 1, as well as the autocorrelation function $\langle A(0)A(t)\rangle$, is shown in Fig. 5.5.

With the means of $p_i(A)$ within each state fixed as above, we considered models for $p_i(A)$ that were either *normal* or *exponential*, using the

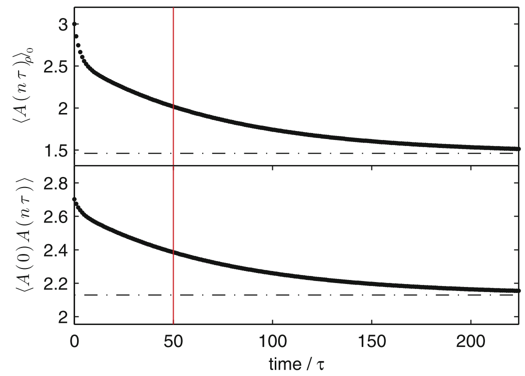


Fig. 5.5 Observables for three-state model system. *Top:* Relaxation of $\langle A(t)\rangle_{\rho_0}$ (*solid line*) from initial distribution $\rho_0 = [100]$ to equilibrium expectation $\langle A\rangle$ (*dash-dotted line*). *Bottom:* Equilibrium autocorrelation function $\langle A(0)A(t)\rangle$ (*solid line*) to $\langle A\rangle^2$ (*dash-dotted line*). The estimates of both $\langle A(t)\rangle_{\rho_0}$ and $\langle A(0)A(t)\rangle$ at 50 timesteps (*red vertical line*) were assessed in the validation tests described here

probability density functions:

$$p_i(A) = (2\pi)^{-1/2}\sigma_i^{-1}\exp\left[-\frac{1}{2\sigma_i^2}(A-\mu_i)^2\right],$$

normal

$$p_i(A) = \mu_i^{-1}\exp[-A/\mu_i],$$

$A \geq 0.$ exponential

While the normal output distribution for $p_i(A)$ corresponds to the hierarchical Bayesian model that forms the basis for our approach, the exponential distribution is significantly different, and represents a challenging test case.

Figure 5.6 depicts the resulting uncertainty estimates for both normal (top) and exponential (bottom) densities for the observable A. In both cases, the confidence intervals are *underestimated* for short trajectory lengths (1 000 steps) where, in many realizations, few samples are ob-

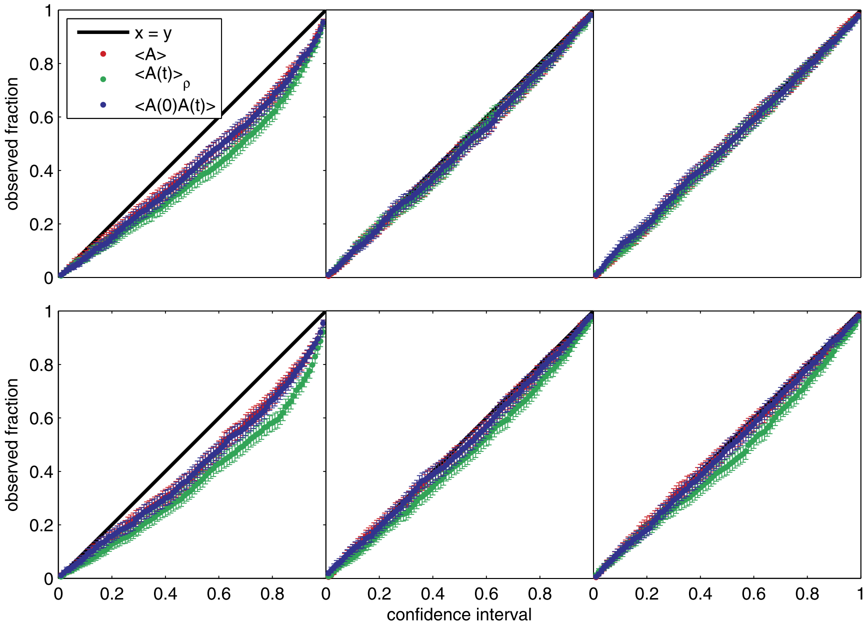


Fig. 5.6 Confidence interval tests for model system. *Top:* Expected and observed confidence intervals for three-state system with normal distribution for observable A with unit variance for simulations of length 1 000 (*left*), 10 000 (*middle*), and 100 000 (*right*) steps. Confidence intervals were estimated from generating 10 000 samples from the Bayesian posterior. Estimates of the fraction of observed times the true value was within the confidence interval estimated from the Bayesian posterior were computed from generating 1 000 independent experimental realizations. The resulting curves are shown for the equilibrium estimate $\langle A\rangle$ (*red*), nonequilibrium relaxation $\langle A\rangle_{\rho_0}$ (*green*), and the equilibrium correlation function $\langle A(0)A(t)\rangle$ (*blue*). *Bottom:* Same as top, except an exponential distribution with the same mean was used for the probability of observing a particular value of A within each state

served in one or more states, so that the variance is underestimated or the effective asymptotic limit has not yet been reached. As the simulation length is increased to 10 000 or 100 000 steps so that it is much more likely there are a sufficient number of samples in each state to reach the asymptotic limit, however, the confidence intervals predicted by the Bayesian posterior become quite good. For the exponential model for observing values of A (which might be the case in, say, fluorescence lifetimes), we observe similar behavior. Except for what appears to be a slight, consistent underestimation of $\langle A(t)\rangle_{\rho_0}$ (much less than half a standard deviation) there appears to be excellent agreement between the expected and observed confidence intervals, confirming that this method is expected to be a useful approach to modeling statistical uncertainties in equilibrium and kinetic observables.

References

1. Anderson TW, Goodman LA (1957) Statistical inference about Markov chains. Ann Math Stat 28:89–110
2. Chodera JD, Noé F (2010) Probability distributions of molecular observables computed from Markov models, II: uncertainties in observables and their time-evolution. J Chem Phys 133:105,102
3. Goyal P (2005) Prior probabilities: an information-theoretic approach. In: Knuth KH, Abbas AE, Morriss RD, Castle JP (eds) Bayesian inference and maximum entropy methods in science and engineering. American Institute of Physics, New York, pp 366–373
4. Hinrichs NS, Pande VS (2007) Calculation of the distribution of eigenvalues and eigenvectors in Markovian state models for molecular dynamics. J Chem Phys 126:244,101
5. Jeffreys H (1946) An invariant form for the prior probability in estimation problems. Proc R Soc A 186:453–461
6. Metzner P, Noé F, Schütte C (2009) Estimation of transition matrix distributions by Monte Carlo sampling. Phys Rev E 80:021,106
7. Noé F (2008) Probability distributions of molecular observables computed from Markov models. J Chem Phys 128:244,103
8. Noé F, Oswald M, Reinelt G (2007) Optimizing in graphs with expensive computation of edge weights. In: Kalcsics J, Nickel S (eds) Operations research proceedings. Springer, Berlin, pp 435–440
9. Noé F, Oswald M, Reinelt G, Fischer S, Smith JC (2006) Computing best transition pathways in high-dimensional dynamical systems: application to the alphaL–beta–alphaR transitions in octaalanine. Multiscale Model Simul 5:393–419
10. Noé F, Schütte C, Vanden-Eijnden E, Reich L, Weikl TR (2009) Constructing the full ensemble of folding pathways from short off-equilibrium simulations. Proc Natl Acad Sci USA 106:19,011–19,016
11. Prinz JH et al (2011) Markov models of molecular kinetics: generation and validation. J Chem Phys 134:174,105
12. Prinz JH, Held M, Smith JC, Noé F (2011) Efficient computation of committor probabilities and transition state ensembles. Multiscale Model Simul 9:545
13. Singhal N, Pande VS (2005) Error analysis and efficient sampling in Markovian state models for molecular dynamics. J Chem Phys 123:204,909

6 Analysis of Markov Models

Frank Noé and Jan-Hendrik Prinz

This chapter follows the following reference which should be used as a reference: Prinz, J.-H. and Keller, B. and Noé, F. (2011) Probing molecular kinetics with Markov models: Metastable states, transition pathways and spectroscopic observables. Phys. Chem. Chem. Phys., 13, pp. 16912–16927.

6.1 Illustrative Protein Folding Model

We use a simple protein folding model throughout this study in order to illustrate the concepts described in this chapter. We consider three structural elements called a, b and c that form independently of another. A simple energy model has been designed in which the folding of each structure element contributes a loss in potential energy and also a loss of entropy (Table 6.1).

The entropic part is chosen that the formation of a structural element decreases the accessible conformational space by a factor $a \to 2$, $b \to 3$ and $c \to 5$ favoring the unfolded state for high temperatures. A small number (0.5) is subtracted from the conformation space volumes in order to break the perfect independence of structure elements. In addition, for each formed structural element the potential energy is lowered so as to favor the folding at low temperatures.

Table 6.1 Energy model of the simple protein folding model. Shown is the potential energy ΔU and the entropy ΔS depending of the folding state. The potential energy drops with the number of structural elements formed, while the entropic part mimics a reduction of conformational space when one of the elements forms (by a factor of $a \to 2$, $b \to 3$ and $c \to 5$)

	U	S
unfolded	0	$10.3804 = \log(60 + 120 - 0.5)$
a	-1.5	$6.76878 = \log(30 - 0.5)$
b	-1.5	$5.94083 = \log(20 - 0.5)$
c	-1.5	$4.88469 = \log(12 - 0.5)$
a/b	-3.75	$4.50258 = \log(10 - 0.5)$
a/c	-3.75	$3.4095 = \log(6 - 0.5)$
b/c	-3.75	$2.50553 = \log(4 - 0.5)$
$a/b/c$	-4.5	$0.81093 = \log(2 - 0.5)$

Thus, at any given temperature T, the free energy $F_i = U_i - TS_i$ for each of the eight possible foldamers $\{0, a, b, c, ab, ac, bc, abc\}$ can be calculated and also the associated stationary distribution

$$\pi_i = \frac{\exp(-F_i/k_B T)}{\sum_j \exp(-F_j/k_B T)}.$$

Assuming furthermore that the model protein can jump between states by forming or breaking one structure element with transition probabilities

$$T_{ij} = \exp\left(-\frac{\Delta + \max(0, F_j - F_i)}{k_B T}\right)$$

with minimum barrier height $\Delta = 4$, we have a consistent dynamical model that can be used for

F. Noé (✉) · J.-H. Prinz
Freie Universität Berlin, Arnimallee 6, 14195 Berlin, Germany
e-mail: frank.noe@fu-berlin.de

G.R. Bowman et al. (eds.), *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology 797,
DOI 10.1007/978-94-007-7606-7_6, © Springer Science+Business Media Dordrecht 2014

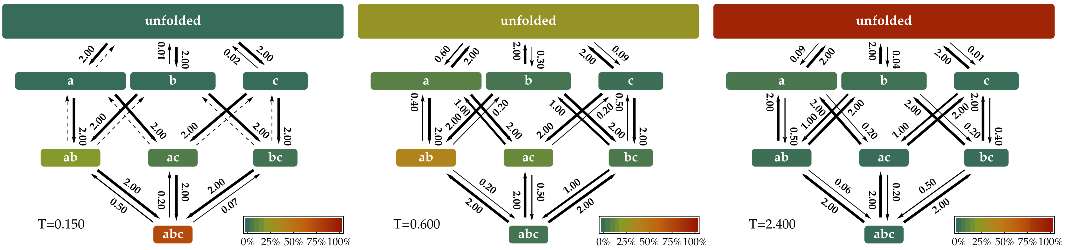


Fig. 6.1 Illustrative protein folding model for low, intermediate and high temperature. The *colours* indicate the stationary probability of states, while the *thickness of the arrows* and the *numbers* next to them quantify transition probabilities (within some fixed but arbitrary timescale)

analysis. Figure 6.1 illustrates this model at low, intermediate and high temperatures, showing that the folded state is stable at low temperatures and the unfolded state is stable at high temperatures.

6.2 Spectral Analysis: Eigenvectors and Eigenvalues

A key concept of Markov modeling is that a lot of the essential information about conformation dynamics is encoded in the eigenvectors and eigenvalues of the transition matrix, which are approximations to the exact eigenfunctions and eigenvalues of the transfer operator (see Theory chapter) that a Markov model attempts to approximate. Although this point is somewhat difficult to understand at first, it is essential in order to see what metastable states are, why some Markov models work better than others, and eventually also how kinetics experiments work. At this point a comparison to another approach that is more commonly used in the Chemical Physics community may be useful: Consider the Principal Component Analysis method [1], where the relative distances of a set of data points (e.g. molecular structures) is captured by a covariance matrix. When performing an eigenvalue decomposition one obtains eigenvectors and eigenvalues. The eigenvectors with the largest eigenvalues are called "principal components" and describe where the directions along which the data set has the greatest spatial extent. The corresponding eigenvalues capture the variance of the data set along these principal directions. Analogously, a transition matrix $\mathbf{T}(\tau)$ can also be decomposed into eigenvectors and eigenvalues. The eigenvectors also represent "principal modes", but since the transition matrix contains probabilities these modes are vectors that contain changes of the probability for each discrete state S_i. The principal modes with the largest eigenvalues are indeed the main modes of probability flow between the system's substates. The corresponding eigenvalues have magnitude expressing how slow or fast the corresponding probability flow occurs. Thus, the eigenvalue decomposition of a transition matrix may be understood as a principal component analysis of the dynamics.

More formally, transition matrices can, as any diagonalizable matrix, be written as a linear combination of their left eigenvectors, their eigenvalues and their right eigenvectors. For the here assumed case of matrices fulfilling detailed balance, the right eigenvalues can be replaced by the left eigenvalues (and vice versa), leading to the decomposition:

$$\mathbf{T}(\tau) = \boldsymbol{\Pi}^{-1} \sum_{i=1}^{n} \lambda_i(\tau) \boldsymbol{\phi}^{(i)} \left(\boldsymbol{\phi}^{(i)}\right)^T \tag{6.1}$$

with the diagonal matrix $\boldsymbol{\Pi}^{-1} = \operatorname{diag}(\pi_1^{-1}, \dots, \pi_n^{-1})$. Thus, for longer timescales:

$$\mathbf{T}^k(\tau) = \boldsymbol{\Pi}^{-1} \sum_{i=1}^{n} \lambda_i^k(\tau) \boldsymbol{\phi}^{(i)} \left(\boldsymbol{\phi}^{(i)}\right)^T. \tag{6.2}$$

The transition matrix $\mathbf{T}(k\tau) = \mathbf{T}^k(\tau)$ which transports an initial probability k time steps forward is again a linear combination of the eigenvectors and eigenvalues. These linear combinations

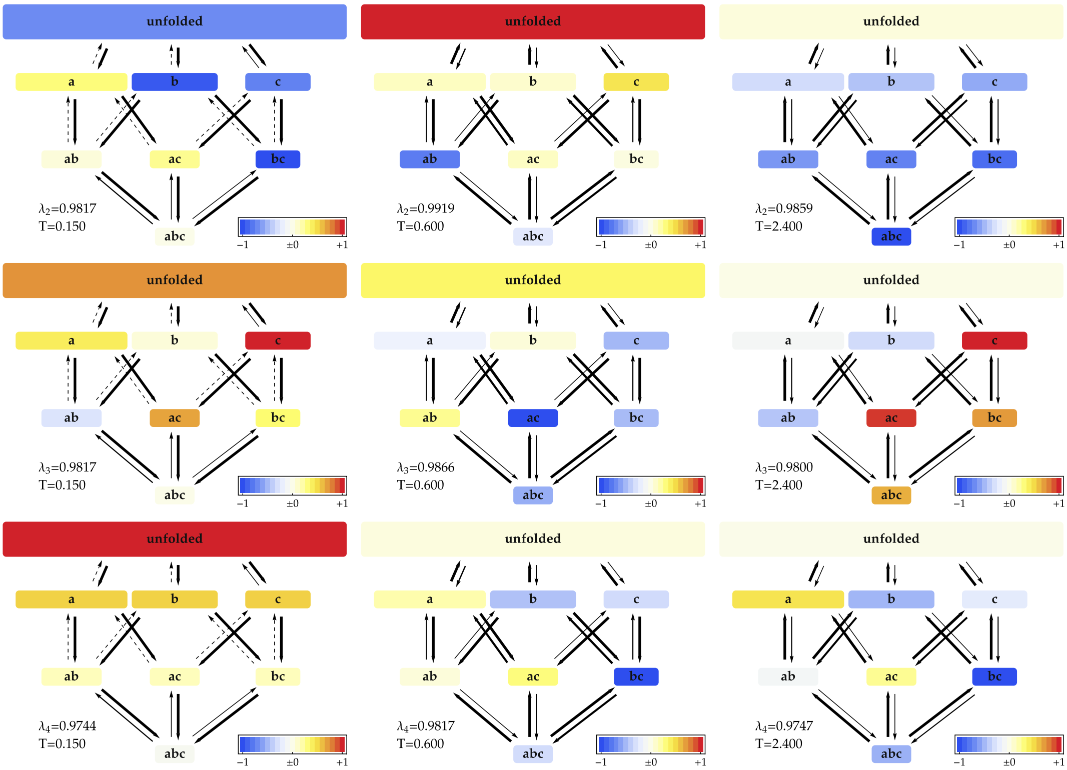


Fig. 6.2 Dominant eigenvectors and eigenvalues of the protein folding model

(Eqs. (6.1) and (6.2)) are known as *spectral decomposition* of the transition matrix. They are very useful for connecting the dynamics of the molecule to experimentally-measured signals, which is described in Sect. 6.5.

Equation (6.2) is the key for understanding how the transition matrix transforms a probability vector. The complete process consists of n subprocesses $\boldsymbol{\phi}^{(i)}(\boldsymbol{\phi}^{(i)})^T$, each of which is weighted by the eigenvalue λ_i raised to the power k. Because the transition matrix is a row-stochastic matrix, it always has one eigenvalue which is equal to one $\lambda_1 = 1$ [10]. Raising this eigenvalue to the power k does not change the weight of the corresponding subprocess $\boldsymbol{\phi}^{(1)}(\boldsymbol{\phi}^{(1)})^T$ which is the stationary process, $\boldsymbol{\phi}^{(1)} = \pi$. All other eigenvalues of the transition matrix are guaranteed to be smaller than one in absolute value [10]. Eigenvectors are approximations of the transfer operator eigenfunctions. To understand the meaning of the eigenfunctions, please refer to the illustration in the theory chapter (see Fig. 3.1).

The weights of the processes hence decay exponentially with the implied timescale t_i of the decay process

$$t_i = -\frac{\tau}{\ln \lambda_i}. \tag{6.3}$$

Since the relaxation timescales t_i are physical properties of the dynamics, they should be invariant under change of the lag time τ used to parametrize the transition matrix [42]. For large enough τ, t_i should converge to their true value (assuming sufficient statistics). Therefore, the convergence of t_i with increasing τ has often been employed as an indicator for selecting τ [9, 32, 34, 42] (see Section on Markov model validation). The smaller the eigenvalue λ_i, the smaller the implied timescale t_i, the faster the corresponding process decays.

Figure 6.2 shows the eigenvectors in the protein folding model. For the low-temperature situation, the folding process is interestingly not the slowest, but the third-slowest process, which exchanges probability between unfolded-a-b-c and states ab-ac-bc-abc. The slowest process corresponds to the formation of a, while the second-slowest process is a more complex transition involving the exchange of unfolded, c and ac with the rest.

The intermediate-temperature situation, the slowest process is the one that most closely resembles folding—it mostly exchanges probability between unfolded—c and ab–abc. The second- and third-slowest processes correspond to the formation of c and b, respectively.

In the high-temperature situation, the slowest process is a folding process which exchanges probability between unfolded and the rest. It is therefore a different kind of folding process than the third-slowest process in the low-temperature case. One might say that the transition state has shifted towards the unfolded side. The second- and third-slowest processes again correspond to the formation of c and b, respectively.

6.3 Metastable States

The protein folding model used here for illustration consists of only 8 states and is thus easy to comprehend. When building Markov models from clustered molecular dynamics data one often requires several thousands of states in order to approximate the system kinetics well. Network approaches have been developed to visualize the network of transitions arising from such a model [35], but especially when the network is dense, this is not straightforward. It is thus desirable to find en effective representation that communicates the essential properties of the kinetics. In this section we describe a way to cluster the large discrete state space into a few metastable sets that have the property that they capture the dynamics for long times before jumping to another set. Let us stress that the purpose of finding these sets is purely illustrative (e.g. for lumping fluxes, see Sect. 6.4). For quantitatively calculating kinetic properties, the full Markov model should be used, as the approximation of the system's kinetics will generally deteriorate when using a lumped Markov model [24, 34, 36].

Let us consider the coarse partition of state space $\Omega = \{C_1, C_2, \dots, C_n\}$ where each cluster C_i consists of a set of states S_j. We are interested in finding a clustering that is maximally metastable. In other words, each cluster C_i should represent a set of structures that the dynamics remains in for a long time before jumping to another cluster C_j. Thus, each cluster C_i can be associated with a free energy basin.

As shown above (Sect. 6.2), we can understand the slow kinetics in terms of probability transport by the dominant eigenvectors of the transition matrix. Consequently, these dominant eigenvectors can also be used in order to decompose the system into metastable sets [37, 47]. Consider the eigenvector corresponding to the slowest process in Fig. 3.1 (yellow line): This eigenvector is almost a step function which changes from negative to positive values at the saddle point. When we take the value of this eigenvector in each state and plot it along one axis, we obtain Fig. 6.3a. Partitioning this line in the middle dissects state space into the two most metastable states of the system (Fig. 6.3b). The two most metastable states exchange at a timescale given by the slowest timescale t_2. If we are interested in differentiating between smaller substates, we may ask for the partition into the three most metastable states. In this case we consider two eigenvectors simultaneously, $\mathbf{r}_2$ and $\mathbf{r}_3$. Plotting the coordinates in these eigenvalues for each state yields the triangle shown in Fig. 6.3c whose corners represent the kinetic centers of metastable states. Assigning each state to the nearest corner partitions state space into the three most metastable states (Fig. 6.3d) that exchange at timescales of t_3 or slower. The same partition can be done using three eigenvectors, $\mathbf{r}_2$, $\mathbf{r}_3$ and $\mathbf{r}_4$, yielding four metastable states exchanging at timescales t_4 and slower, and so on (Figs. 6.3e, f). Generally, it can be shown that when n eigenvectors are considered, their coordinates lie in an n-dimensional simplex with $n + 1$ corners called *vertices* which allow the dynamics to be partitioned into $n + 1$ metastable sets [32, 47].

Each of these partitionings is a valid selection in a hierarchy of possible decompositions of the system dynamics. Moving down this hierarchy

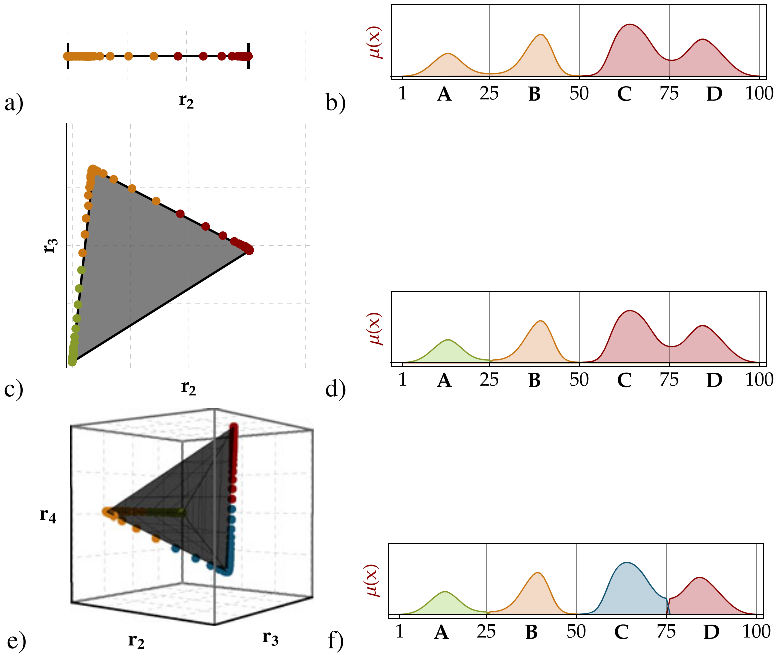


Fig. 6.3 Metastable states of the one-dimensional dynamics (see Fig. 3.1) identified by PCCA+. (**a**), (**c**), (**e**): Plot of the eigenvector elements of one, two, and three eigenvectors. The colors indicate groups of elements (and thus conformational states) that are clustered together. (**b**), (**d**), (**f**): Clustering of conformation space into two, three, and four clusters, respectively

means that more states are being distinguished, revealing more structural details and smaller timescales. For the system shown in Fig. 3.1, two to four states are especially interesting to distinguish. After four states there is a gap in the timescales ($t_5 \ll t_4$) induced by a gap after the fourth eigenvalue Fig. 3.1c). Thus, for a qualitative understanding of the system kinetics, it is not very interesting to distinguish more than four states. However, note that for quantitatively modeling the system kinetics, it is essential to maintain a fine discretization as the MSM discretization error will increase when states are lumped.

Figure 6.4 shows the metastable states of the protein folding model. Interestingly, there is no simple partition that splits unfolded and folded states. In the intermediate temperature case this is most closely the case as the unfolded state is a metastable state and separated from all other states with partial structure. The remaining space and the conformation space at other temperatures is clustered in a non-obvious manner. Sometimes these clusters are defined by the presence of particular structural elements (e.g. red cluster in the high-temperature case is characterized by having c formed.

6.4 Transition Pathways

Understanding the folding mechanism of macromolecules, and proteins in particular, is one of the grand challenges in biophysics. The field was driven by questions such as [11]: How does an ensemble of denatured molecules find the same native structure, starting from different conformations? Is folding hierarchical [3, 4]? Which forms first: secondary or tertiary structure [16, 49]? Does the protein collapse to compact structures before structure formation, or concurrently [2, 19, 39]? Are there folding nuclei [21]? Is there a particular sequence in which secondary structure elements are formed?

Heterogeneity in folding pathways has been found in a number of experimental studies. For example, using time-resolved FRET with four different intramolecular distances, it was found in Barstar [40] that there are multiple folding routes, and that different routes dominate under different folding conditions. Moreover, changing the denaturant can change the dominant pathway [25]. Extensive mutational analysis of the seven ankyrin sequence repeats of the Notch ankyrin repeat domain has revealed its funnel landscape [7, 27, 41]. Some folding is sequential, as in

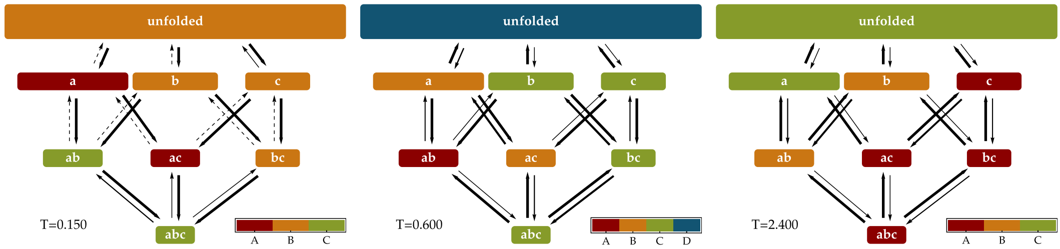


Fig. 6.4 Metastable sets of the folding model

FynSH3 [23], cytochrome [14], T4 lysozyme [8], and Im7 [15], and some folding is parallel, as in cytochrome C [17] and HEW lysozyme [26].

Formally, the question about folding pathways boils down to the following: Let A and B be two subsets of state space, defined so as to specify the transition process one wants to investigate. For example, A may correspond to the strongly denatured set of sets while B is the metastable set around the crystal structure when known [33]. All remaining states are unassigned "intermediate" states I. What is the probability distribution of the trajectories leaving A and continuing on to B? I.e., what is the typical sequence of I states used along the transition pathways?

When an MSM is already available, the information of transition pathways is easily accessible via Transition Path Theory (TPT). The concepts of TPT and the TPT equations for continuous Markov processes were introduced in [45]. See also [46] for a review. TPT was extended to discrete-space Markov jump processes (i.e. for Master equation dynamics) in [28] and for Markov chains in [33]. Transition path theory is related to Transition Path Sampling (TPS) in the sense that both are trying to generate statistical information about the ensemble of $A \rightarrow B$ pathways. TPS is a direct approach to sampling pathways directly [6] and could in principle be used to sample folding pathways. However, in TPS the sampled trajectories are in practice of limited length and it is thus unpractical to use TPS when the intermediate states I contain metastabilities. One can run multiple TPS-samplings between pairs of metastable states after having identified them [44]. More information on the relation of TPT and TPS can be found in [46].

6.4.1 Discrete Transition Path Theory

We give a brief introduction to TPT for Markov chains as described in [33], while Sect. 6.5 gives a more thorough theoretical description. The essential ingredient required to compute the statistics of transition pathways is the committor probability q_i^+. q_i^+ is the probability when being at state i, the system will reach the set B next rather than A [6, 13, 43]. In protein folding contexts, it is the probability of folding [13]. By definition, all states in A have $q_i^+ = 0$ while all states in B have $q_i^+ = 1$. For all intermediate states, the committor gradually increases from A to B (see Fig. 6.5), and its value can be calculated by solving the following system of equations:

$$-q_i^+ + \sum_{k \in I} T_{ik} q_k^+ = -\sum_{k \in B} T_{ik} \quad \text{for } i \in I$$

(see SI appendix of [33] for derivation). Figure 6.5 shows the committor (color-coding) for the protein folding model: At low temperatures, the committor changes rapidly after leaving the unfolded state and forming the first structure elements. At high temperatures, it changes rapidly when entering the full-structured native state. At both temperatures, the folding process has thus essentially two-state character, although with different definitions of the two states. At intermediate temperatures, the commitor increases gradually from the unfolded to the native state, indicating that it is important to consider the intermediate states in the folding process.

We further need the backward-committor probability, q_i^-. q_i^- the probability, when being at state i, that the system was in set A previously

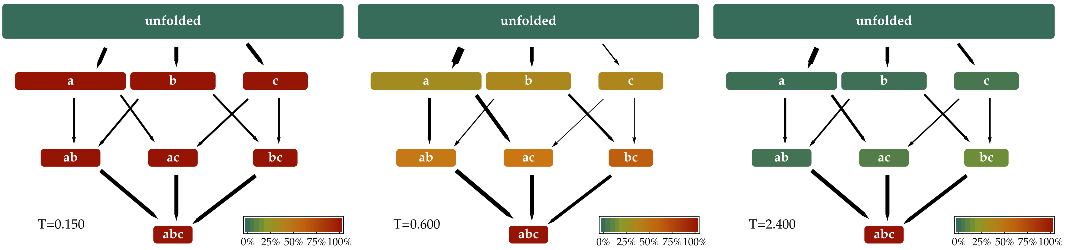


Fig. 6.5 Committor and net flux from unfolded to folded state

rather than in B. For dynamics obeying detailed balance (which is assumed here) this is simply

$$q^- = 1 - q^+.$$

Consider the probability flux between two states i and j, given by $\pi_i T_{ij}$ (absolute probability of finding the system at this transition). We are only interested in trajectories that successfully move from A to B without recurring to A beforehand. The flux pertaining to these *reactive* trajectories only is given by multiplying the flux by the probability to come from A and to move on to B:

$$f_{ij} = \pi_i q_i^- T_{ij} q_j^+.$$

This flux is the quantity that could be obtained directly from a converged TPS sampling by counting transitions of the reactive path ensemble. However, we further want to remove contributions that come from recrossings or detours. For example, a trajectory that would jump on its way from A to B multiple times between two substates i and j would produce an increase in the flux $i \to j$ and the backward flux $j \to i$. However, we only want to consider a single transition per pathway and thus define the net flux, given by:

$$f_{ij}^+ = \max\{0, f_{ij} - f_{ji}\}.$$

Considering detailed balance dynamics and when ordering states along the reaction coordinate q_i^+ such that $q_i^+ \leq q_j^+$, an equivalent expression is [5, 28]:

$$f_{ij}^+ = \pi_i T_{ij} \left(q_j^+ - q_i^+\right).$$

f_{ij}^+ defines the net flux and is a network of fluxes leaving states A and entering states B (see Fig. 6.5). This network is flux-conserving, i.e. for every intermediate state i, the input flux equals the output flux (see [28, 33] for proof). The only set in the network that produces flux is A and the only set that consumes flux is B. Due to flux conservation, these amounts of flux are identical and are called total flux F of the transition $A \to B$:

$$F = \sum_{i \in A} \sum_{j \notin A} \pi_i T_{ij} q_j^+ = \sum_{i \notin B} \sum_{j \in B} \pi_i T_{ij} \left(1 - q_i^+\right).$$

The value of F gives the expected number of observed $A \to B$ transitions per time unit τ that an infinitely long trajectory would produce. Of special interest is the reaction rate constant k_{AB} (see [33] for derivation):

$$k_{AB} = F \bigg/ \left(\tau \sum_{i=1}^{m} \pi_i q_i^- \right). \tag{6.4}$$

Note that all states that trap the trajectory for some time will reduce k_{AB}. The effect of these traps is properly accounted for in the folding flux, even if they do not contribute to productive pathways.

6.4.2 Transition Paths Between Macrostates

Since the number of n conformational states used to construct a Markov model is often very large, it is convenient for illustration purposes to compute the net flux of $A \to B$ trajectories amongst only a few coarse sets of conformations. We

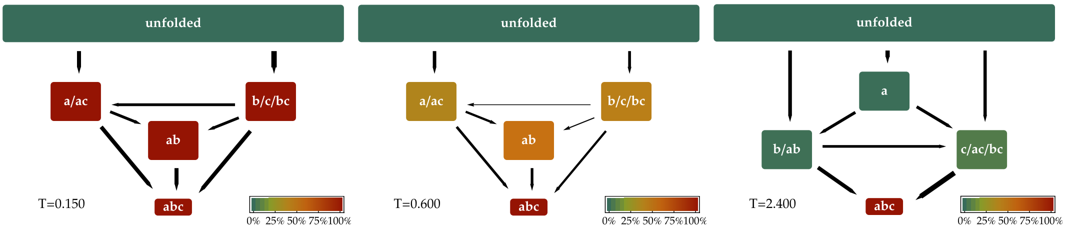


Fig. 6.6 Coarse-grained folding fluxes

consider a coarse partition of state space $S = \{C_1, C_2, \dots, C_n\}$, which may be based on a decomposition into metastable states as described in Sect. 6.3, or another partition that the user defines e.g. based on order parameters of interest. We make the restriction, however, that this decomposition preserves the boundaries of sets A, B and I, i.e. A and B are either identical to individual C_i, or to a collection of multiple C_i.

The coarse-grained flux between two sets is then given by:

$$F_{ij} = \sum_{k \in C_i, l \in C_j} f_{kl}.$$

and the net flux by

$$F_{ij}^{+} = \max\{0, F_{ij} - F_{ji}\}.$$

We note a technicality here: the second step of again removing backfluxes to obtain a coarse-grained net flux is necessary only if the clusters used do not partition state space along the isocommitor surfaces. Thus it may be desirably to use a partition that only groups states with similar committor values.

Figure 6.6 shows the coarse-grained fluxed from the unfolded to the folded states where the coarse-graining has been done according to metastable states. At low and intermediate temperatures, the topology of the folding network is equal, but the flux becomes smaller and the *ab* intermediate is used less. At higher temperatures, the topology of the folding network changes due to a change in the boundaries of metastable states and the unfolded state first splits into three intermediate states before converging to *abc*.

Coarse-graining generates a simplified view but correct on the folding flux. The actual dynamics, represented by the Markov model $\mathbf{T}(\tau)$ cannot easily be coarse-grained without loosing information, and no statement is made here about the transition probability between two coarse sets C_i and C_j.

6.4.3 Pathway Decomposition

The flux network can be decomposed into pathways from $A \to B$. When the dynamics are reversible, then the flux can be completely decomposed into such $A \to B$ pathways and no cycles will remain. Consider a pathway consisting of k nodes

$$P = (i_1 \in A \to i_2 \to \cdots \to i_{k-1} \to i_k \in B)$$

Along each of its edges, say $i_l \to i_{l+1}$, the flux network can carry a flux of up to $f_{i_l i_{l+1}}^{+}$. Thus, the capacity or flux of the pathway is given by the minimum of these fluxes:

$$f(P) = \min\left\{ f_{i_l i_{l+1}}^{+} \;\middle|\; l = 1 \dots k \right\}$$

A pathway decomposition consists of choosing a pathway P_1, and then removing its flux $f(P_1)$ from the flux along all the edges of P_1. This may be repeated until the total flux F has been subtracted and the network is thus free of $A \to B$ pathways. Note that while the flux network is unique, such a decomposition is not unique, because one may choose different strategies to select pathways. Nevertheless pathway decompositions are useful in at least the following aspects:

1. The strongest pathway, i.e. the pathway whose minimum flux $f(P)$ is largest of all pathways, is of special interest. Especially so, if $f(P)$ is not much smaller than the total flux F.
2. One reasonable way to perform a pathways decomposition is to first remove the strongest pathway, then remove the strongest pathway of the remaining network, and so on [29]. This decomposition is useful to estimate how many $A \to B$ are necessary to obtain a certain percentage of the flux [33].
3. Any pathway decomposition, even a decomposition in which pathways are chosen randomly, gives the same answer when calculating the probability of certain events. Let us consider the probability that, in the protein folding model, one of the three structural elements, a, b, and c, is formed before the other ones in the intermediate-temperature case. The network can, e.g. be decomposed into the pathways with corresponding fluxes:

$$\begin{aligned}
&\text{unfolded} \to a \to ab \to abc \;\; 0.000241655,\\
&\text{unfolded} \to a \to ac \to abc \;\; 0.000276008,\\
&\text{unfolded} \to b \to ab \to abc \;\; 0.0000782191,\\
&\text{unfolded} \to b \to bc \to abc \;\; 0.000175341,\\
&\text{unfolded} \to c \to ac \to abc \;\; 0.0000306848,\\
&\text{unfolded} \to c \to bc \to abc \;\; 0.0000592429
\end{aligned}$$

and the probability of forming a, b or c first is given by the flux fraction of pathways where this occurs:

$$\begin{aligned}
\mathbb{P}(a \text{ first}) &= \frac{1}{F}\sum_i f(P_i)\chi_i(a \text{ first})\\
&= 60.11\ \%,\\
\mathbb{P}(b \text{ first}) &= \frac{1}{F}\sum_i f(P_i)\chi_i(b \text{ first})\\
&= 29.44\ \%,\\
\mathbb{P}(c \text{ first}) &= \frac{1}{F}\sum_i f(P_i)\chi_i(c \text{ first})\\
&= 10.44\ \%
\end{aligned}$$

where χ_i is 1 if $a/b/c$ forms first in pathway P_i, respectively, and 0 otherwise.

The pathway decomposition is usually done on the original flux network. It can also be done on a coarse-grained flux network, provided that the coarse-graining does not lump states which need to be distinguished in order to calculate the probabilities of the events investigated.

6.4.4 PinWW Example

In order to illustrate the utility of our approach for studying folding mechanisms, the folding dynamics of the PinWW domain [20] is studied here. The text and figures from this section have been published in [33].

180 molecular dynamics simulations were started, 100 from near-native, 80 from different denatured conformations, and run for 115 ns each at a temperature of 360 K. The simulations were conducted with the GROMACS program [38] using explicit SPC solvent, the GROMOS96 force field [18] and the reaction field method for computing nonbonded forces. The simulation setup is described in detail in the Supplementary Information. The simulated structures were aligned onto the native structure and then clustered finely into 1734 kinetically connected and well-populated clusters. A transition matrix $\mathbf{T}(\tau)$ was constructed by counting transitions between these clusters at a lagtime of $\tau = 2$ ns. It was verified that $\mathbf{T}(\tau)$ is a good model for the long-time kinetics by conducting a Chapman-Kolmogorov test (see Supplementary Information of [33]). All properties computed from the Markov model are associated with statistical uncertainty resulting from that fact than only a finite amount of simulation data has been used to construct the model. These uncertainties are computed using a Bayesian inference method described in [30], the details are given in the Supplementary Information of [33]. The slowest timescale, corresponding to the second-largest eigenvalue of the Markov model, was 26 µs (confidence intervals 8–78 µs), compared to 13.2 µs measured in a temperature-jump experiment [20].

In order to study the folding mechanism, a folded set, B, was defined to be the set of clusters with average backbone root mean square difference to the X-ray structure of less than 0.3 nm.

The denatured set, A, was defined to be the set of all clusters with little β-structure (having a mean of 3 h-bonds in hairpin 1 which has 6 h-bonds in the native state and 1 h-bonds in hairpin 2 which has 3 h-bonds in the native state). Based on these definitions and the transition matrix $\mathbf{T}(\tau)$ between the 1734 clusters, the committor probabilities and the folding flux were computed as described in the Theory section.

In order to obtain a view of the sequence of events that is unbiased by defining reaction-coordinates, the folding pathways must be considered individually. Therefore, the folding flux was decomposed into individual pathways (see Theory section) and for each of them the times when hairpin 1 or 2 forms and remains stable were computed. "Formation" was defined as having 80 % of the average number of hydrogen bonds that are present in the native state, but variations of this threshold did not change the results qualitatively. The probability that hairpin 1 forms before hairpin 2 was computed by calculating the fraction of individual path fluxes in which this occurred (see previous section): In 30 % of the folding trajectories, hairpin 1 forms before hairpin 2 (confidence interval 18–34 %), and in 70 % the other way around. Thus, there is no unique mechanism in terms of the order of secondary structure formation, which is in qualitative agreement with a structural interpretation of mutational Φ-values for the pin WW domain [48].

In order to visualize the "essential folding pathways", coarse conformational sets were defined onto which the folding flux was projected (see Theory section). We employed a definition of 50 sets that separate the most slowly converting ("metastable") parts of state space. The number of sets can be chosen by fixing a timescale of interest (here 100 ns), then the number of metastable sets are given by the number of implied timescales of the transition matrix slower than that timescale of interest. The definition of metastable states was obtained by PCCA+. Figure 6.7 shows the network of the 70 % most relevant pathways, which involves only 21 of these 50 conformational sets. The remaining 30 % of the flux is mainly in small pathways between the structures shown in Fig. 6.7 and is omitted here for clarity of the visualization. The 29 of the 50 conformational sets not shown in the figure are only weakly involved in the $A \to B$ flux.

The denatured set (A) consists of mostly globular structures. No completely stretched structures are observed in the simulation. The figure suggests a relatively large number of kinetically separated unfolded states. Note that this does not necessarily mean that there are large energy barriers between them, but only that the energy barriers between them are not smaller than the ones that are overcome when proceeding towards the intermediate states.

The coarse-grained folding flux suggests that there is a large number of unfolded states and early intermediates that narrow down when coming closer to the native state. The picture reemphasizes the existence of many structurally different parallel pathways. Pathways where hairpin 1 forms first are shown on the right, pathways where hairpin 2 forms first on the left. It is apparent that the pathways in which hairpin 1 forms first also include some partially helical structures formed by the sequence that will later become the third β strand.

Figure 6.7 also indicates whether a set of structures with hairpins formed has the same register pattern as in the native state (0) or is register-shifted by one or two residues $(1, 2)$. Most of the productive folding pathways proceed from no hairpins over on-register intermediates to the native state. Some of the folding-efficient structures have the smaller hairpin 2 register-shifted, but none of them have hairpin 1 register-shifted. A special case is a structure which has both chain ends curled in such a way that they are on-register near the termini, but register-shifted by 2 residues in between (indicated by "0–2").

For the 50 coarse states defined here, the coarse flux network was decomposed into individual pathways according to decreasing flux as described in the Theory section. Figure 6.8 top shows the cumulative flux depending on the number of pathways, showing that about 3–5 pathways are needed to carry 50 % of the total flux and about 11–20 pathways are needed to carry 90 % of the total flux. Although the absolute number of parallel pathways depends on the number

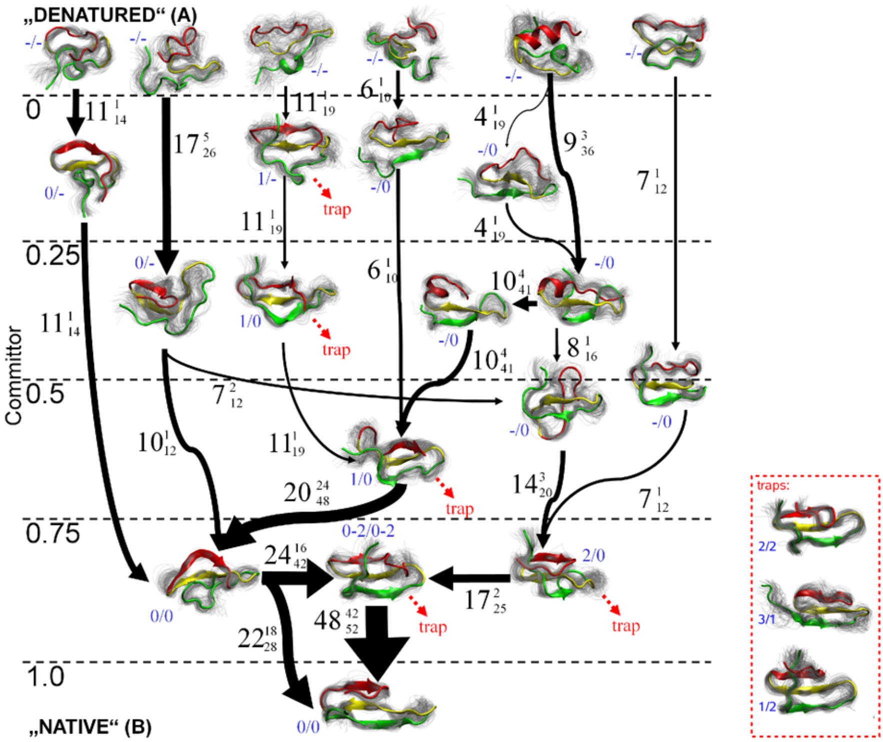


Fig. 6.7 *Left:* The network of the 70 % most relevant folding pathways for PinWW. The *numbers on the left* indicate the committor probabilities, the *thickness of the arrow* indicates the flux of folding trajectories between each pair of conformations. For each conformation, a representative mean structure is shown in color along with an overlay of equilibrium distributed structures from that conformation indicating the structural flexibility (*gray cloud*). The numbers next to the arrows give the normalized net flux (*large number*) and the 80 % confidence interval limits (*small numbers*) in percent. The *blue numbers* next to the structures indicate if the first/second hairpin has the native register (0), is register-shifted by one or two residues (1, 2) or not formed at all (−). *Right:* register-shifted trap states that do not carry significant folding flux but reduces the folding speed by nearly a factor of 2. Figure reprinted from [33]

of states one defines, i.e. on the amount of coarse-graining, the structural differences between the 50 sets defined here implies a remarkable degree of parallelism of the folding mechanism in the present system.

The six pathways which carry most of the total flux are depicted in Fig. 6.8 bottom, highlighting that there are routes where hairpin 1 forms first (paths 3, 4, 6), where hairpin 2 forms first (paths 1, 2), and where there is a more or less concurrent formation of both (path 5). Note that the percentages of individual pathways given in Fig. 6.8 should not be misinterpreted as the absolute probability of finding this exact sequence of conformations, as these pathways do, e.g., not consider the possibility of unproductive recrossing events or changing between different paths. However, they do provide the relative probabilities of choosing each one folding pathway from the ensemble of productive folding pathways. For example, pathway 1 is nearly twice as probable as pathway 6.

Interestingly, there are three metastable sets that contribute almost no folding flux (<5 %), but the system still spends a significant fraction of the time in them (stationary probability 18 % with confidence intervals 3–45 %). These “trap” states, depicted in Fig. 6.7, have almost full β content, but the hairpins are register shifted with respect to the native structure, in particular at hairpin 1 which is not fully shifted in any of the intermediates that significantly contribute to the folding

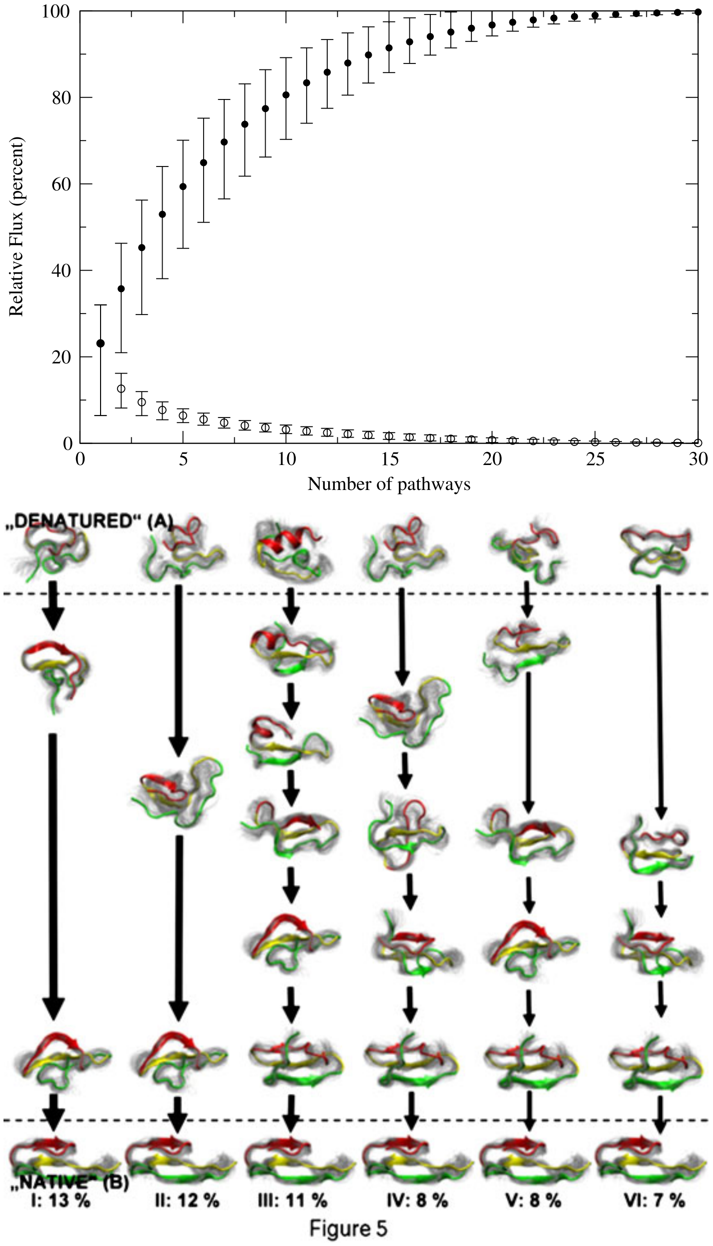


Fig. 6.8 *Top:* Fluxes of individual pathways and the cumulative flux. The *bullets* indicate the mean of the distribution, the error bars mark the 80 %-confidence interval. *Bottom:* The six individual pathways which carry most of the total flux (nearly 60 %). Figure reprinted from [33]

flux. The effective flux reveals that these traps are accessible from different metastable states, all of which already have a register shift in hairpin 2 or a partial register shift in hairpin 1 (see Fig. 6.7).

Removing the trap states from the Markov model increases the absolute folding rate k_{AB} (Eq. (6.4)) by almost a factor of 2, showing that there is a significant probability that the system gets stuck in one of the trap states for some time.

6.5 Experimental Observables/Dynamical Fingerprints

In experimental studies of protein folding, the conformational dynamics is mapped onto an observable a which is measured. a could be a fluorescence or transfer efficiency in a fluorescence experiment, the chemical shift in an NMR experiment, the intensity of a given spectral peak in an IR experiment, the distance in a pulling experiment, and so forth. In the following we assume, that a has a scalar value for every state S_i, i.e. there is a mapping $S_i \rightarrow a_i$, where a_i is the mean values of a over the state S_i. We note that vector- or function-valued observables (such as entire spectra in IR or NMR data) could be treated in a similar way, although this is not done here. Given the observable vector, various experimental measurements can be expressed as derived in [31] and [22].

In equilibrium experiments, the observed molecule is in equilibrium with the current conditions of the surroundings (temperature, applied forces, salt concentration etc.), and the mean value of an observable a, $\mathbb{E}_\pi[a]$, is recorded. This may be either done my measuring $\mathbb{E}_\pi[a]$ directly from an unperturbed ensemble of molecules, or by recording sufficiently many and long single molecule traces $a(t)$ and averaging over them. The expected measured signal is

$$\mathbb{E}_\pi[a] = \sum_{i=1}^{n} a_i \pi_i = \langle \mathbf{a}, \boldsymbol{\pi} \rangle. \tag{6.5}$$

where $\mathbb{E}[x]$ denotes the expectation value of an observable $x(t)$ and $\langle \mathbf{x}, \mathbf{y} \rangle$ denotes the scalar product between two vectors $\mathbf{x}$ and $\mathbf{y}$. Since $\boldsymbol{\pi}$ is the eigenvector to eigenvalue 1 of the transition matrix $\mathbf{T}(\tau)$, it can easily be calculated from the MSM. $\mathbb{E}_\pi[a]$ does not depend on time and therefore bears no kinetic information.

Kinetic information is available through time-correlation experiments. These may be realized by taking trajectories from time-resolved single molecule experiments, such as single molecule fluorescence or pulling experiments, and computing time correlations from these trajectories. Given a partition into states S_i, the autocorrelation of a for time $k\tau$ can be expressed as:

$$\begin{aligned}\mathbb{E}\big[a(t)a(t+k\tau)\big] &= \sum_{i=1}^{n}\sum_{j=1}^{n} a_i \mathbb{P}(s_t = S_i) \\ &\quad \cdot a_j \mathbb{P}(s_{t+k\tau} = S_j \mid s_t = S_i). \end{aligned} \tag{6.6}$$

The terms under the summation signs contain the product of the signal in state i and the signal in state j, $a_i a_j$, where a_i is weighted by the probability of finding the system in state S_i, and a_j is weighted by the conditional probability of finding the system in state j given that it has been in state i at k timesteps τ earlier. In equilibrium, the former probability is given by the equilibrium probability π. Assuming that the process is Markovian, the latter probability is given by the transition matrix element of the corresponding transition matrix. Equation (6.6) can be rewritten as a matrix equation in which $\mathbf{T}(\tau)$ appears explicitly

$$\begin{aligned}\mathbb{E}\big[a(t)a(t+k\tau)\big] &= \sum_{i=1}^{n}\sum_{j=1}^{n} a_i \pi_i \cdot a_j \big[\mathbf{T}^k(\tau)\big]_{ij} \\ &= \mathbf{a}^T \mathit{\Pi} \mathbf{T}^k(\tau)\mathbf{a}. \end{aligned} \tag{6.7}$$

Replacing $\mathbf{T}^k(\tau)$ by its spectral decomposition (Eq. (6.2)), one obtains

$$\begin{aligned}&\mathbb{E}\big[a(t)a(t+k\tau)\big] \\ &\quad = \mathbf{a}^T \left[\sum_{i=1}^{n} \exp\left(-\frac{k\tau}{t_i}\right) \boldsymbol{\phi}^{(i)} \boldsymbol{\phi}^{(i)T}\right] \mathbf{a} \\ &\quad = \langle \mathbf{a}, \pi \rangle^2 + \sum_{i=2}^{n} \exp\left(-\frac{k\tau}{t_i}\right) \big\langle \mathbf{a}, \boldsymbol{\phi}^{(i)} \big\rangle^2. \end{aligned} \tag{6.8}$$

Likewise, cross-correlation functions can be computed as

$$\begin{aligned}&\mathbb{E}\big[a(t)b(t+k\tau)\big] \\ &\quad = \langle \mathbf{a}, \pi \rangle \langle \mathbf{b}, \pi \rangle \\ &\qquad + \sum_{i=2}^{n} \exp\left(-\frac{k\tau}{t_i}\right) \big\langle \mathbf{a}, \boldsymbol{\phi}^{(i)} \big\rangle \big\langle \mathbf{b}, \boldsymbol{\phi}^{(i)} \big\rangle. \end{aligned} \tag{6.9}$$

Equations (6.8) and (6.9) have the form of a multiexponential decay function

$$f(t) = \gamma_1^{\mathrm{corr}} + \sum_{i=2} \gamma_i^{\mathrm{corr}} \exp\left(-\frac{t}{t_i}\right), \tag{6.10}$$

with amplitudes

$$\gamma_i^{\mathrm{corr}} = \big\langle \mathbf{a}, \boldsymbol{\phi}^{(i)} \big\rangle \big\langle \mathbf{b}, \boldsymbol{\phi}^{(i)} \big\rangle. \tag{6.11}$$

Table 6.2 Overview of the expressions for the amplitudes in correlation experiments

	equilibrium correlation experiment	relaxation experiment
relaxation experiment	–	$\gamma_i^{\text{relax}} = \langle \mathbf{a}, \boldsymbol{\phi}^{(i)} \rangle \langle \mathbf{p}'^T(0), \boldsymbol{\phi}^{(i)} \rangle$
autocorrelation	$\gamma_i^{\text{eq,ac}} = \langle \mathbf{a}, \boldsymbol{\phi}^{(i)} \rangle^2$	$\gamma_i^{\text{jump,ac}} = \langle \mathbf{a}, \mathbf{P}'(0)\boldsymbol{\phi}^{(i)} \rangle \langle \mathbf{a}, \boldsymbol{\phi}^{(i)} \rangle$
cross-correlation	$\gamma_i^{\text{eq,cc}} = \langle \mathbf{a}, \boldsymbol{\phi}^{(i)} \rangle \langle \mathbf{b}, \boldsymbol{\phi}^{(i)} \rangle$	$\gamma_i^{\text{jump,cc}} = \langle \mathbf{a}, \mathbf{P}'(0)\boldsymbol{\phi}^{(i)} \rangle \langle \mathbf{b}, \boldsymbol{\phi}^{(i)} \rangle$

Each of the amplitudes is associated with an eigenvector of the transition matrix and the decay constant t_i is the implied time scale of this eigenvector, $t_i = -\tau / \ln \lambda_i$.

Alternatively, relaxation experiments can be used to probe the molecules' kinetics. In these experiments, the system is allowed to relax from a non-equilibrium starting state with probability distribution $\mathbf{p}(0)$. Examples are temperature-jump, pressure-jump, or pH-jump experiments, rapid mixing experiments, or experiments where measurement at $t = 0$ starts from a synchronized starting state, such as in processes that are started by an external trigger like a photoflash. After time $t = 0$ the conditions are governed by a transition matrix $\mathbf{T}(\tau)$ with stationary distribution $\boldsymbol{\pi} \neq \mathbf{p}(0)$. The ensemble average $\mathbb{E}_{\boldsymbol{p}(0)}[a(t)]$ is recorded while the system relaxes from the initial distribution $\mathbf{p}(0)$ to the new equilibrium distribution π. The expectation value of the signal at time $t = k\tau$ depends on the current probability distribution $\mathbf{p}(k\tau)$ and is given by

$$\mathbb{E}_{\boldsymbol{p}(0)}\big[a(k\tau)\big] = \sum_{i=1}^{n} a_i p_i(k\tau) = \big\langle \mathbf{a}, \mathbf{p}(k\tau) \big\rangle. \tag{6.12}$$

Equation (6.12) is analogous to Eq. (6.9). $\mathbf{p}(k\tau)$ evolves under the influence of the transition matrix $\mathbf{T}(\tau)$. Using the spectral decomposition of $\mathbf{T}(\tau)$ (Eq. (6.2)) and expressing λ_i^k via implied timescales t_i, we obtain

$$\begin{aligned} &\mathbb{E}_{\boldsymbol{p}(0)}\big[a(k\tau)\big] \\ &\quad = \big\langle \mathbf{p}'(0), \pi \big\rangle \langle \mathbf{a}, \pi \rangle \\ &\qquad + \sum_{i=2}^{n} \exp\left(-\frac{k\tau}{t_i}\right) \big\langle \mathbf{p}'(0), \boldsymbol{\phi}^{(i)} \big\rangle \big\langle \mathbf{a}, \boldsymbol{\phi}^{(i)} \big\rangle \end{aligned} \tag{6.13}$$

where $\mathbf{p}'(0)$ is the *excess probability distribution* $\mathbf{p}'(0) = \Pi^{-1}\mathbf{p}(0)$. $\mathbb{E}_{\boldsymbol{p}(0)}[a(k\tau)]$ is again a multi-exponential decay function with amplitudes

$$\gamma_i^{\text{relax}} = \big\langle \mathbf{p}'(0), \boldsymbol{\phi}^{(i)} \big\rangle \big\langle \mathbf{a}, \boldsymbol{\phi}^{(i)} \big\rangle. \tag{6.14}$$

A summary of the amplitudes of various types of experiments is given in Table 6.2.

These equations are useful to calculate based on simulations which processes a given experiment will be sensitive to. To illustrate this, consider again the protein folding model and let us consider three different observables. In observable A, we measure the formation of structure element a, i.e. $a = 1$ for states in which a is formed while $a = 0$ for states in which a is not formed. Likewise observables B and C measure the formation of structure elements b and c. This can be realized e.g. with a fluorophor and a specific quencher at appropriate positions [12]. We also consider three ways of measuring each of these three constructs, namely temperature jump experiments at three different temperatures from 0.15 to 0.2, from 0.6 to 0.65, and from 2.4 to 2.45. We calculate the amplitude that is in the slowest and second-slowest processes and report the normalized results in Fig. 6.9.

It is apparent that the processes that can be measured drastically depends on the way the measurement is done and the observable used. For example, at high temperatures, all observables yield nearly single-exponential kinetics with the timescale of moving between the unfolded state and the partially structured state. At low temperature, the kinetics may appear biexponential, provided that measurement noise is sufficiently small, with the main amplitude being in the formation of a (γ_2) and c (γ_3).

The combination of Markov models and the spectral theory given is useful to compare simulations and experiments via the *dynamical finger-*

		Obs A	Obs B	Obs C
T-Jump 0.15 → 0.20	γ_2	0.71	0.19	0.13
	γ_3	0.29	0.81	0.87
T-Jump 0.60 → 0.65	γ_2	0.94	0.89	0.17
	γ_3	0.06	0.11	0.83
T-Jump 2.40 → 2.45	γ_2	0.98	0.95	0.89
	γ_3	0.02	0.05	0.11

Fig. 6.9 Normalized amplitudes of the slowest and second-slowest processes of simulated temperature-jump experiments of the folding model

print representation of the system kinetics [31]. Furthermore, this approach permits to design experiments that are optimal to probe individual relaxations [31].

References

1. Amadei A, Linssen AB, Berendsen HJC (1993) Essential dynamics of proteins. Proteins 17:212–225
2. Bachmann A, Kiefhaber T (2001) Apparent two-state tendamistat folding is a sequential process along a defined route. J Mol Biol 306(2):375–386. doi:10.1006/jmbi.2000.4399
3. Baldwin RL, Rose GD (1999) Is protein folding hierarchic? I. Local structure and peptide folding. Trends Biochem Sci 24(1):26–33. http://view.ncbi.nlm.nih.gov/pubmed/10087919
4. Baldwin RL, Rose GD (1999) Is protein folding hierarchic? II. Folding intermediates and transition states. Trends Biochem Sci 24(2):77–83. http://view.ncbi.nlm.nih.gov/pubmed/10098403
5. Berezhkovskii A, Hummer G, Szabo A (2009) Reactive flux and folding pathways in network models of coarse-grained protein dynamics. J Chem Phys 130(20). doi:10.1063/1.3139063
6. Bolhuis PG, Chandler D, Dellago C, Geissler PL (2002) Transition path sampling: throwing ropes over rough mountain passes, in the dark. Annu Rev Phys Chem 53(1):291–318. doi:10.1146/annurev.physchem.53.082301.113146
7. Bradley C, Barrick D (2006) The notch ankyrin domain folds via a discrete, centralized pathway. Structure 14(8):1303–1312. doi:10.1016/j.str.2006.06.013
8. Cellitti J, Bernstein R, Marqusee S (2007) Exploring subdomain cooperativity in T4 lysozyme II: uncovering the C-terminal subdomain as a hidden intermediate in the kinetic folding pathway. Protein Sci 16(5):852–862. doi:10.1110/ps.062632807
9. Chodera JD, Dill KA, Singhal N, Pande VS, Swope WC, Pitera JW (2007) Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics. J Chem Phys 126:155,101
10. Deuflhard P, Weber M (2003) Robust Perron cluster analysis in conformation dynamics. ZIB report 03-09
11. Dill KA, Ozkan SB, Shell MS, Weikl TR (2008) The protein folding problem. Annu Rev Biophys 37(1):289–316. doi:10.1146/annurev.biophys.37.092707.153558
12. Doose S, Neuweiler H, Sauer M (2009) Fluorescence quenching by photoinduced electron transfer: a reporter for conformational dynamics of macromolecules. Chem Phys Chem 10(9–10):1389–1398. doi:10.1002/cphc.200900238
13. Du R, Pande VS, Yu A, Tanaka T, Shakhnovich ES (1998) On the transition coordinate for protein folding. J Chem Phys 108(1):334–350. doi:10.1063/1.475393
14. Feng H, Zhou Z, Bai Y (2005) A protein folding pathway with multiple folding intermediates at atomic resolution. Proc Natl Acad Sci USA 102(14):5026–5031. doi:10.1073/pnas.0501372102
15. Friel CT, Beddard GS, Radford SE (2004) Switching two-state to three-state kinetics in the helical protein Im9 via the optimisation of stabilising non-native interactions by design. J Mol Biol 342:261–273
16. Gilmanshin R, Williams S, Callender RH, Woodruff, Dyer RB (1997) Fast events in protein folding: relaxation dynamics of secondary and tertiary structure in native apomyoglobin. Proc Natl Acad Sci USA 94:3709–3713
17. Goldbeck RA, Thomas YG, Chen E, Esquerra RM, Kliger DS (1999) Multiple pathways on a protein-folding energy landscape: kinetic evidence. Proc Natl Acad Sci USA 96(6):2782–2787. http://view.ncbi.nlm.nih.gov/pubmed/10077588
18. van Gunsteren WF, Berendsen HJC (1990) Computer simulation of molecular dynamics: methodology, applications and perspectives in chemistry. Angew Chem, Int Ed Engl 29:992–1023
19. Hoang L, Bédard S, Krishna MMG, Lin Y, Englander SW (2002) Cytochrome c folding pathway: kinetic native-state hydrogen exchange. Proc Natl Acad Sci USA 99(19):12,173–12,178. doi:10.1073/pnas.152439199
20. Jäger M, Nguyen H, Crane JC, Kelly JW, Gruebele M (2001) The folding mechanism of a beta-sheet: the WW domain. J Mol Biol 311(2):373–393. doi:10.1006/jmbi.2001.4873
21. Jane Wright PEE, Scheraga HAA (2006) The role of hydrophobic interactions in initiation and propagation of protein folding. Proc Natl Acad Sci USA 103(35):13,057–13,061. doi:10.1073/pnas.0605504103
22. Keller B, Prinz JH, Noé F (2012) Markov models and dynamical fingerprints: unraveling the complexity of molecular kinetics. Chem Phys 396:92–107
23. Korzhnev DM, Salvatella X, Vendruscolo M, Di Nardo AA, Davidson AR, Dobson CM, Kay LE (2004) Low-populated folding intermediates of Fyn SH3 characterized by relaxation dispersion NMR. Nature 430(6999):586–590. doi:10.1038/nature02655

24. Kube S, Weber M (2007) A coarse graining method for the identification of transition rates between molecular conformations. J Chem Phys 126(2):024,103+. doi:10.1063/1.2404953
25. Lindberg MO, Oliveberg M (2007) Malleability of protein folding pathways: a simple reason for complex behaviour. Curr Opin Struct Biol 17(1):21–29. doi:10.1016/j.sbi.2007.01.008
26. Matagne A, Radford SE, Dobson CM (1997) Fast and slow tracks in lysozyme folding: insight into the role of domains in the folding process. J Mol Biol 267(5):1068–1074. doi:10.1006/jmbi.1997.0963
27. Mello CC, Barrick D (2004) An experimentally determined protein folding energy landscape. Proc Natl Acad Sci USA 101(39):14,102–14,107. doi:10.1073/pnas.0403386101
28. Metzner P, Schütte C, Eijnden EV (2009) Transition path theory for Markov jump processes. Multiscale Model Simul 7:1192–1219
29. Metzner P, Schütte C, Vanden-Eijnden E (2006) Illustration of transition path theory on a collection of simple examples. J Chem Phys 125(8). doi:10.1063/1.2335447
30. Noé F (2008) Probability distributions of molecular observables computed from Markov models. J Chem Phys 128:244,103
31. Noé F, Doose S, Daidone I, Löllmann M, Chodera JD, Sauer M, Smith JC (2011) Dynamical fingerprints for probing individual relaxation processes in biomolecular dynamics with simulations and kinetic experiments. Proc Natl Acad Sci USA 108:4822–4827
32. Noé F, Horenko I, Schütte C, Smith JC (2007) Hierarchical analysis of conformational dynamics in biomolecules: transition networks of metastable states. J Chem Phys 126:155,102
33. Noé F, Schütte C, Vanden-Eijnden E, Reich L, Weikl TR (2009) Constructing the full ensemble of folding pathways from short off-equilibrium simulations. Proc Natl Acad Sci USA 106:19,011–19,016
34. Prinz JH, Wu H, Sarich M, Keller B, Fischbach M, Held M, Chodera JD, Schütte C, Noé F (2011) Markov models of molecular kinetics: generation and validation. J Chem Phys 134:174,105
35. Rao F, Caflisch A (2004) The protein folding network. J Mol Biol 342:299–306
36. Sarich M, Noé F, Schütte C (2010) On the approximation error of Markov state models. Multiscale Model Simul 8:1154–1177
37. Schütte C, Fischer A, Huisinga W, Deuflhard P (1999) A direct approach to conformational dynamics based on hybrid Monte Carlo. J Comput Phys 151:146–168
38. van der Spoel D, Lindahl E, Hess B, Groenhof G, Mark AE, Berendsen HJC (2005) GROMAC: fast, flexible and free. J Comput Chem 26:1701–1718
39. Sridevi K (2000) The slow folding reaction of barstar: the core tryptophan region attains tight packing before substantial secondary and tertiary structure formation and final compaction of the polypeptide chain. J Mol Biol 302(2):479–495. doi:10.1006/jmbi.2000.4060
40. Sridevi K, Lakshmikanth GS, Krishnamoorthy G, Udgaonkar JB (2004) Increasing stability reduces conformational heterogeneity in a protein folding intermediate ensemble. J Mol Biol 337(3):699–711. doi:10.1016/j.jmb.2003.12.083
41. Street TO, Bradley CM, Barrick D (2007) Predicting coupling limits from an experimentally determined energy landscape. Proc Natl Acad Sci USA 104(12):4907–4912. doi:10.1073/pnas.0608756104
42. Swope WC, Pitera JW, Suits F (2004) Describing protein folding kinetics by molecular dynamics simulations, 1: theory. J Phys Chem B 108:6571–6581
43. Vanden-Eijnden E (2006) Transition path theory. In: Computer simulations in condensed matter systems: from materials to chemical biology, vol 1. Springer, Heidelberg, pp 453–493. doi:10.1007/3-540-35273-2_13
44. Vreede J, Juraszek J, Bolhuis PG (2010) Predicting the reaction coordinates of millisecond light-induced conformational changes in photoactive yellow protein. Proc Natl Acad Sci USA 107:2397–2402
45. Vanden-Eijnden E (2006) Towards a theory of transition paths. J Stat Phys 123(3):503–523. doi:10.1007/s10955-005-9003-9
46. Vanden-Eijnden E (2010) Transition-path theory and path-finding algorithms for the study of rare events. Annu Rev Phys Chem 61:391–420
47. Weber M (2003) Improved Perron cluster analysis. ZIB report 03-04
48. Weikl TR (2008) Transition states in protein folding kinetics: modeling phi-values of small beta-sheet proteins. Biophys J 94(3):929–937. http://www.cell.com/biophysj/abstract/S0006-3495(08)70691-X
49. Yeh SR, Rousseau DL (2000) Hierarchical folding of cytochrome c. Nat Struct Biol 7(6):443–445. doi:10.1038/75831

7 Transition Path Theory

Eric Vanden-Eijnden

7.1 Introduction

Markov State Models (MSMs) are meant to be a way to analyze complex time-series data from molecular dynamics (MD) simulations. But MSMs can be complicated themselves and require analysis tools that go beyond simple "look-and-see" techniques. In this chapter we describe a set of such tools based on the framework of Transition Path Theory (TPT) originally introduced in [4] and further developed in [1, 5, 7, 8, 12]. In a nutshell, these tools are aimed at understanding the mechanism and rate of specific reactions in the system, that is, transitions between any particular states or group of states of interest in the MSM. Beside the theory recalled in Sect. 7.2, the main upshots of this chapter are two algorithms presented in Sect. 7.4 that permit to generate directly reactive trajectories (i.e. trajectories by which a specific transition of interest occurs) and loop-erased reactive trajectories (i.e. reactive trajectories from which we have extracted the productive pieces when they progress from the reactant to the product state).

We begin by setting up notation and recalling a few basic concepts of the theory of discrete-time Markov chains that will prove useful in the sequel [10]. Let $T_{ij} \equiv T_{ij}(\tau)$, $i, j = 1, \ldots, N$ denote the entries of the probability transition matrix over the N states of the MSM and let us assume that this matrix satisfies a detailed balance condition (time-reversibility) with respect to the equilibrium probability distribution π_i, i.e.

$$\pi_i T_{ij} = \pi_j T_{ji}, \quad \forall i, j = 1, \ldots, N. \tag{7.1}$$

This condition implies that the transition matrix T_{ij} admits the spectral decomposition

$$T_{ij} = \sum_{k=1}^{N} \lambda_k \psi_i^{(k)} \psi_j^{(k)} \pi_j \tag{7.2}$$

where the eigenvalue/eigenvector pairs $(\lambda_k, \psi^{(k)})$ satisfy

$$\sum_{j=1}^{N} T_{ij} \psi_j^{(k)} = \lambda_k \psi_i^{(k)}, \quad k = 1, \ldots, N \tag{7.3}$$

with the normalization condition

$$\sum_{i=1}^{N} \psi_i^{(k)} \psi_i^{(l)} \pi_i = \delta_{k,l}, \quad k, l = 1, \ldots, N. \tag{7.4}$$

Assuming ergodicity, the detailed balance condition (7.1) also implies that all the eigenvectors but the first $\lambda_1 = 1$ (associated with $\psi_i^{(1)} = 1$) are in the interval $(-1, 1)$ and can be ordered as $1 = \lambda_1 > |\lambda_2| \geq \cdots \geq |\lambda_N| \geq 0$. If we denote by $\mu_i(n)$ the probability to observe the system in state i after n steps (i.e. at time $n\tau$), this proba-

E. Vanden-Eijnden (✉)
Courant Institute, New York University, 251 Mercer street, New York, NY 10012, USA
e-mail: eve2@cims.nyu.edu

G.R. Bowman et al. (eds.), *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology 797,
DOI 10.1007/978-94-007-7606-7_7, © Springer Science+Business Media Dordrecht 2014

bility also admits a spectral decomposition

$$\mu_i(n) = \sum_{k=1}^{N} c_k \lambda_k^n \psi_i^{(k)} \pi_i, \quad c_k = \sum_{j=1}^{N} \psi_j^{(k)} \mu_j(0) \tag{7.5}$$

and so does the probability current $F_{ij}(n)$ entering the forward Kolmogorov equation for $\mu_i(n)$:

$$\mu_i(n+1) - \mu_i(n) = \sum_{j=1}^{N} F_{ij}(n),$$
$$F_{ij}(n) = \mu_j(n) T_{ji} - \mu_i(n) T_{ij}. \tag{7.6}$$

Explicitly:

$$F_{ij}(n) = \sum_{k=1}^{N} c_k \lambda_k^n F_{ij}^{(k)},$$
$$F_{ij}^{(k)} = \pi_i T_{ij} \big(\psi_j^{(k)} - \psi_i^{(k)}\big). \tag{7.7}$$

The spectral decompositions above permit to analyze how the system relaxes to equilibrium and they are most useful in systems displaying metastability [2, 3]. By definition, a metastable system is one in which there is a group of eigenvalues very close to 1 whereas the ones outside this group are much less so, i.e. there is a $M < N$ such that $1 - |\lambda_M| \le \varepsilon \ll 1$ and $1 - |\lambda_{M+1}| \ge \delta \gg \varepsilon$. The eigenvalues and eigenvectors in this low-lying group capture the slow relaxation processes in the system, in the sense that there are values of n such that λ_{M+1}^n has decayed to values very close to zero but λ_M^n has not: on time-scales reached after such n steps, the sum over k in the spectral decompositions of the probability in (7.5) and the current in (7.7) can be truncated to the first M terms. It is well-known [11] that the low-lying eigenvectors, $\psi_i^{(1)}, \dots, \psi_i^{(M)}$, permit to identify the metastable regions in phase space where the system gets trapped for long period of times, and the low-lying currents, $F_{ij}^{(1)}, \dots, F_{ij}^{(M)}$, indicate how the probability slowly flows between these regions to eventually reach equilibrium—in this sense, they explain the mechanism of transition between the metastable states.

The discussion above suggests that a way to analyze an MSM could be to calculate its spectrum and focus on its low-lying part. This idea, however, is not a very practical one in many systems of interest whose complexity goes far beyond the numerical examples usually chosen to illustrate them. Indeed, it is often the case that the low-lying part of the spectrum is quite complicated, with subgroups of metastable eigenvalues into metastable groups, subsubgroups into the subgroups, etc. Or that the system is in fact not very metastable in the sense that distinguishing between these groups is not straightforward. In some sense, the spectral decomposition of the chain contains all the information about the MSM but this is often too much of it to be useful. In many situations, it would be preferable to have at one's disposal a set of tools that permit to understand more specifically how the system makes transitions between arbitrary states of interest that one picks *a priori*. These states could for example be those associated with the unfolded structure of a protein on the one hand, and its native structure on the other. To build such tools, one could introduce a source and a sink on these states and see how the system evolves after such modification. In essence, this is the idea behind TPT, except that the source and the sink are introduced in a natural way that do not alter the mechanism of the reaction between states of interest. How this is done is explained next.

7.2 The Basic Concepts and Main Outputs of TPT

Suppose that we have identified two states or group of states in the MSMs, which we will denote by A and B, and we would like to understand the mechanism by which the system transits from A to B or vice-versa—by detailed balance these two types of transitions are the time-reversed of one another. One way to think about this problem is to imagine that we have at our disposal a very long equilibrium trajectory of the MSM out of which we prune the successive pieces during which the system has last left A before entering B—a schematic representation of this procedure is shown in Fig. 7.1. We call these pieces the 'reactive trajectories' (they are shown in red in Fig. 7.1) and we ask about their statistical properties, such as their probability distribution, their

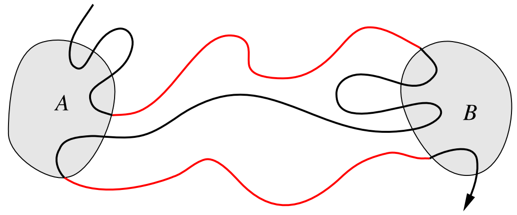


Fig. 7.1 Schematic representation of a long equilibrium trajectory in the MSM oscillating between set A and set B. The *red* pieces of this trajectory, during which it has last left A before entering B, form the ensemble of reactive trajectories. In this cartoon the trajectory looks continuous but in the MSM it is discrete both in space and time

probability current, etc. To answer these questions, let us first estimate the probability that a trajectory visiting state i be reactive, i.e. that this visit occurs while the system is on its way from A to B. Clearly this requires that after leaving i the trajectory will first reach B rather than A, and that before arriving i it last left A rather than B. By time-reversibility, the probability of the second event is one minus the probability of the first, which we denote by q_i and call the committor function for the reaction from A to B or, in short, the committor. It is easy to write down an equation for q_i:

$$q_i = \sum_{j\in B} T_{ij} + \sum_{j\notin A\cup B} T_{ij}q_j, \quad i \notin A\cup B. \quad (7.8)$$

This equation uses the Markov property and simply says that the probability q_i that after leaving i the trajectory will first reach B rather than A is the sum of the probabilities of distinct events: either the trajectory goes to B rather than A in one step after leaving i, which leads to the first term at the right hand-side of (7.8), or it goes to some intermediate state j not in A or B and then to B rather than A after leaving that intermediate state, which leads to the second term at the right hand-side of (7.8). Note that since $q_i = 0$ if $i \in A$ and $q_i = 1$ if $i \in B$, (7.8) can also be written more compactly as

$$q_i = \sum_{j=1}^{N} T_{ij}q_j, \quad i \notin A\cup B. \quad (7.9)$$

In terms of the committor q_i, the probability that a trajectory visiting state i be reactive is simply given by $q_i(1-q_i)$, which also means that the equilibrium probability to observe a trajectory at state i and that it be reactive is given by

$$\pi_i^R = (1-q_i)\pi_i q_i. \quad (7.10)$$

Note that this distribution is not normalized to 1. In fact

$$\rho_R = \sum_{i=1}^{N} \pi_i^R = \sum_{i=1}^{N}(1-q_i)\pi_i q_i \quad (7.11)$$

gives the equilibrium probability for the trajectory to be reactive, that is, the proportion of time it spends on the pieces where it has last left A before entering B.

By a similar argument we can derive an expression for the joint equilibrium probability that the trajectory visits states i and j consecutively and that it be reactive:

$$\pi_{ij}^R = (1-q_i)\pi_i T_{ij} q_j. \quad (7.12)$$

If we anti-symmetrize this quantity we obtain the probability current of the reactive trajectories that indicates how, on average, reactive trajectories flow from A to B:

$$F_{ij}^R = \pi_{ij}^R - \pi_{ji}^R = \pi_i T_{ij}(q_j - q_i). \quad (7.13)$$

This current should be compared to the $F_{ij}^{(k)}$ defined in (7.7). It is known [3] that the low-lying eigenvectors can be approximated by the committor function for the reaction between appropriately chosen sets A and B. Equation (7.13) can be used to make the same statement at the level of currents. Of course, (7.13) is exact and does not require the system to be metastable: the current F_{ij}^R permits to analyze the mechanism of the transition between any two sets A and B, as will be illustrated in Sects. 7.3 and 7.4.

Another quantity of interest is the average frequency at which the trajectory makes transition between A and B. Since by definition every reactive trajectory that leaves A goes to B next, this average frequency can be expressed as the total current of reactive trajectories out of A or, equivalently, their total current into B:

$$\begin{aligned}\nu_R &= \sum_{i\in A}\sum_{j\notin A} F_{ij}^R = \sum_{i\in A}\sum_{j\notin A} \pi_i T_{ij} q_j \\ &= \sum_{i\notin B}\sum_{j\in B} F_{ij}^R = \sum_{i\notin B}\sum_{j\in B} (1-q_i)\pi_i T_{ij}\end{aligned} \tag{7.14}$$

where we used $q_i = 0$ if $i \in A$ and $q_i = 1$ if $i \in B$. It is a simple exercise to show from (7.9) that these expressions for ν_R can be reorganized into

$$\nu_R = \tfrac{1}{2}\sum_{i,j=1}^N \pi_i T_{ij}(q_j - q_i)^2. \tag{7.15}$$

If N_T denotes the number of reactive pieces along a trajectory of length T, ν_R is the limit of N_T/T as $T \to \infty$ and it should not be confused with the rates of transition from A to B and B to A, which we will denote by $k_{A,B}$ and $k_{B,A}$, respectively. Indeed, ν_R is symmetric with respect to A and B: the number of transitions from A to B is the same as the one from B to A, since each of the first is followed by one of the second. The rates $k_{A,B}$ and $k_{B,A}$, on the other hand, account for the fact that between these transitions the system can spend more time in one set than in the other: If T_A and T_B denote the sum of all times during which the last set visited by a trajectory of length T was A or B, respectively, by definition $k_{A,B}$ is the limit of N_T/T_A as $T \to \infty$ and $k_{B,A}$ that of N_T/T_B. Since the probabilities to find the system in state i and that it came last from A or B are given by $\pi_i(1-q_i)$ and $\pi_i q_i$, respectively, the proportions of time that the system was last in A or B are

$$\begin{aligned}\rho_A &= \sum_{i=1}^N \pi_i(1-q_i), \\ \rho_B &= \sum_{i=1}^N \pi_i q_i \qquad (\rho_A + \rho_B = 1),\end{aligned} \tag{7.16}$$

ρ_A is the limit of T_A/T as $T \to \infty$ and ρ_B that of T_B/T. To obtain the transition rates, we must then divide ν_R by these quantities:

$$k_{A,B} = \nu_R/\rho_A, \qquad k_{B,A} = \nu_R/\rho_B. \tag{7.17}$$

The inverse of these rates, $1/k_{A,B}$ and $1/k_{B,A}$, give, respectively, the mean times between successive visits of A then B or B then A. We could also ask ourselves what is the average duration of the reactive trajectories, i.e. the mean time the trajectory spends to go to B after leaving A last. This mean time is simply given by

$$\tau_R = \rho_R/\nu_R. \tag{7.18}$$

Other statistical properties of the reactive trajectories can be derived from arguments similar to the ones above. We will, however, conclude here our little tour of TPT and henceforth focus on tools to analyze and interpret the main output of TPT, namely (7.10), (7.12) and (7.13).

7.3 Illustrative Example

It is useful to illustrate the main outputs of TPT on an example where these quantities can simply be plotted directly. This will help us understand the meaning of these objects. To this end, consider a system whose state space are the nodes on the square grid shown in the left panel of Fig. 7.2. To every node on the grid, we associated an energy, E_i, and we assume that the system can hop from a node to one of its direct neighbors on the grid with a probability consistent with Metropolis-Hasting rule. Specifically, if $a_{ij} = a_{ji}$ denotes the adjacency matrix of the grid (i.e. $a_{ij} = 1$ if i and j are direct neighbors and 0 otherwise), we take as transition matrix

$$T_{ij} = \hat{p}_{ij} \min\left(\frac{e^{-\beta E_j}\,\hat{p}_{ji}}{e^{-\beta E_i}\,\hat{p}_{ij}}, 1\right) \quad (i \neq j) \tag{7.19}$$

and $T_{ii} = 1 - \sum_{j\neq i} T_{ij}$; here $\hat{p}_{ij} = a_{ij}/\sum_j a_{ij}$ plays the role of the proposal distribution in the Metropolis-Hasting Monte-Carlo algorithm. This choice guarantees that the chain is ergodic with respect to the Boltzmann-Gibbs equilibrium distribution

$$\pi_i = C^{-1} e^{-\beta E_i}, \qquad C = \sum_{i=1}^N e^{-\beta E_i} \tag{7.20}$$

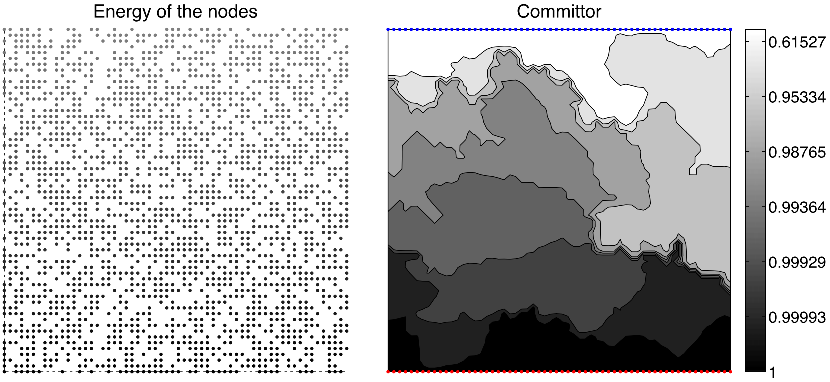


Fig. 7.2 *Left panel*: Each of the nodes on this 60 × 60 grid is a state of the Markov chain, and the grayscale of the node indicates its energy: the darker it is, the lower its energy, i.e. the higher its equilibrium probability (7.20). The white nodes (invisible on the figure) are accessible too, but their energy is higher and hence their equilibrium probability is low. *Right panel*: Isocontour plot of the committor q_i if A is made of the nodes in the top row shown in blue and B is made of the nodes in the bottom row shown in red. To create this figure, instead of plotting the values of the committor at the individual nodes, we interpolated a function between them—this improves the readability of the figure

where N is the number of nodes on the grid. The specific example shown in the left panel of Fig. 7.2 was constructed by first picking a random set of nodes, and assigning them high energies and hence low equilibrium probability (these nodes are white in the figure, making them invisible). For the remaining nodes, the smaller their y-coordinate on the grid, the lower we make their energy, so that at equilibrium the system spends slightly more time at the bottom than at the top. Here we will analyze how transitions between top and bottom arise. To make such a transition, it is natural to suspect that the system will find its way through the grid by hopping from low energy node to low energy node (i.e. avoiding the invisible nodes in the left panel of Fig. 7.2), and an interesting question is whether there is a preferential way to do this. Next we use TPT to answer this question. Note that this is an example in which spectral analysis is difficult to perform and not very enlightening: there are no low-lying eigenvalues, and the eigenvectors are complicated and hard to interpret.

If we pick the top row of nodes as set A, and the bottom row as set B, the committor has the structure shown in the right panel of Fig. 7.2. This figure alone does not explain much of the mechanism of the reaction, but it already indicates how the system can be stratified using the committor value of the nodes as reaction coordinate—in fact, this is the best reaction coordinate we can pick to describe the transition and the results in the right panel of Fig. 7.2 show that y-coordinate of the node on the grid is not a good indicator of how close this node is from B in terms of the transition. Indeed, the committor is quite different on the left and right halves of the system.

Shown in Fig. 7.3 is the probability distribution of reactive trajectories defined in (7.10) (left panel) and the total probability current (or total flux) of reactive trajectories going through each node defined as (right panel):

$$\frac{1}{2}\sum_{j=1}^{N}|F_{ij}^{R}|. \tag{7.21}$$

As can be seen, the probability distribution of reactive trajectories takes appreciable values on many of the nodes with low energy in the system, meaning that the reactive trajectories venture on all of these with comparable frequency. At the same time, however, only a small subset of nodes carry most of the current of reactive trajectories: from the right panel of Fig. 7.3 we see that these nodes are in a narrow tube at the top that branches into two tubes at the bottom. This means

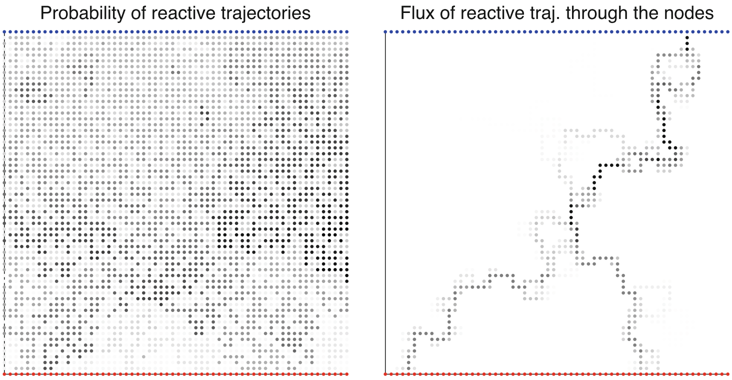


Fig. 7.3 *Left panel*: Probability distribution of the reactive trajectories (7.10). The darker the node, the higher its probability weight. *Right panel*: Every node is colored according to the total current of reactive trajectories going through it, $\frac{1}{2}\sum_j |F^R_{ij}|$, where F^R_{ij} is the current of reactive trajectories (7.13)

that the reactive trajectories themselves are quite complicated, and lose themselves in the 'maze' of the system, looping on themselves, going back and forth, etc. However there are portions of their paths during which they do advance productively from A to B, much in the same way as a random walker in a maze would make many detours and yet from time to time advance on the correct shortest path from entrance to exit. These portions involve mostly motion through the tubes shown in the right panel of Fig. 7.3, which is quite a remarkable conclusion considering how uniform the system looks to the naked eye. This indicates that the probability current of reactive trajectories is the object that is most useful to explain the mechanism of the transition, in the sense that it permits to identify the productive paths from A to B. How complicated the reactive trajectories themselves are, how much do they wander away from these productive paths and where, etc. can then be explained by the committor and the probability distribution of reactive trajectories which permit to identify dead-ends, dynamical traps, etc. as regions that are visited by the reactive trajectories but through which little of their probability current goes. Next we explain how to carry on such an analysis in situations where plotting directly the probability distribution and current of reactive trajectories is more difficult.

7.4 Analysis Tools Based on TPT

In this section we introduce a few tools to analyze the outputs of TPT in situations where we cannot simply plot them. To this end, we revisit a question that we asked earlier, namely how to modify the chain by introducing a source at A and a sink at B in such a way that the trajectories in this modified chain tell us something about the mechanism of the transition. We can use TPT as guide to design several ways to do this, each of which giving us a different type of information. Here we discuss two of them. For simplicity, we assume that $T_{ij}=0$ if $i\in A$ and $j\in B$, i.e. there are no direct jumps from A to B—if this assumption fails, some of the formulae below need to be modified.

The first way permits to generate directly a set of trajectories that are statistically identical to the set of reactive trajectories. This can be done by generating each trajectory via the following two-step procedure:

1. Pick a state i outside of A with probability

$$p'_i = \frac{\sum_{j\in A}\pi_j T_{ji} q_i}{\sum_{j\in A}\sum_{k\notin A}\pi_j T_{jk} q_k}. \tag{7.22}$$

2. Out of this state generate a trajectory using as transition matrix

$$T'_{ij} = \frac{T_{ij}q_j}{\sum_{k=1}^N T_{ik}q_k} = \frac{T_{ij}q_j}{q_i} \qquad (i\notin A\cup B) \tag{7.23}$$

until it reaches B, and keep only the sequence of states outside of A and B (i.e. disregard the last state it reached in B and instead pick again a new state according to p'_i to start a new trajectory).

Because of the presence of the factor q_j at the numerator of T'_{ij} the trajectories generated by this procedure can never go back to A and they eventually have to reach B. This make them good candidates for reactive trajectories. To show that they are indeed true reactive trajectories, let us check that their stationary distribution is given by the normalized version the probability distribution of reactive trajectories π_i^R defined in (7.10)

$$\pi_i' = \frac{\pi_i^R}{\sum_{j=1}^N \pi_j^R} = \frac{\pi_i^R}{\rho_R}. \tag{7.24}$$

We can verify this statement by checking that π_i' satisfies the right balance equation for the process defined by the procedure above based on (7.22) and (7.23):

$$\pi_i' \sum_{j\neq i} T_{ij}' = f_i' + \sum_{j\neq i} \pi_j' T_{ji}' \quad (i \notin A \cup B). \tag{7.25}$$

Here the left hand-side is the total probability flux outside state i and the right hand-side is the total probability flux into state i, including the flux coming from A which is accounted for by the term f_i' defined as

$$f_i' = \left(\sum_{k\notin B}\sum_{l\in B} \pi_k' T_{kl}'\right) p_i'. \tag{7.26}$$

To understand the form of this term, note that the generating procedure described above implies that a new trajectory is started each time the previous one reaches B. Thus the total flux out of A is the same as the total flux into B, which is $\sum_{k\notin B}\sum_{l\in B}\pi_k' T_{kl}'$: since this flux is distributed among states according to p_i', this gives (7.26). It is easy to see using their explicit expressions for p_i', T_{ij}', π_i' and f_i' in (7.25) that the left and right hand-sides in this equation balance each other, confirming that the procedure above does indeed generate reactive trajectories—this calculation is done at the end of this chapter.

While this is nice, from the example in Sect. 7.3 we know that reactive trajectories can be quite complicated and get caught in dynamical traps or dead-ends, loop on themselves, etc. For this reason it is more useful to generate loop-erased reactive trajectories, that is, paths that only contain the portions of the reactive trajectories when they advance productively from A to B. This can be done by modifying the procedure above as follows:

1. Pick a state i outside of A with probability p_i' given in (7.22).
2. Out of this state generate a trajectory using as transition matrix

$$T_{ij}^R = \begin{cases} T_{ij}(q_j - q_i)_+ & \text{if } i \neq j, \\ 1 - \sum_{k\neq i} T_{ik}(q_k - q_i)_+ & \text{if } i = j \end{cases} \quad (i \notin A \cup B) \tag{7.27}$$

until it reaches B, and keep only the sequence of states outside of A and B (i.e. disregard the last state it reached in B and instead pick again a new state according to p_i' to start a new trajectory): here $(q_j - q_i)_+ = \max\{(q_j - q_i), 0\}$.

Because of the presence of the factor $(q_j - q_i)_+$ the trajectories generated this way can only move from i to j if the committor value of j is higher than that of i, i.e. they can only take productive steps towards B. The stationary distribution of the trajectories generated this way is simply the equilibrium distribution conditioned on the region outside of A and B, i.e.

$$\pi_i^C = \frac{\pi_i}{\sum_{j\notin A\cup B}\pi_j} \tag{7.28}$$

if $i \notin A \cup B$ and $\pi_i^C = 0$ otherwise. This can be checked from the equivalent of (7.25), which in the present case reads

$$\pi_i^C \sum_{j\neq i} T_{ij}^R = f_i^R + \sum_{j\neq i} \pi_j^C T_{ji}^R \quad (i \notin A \cup B) \tag{7.29}$$

where

$$f_i^R = \left(\sum_{k\notin B}\sum_{l\in B} \pi_k^C T_{kl}^R\right) p_i' \tag{7.30}$$

(7.29) is checked at the end of this chapter. The form of the transition matrix in (7.27) and of the stationary distribution in (7.28) imply, in particular, that the probability current of these loop-erased reactive trajectories is

$$\pi_i^C T_{ij}^R \propto \pi_i T_{ij}(q_j - q_i)_+, \tag{7.31}$$

i.e. it is proportional to $(F_{ij}^R)_+$, where F_{ij}^R is the current of reactive trajectories defined in (7.13). Therefore, the loop-erased reactive trajectories carry the same amount of current as the reactive trajectories themselves, except that they only take productive steps from A to B, meaning in particular that they would concentrate mostly in the tube shown in the right panel of Fig. 7.3 in the example of Sect. 7.3. Note also that if we only care about the sequence of states visited along these paths (and not about the number of steps they stay in one given state), instead of T^R_{ij} we can generate paths using

$$\bar{T}_{ij} = \frac{T_{ij}(q_j - q_i)_+}{\sum_{k=1}^N T_{ik}(q_k - q_i)_+} \qquad (i \notin A \cup B). \tag{7.32}$$

Because $\bar{T}_{ii} = 0$, the system moves at every step, but it traces the same paths as the loop-erased reactive trajectories (i.e. these paths would again allow to identify the tubes in the right panel of Fig. 7.3).

The processes defined above by (7.22) and (7.23) or (7.22) and (7.27) or (7.22) and (7.32) can be used to perform statistical analysis of the reactive trajectories or the productive portions along them. For example, we can estimate what is the distribution of length of these paths, what is the highest energy point along them, what is the probability to observe a given state along them, etc. This analysis can be done globally, or by stratifying the system into regions where the committor function is between two given values (for example to assess the width of tubes or their multiplicity), etc.

Finally we note that other tools for the analysis of the output of TPT involve finding the dynamical bottlenecks and representative dominant pathways. These tools were originally introduced in [8, 9] (see also [6]) and they are discussed in Chap. 6 of this book.

Verification of (7.25)

First notice that we can augment the sums over j in each sides of (7.25) to all the states since the added terms involving $j = i$ are the same. Let us then consider all three terms separately using their explicit expressions for p'_i, T'_{ij}, π'_i and f'_i. First

$$\begin{aligned}\pi'_i \sum_{j=1}^N T'_{ij} &= \rho_R^{-1}(1-q_i)q_i\pi_i \sum_{j=1}^N \frac{T_{ij}q_j}{q_i}\\ &= \rho_R^{-1}(1-q_i)q_i\pi_i\end{aligned} \tag{7.33}$$

where we used (7.9) which is legitimate since $i \notin A \cup B$. Second

$$\begin{aligned}f'_i &= \rho_R^{-1} \sum_{k\notin B}\sum_{l\in B} q_k(1-q_k)\pi_k\\ &\qquad \times \frac{T_{kl}q_l}{q_k}\,\frac{\sum_{j\in A}\pi_j T_{ji} q_i}{\sum_{j\in A}\sum_{k\notin A}\pi_j T_{jk} q_k}\\ &= \rho_R^{-1} \sum_{k\notin B}\sum_{l\in B} (1-q_k)\pi_k T_{kl}\\ &\qquad \times \frac{\sum_{j\in A}\pi_j T_{ji} q_i}{\sum_{j\in A}\sum_{k\notin A}\pi_j T_{jk} q_k}\\ &= \rho_R^{-1} \sum_{j\in A} \pi_j T_{ji} q_i\\ &= \rho_R^{-1} q_i \pi_i \sum_{j\in A} T_{ij}\end{aligned} \tag{7.34}$$

where we used $q_l = 1$ if $l \in B$ to get the second equality, the property that the flux of reactive trajectories out of A is the same as the one into B (see (7.14)) to get the third, and the detailed balance condition (7.1) to get the fourth. Third

$$\begin{aligned}\sum_{j=1}^N \pi'_j T'_{ji} &= \rho_R^{-1} \sum_{j\notin A} (1-q_j)q_j\pi_j \frac{T_{ji}q_i}{q_j}\\ &= \rho_R^{-1} \sum_{j\notin A} (1-q_j)\pi_j T_{ji} q_i\\ &= \rho_R^{-1} q_i\pi_i \sum_{j\notin A} T_{ij}(1-q_j).\end{aligned} \tag{7.35}$$

If we add up (7.34) and (7.35), we obtain the right hand-side of (7.25):

$$\begin{aligned}&f'_i + \sum_{j=1}^N \pi'_j T'_{ji}\\ &\quad = \rho_R^{-1} q_i \pi_i \Bigl(\sum_{j\in A} T_{ij} + \sum_{j\notin A} T_{ij}(1-q_j)\Bigr)\end{aligned}$$

$$
\begin{aligned}
&= \rho_R^{-1} q_i \pi_i \left(\sum_{j=1}^{N} T_{ij}(1-q_j) \right) \\
&= \rho_R^{-1} q_i \pi_i (1-q_i) \qquad (7.36)
\end{aligned}
$$

where we used $q_j = 0$ if $j \in A$ to get the second equality, and (7.9) and $\sum_{j=1}^N T_{ij} = 1$ to get the third. This complete our verification since (7.36) is identical to the expression in (7.33) for the left hand side of (7.25).

Verification of (7.29)

We proceed similarly as in the verification of (7.25). First,

$$
\pi_i^C \sum_{j\neq i} T_{ij}^R = C^{-1} \pi_i \sum_{j=1}^{N} T_{ij}(q_j - q_i)_+ \qquad (7.37)
$$

where we defined $C = \sum_{i\notin A\cup B} \pi_i$ and we augmented the sum since $(q_j - q_i)_+ = 0$ anyway if $i = j$. Second,

$$
\begin{aligned}
f_i^R &= C^{-1} \sum_{k\notin B} \sum_{l\in B} \pi_k T_{kl}(q_l - q_k)_+ \\
&\quad \times \frac{\sum_{j\in A} \pi_j T_{ji} q_i}{\sum_{j\in A} \sum_{k\notin A} \pi_j T_{jk} q_k} \\
&= C^{-1} \sum_{k\notin B} \sum_{l\in B} \pi_k T_{kl}(1 - q_k) \\
&\quad \times \frac{\sum_{j\in A} \pi_j T_{ji} q_i}{\sum_{j\in A} \sum_{k\notin A} \pi_j T_{jk} q_k} \\
&= C^{-1} \sum_{j\in A} \pi_j T_{ji} q_i \\
&= C^{-1} \pi_i \sum_{j\in A} T_{ij}(q_i - q_j)_+ \qquad (7.38)
\end{aligned}
$$

property that the flux of reactive trajectories out of A is the same as the one into B (see (7.14)) to get the third, and the detailed balance condition (7.1) as well as the property that $q_i = q_i - q_j = (q_i - q_j)_+$ if $j \in A$ (since $q_j = 0$ then) to get the fourth. Third,

$$
\begin{aligned}
\sum_{j\neq i} \pi_j^C T_{ji}^R &= C^{-1} \sum_{j\notin A\cup B} \pi_j T_{ji}(q_i - q_j)_+ \\
&= C^{-1} \sum_{j\notin A} \pi_j T_{ji}(q_i - q_j)_+ \\
&= C^{-1} \pi_i \sum_{j\notin A} T_{ij}(q_i - q_j)_+ \qquad (7.39)
\end{aligned}
$$

where we used that $\pi_j^C = 0$ if $j \in A\cup B$ in the first equality, the property that $(q_i - q_j)_+ = 0$ if $j \in B$ to augment the sum from $j \notin A\cup B$ to $j \notin B$ and get the second equality, and the detailed balance condition (7.1) to get the third. If we subtract the sum of (7.38) and (7.39) to (7.37) (i.e. if we subtract the right to the left hand-sides of (7.29)), we therefore arrive at

$$
\begin{aligned}
& C^{-1} \pi_i \left(\sum_{j\in A} T_{ij}(q_i - q_j)_+ + \sum_{j\notin A} T_{ij}(q_i - q_j)_+ \right. \\
& \qquad \left. - \sum_{j=1}^{N} T_{ij}(q_j - q_i)_+ \right) \\
&= C^{-1} \pi_i \sum_{j=1}^{N} T_{ij}(q_i - q_j) = 0 \qquad (7.40)
\end{aligned}
$$

where the last equality follows from $\sum_{j=1}^N T_{ij} = 1$ and (7.9). This concludes the verification of (7.29).

References

1. Berezhkovskii A, Hummer G, Szabo A (2009) Reactive flux and folding pathways in network models of coarse-grained protein dynamics. J Chem Phys 130(20):205,102. doi:10.1063/1.3139063
2. Bovier A, Eckhoff M, Gayrard V, Klein M (2000) Metastability and small eigenvalues in Markov chains. J Phys A 33(46):L447–L451
3. Bovier A, Eckhoff M, Gayrard V, Klein M (2002) Metastability and low lying spectra in reversible Markov chains. Commun Math Phys 228(2):219–255
4. E W, Vanden-Eijnden E (2006) Towards a theory of transition paths. J Stat Phys 123:503–523
5. E W, Vanden-Eijnden E (2010) Transition-path theory and path-finding algorithms for the study of rare events. Annu Rev Phys Chem 61:391–420. http://www.hubmed.org/fulltext.cgi?uids=18999998. doi:10.1146/annurev.physchem.040808.090412
6. Kirmizialtin S, Elber R (2011) Revisiting and computing reaction coordinates with directional milestoning. J Phys Chem A 115(23):6137

7. Metzner P, Schütte C, Vanden-Eijnden E (2006) Illustration of transition path theory on a collection of simple examples. J Chem Phys 125:084,110
8. Metzner P, Schütte C, Vanden-Eijnden E (2009) Transition path theory for Markov jump processes. Multiscale Model Simul 7(3):1192–1219
9. Noé F, Schütte C, Vanden-Eijnden E, Reich L, Weikl TR (2009) Constructing the equilibrium ensemble of folding pathways from short off-equilibrium simulations. Proc Natl Acad Sci USA 106(45):19,011–19,016
10. Norris JR (2004) Markov chains. Cambridge series in statistical and probabilistic mathematics. Cambridge University Press, Cambridge
11. Schütte C, Huisinga W (2003) Biomolecular conformations can be identified as metastable sets of molecular dynamics. Handb Numer Anal 10:699–744
12. Vanden-Eijnden E (2006) Transition path theory. In: Ferrario M, Ciccotti G, Binder K (eds) Computer simulations in condensed matter: from materials to chemical biology, vol 1. Springer, Berlin, pp 439–478

Understanding Protein Folding Using Markov State Models

8

Vijay S. Pande

8.1 Introduction

8.1.1 What Is Protein Folding?

Proteins play a central role in biology, acting as catalysts, sources of molecular recognition, structural elements, among many other roles. But before they can carryout these functions, proteins must first assemble themselves, or "fold," into their biologically functional or "native" state. As proteins are long chain molecules constituting of tens to thousands of amino acids, the fact that proteins fold to essentially a unique fold is a triumph of natural selection, considering the enormous amount of conformational entropy that folding must overcome.

This leads to the natural question: how does this process occur? Answering this question would be a resolution to one of the greatest outstanding questions in molecular biophysics. Moreover, as self-assembly is at the heart of many biological processes as well as the inspiration for modern nanotechnology, understanding how proteins fold could have an impact on many other fields. Finally, how proteins fold has emerged as a central part of the molecular mechanism of many diseases, such as Alzheimer's Disease or Huntington's Disease, where it is believed that proteins fold incorrectly—or misfold—as a critical part of the disease pathology.

8.1.2 Why Simulate Protein Folding?

The biophysical and biomedical aspects of protein folding has highlighted many challenges in understanding folding. First, we have found that even small changes, such as a mutation of a single amino acid, can lead to changes in how a protein folds or whether it even folds at all.

Studying protein folding experimentally is fraught with many challenges. In particular, we wish to understand folding at the atomic scale. This is particularly challenging for experimental methods, given the stochastic and heterogeneous nature of an ensemble of proteins folding in an experiment.

Therefore, this challenge suggests an opportunity—*simulating* protein folding is a means to gain new insight into this challenging problem. Ideally, simulations can shed new insight into how proteins fold, suggest new hypotheses, as well as suggest new interpretations of experiments. When tightly combined with experiments, simulations have the hope to address the ultimate question of how proteins fold. Below, we present recent advances deriving from MSM approaches.

8.1.3 Challenges in Simulating Protein Folding

There are three primary challenges in any simulation. First, is our model for interatomic interactions (i.e. the "force field") sufficiently accurate to predict the behavior of the system of

V.S. Pande (✉)
Stanford University, Stanford, CA 94305, USA
e-mail: pande@stanford.edu

G.R. Bowman et al. (eds.), *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology 797,
DOI 10.1007/978-94-007-7606-7_8, © Springer Science+Business Media Dordrecht 2014

interest. This has been a challenge for decades, but recent work has suggested that current force fields are sufficiently accurate for the quantitative prediction of a wide-range of bimolecular properties, but within certain known limitations [1] (see Fig. 8.2). Second, can one simulate the timescales relevant for the phenomena of interest? This has been a central challenge, since until recently, experimentally relevant timescales (microseconds to milliseconds) could not be reached with modern computer power using sufficiently accurate, atomically detailed models. Finally, a third challenge arises now that one can simulate long timescales with sufficiently accurate models: how can one use the resulting sea of data to gain some new insight? With the first two challenges now within reach for small, fast folding proteins, the third challenge of gaining new insight has come into the forefront.

As we discuss below, MSMs can aid in both the push for longer timescales as well as for the development of means to gain new insight from the resulting simulation data, even for more conventional simulation methods. Moreover, we will see that there are some potentially unique challenges associated with the construction of MSMs for protein folding. In particular, the unfolded state of a protein is huge (many conformations) and thus sampling it can be a challenge for the construction of an MSM. Also, the potential exponential growth in the number of relevant MSM microstates is also a potential challenge for MSM construction of protein folding, as simple arguments suggest that the number of structures grows exponentially length of the chain.

8.1.4 Unanswered Questions to Which MSMs Can Yield Insight

The end goal of a simulation of protein folding is the elucidation of the mechanism by which a protein folds, i.e. what are the steps a protein takes in assembling itself. There are several questions associated with this, including

1. **Does a protein fold in a single pathway or in many parallel paths?** This question is both relevant for the basic biophysics of folding, but also potentially relevant for the biochemistry of chaperonins, which catalyze the folding of some protein substrates. If folding occurs via a single, well-defined path, then catalysis could naturally take the form of the recognition of some well-defined transition state in the folding process. If folding occurs via multiple paths, then the resolution of the mechanism of catalysis is considerably more complex.
2. **Are there intermediates along the way to folding?** A common paradigm in the protein folding field is that simple proteins fold in a “two-state” manner, i.e. with just the unfolded and folded states and no intermediates in-between. Another way to rephrase this question is to study the separation of timescales between the slowest timescale (corresponding to folding) and the next slowest timescale; is this gap large compared to the folding time itself (for “two-state” system) or not? Simulations can help probe this hypothesis in a way that experiments cannot, due to their limitations of signal to noise of accumulating intermediates.
3. **Is the protein folding mechanism robust?** The entire discussion of a protein folding “mechanism” is hinged on the concept that such details are robust to subtle changes in the experimental environment (pH, temperature, co-solvents, etc.) as well as to variations in force fields used to simulate folding. Mechanistic properties which are robust have the hope to be comparable to experiment and free of variations caused by either experimental or computational variations. Moreover, the identification of non-robust properties is itself an important contribution as well. For example folding rates are robust (Fig. 8.2).

8.2 MSMs Have Allowed the Direct Simulation of Protein Folding

Given that the sampling at millisecond timescales has been possible for only two years (see Fig. 8.1), and analysis methodology is still immature, unambiguous scientific results learned from atomic simulation have thus far been modest. It will be

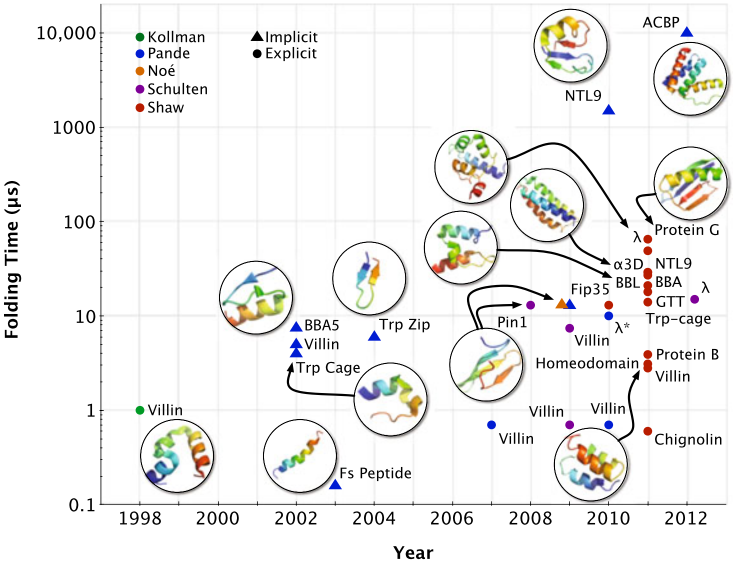


Fig. 8.1 The folding times accessible by simulation have increased exponentially over the past decade. Shown are all protein folding simulations conducted using unbiased, all-atom MD in empirical force-fields reported in the literature. Some folding times for the same protein differ, due to various mutations. For lambda marked with a (*), the longest timescale seen in that simulation, which was not the folding time, occurred on the order of 10 ms

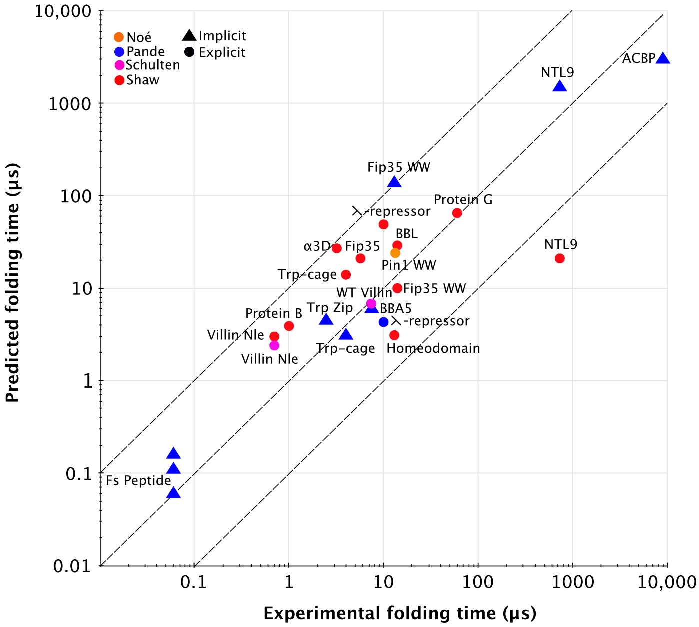


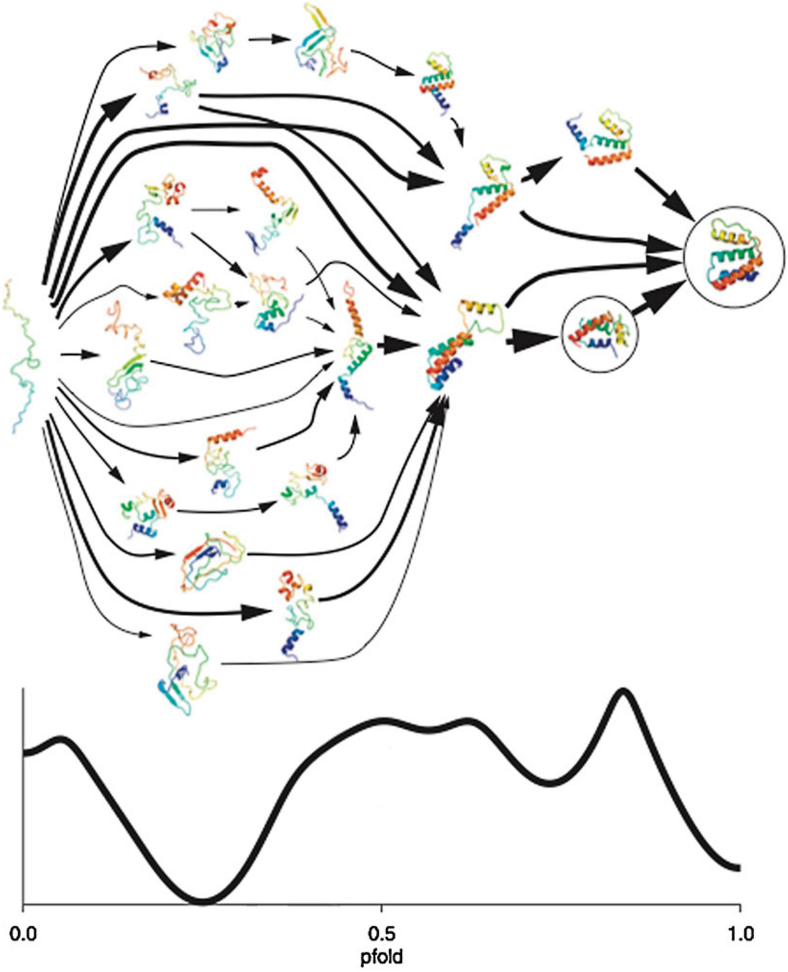
Fig. 8.2 Comparison of predicted and experimentally measured folding times. *Central dashed line* is perfect agreement, *outside lines* are within one order of magnitude of perfect agreement. Given that experimental folding times can vary over more than an order of magnitude given different conditions (temperature, salt, pH, etc.), as well as uncertainties associated with measuring experimental and simulated folding times, an order of magnitude agreement is close to the upper limit of accuracy one might expect

a major challenge in the next five years to turn advances in sampling and accuracy into scientific insight about how proteins fold.

Despite this relative immaturity, MSMs built from atomistic simulation have already begun to influence our view of protein folding. Detailed comparisons to experiment have been performed for MSMs of many specific proteins, including villin [3], NTL9 [9], WW domains [6–8], lambda repressor [5], and ACBP [10]. Universally accepted generalities amongst these specific protein simulations have not yet emerged, though some have been suggested, for instance that folding kinetics might be hub-like [4].

8.3 What Have We Learned?

With the ability to simulate proteins which fold on long timescales (milliseconds) and for non-trivial sizes (approaching 100 amino acids), MSM



Current Opinion in Structural Biology

Fig. 8.3 An MSM for the dynamics of ACBP, an 86 residue protein that folds on the 10 millisecond timescale. The size and long timescale (100 times longer than can be reached by traditional methods) make this calculation a landmark calculation in the simulation of protein folding. This diagram highlights the complexity of protein folding, showing the multiple paths that a protein can take to go from unfolded (*left*) to folded (*right*). The widths of the *arrows* denote how much flux each path carries

simulations have the hope to shed new insight into how do proteins fold. Below, we summarize three key results that have been seen so far.

8.3.1 Proteins Fold via Parallel Pathways Comprised of Metastable States

One of the principal results we have seen is that the mechanism of protein folding appears to be comprised of the interconversion of many metastable states. While an overall reaction may be dominated by a single slow timescale, leading to apparent "two-state" folding, more microscopically, folding looks much more detailed and complex. Where does this complexity go when examined experimentally? This complexity easily can be hidden when projected to a given reaction coordinate.

For example, consider Fig. 8.3, which shows an MSM for ACBP, which folds on the 10 millisecond timescale. While the MSM is complex, comprised of numerous states, ACBP appears to be only a three-state folder experimentally. However, when the MSM is projected to the fold reaction coordinate, we see that the MSM simplifies to look very much like a three-state folder [10]. This also opens the door to folding simulations helping predict new experiments which can more easily reveal this complexity.

8.3.2 These States Have Non-native Structural Elements: Register Shifts and Intramolecular Amyloids

With the illumination of these metastable states, one can interrogate the structural nature of these states to gain new insight into how proteins fold. One general property we find is that these states have an abundant degree of non-native structure. In particular, there are three forms of non-native structure which seems particularly common:

First, in beta sheet proteins, we often see states with register shifts. In these cases, the natural turn of a beta sheet is misplaced, leading to a different beta sheet structure. As turns can be formed in many places, sequences permit this reasonably easily in many cases [2].

Second, we often see elongated helices. In this case, a helix in a given intermediate state may be longer than in the native state. This is also natural given the commonality of helical propensity in amino acids, even in cases where the structure is not a helix natively.

Finally, and perhaps most strikingly, we have seen intramolecular amyloids,—cases where beta sheets form in alpha-helical proteins. This formation is not unlike the formation of intermolecular amyloids, where proteins spontaneously form beta sheet structures. Once a protein gets to be sufficiently long, we argue that it can act in the same fashion, *intramolecularly.*

8.3.3 The Connectivity of These States Suggest that the Native-State is a Kinetic Hub

Finally, how are these states "connected," i.e. that have non-zero conditional probabilities to go from one state to another? Addressing this question yields another aspect of the mechanism of protein folding. In MSM studies of protein folding, the native state has appeared to be a kinetic hub, i.e. there are many paths into the hub, compared with other states. This particular topology is common in other types of networks and suggest that the intrinsic kinetics of protein folding may have been evolutionarily optimized for kinetic properties including the kinetic network.

8.4 Next Challenges

MSM methods are sufficiently well developed to pursue many exciting applications. However, there is still a great deal of room for further methodological improvements. Here, we list a few of them.

1. **Longer timescales.** While MSMs have been able to simulate protein folding on the 10 millisecond timescale, proteins of interest for understanding how proteins fold can fold up to 1000× longer. This could present new challenges for MSM sampling.
2. **Larger proteins.** Similarly, the largest proteins studied so far are just under 100 amino acids, while proteins of interest can be up to 2× to 3× longer in length. Larger proteins may present new challenges for MSM building due to the potential exponential growth in the configurational space involved.
3. **Better state decomposition.** One way to handle these challenges is to determine better methods for building states, allowing for fewer states to be used and thus enabling the ability to build more complex MSMs.

In the coming years, we expect that these challenges as well will be reached, yielding both new insights into how proteins fold but also new MSM methods which could be broadly applicable to many other applications as well.

References

1. Beauchamp KA, Lin YS, Das R, Pande VS (2012) Are protein force fields getting better? A systematic benchmark on 524 diverse NMR measurements. J Chem Theory Comput 8(4):1409–1414
2. Beauchamp KA, McGibbon R, Lin YS, Pande VS (2012) Simple few-state models reveal hidden complexity in protein folding. Proc Natl Acad Sci USA 109(44):17,807–17,813
3. Bowman GR, Beauchamp KA, Boxer G, Pande VS (2009) Progress and challenges in the automated construction of Markov state models for full protein systems. J Chem Phys 131(12):124,101

4. Bowman GR, Pande VS (2010) Protein folded states are kinetic hubs. Proc Natl Acad Sci USA 107(24):10,890–10,895
5. Bowman GR, Voelz VA, Pande VS (2011) Atomistic folding simulations of the five-helix bundle protein (685). J Am Chem Soc 133(4):664–667
6. Lane TJ, Bowman GR, Beauchamp K, Voelz VA, Pande VS (2011) Markov state model reveals folding and functional dynamics in ultra-long MD trajectories. J Am Chem Soc 133(45):18,413–18,419
7. Morcos F, Chatterjee S, McClendon CL, Brenner PR, Lopez-Rendon R, Zintsmaster J, Ercsey-Ravasz M, Sweet CR, Jacobson MP, Peng JW, Izaguirre JA (2010) Modeling conformational ensembles of slow functional motions in Pin1-WW. PLoS Comput Biol 6(12):e1001, 015
8. Noe F, Schutte C, Vanden-Eijnden E, Reich L, Weikl TR (2009) Constructing the equilibrium ensemble of folding pathways from short off-equilibrium simulations. Proc Natl Acad Sci USA 106(45):19,011–19,016
9. Voelz VA, Bowman GR, Beauchamp K, Pande VS (2010) Molecular simulation of ab initio protein folding for a millisecond folder NTL9(1-39). J Am Chem Soc 132(5):1526–1528
10. Voelz VA, Jager M, Yao S, Chen Y, Zhu L, Waldauer SA, Bowman GR, Friedrichs M, Bakajin O, Lapidus LJ, Weiss S, Pande VS (2012) Slow unfolded-state structuring in Acyl-CoA binding protein folding revealed by simulation and experiment. J Am Chem Soc 134(30):12,565–12,577

Understanding Molecular Recognition by Kinetic Network Models Constructed from Molecular Dynamics Simulations

9

Xuhui Huang and Gianni De Fabritiis

9.1 Introduction

Molecular recognition is the process by which macromolecules selectively interact. Virtually all biological phenomena depend in some way on specific molecular recognition, and thus an understanding of the process is of central importance in the study of biology. One critically important factor is that proteins exist as a statistical ensemble of conformers, which are transitory excited-states (having higher free energy) in the protein in normal solvated conditions; however, these excited states can become preferred upon binding, by shifting the equilibrium distribution towards them. For example, a thermally-accessible conformer that is 2 kBT higher in free energy would exist in just 13 % of the molecules in solution (according to Boltzmann probability), yet upon binding could become the most favored state.

There are two popular models aiming to explain the mechanisms of molecular recognition based on a dual dynamic mechanism: "induced-fit" (see reaction R4–R3 in Fig. 9.1) and "conformational selection" (see reaction R1–R2 in Fig. 9.1). In the induced fit model introduced by Koshland [1], the apo protein only exists in the unbound form and the interactions with the ligand induce the protein to reach the bound state. In the conformational selection model [2–8], the protein's intrinsic dynamics may lead it to sample not only the unbound state but also the minor bound state. The ligand may then selectively bind to the pre-existing bound conformation and further increase its population. These two models are not mutually exclusive and both mechanisms may play a role as binding and folding are both search processes over a rugged free energy surface. For example, by binding to protein A, protein B may be stabilized in an excited conformation B* which can facilitate binding to other proteins or ligands determining a cellular signaling cascade.

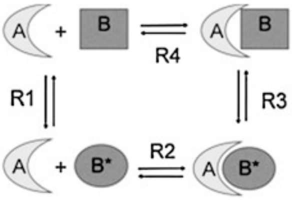


Fig. 9.1 Conformational selection (R1–R2) v.s. induced fit (R3–R4). A schematic diagram for the two popular binding mechanisms are displayed

Many molecular recognition processes involve significant conformational changes of one or both binding partners. For example, Periplasmic Binding Proteins (PBPs) can undergo a large-scale

X. Huang (✉)
Department of Chemistry, Division of Biomedical Engineering, Center of Systems Biology and Human Health, Institute for Advance Study, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong
e-mail: xuhuihuang@ust.hk

G. De Fabritiis
Computational Biophysics Laboratory (GRIB-IMIM), Universitat Pompeu Fabra, Barcelona Biomedical Research Park (PRBB), C. Doctor Aiguader 88, 08003 Barcelona, Spain

G.R. Bowman et al. (eds.), *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology 797,
DOI 10.1007/978-94-007-7606-7_9, © Springer Science+Business Media Dordrecht 2014

hinge bending motion between two domains from an open to a closed state upon substrate binding [9–12]. In these systems, the interplay between protein structure and dynamics upon substrate binding may ultimately determine the binding mechanisms. Computer modeling has been shown to be a valuable approach to complement experimental techniques to reveal the chemical details of molecular recognition mechanisms. Markov state models (MSMs) are kinetic network models that hold great potential for understanding the mechanisms of molecular recognition events from computer simulations.

Although MSMs have been successfully applied to study conformational dynamics of one-body systems such as a single protein or RNA [13–17], constructing MSMs to investigate the protein-ligand binding process is challenging because the ligand dynamics normally occurs on two significantly different timescales due to its interactions with the protein. In particular, a ligand's dynamics tend to be very slow when interacting with a protein, but ligands typically diffuse very quickly in solution. Therefore, the standard methods for constructing MSMs through a uniform clustering at a single resolution are often insufficient for properly describing ligand binding. In this chapter, we will review some recent progress on constructing MSMs for two-body systems associated with large conformational changes where the ligand dynamics occurs at a mixture of different resolutions.

9.2 Methodology

9.2.1 Projected Dynamics MSMs

The use of reaction coordinates to project the high dimensional space of a molecular systems into a small dimensional space has been used for many years, especially in the setting of biased dynamics [18]. These biased dynamics schemes offer the advantage of speeding up the global dynamics provided that the reaction coordinates is a good one (no other degree of freedom is slower). A good reaction coordinate also provides a way to compute realistic energetic maps of the phenomena.

A different approach is to use MSM to analyze a set of unbiased trajectories using a low dimensional space to build the Markov model. In this case, the reaction coordinate does not have to be perfect as the dynamics is only projected into this space but the kinetics are well recovered provided that the runs are long enough. This is the approach for instance of Ref. [19], in which the binding pathway of a small molecule is constructed by using a simple reaction coordinate, the three-dimensional position of one of its atoms.

9.2.2 Automated Methods for Constructing MSMs for One-Body Systems

In many studies, MSMs are constructed by grouping conformations into a number of metastable states and then counting the transitions between these states without projecting the dynamics onto certain reaction coordinates. Automated methods based on a splitting-and-lumping scheme have been developed to construct MSMs for one-body systems [20, 21]. Since these methods have been discussed in detail in other chapters, we briefly review the general procedure here: first, a geometric clustering is applied to divide the MD conformations into a large number of small clusters. This assumes that conformations within the same cluster are kinetically similar because of their structural similarity. Next, clusters that can interconvert quickly are grouped together into the same metastable state to construct an MSM model. Finally, we can calculate thermodynamic and kinetic properties of interest if the model is Markovian.

9.2.3 Constructing MSMs for Two-Body Systems

The above splitting-and-lumping algorithm for one-body systems is often unideal for two-body systems because the dynamics in these systems occur at a mixture of different timescales due to the interactions between the binding partners (see

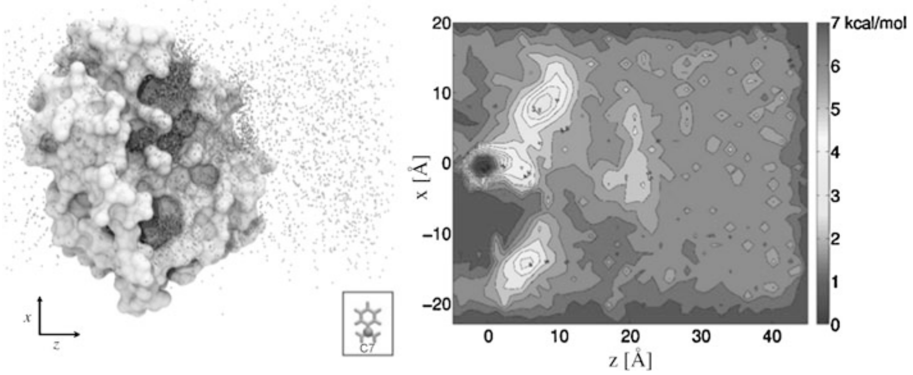


Fig. 9.2 The spatial positions visited by diffusion of Benzamidine show clearly few metastable states, but only a MSM analysis of the trajectories can recover the free energy profile in three-dimension (here projected in two dimensions for clarity). These figures are adapted from [19]

Fig. 9.5). For example, the ligand diffuses freely in the solvent in the un-bound state. While in the bound state, the ligand forms stable interactions with the protein and its dynamics is slow and strongly correlated with the protein conformation. A kinetically-relevant, uniform clustering at a single resolution as in the splitting-and-lumping algorithm is often difficult to achieve for these two-body systems. If the resolution of the clustering is too low, one cannot split enough in the region where the ligand binds to the protein. On the other hand, if the clustering resolution is too high, there may not be a sufficient number of conformations in each cluster in the unbound region (e.g. many clusters in the unbound region end up containing a single conformation).

In order to address this issue, Silva et al. [22] have performed independent clustering at two different resolutions: a high-resolution clustering (or larger number of clusters) on conformations where the ligand binds to the protein and a low-resolution clustering (or smaller number of clusters) on conformations where the ligand diffuses in solution. Kinetic lumping was then used within each region to generate a set of metastable states. Finally, the two sets of metastable states were combined into a single MSM. In this algorithm, a hard distance cut-off (5 Å between the ligand and protein) is set to separate the fast and slow motion regions for the ligand. This algorithm was shown to be useful for dealing with protein-ligand binding systems, but it may introduce errors on the boundary between the two regions due to the hard distance separation.

9.3 Example Trypsin-Benzamidine Binding

In this section, we use the molecular recognition process of trypsin-benzamine as an exemplary case of rigid binding. In Ref. [19], a kinetic model for the binding process of serine protease beta-trypsin inhibitor benzamidine was obtained from extensive high-throughput all-atom MD simulations using the ACEMD [23] software on the GPUGRID distributed computing network [24].

The analysis of 495 trajectories of free diffusion of benzamidine around trypsin each of 100 ns of length lead to 187 trajectories (37 %) which successfully recovered the bound pose in the binding pocket with an RMSD compared to the crystal structure of less than 2 Å. Several clusters of benzamidine on the surface of trypsin can be observed in Fig. 9.2, which indicates a rather more complex pathway of binding than expected instead of a of simple pathway directly from the bulk. Some trajectories reach the bound crystallographic pose just after 10–15 ns of simulation while some reach the binding pocket only after 90 ns, but the majority of the trajectories do not enter the binding site within 100 ns, as should

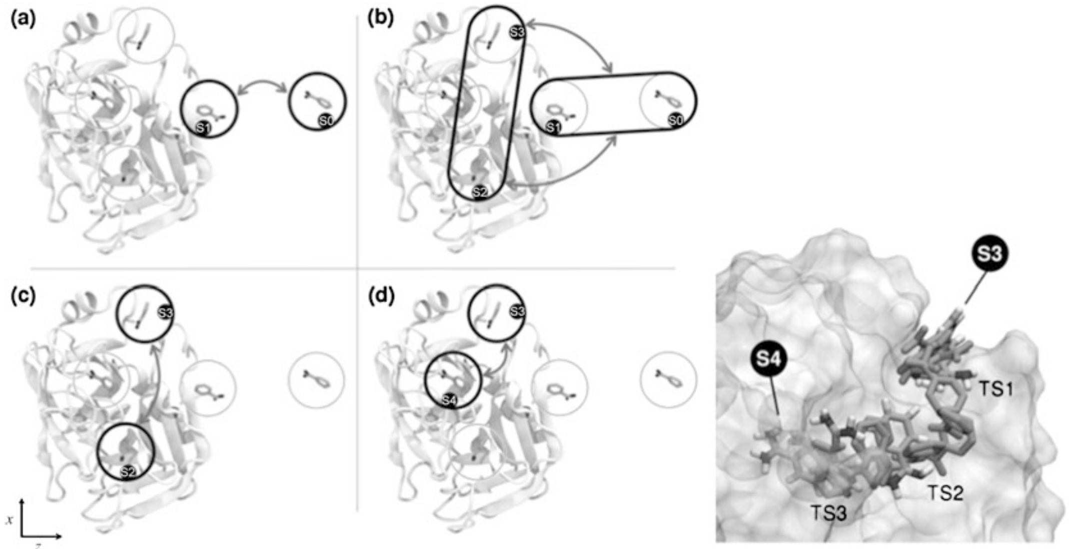


Fig. 9.3 Main binding modes for benzamidine on trypsin. (**a**) The encounter of the ligand with the protein, (**b**) binding to two secondary binding sites, (**c**) exchange between the secondary binding sites, (**d**) final pathway of binding into the catalytic site, which shows a curious rolling of the ligand on the surface of the protein as the most probable path. These figures are adapted from [19]

be expected in such a short time frame. Nevertheless, these simulations provide enough data to carry out a detailed quantitative analysis of the binding pathway.

An aggregate of 50 microseconds of trajectory data have been used to construct a MSM of the binding process of benzamidine to trypsin. The MSM was built using the three-dimensional reaction coordinate defined by the coordinates of the C7-atom of benzamidine (Fig. 9.2). A projection in two dimensions of the energetic profile is also shown highlighting the secondary and the main binding sites (Fig. 9.2). This surface is recovered directly by solving the stationary distribution of the MSM. Using a formula derived in [19], it is possible to compute directly from the energetic map the standard free energy of binding of the ligand of approximately 5.2 kcal/mol compared to an experimental one of 6.2 kcal/mol. A kinetic model can also be built to measure on and off rates which compare well with experiments [19].

An analysis of the slowest eigenvectors of the MSM also allows the reconstruction of the binding pathway. Considering the slowest modes, we see transitions from site S0 to S1 (Fig. 9.3a) and collectively from sites S0/S1 to sites S3/S4 (Fig. 9.3b), corresponding to the diffusion of the ligand from bulk to the first structural contact with the protein. At a slower timescale, there are transitions between sites S2 and S3 (Fig. 9.3c). Site S2 is a secondary binding pocket but not directly involved in the binding pathway. Finally, the rate-limiting step of the process is the transition to the bound site S4 (Fig. 9.3d) and preferentially coming from S3 interestingly rolling on the surface of the protein.

The case of Trypsin-Benzamidine represents a best case scenario where both the ligand and protein are relatively inflexible. While the methodology is not limited to this case, more flexible ligands would require substantially more time to bind. Conformational changes in the protein could also forbid binding all-together until certain loops open. All these factors imply that while the current methodology is very promising, more work is necessary in order to efficiently resolve complex molecular recognition processes.

9.4 Example LAO Protein Binding

In this section, we use the Lysine-, Arginine-, Ornithine-binding (LAO) protein as an example to demonstrate the power of MSMs for studying protein-ligand binding mechanisms. The LAO protein is one of the Periplasmic Binding Proteins (PBPs), which is an attractive class of systems for studying the mechanisms of molecular recognition events [25, 26]. With more than 100 crystal structures available, different PBPs can bind to a large variety of substrates including amino acids, sugars, small peptides, etc. However, all PBPs share similar tertiary structures containing two globular domains connected by a hinge region with the binding site at the domain-domain interface. They can undergo a large-scale hinge bending motion from an open to a closed state upon ligand binding (see Fig. 9.4). These features make PBPs a good model system to investigate the coupling between ligand binding and protein conformational changes.

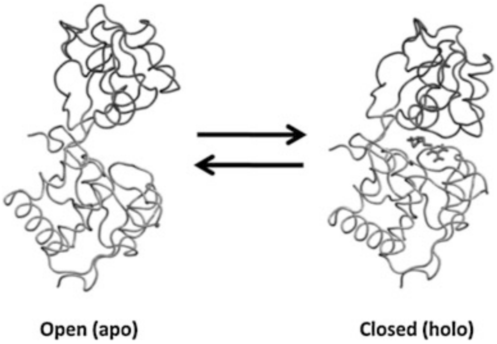


Fig. 9.4 The Lysine-, Arginine-, Ornithine-binding (LAO) Protein undergoes large domain displacement from the open (*left*, PDB id: 2LAO) to the closed (*right*, PDB id: 1LAF) state upon the binding of Arginine (sticks). This figure is reproduced from [22]

MD simulations have shown that the ligand dynamics in the LAO system indeed displays a mixture of different timescales. Silva et al. [22] performed a set of sixty-five 200-ns MD simulations of the ligand Arginine binding to the LAO protein. From these simulations, they calculated the ligand rotational autocorrelation functions for three conformational states: unbound state, encounter complex, and bound state. As shown in Fig. 9.5, the ligand can rotate quickly when it undergoes free diffusion in the solvent, but the ligand rotation is largely restrained when it binds to the protein. Therefore, when they later constructed MSMs from these MD simulations, they performed structural clustering at two different resolutions in the "splitting" stage of the splitting-and-lumping algorithm. In the low-resolution (or fewer clusters) region, the dynamics of the ligand is fast, so that only the center of mass motion of the ligand is considered. In the high-resolution (or more clusters) region, the dynamics of the ligand is constrained due to its strong interactions with the protein, so that motions of all ligand heavy atoms are considered. Finally, they performed kinetic lumping at each region to generate a set of metastable states, and combine them into a single 54-state MSM. This MSM is shown to reproduce the structure of the bound state, experimental binding free energy, and association rate with reasonable accuracy [22].

MSMs [22] suggest a two-step binding mechanism for the LAO protein with a number of intermediate states and parallel binding pathways (see the ten most probable binding pathways predicted by the MSM as shown in Fig. 9.6). In the first step, the ligand binds to the protein to form an encounter complex. In the encounter complex state, the protein is partially closed and only weakly interacts with the ligand. RMSD analysis shows that the structure of individual protein domains in the encounter complex is very similar to those in the unbound and bound X-ray structures (with RMSD mostly <2 Å). Therefore the conformational change from either unbound or bound state to the encounter complex conformation may be achieved through domain rigid body rotations. All major pathways pass through the encounter complex state, which serves as a gatekeeper for binding. This process is dominated by conformational selection. In the second step, the protein-ligand interactions induce conformational changes to reach the bound state.

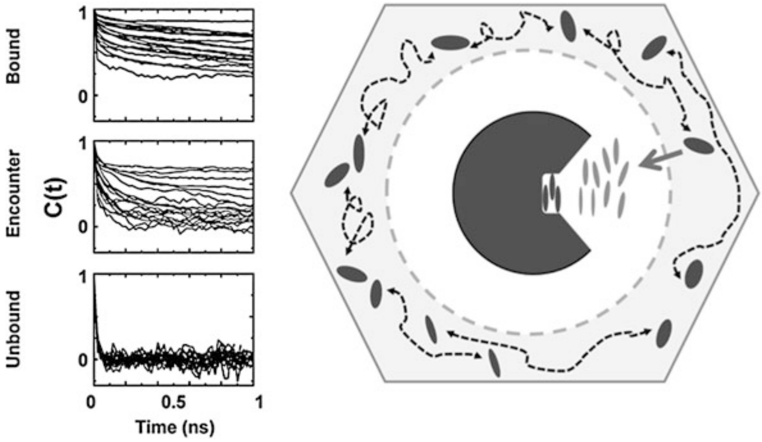


Fig. 9.5 A figure demonstrating the challenge for constructing kinetic network models for two-body systems where the ligand dynamics occur at a mixture of different timescales as shown by the rotational autocorrelation functions of the ligand in the LAO protein system. The decay of ligand rotational autocorrelation functions is much faster in the unbound state (*inside hexagon* in *right panel* and *bottom* of the *left panel*) than the encounter complex (*inside dashed circle* in *right panel* and *middle* of the *left panel*) and the bound state (in *binding pocket* in *right panel* and *top* of the *left panel*). On the *right panel*, a schematic figure illustrates the ligand positions in different states. This figure is reproduced from [22]

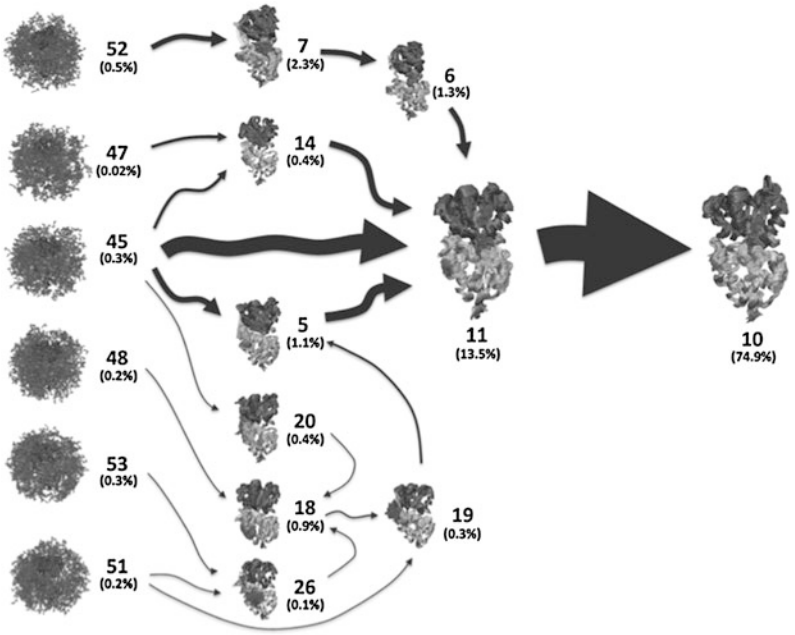


Fig. 9.6 Ten highest flux binding pathways from the unbound states (*left*) to the bound state (*right*) of the LAO protein are superimposed. The arrow sizes are proportional to the flux. State numbers and their equilibrium population calculated from a 54-state Markov State Model are also shown. This figure is reproduced from [22]

9.5 Discussion and Future Perspective

One major advantage of MSMs is that they can dissect atomistic details of molecular recognition. For instance, Silva et al. [22] have observed roles for both conformational selection and induced fit in LAO binding, as well as an encounter complex intermediate state. Recent NMR studies by Tang et al. [27] have also suggested the duality of conformational selection and induced fit for the binding of PBPs. Using NMR with paramagnetic relaxation enhancement (PRE), they have identified a minor (5 %) partially closed form in equilibrium with the major open form for another PDB, the maltose-binding protein [27]. Based on these observations, they proposed that this partially closed state may be available for the binding of the ligand through conformational selection and this binding could then facilitate the transition to the bound state via the induced fit mechanism. This model was proposed mainly based on experiments in the absence of the ligand. MSMs have the advantage that they can directly observe the interplay between protein conformational changes and ligand dynamics from simulations of ligand binding at atomic resolution. The other main advantage of MSMs is that they can help bridge the timescale gap between the experiments and atomistic MD simulations. For many two-body systems such as protein-ligand binding and protein-protein interactions, the association timescales (millisecond or longer) would be too long to be reached by straightforward atomistic MD simulations. MSMs built from many independent microsecond simulations, however, have already proven capable of capturing protein-folding events that occur at tens of milliseconds timescales [16]. They can thus likely be applied to study slow protein-ligand binding events too. For the LAO protein discussed above, the timescale is fast enough to observe multiple binding and unbinding events within our sixty-five 200-ns simulations. Even for this case, it would still be challenging to extract a complete picture of the binding mechanism from a single long simulation, because one would need this single simulation to be at least tens of microseconds long so that many binding/unbinding transitions occur (the average transition time from the unbound to the bound state is 2 microseconds). While such a trajectory could be run, scaling to study even slower events (i.e. at millisecond timescales) would not be possible.

In the future, new algorithms are needed for better integrating different timescales of ligand dynamics when constructing the kinetic network models. Silva et al. [22] used a hard distance cut-off (5 Å between the ligand and protein) to separate the slow and fast motion regions for the ligand, and then performed independent kinetic lumping for each before recombining the two sets of metastable states into a single MSM. As we discussed above, this algorithm may introduce errors on the boundary between the two regions due to this sharp distance cut-off. One potential way to avoid this problem is to directly integrate the geometric “splitting” and kinetic “lumping” steps during model construction. This may require the consideration of both the structural similarity and the kinetic connectivity when performing the clustering. Moreover, kinetic network models containing nodes at transition states could also greatly aid in understanding the mechanisms of molecular recognition events, even though these models are no longer Markovian. They are particularly useful for systems where sufficient sampling can already be achieved by straightforward MD simulations.

References

1. Koshland DE (1958) Application of a theory of enzyme specificity to protein synthesis. Proc Natl Acad Sci USA 44(2):98–104
2. Kumar S, Ma B, Tsai CJ, Sinha N, Nussinov R (2000) Folding and binding cascades: dynamic landscapes and population shifts. Protein Sci 9(1):10–19
3. Ma B, Kumar S, Tsai CJ, Nussinov R (1999) Folding funnels and binding mechanisms. Protein Eng 12(9):713–720
4. Ma B, Shatsky M, Wolfson HJ, Nussinov R (2002) Multiple diverse ligands binding at a single protein site: a matter of pre-existing populations. Protein Sci 11(2):184–197
5. Tsai CJ, Kumar S, Ma B, Nussinov R (1999) Folding funnels, binding funnels, and protein function. Protein Sci 8(6):1181–1190
6. Tsai CJ, Ma B, Nussinov R (1999) Folding and binding cascades: shifts in energy landscapes. Proc Natl Acad Sci USA 96(18):9970–9972
7. Arora K, Brooks CL (2007) Large-scale allosteric conformational transitions of adenylate kinase appear to involve a population-shift mechanism. Proc Natl Acad Sci USA 104(47):18496–18501
8. Bahar I, Chennubhotla C, Tobi D (2007) Intrinsic dynamics of enzymes in the unbound state and relation to allosteric regulation. Curr Opin Struct Biol 17(6):633–640
9. Oh BH, Ames GF, Kim SH (1994) Structural basis for multiple ligand specificity of the periplasmic lysine-, arginine-, ornithine-binding protein. J Biol Chem 269(42):26323–26330

10. Ames GF (1986) Bacterial periplasmic transport systems: structure, mechanism, and evolution. Annu Rev Biochem 55:397–425
11. Pang A, Arinaminpathy Y, Sansom MS, Biggin PC (2005) Comparative molecular dynamics-similar folds and similar motions? Proteins 61(4):809–822
12. Stockner T, Vogel H, Tieleman D (2005) A salt-bridge motif involved in ligand binding and large-scale domain motions of the maltose-binding protein. Biophys J 89(5):3362–3371
13. Buchete NV, Hummer G (2008) Coarse master equations for peptide folding dynamics. J Phys Chem B 112(19):6057–6069
14. Noe F, Schutte C, Vanden-Eijnden E, Reich L, Weikl TR (2009) Constructing the equilibrium ensemble of folding pathways from short off-equilibrium simulations. Proc Natl Acad Sci USA 106(45):19011–19016
15. Huang X, Yao Y, Bowman GR, Sun J, Guibas LJ, Carlsson G, Pande VS (2010) Constructing multi-resolution markov state models (msms) to elucidate RNA hairpin folding mechanisms. Pac Symp Biocomput, 228–239
16. Voelz VA, Bowman GR, Beauchamp K, Pande VS. Molecular simulation of ab initio protein folding for a millisecond folder NTL9(1-39). J Am Chem Soc 132(5):1526–1528
17. Morcos F, Chatterjee S, McClendon CL, Brenner PR, Lopez-Rendon R, Zintsmaster J, Ercsey-Ravasz M, Sweet CR, Jacobson MP, Peng JW, Izaguirre JA (2010) Modeling conformational ensembles of slow functional motions in Pin1-WW. PLoS Comput Biol 6(12):e1001015
18. Buch I, Sadiq SK, De Fabritiis G (2011) Optimized potential of mean force calculations for standard binding free energies. J Chem Theory Comput 7(6):1765–1772
19. Buch I, Giorgino T, De Fabritiis G (2011) Complete reconstruction of an enzyme-inhibitor binding process by molecular dynamics simulations. Proc Natl Acad Sci USA 108(25):10184–10189
20. Bowman GR, Huang X, Pande VS (2009) Using generalized ensemble simulations and Markov state models to identify conformational states. Methods 49(2):197–201
21. Huang X, Bowman GR, Bacallado S, Pande VS (2009) Rapid equilibrium sampling initiated from nonequilibrium data. Proc Natl Acad Sci USA 106(47):19765–19769
22. Oh BH, Pandit J, Kang CH, Nikaido K, Gokcen S, Ames GF, Kim SH (1993) Three-dimensional structures of the periplasmic lysine/arginine/ornithine-binding protein with and without a ligand. J Biol Chem 268(15):11348–11355
23. Harvey MJ, Giupponi G, De Fabritiis G (2009) ACEMD: accelerating biomolecular dynamics in the microsecond time scale. J Chem Theory Comput 5(6):1632–1639
24. Buch I, Harvey MJ, Giorgino T, Anderson DP, De Fabritiis G (2010) High-throughput all-atom molecular dynamics simulations using distributed computing. J Chem Inf Model 50(3):397–403
25. Bucher D, Grant BJ, Markwick PR, McCammon JA (2011) Accessing a hidden conformation of the maltose binding protein using accelerated molecular dynamics. PLoS Comput Biol 7(4):e1002034
26. Quiocho FA, Ledvina PS (1996) Atomic structure and specificity of bacterial periplasmic receptors for active transport and chemotaxis: variation of common themes. Mol Microbiol 20(1):17–25
27. Tang C, Schwieters CD, Clore GM (2007) Open-to-closed transition in apo maltose-binding protein observed by paramagnetic NMR. Nature 449(7165):1078–1082

Markov State and Diffusive Stochastic Models in Electron Spin Resonance 10

Deniz Sezer and Benoît Roux

10.1 Introduction

Continuous-wave electron spin resonance (cw-ESR) experiments combined with site-directed spin labeling (SDSL) techniques are a rich source of information about the local structure and dynamics of biomolecules in native-like environments [6, 10, 16]. Partly owing to its high sensitivity and the relative ease with which biomolecules can be systematically labeled at almost any desired location (e.g., by introducing cysteine mutations in proteins), ESR plays an increasingly important role in studies of proteins [26], nucleic acids [27], and membrane systems [37]. An exciting aspect of cw-ESR is the extreme sensitivity of the technique to the details of the dynamical processes occurring at the molecular level. Cw-ESR spectra are sensitive to dynamics over a broad range of time scales, from hundreds of picoseconds to tens of nanoseconds, depending on the strength of the constant magnetic field or the corresponding microwave frequency. While the slower end of this temporal range is routinely covered by lower fields/frequencies, like the most commonly used X-band, the faster end has recently become accessible as a result of vigorous developments in high-field/high-frequency ESR [5, 13, 20]. However, despite the great progress with the experimental cw-ESR and SDSL techniques, the development of a comprehensive set of theoretical methods able to account quantitatively for all the spectral features in terms of atomic models remains challenging.

A qualitative interpretation of cw-ESR spectra is relatively straightforward when the overall purpose of an experimental investigation is to delineate broad structural features of a protein. For instance, secondary structural elements can be mapped out by systematically scanning the protein by SDSL, and comparing the resulting spectra. Information about the overall position of the subunits of a membrane protein relative to the lipid bilayer can also be obtained with the aid of additional water- or membrane-soluble paramagnetic agents. The situation is very different when detailed structural and dynamic information about a system is sought, and the specific features of the spectral line shapes need to be properly interpreted. In this case, understanding the link between the conformational dynamics of the spin label and the observed spectra becomes of paramount importance. Unambiguously inferring the underlying molecular processes from spectra is difficult, however, even when using high-field/high-frequency ESR. The task becomes particularly challenging for spin-labeled biomolecules because they undergo complex motions occurring over a multitude of overlapping time scales. The internal dynamics of the spectroscopic reporter, i.e., the spin label, adds

D. Sezer (✉)
Faculty of Engineering and Natural Sciences, Sabancı University, Istanbul, Turkey
e-mail: dsezer@sabanciuniv.edu

B. Roux
Department of Biochemistry and Molecular Biology, The University of Chicago, Chicago, USA
e-mail: roux@uchicago.edu

G.R. Bowman et al. (eds.), *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology 797,
DOI 10.1007/978-94-007-7606-7_10, © Springer Science+Business Media Dordrecht 2014

another layer of complexity that must also be accounted for.

Currently, the most established theoretical/computational method for quantifying cw-ESR spectra is based on the stochastic Liouville equation (SLE). Spectral line shapes are simulated by generating a matrix representation for the Liouville operator corresponding to the relevant spin dynamics coupled to the hypothesized stochastic processes [19, 46]. Among the various stochastic models, most sophisticated are the MOMD (microscopic order macroscopic disorder) [38] and SRLS (slowly relaxing local structure) [40, 41] models, in which the dynamics of the spin label relative to the magnetic fields applied in the laboratory is described as a collection of nested rotational transformations evolving in a diffusive manner (see Sect. 10.2.2). The necessary matrix diagonalization and simulation of cw-ESR spectra is done in a very efficient way by the SLE numerical solver—a suite of programs for simulating and fitting slow-motional ESR spectra—developed in the laboratory of Jack Freed [7]. The outcome of an SLE analysis is typically a small set of phenomenological parameters associated with the rate and the range of motion of the spin label in some local mean-field potential. In practice it is necessary to keep the internal structure of the phenomenological stochastic model relatively simple to maintain the size of the problems within approachable numerical limits for the SLE solver. To avoid this issue, an alternative route is to bypass the construction of SLE altogether and simulate cw-ESR spectra directly from stochastic trajectories [14, 15, 17, 44, 52, 55]. The most important limitation of such phenomenological approaches, which thwarts developing insight about the molecular factors reported by the details of the spectra, remains the difficulty to understand the correspondence between the fitted mean-field parameters of the stochastic model and the underlying atomistic motions.

In this regard, all-atom molecular dynamics (MD) simulations with explicit solvent offer, perhaps, one of the most promising approaches for calculating cw-ESR spectra directly without extra assumptions about phenomenological models [4, 12, 30, 53]. In principle, MD simulations provide a "virtual route" to unambiguously link the atomistic dynamics to the experimentally observed cw-ESR spectra. Such an approach is, after all, routinely used to analyze and interpret results from nuclear magnetic resonance (NMR) [34]. However, a straightforward all-atom MD strategy for calculating ESR spectra remains challenging, even with current computational resources [47, 48]. The reason for this is both simple and complex. In NMR, the spin of the nuclei are only weakly coupled to their surrounding, therefore most magnetic relaxation coefficients can be calculated accurately from nanosecond trajectories using Redfield theory [43], which follows from time-dependent perturbation theory in quantum mechanics carried to second order. In contrast, the coupling of an electron spin to its environment is almost three orders of magnitude stronger than the coupling of nuclear spins. As a result, in most ESR experiments with spin-labeled macromolecules a perturbative treatment is not applicable. For this reason, the quantal degrees of freedom must be propagated for hundreds of nanoseconds to calculate spectra with a reasonable resolution of detail. Paradoxically, all issues of statistical convergence are not immediately resolved even when one trajectory is sufficiently long to allow the spin label to explore all accessible configurations and lose its correlation. The problem is that a large number of independent "samples" are necessary for a reliable estimate of the ESR spectrum. To clarify this point, it is useful to consider that the effective error of an ensemble average normally goes as $\sigma^2/\sqrt{N}$, where N is the number of independent samples and σ^2 is the intrinsic variance of the signal. When the averaging process is carried out from a trajectory of total length $\mathscr{T}$ the number of independent samples is typically understood as $\mathscr{T}/\tau_c$, where τ_c is the correlation time. In the case of biomolecular ESR, τ_c can be on the order of tens or hundreds of nanoseconds, which does not correspond to exceedingly long trajectories with current standards. However, due to the strong coupling of the electron spin to the orientation of the nitroxide label, the effective σ^2 of cw-ESR spectra in the presence of such slow motions is very large. As a consequence, small changes in one classical trajectory of length T can lead to considerable variations in the resulting spectrum. For this reason, one needs a very large N to get a reliable spectrum. If the averaging process is carried out from an ensemble, then a large number of sample trajectories is required. Alternatively, if the averaging process is carried out from a single trajectory then the latter has to be much longer than a single correlation time.

With the aim of establishing a flexible computational formalism for simulating cw-ESR spectra, we have developed a framework that circumvents these difficulties. Relevant information about the spin-label dynamics is first extracted from (relatively short) MD trajectories and mapped onto a Markov state model (MSM). Extremely long and computationally inexpensive stochastic state-hoping trajectories are then generated, while global tumbling of the macromolecule can be incorporated via a rotational diffusion model. Finally, the quantal degrees of freedom can be propagated along these trajectories to calculate cw-ESR spectra accurately [47–49]. The feasibility of this approach was demonstrated in Ref. [50], where it was successfully applied to the simulation of multifrequency spectra of spin-labeled T4 Lysozyme [56]. In this chapter, we review the main theoretical and practical elements of the method.

10.2 General Overview

Before discussing the details of the approach, we start by giving a quick overview of the problem. To this end, we first introduce the quantum mechanical aspect of the problem (Sect. 10.2.1), then look at various models of the classical molecular motion (Sect. 10.2.2), and finally combine the two by illustrating the effect of molecular tumbling on cw-ESR spectra (Sect. 10.2.3). This structure reflects the overall organization of the chapter: Sect. 10.3 is concerned with the MSM modeling of the classical dynamics of a protein-attached spin label, Sect. 10.4 presents the details of the quantal spin dynamics, and Sect. 10.5 discusses the combination of the two aspects. Simulations of multifrequency cw-ESR spectra of spin-labeled T4 Lysozyme (Sect. 10.5.3) demonstrate the power of this novel methodology in practical applications.

10.2.1 The Nitroxide Spin Hamiltonian

Cw-ESR spectroscopy consists in measuring the transverse magnetization from a bulk system in which nitroxide spin labels have been introduced. A nitroxide has an unpaired electron of spin $S = 1/2$ and a nitrogen nucleus with spin $I = 1$ (for ^{14}N) or $I = 1/2$ (for ^{15}N). The spin Hamiltonian of the nitroxide spin label, accounting for the interactions of the electron and nuclear spins, is

$$\hat{H}(t) = |\gamma_e| \big[\mathbf{B} \cdot \mathbf{G}(t) \cdot \hat{\mathbf{S}} + \hat{\mathbf{I}} \cdot \mathbf{A}(t) \cdot \hat{\mathbf{S}} \big] \tag{10.1}$$

in units of angular frequency. (Bold letters are used to denote vectors and matrices in physical space, Hilbert space operators are indicated with a caret.) Here, $\gamma_e = -1.76086 \times 10^{-2}$ rad ns^{-1} G^{-1} is the electron gyromagnetic ratio, $\hat{\mathbf{S}}$ and $\hat{\mathbf{I}}$ are the electron and nuclear spin operators, $\mathbf{A}$ is the hyperfine tensor (expressed in units of magnetic field) and

$$\mathbf{G}(t) \equiv \mathbf{g}(t)/g_e \tag{10.2}$$

is the electronic g tensor, $\mathbf{g}$, divided by the free electron g-factor, $g_e = 2.0023193$. The electron Zeeman interaction and the electron-nucleus hyperfine interaction are explicitly accounted for in the Hamiltonian (10.1). In contrast, the nuclear Zeeman and quadrupolar (in the case of $I = 1$) interactions have been neglected.

To a very good approximation, the coupling tensors $\mathbf{G}$ and $\mathbf{A}$ are diagonal in the same nitroxide-fixed coordinate frame $\mathtt{N}$. The standard choice of axes in $\mathtt{N}$ with respect to the nitroxide structure is shown in Fig. 10.1. Typical magnetic tensor values for nitroxide spin labels on biomolecules are

$$\begin{aligned} \mathbf{g}^{\mathrm{N}} &= \mathrm{diag}(2.008, 2.006, 2.0022), \\ \mathbf{A}^{\mathrm{N}} &= \mathrm{diag}(5.0, 5.0, 37.0)\ \text{Gauss}. \end{aligned} \tag{10.3}$$

In this picture, the explicit time dependence of the magnetic tensors in (10.1) is due to the *classical* rotational dynamics of the coordinate system $\mathtt{N}$ with respect to the stationary laboratory

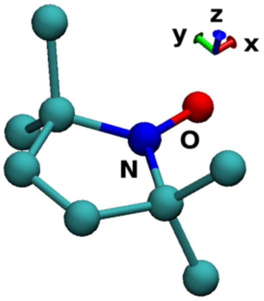


Fig. 10.1 The direction of the principle axes in the coordinate system $\mathtt{N}$ attached to the nitroxide ring. The x direction is along the N—O bond, the z direction is perpendicular to the ring, and the y direction is such that a right-handed system of axes is formed

frame $\mathtt{L}$, in which the constant magnetic field $\mathbf{B} = (0, 0, B)$ is applied.[1] Since the spins are quantized along $\mathbf{B}$ all the vector and tensor components in the Hamiltonian are with respect to $\mathtt{L}$.

The dynamics that cw-ESR experiments aim to probe leaves its mark on the spin dynamics described above by modulating the components G^{L}_{ij} and A^{L}_{ij} $(i, j = x, y, z)$ of the Zeeman and hyperfine magnetic tensors in the laboratory frame. Since these tensors remain unchanged in a coordinate system fixed on the nitroxide (cf. Fig. 10.1), the molecular motion is encoded in the form of rotation matrices $R^{\mathrm{LN}}(t)$ that transform the tensor components from the $\mathtt{N}$ to the $\mathtt{L}$ system of axes according to

$$\begin{aligned} G^{\mathrm{L}}_{ij}(t) &= \sum_{k=x,y,z} R^{\mathrm{LN}}_{ik}(t) G^{\mathrm{N}}_{kk} R^{\mathrm{LN}}_{jk}(t), \\ A^{\mathrm{L}}_{ij}(t) &= \sum_{k} R^{\mathrm{LN}}_{ik}(t) A^{\mathrm{N}}_{kk} R^{\mathrm{LN}}_{jk}(t). \end{aligned} \tag{10.4}$$

The time dependence of the transformation matrices is due to the dynamics of the nitroxide, as well as the global and internal dynamics of the molecule to which the spin label is covalently attached. Different motional models assumed for these dynamical modes result in different sequences $R^{\mathrm{LN}}(t)$, as discussed next.

10.2.2 Stochastic Models of Molecular Motion

The transformation from the nitroxide-fixed to the laboratory system of coordinate axes can be achieved as a sequence of nested rotational transformations. For example, the MOMD model mentioned in Sect. 10.1 can be illustrated schematically as

$$\mathtt{L} \xrightarrow{\text{powder}} \mathtt{M} \xrightarrow[\text{diffusion}]{\text{restricted}} \mathtt{S} \xrightarrow{\text{fixed}} \mathtt{N}. \tag{10.5}$$

Here $\mathtt{S}$ refers to the system of coordinate axes fixed on the spin label and $\mathtt{M}$ to the coordinate system attached to the macromolecule (e.g., protein) to which the spin label is covalently bonded. The model (10.5) represents the dynamics of the spin label with respect to the protein as restricted rotational diffusion. The coordinate system $\mathtt{S}$ attached to the spin label is defined by the principle axes of its diffusion tensor. In general, these axes do not need to overlap with the principle axes of the magnetic tensors, defining the coordinate frame $\mathtt{N}$. The model (10.5) can therefore account for the possibility that the nitroxide frame $\mathtt{N}$ has a fixed, time-independent orientation with respect to $\mathtt{S}$. In addition to the spin-label dynamics relative to the protein, in (10.5) the protein is allowed to be randomly orientated with respect to the laboratory frame. This would be the case for a large, relatively immobilized macromolecule in solution or a frozen (powder) sample.[2]

Clearly, more complex motional models can be constructed by combining independent or coupled nested rotations. Very attractive, however, is

[1] Although not rigorously correct, the assumption that the classical dynamics is completely uninfluenced by the states of the quantum system is typically an excellent approximation for room-temperature magnetic resonance. One minor inconvenience is that the equilibrium population of the states of the spin system corresponds to an infinite temperature. This, however, affects only the longitudinal magnetization but not the transverse magnetization whose evolution is calculated.

[2] A time-dependent or constant rotation matrix is associated with each successive transformation in a motional model like (10.5). The matrix for the net transformation from $\mathtt{L}$ to $\mathtt{N}$, to be employed in (10.4), is obtained as the product of the successive rotation matrices: $R^{\mathrm{LN}}(t) = R^{\mathrm{LM}} R^{\mathrm{MS}}(t) R^{\mathrm{SN}}$.

the alternative to forgo completely any stochastic model, and use the time-dependent dynamics of $\mathtt{N}$ relative to $\mathtt{L}$ extracted directly from atomistic MD simulations. This approach has been pursued by many [4, 8, 12, 22, 30, 53], following the pioneering work of Steinhoff and Hubbell from more than a decade and a half ago [52]. In this approach, the time-dependent rotation matrices $R^{\mathtt{LN}}(t)$ can be obtained directly from the snapshots of the classical MD simulations, which can be represented as

$$\mathtt{L} \xrightarrow[]{\substack{\text{MD}\\ \text{simulation}}} \mathtt{N}. \tag{10.6}$$

Although MD simulations of a spin-labeled macromolecule are expected to offer insight into the detailed dynamics of the spin label and its environment, there are important shortcomings to such an approach. In particular, extremely long trajectories are needed to sample the global tumbling of the macromolecule in solution. Without proper sampling of this relatively simple motion, the MD trajectories will not reflect the experimental situation realistically and cw-ESR spectra simulated from them will fail to reproduce the observed spectra. Thus, for the quantitative comparison of simulated and recorded spectra, it becomes necessary to be able to directly account for rotational diffusion by relying on a stochastic model. This can be achieved by modeling the dynamics of the coordinate frame $\mathtt{N}$ with respect to the macromolecule with atomistic MD simulations, while generating the dynamics of $\mathtt{M}$ relative to $\mathtt{L}$ using a stochastic model of isotropic or anisotropic rotational diffusion:[3]

$$\mathtt{L} \xrightarrow{\substack{\text{rotational}\\ \text{diffusion}}} \mathtt{M} \xrightarrow{\substack{\text{MD}\\ \text{trajectories}}} \mathtt{N}. \tag{10.7}$$

Cw-ESR spectra simulated from such a combination of stochastic rotational diffusion and MD trajectories were presented in Refs. [47, 49]. Essentially the same approach has been used by DeSensi et al. [12].

Even as MD simulations become more and more routine, the demand on the number and duration of the MD trajectories may become rapidly wasteful and inefficient when the purpose is to insert these into the model (10.7). In particular, the spin dynamics must be propagated over multiple molecular correlation times to explore all the possible orientations and yield a converged ESR spectrum. For this reason, it is important to develop alternative stochastic models able to provide a realistic "mimic" of the long-time dynamics of the spin label relative to the protein. When this dynamics is dominated by rotameric isomerization, the intermittent nature of the transitions between the various rotamers suggests that a Markov state model (MSM) shall provide an ideal framework to encode the internal spin-label dynamics available from MD simulations. Once its parameters have been properly estimated, the so-constructed MSM allows for the generation of computationally inexpensive and arbitrarily long stochastic trajectories. Combining the MSM dynamics with a diffusive model of the tumbling of the protein, cw-ESR spectra can be simulated in time domain according to the scheme:

$$\mathtt{L} \xrightarrow{\substack{\text{rotational}\\ \text{diffusion}}} \mathtt{M} \xrightarrow{\substack{\text{MSM}\\ \text{trajectories}}} \mathtt{N}. \tag{10.8}$$

Such simulations were performed in Refs. [49, 50].

10.2.3 Isotropic and Anisotropic Rotational Diffusion

To illustrate the impact of rotational diffusion on cw-ESR spectra, we consider the motional model

$$\mathtt{L} \xrightarrow{\substack{\text{rotational}\\ \text{diffusion}}} \mathtt{M} \xrightarrow{\text{fixed}} \mathtt{N}, \tag{10.9}$$

which describes a spin label rigidly tethered to a biological macromolecule tumbling in solution. For concreteness, let us take a double-helical B-DNA consisting of 20 base pairs (Fig. 10.2, left),

[3] Splitting the molecular motion according to (10.7) assumes that the overall molecular tumbling and the motion of the spin label with respect to the global molecular frame are independent [23]. Clearly, this approximation may break in some cases, e.g., when an internal structural rearrangement changes the overall structure—and hence the rotational diffusion tensor—of the whole molecule. Nevertheless, in many instances with spin-labeled biomacromolecules the approximation of decoupled global and internal motions is well justified.

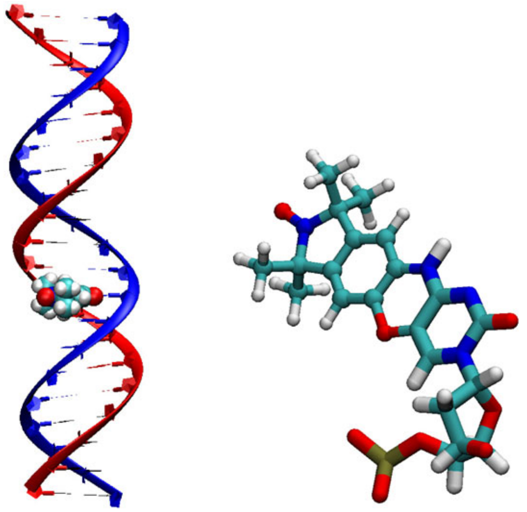


Fig. 10.2 *Left*: Double-helical B-DNA containing 20 base pairs and labeled with a single nitroxide spin label. *Right*: Nitroxide spin label (balls and thin sticks) covalently attached to the cytosine nucleotide (thick sticks) of DNA or RNA [9]

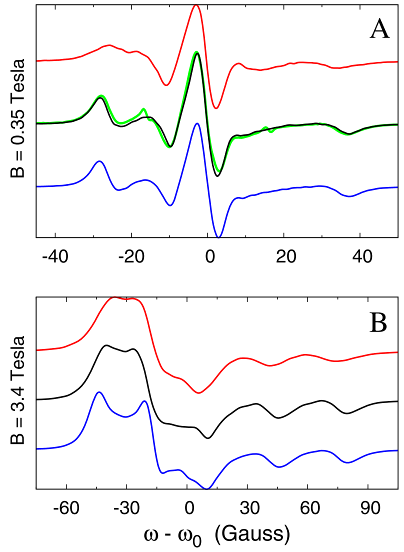


Fig. 10.3 Experimental X-band cw-ESR spectrum (*green*) and spectra calculated at two different magnetic fields (**A**) $B = 0.35$ T and (**B**) $B = 3.4$ T. At each field, three different tumbling rates with the following rotational diffusion tensors are compared: $D = \mathrm{diag}(10, 10, 10) \times 10^6\ \mathrm{s}^{-1}$ (*blue*), $D = \mathrm{diag}(10, 10, 40) \times 10^6\ \mathrm{s}^{-1}$ (*black*), and $D = \mathrm{diag}(25, 25, 40) \times 10^6\ \mathrm{s}^{-1}$ (*red*). For visual purposes the three spectra are systematically shifted in the vertical direction, which corresponds to the spectral intensity (in arbitrary units)

one cytosine base of which is labeled with a nitroxide spin label (Fig. 10.2, right) [9]. From the labeling geometry, the z axis of the nitroxide frame (defined in Fig. 10.1) is seen to be perpendicular to the plane of the base and collinear with the helix axis of the DNA. Choosing the z axis of the macromolecular coordinate system as the helix axis, and taking into account the symmetry of the double helix under rotation about this axis, we can take the two coordinate systems $\mathtt{M}$ and $\mathtt{N}$ to be identical.

In general, a 3 × 3 diffusion tensor needs to be specified for the tumbling of the molecular frame $\mathtt{M}$ with respect to the laboratory system of axes $\mathtt{L}$. With the above choice of the coordinate axes on the macromolecule the diffusion tensor is expected to be diagonal and of the form $D = \mathrm{diag}(D_\perp, D_\perp, D_\parallel)$, where $D_\parallel$ and $D_\perp$ are the diffusion coefficients for rotation about directions, respectively, parallel and perpendicular to the helix axis. Since the length of the double helix (≈70 Å) is several times larger than its diameter (≈20 Å), we expect to have $D_\parallel > D_\perp$.

Simulations of cw-ESR spectra at two different magnetic fields ($B = 0.35$ and $B = 3.4$ Tesla) for the spin-labeled B-DNA tumbling in solution are presented in Fig. 10.3. (For further details about the geometry and the simulation parameters the reader is referred to Ref. [51].) The bottom spectrum (blue), simulated using $D = \mathrm{diag}(10, 10, 10) \times 10^6\ \mathrm{rad}^2/\mathrm{s}$, corresponds to isotropic rotational diffusion expected from a spherical macromolecule. The spectrum in the middle (black), simulated using $D = \mathrm{diag}(10, 10, 40) \times 10^6\ \mathrm{rad}^2/\mathrm{s}$, takes into account the faster diffusion of the elongated DNA molecule about its helix axes. In fact, it compares very well with the experimental spectrum (green) on top of which it is overlaid. Notice, however, that the two spectra (black and blue) at $B = 0.35$ T—the magnetic field most commonly used in studies of biomacromolecules—are indistinguishable in these two cases. For the spin labeling geometry considered in this example, cw-ESR experiments at the higher field of $B = 3.4$ T are necessary to pick up the elongated shape of the molecule. The (red) spectrum at the top in Figs. 10.3A

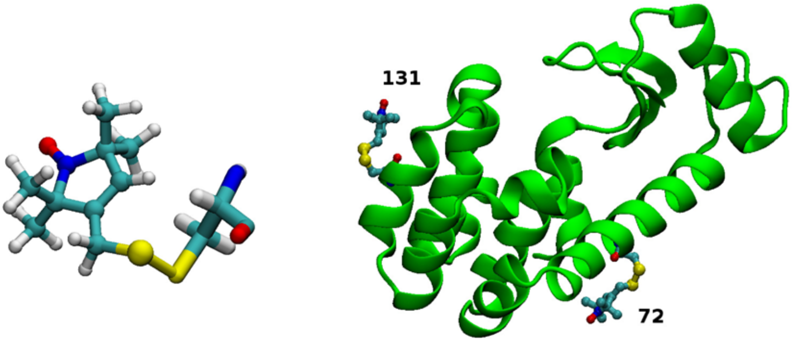


Fig. 10.4 *Left*: Nitroxide spin label R1 (balls and thin sticks) covalently bonded to the cysteine amino acid (thick sticks) of a protein. *Right*: T4 Lysozyme (T4L) labeled with R1 at positions 72 and 131. The former is located on a long α-helix, whereas the latter is on a short α-helix. At both sites the spin labels are at the surface of the protein nicely exposed to the solvent, i.e., solvent-exposed helix surface (SEHS) spin labels. In experiments the labels are present one at a time

and B, simulated using $D = \text{diag}(25, 25, 40) \times 10^6$ rad^2/s, is intended to illustrate the effect of shortening the length of the DNA double helix. Keeping the diffusion coefficient for rotation about the helix axis the same as in the (black) spectrum in the middle, we have increased the rate of diffusion about the perpendicular axis from 10 to 25×10^6 rad^2/s. In this case, the spectra at both fields are sufficiently different from the spectra simulated using $D = \text{diag}(10, 10, 40) \times 10^6$ rad^2/s (black in Fig. 10.3), showing the sensitivity of the experiments to the length of the DNA double helix.

The simulations in Fig. 10.3 rest on the assumption that the only motion experienced by the spin label is anisotropic rotational diffusion. In reality, other motions—like the libration of the base to which the spin label is covalently attached—are expected to take place in addition to the global molecular tumbling. However, it should be clear from the presented evidence that the internal motions can be unambiguously inferred from the experimental spectra only if the effect of the global motion is carefully accounted for along the lines illustrated in Fig. 10.3.

10.3 MSM of Spin-Label Dynamics

In this section, we observe that the internal dynamics of a solvent-exposed spin label on the surface of a protein is dominated by the isomerization of its linker (Sect. 10.3.1). Such motion is perfectly suited for modeling by MSMs that can be subsequently used to simulate cw-ESR spectra according to the model (10.8). The construction of MSMs of the spin-label dynamics from MD trajectories is illustrated in Sect. 10.3.2 for two spin labels at solvent-exposed positions on the protein T4 Lysozyme. Multifrequency spectra for these two positions will be considered in Sect. 10.5.3.

10.3.1 Side Chain Isomerization as Intermittent Dynamics

The covalent attachment of the spin label in Fig. 10.2 to the cytosine base of DNA lacks any rotatable bonds. Hence, its internal dynamics is expected to be tightly coupled to the internal dynamics of the entire DNA fragment. The situation is different for ESR studies of proteins, in which the spin label referred to as R1 is most commonly used [10, 16]. This nitroxide spin label is covalently bonded to the side chain of the amino acid cysteine through a linker consisting of five sequential chemical bonds (Fig. 10.4, left). In principle, rotations around each one of these bonds are sterically permitted, which should allow for rich internal spin label dynamics largely independent from the dynamics of the protein backbone.

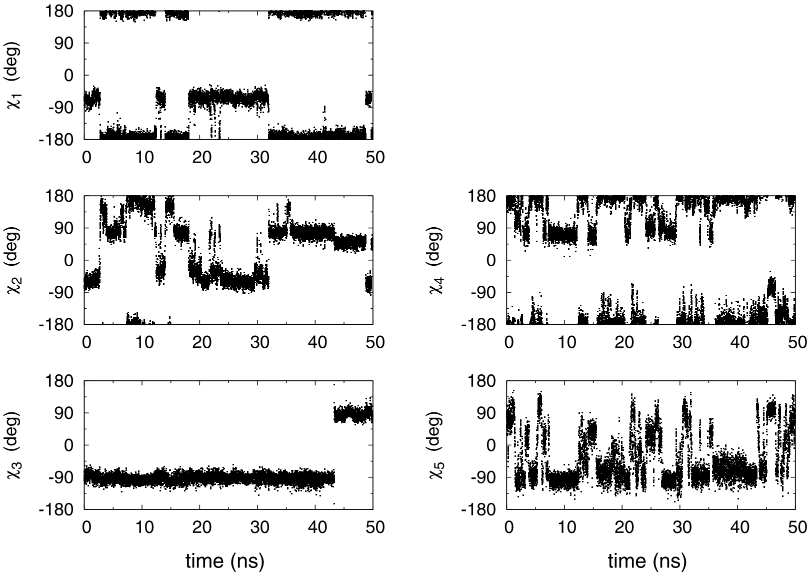


Fig. 10.5 Isomerization dynamics of the nitroxide spin label R1 at position 72 on T4 Lysozyme

Because of the additional complexity introduced by the internal freedom of R1 and other similar spin labels attached to proteins, extensive efforts have been dedicated to elucidate the microscopic factors affecting their conformation and dynamics. Particularly informative are the experimental studies with the well-characterized protein T4 Lysozyme (T4L). A wealth of results, ranging from X-ray crystallography of spin-labeled T4L [18, 21, 31], to X-band [11, 35, 36] or multifrequency [33, 56] cw-ESR experiments, are now available. From those studies, two positions—72 and 131—situated in the middle of, respectively, a long and a short helix, have emerged as prototypical solvent-exposed helix surface (SEHS) sites (Fig. 10.4, right). The dynamics of R1 at these positions has been rationalized in terms of the "χ_4/χ_5 model" for SEHS sites [10, 11], which assumes that the internal motion of R1 is largely limited to rotations about the last two dihedrals of the side chain. According to this model, the remaining dihedrals are effectively "locked": the χ_3 disulfide torsion is opposed by a large energy barrier [25], while the χ_1 and χ_2 torsions of the cysteine side chain are hindered by the formation of a hydrogen bond between the sulfur atom of R1 and the backbone amide [36] or C_α [10]. Such sulphur-backbone contacts are indeed observed in a number of X-ray crystal structures of T4L with spin labels [18, 21, 31], in support for the χ_4/χ_5 model. Furthermore, the χ_4/χ_5 model offers an atomistic rationalization of the fitting parameters of the diffusional models MOMD and SRLS, which can produce simulated spectra in quantitative agreement with experiments [56].

According to the current understanding, 72R1 and 131R1 in T4L are believed to exemplify the internal R1 dynamics at SEHS sites in maximum isolation. The differences in the X-band cw-ESR spectra at these two positions are thought to reflect the effect of backbone motion on the mobility of the spin label side chain [11]. Nevertheless, a number of issues remain. For instance, the spin label is partly disordered and unresolved in several X-ray structures [18], suggesting that multiple conformations are energetically accessible. Therefore, in spite of the large amount of data available, a definitive characterization of the spin label dynamics at SEHS sites at the atomic level has not been achieved.

The values of the spin label dihedral angles during an atomistic simulation of T4L labeled at position 72 are shown in Fig. 10.5. The

Table 10.1 Information about the two sets of MD simulations of spin-labeled T4 Lysozyme

	72R1 (set 1)	131R1 (set 2)
Number of independent trajectories	18 (χ_1, χ_2, χ_3)[a]	54 ($\chi_1, \chi_2, \chi_3, \chi_4$)[a]
Duration of single trajectory	32.3 ns	12.7 ns
Total simulation time (analyzed)[b]	581 (563) ns	686 (632) ns

[a] In each independent trajectory of a given set the spin label was initialized to be in a different rotameric state by restraining the dihedral angles given in the parenthesis to their canonical values. The number of different rotamers was determined using the multiplicity of the dihedral angles, χ_1:3, χ_2:3, χ_3:2, χ_4:3

[b] The first 1 ns of every trajectory was treated as equilibration period and not analyzed

time traces of these angles do seem to undergo jump like dynamics between different discrete states, exemplifying intermittent internal dynamics. Thus, the transitions between the rotameric conformations of the spin label side chain should be amenable to modeling by MSMs.

Before embarking on a rigorous MSM modeling, a few observations about the internal isomerization dynamics of the protein spin label can be made on the basis of the time traces in Fig. 10.5. First, the lifetimes of the states corresponding to the different values that the five torsions preferentially adopt show great variation. At one extreme is the disulfide dihedral angle χ_3, which has undergone a single transition from $\chi_3 \approx -90°$ to $\chi_3 \approx +90°$ during the entire simulation of 50 ns. At the other extreme is the dihedral angle closest to the nitroxide ring, χ_5, which has moved several times between the values $\chi_5 \approx -90°$, $\chi_5 \approx +90°$ and $\chi_5 \approx 0°$. Second, χ_1 and χ_2—the dihedral angles closest to the protein backbone—are seen to undergo a transition every 5 to 10 ns. As a result, χ_2 has visited the three canonical values of $\pm 60°$ and $180°$, even if only a few times. Similarly, χ_1 has exchanged between two of these canonical values a few times.[4] Third, in many of the transitions χ_1 and χ_2 seem to flip simultaneously in a concerted manner. Occasionally, all the four dihedral angles, with the exception of χ_3, are seen to undergo concerted transitions. Furthermore, the rate of isomerization of a given dihedral appears to depend on the values adopted by all the other dihedral angles. (This is perhaps most clearly seen for χ_5.) Hence, it is not justified to assume that the dynamics of the torsion angles is independent [54]. Instead, the conformation of the entire spin label side chain has to be considered when trying to identify the states of the intermittent motion and the rates of exchange between them. In the light of these observations we now turn to the systematic construction of MSMs for the dynamics of R1 at positions 72 and 131 in T4L.

10.3.2 MSM from MD Trajectories

Extensive all-atom MD simulations of fully-solvated spin-labeled T4L were performed for the two SEHS positions of interest—72 and 131—with the purpose of mapping the R1 isomerization dynamics from the MD trajectories to MSMs. To enhance the sampling of the possible spin-label conformations several independent trajectories were generated starting from different R1 conformations. Information about the number and duration of the trajectories is summarized in Table 10.1.

To proceed with building MSMs of the spin-label dynamics relative to the protein from the MD simulations, a set of observables, called order parameters, has to be selected among the large collection of variables contained in the trajectories. Here, we assume that the dihedral angles of the spin label side chain constitute an adequate set of order parameters—a natural choice based on physical insight about the system. Then,

[4] In protein crystal structures the side chain of cysteine is very rarely seen to adopt a conformation with $\chi_1 \approx +60°$ when located on α helices since this places the cysteine sulfur atom in unfavorable steric contact with the backbone atoms of the helix.

the five-dimensional space of the order parameters is divided into 120 regions[5] (microstates) using K-means clustering [24]. At this point, it is hoped that if the microstates are chosen to be narrow enough, such that intrastate relaxation is fast, the kinetics of jumping out of a microstate will be approximately Markovian.

More formally, let $X(t)$ be a random variable indicating the state of an N-state Markov chain model at time t. The probabilities $p_i(t) = \mathbb{P}\{X(t) = i\}$, to observe the chain in state i at time t, form a (row) vector $\langle p(t)| = [p_i(t)]$, whose evolution is governed by the Master equation

$$\dot{p}_j(t) = \sum_{i=1}^{N} p_i(t) K_{ij}. \tag{10.10}$$

Here, a derivative with respect to time has been denoted with a dot. The matrix $K = [K_{ij}]$ is referred to as the "rate matrix". Its off-diagonal entries are larger or equal to zero. For a conservative process, its diagonal elements are negative and given as $K_{ii} = -\sum_{j \neq i} K_{ij}$. They are directly related to the lifetime [39]

$$\nu_i = -1/K_{ii} \tag{10.11}$$

of each state. The stationary probability distribution of the chain $\langle \pi|$, is the left eigenvector of K with eigenvalue zero, i.e., $\langle \pi|K = 0$. A system in thermal equilibrium satisfies the detailed balance condition $\pi_i K_{ij} = \pi_j K_{ji}$, which implies that K can be transformed to a symmetric form by a similarity transformation with the diagonal matrix $D = [\sqrt{\pi_i}\delta_{ij}]$. Thus all the eigenvalues of K are real. When written as $-1/\tau_i$, the nonzero eigenvalues give the relaxation time scales τ_i of the stochastic dynamics generated by K (cf. Eq. (10.12)). Note that, in general, $\tau_i \neq \nu_i$.

If the observed time series were generated from a continuous-time Markov chain, one could easily estimate the rate matrix by counting the number of $i \to j$ jumps and the total time spent in state i, i.e., lifetime ν_i. This is not possible when the trajectories of the order parameters are coming from MD simulations, since the short-time dynamics of the order parameters are not necessarily Markovian. For instance, the time-series of the spin-label torsion angles in Fig. 10.5 contain "spurious" transitions back and forth between states i and j before a "real" transition occurs. This would lead to an unreliable estimate of K from the MD trajectories. This problem is partially alleviated by observing the system only at instances separated by a long enough time interval τ—referred to as lag time—such that the dynamics is more likely to appear memoryless from one observation to the next. Such an approach, however, leads to a discretization of the time axis, thus deviating from the continuous-time Markov chain model (10.10). By counting the number of times the chain in state i goes to state j after time τ, a transition probability matrix, $T(\tau)$, with matrix elements $T_{ij}(\tau) = \mathbb{P}\{X_{t+\tau} = j \mid X_t = i\}$ can be estimated. This matrix determines the evolution of the state probabilities for times spaced by τ: $\langle p(t+\tau)| = \langle p(t)|T(\tau)$.

To further reduce the miscounting of very short-lived excursions in the values of the torsion angles as genuine transition events between distinct conformational states, the time series of the spin label dihedral angles were analyzed with a hidden Markov model. The latter allows for the observed dihedral angle values to be dispersed about the values defining the state of the MSM according to a distribution with a mean and a standard deviation. In this sense, what is analyzed are the values of the dihedral angles "emitted" from a rotameric state that is not directly accessible (i.e., hidden). The mean values of the observed torsion angles in each state as well as the state-to-state jump probabilities were inferred in an iterative manner using the Viterbi algorithm [42]. Detailed description of the followed procedure is available in Ref. [49].

Ideally, if the modeled process is indeed Markovian, the transition matrices estimated using different lag times should be consistent in the sense that $T(\tau_1)T(\tau_2) = T(\tau_1 + \tau_2)$. If the underlying process being observed at discrete instances in time is in fact a continuous-time Markov chain

[5] The observation that, considering the multiplicity of its dihedral angles, the R1 side chain can adopt 108 different rotameric states motivated the choice of number of microstates.

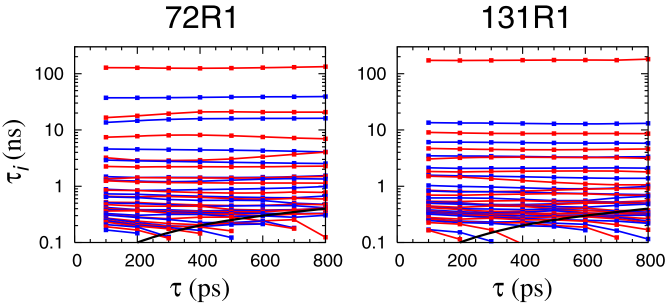


Fig. 10.6 Relaxation time scales calculated according to (10.12) using the eigenvalues of $T(\tau)$ estimated from the data at various lag times τ. The *thick black curves* correspond to 2τ. τ_i's that fall under these curves are essentially zero and are poorly estimated

with rate matrix K, the relation $T(\tau) = \exp(\tau K)$ should hold. This implies that the eigenvalues $\lambda_i^T(\tau)$ of $T(\tau)$ are related to the eigenvalues λ_i^K of K through $\lambda_i^T(\tau) = \exp(\tau\lambda_i^K)$. Therefore, the relaxation time scales of the MSM are

$$\tau_i = -\frac{1}{\lambda_i^K} = -\frac{\tau}{\ln \lambda_i^T(\tau)}. \qquad (10.12)$$

Equation (10.12) is a necessary, although not sufficient, condition for the matrices $T(\tau)$ estimated using different lag times τ to describe a genuinely Markovian process. If the relaxation time scales τ_i calculated from the eigenvalues of such $T(\tau)$ are not independent of τ, it is certain that the process is not Markovian. To assess the Markovian nature of the estimated process corresponding to the isomerization of the spin label R1 at positions 72 and 131, the analysis of the dihedral time series using the Viterbi algorithm was repeated for several different lag times τ, ranging from 100 to 800 ps. The relaxation time scales obtained from the estimated microstate transition matrices according to (10.12) are shown in Fig. 10.6. The fact that the lines are approximately horizontal (i.e., independent of the lag time) indicates that, on time scales larger than the lag time, the time series for dihedral dynamics of both 72R1 and 131R1 are faithfully modeled by MSMs of jumps between 120 discrete states. Further analysis is limited to the transition probability matrices for the shortest lag time, $\tau = 100$ ps.

The choice of microstates based on K-means clustering is purely geometrical. Using the microstate transition probability matrix the microstates can be lumped into several groups of kinetic significance (macrostates). The resulting macrostates are intended to correspond to the rarely exchanging, metastable conformations of the spin label, and in the end, it is the Markovian kinetics among the macrostates that constitutes a model of the slow spin-label dynamics relative to the protein. The degree of lumping, which determines the final number of macrostates, is decided on the basis of the desired temporal resolution (which is related to the width of the cw-ESR spectrum, as discussed in Sect. 10.5.2).

The final MSMs constructed for 72R1 and 131R1 on T4L contained 37 and 38 (macro)states, respectively (Table 10.2). It is important to note that a critical ingredient of the MSMs at this stage is the usage of pre-averaged magnetic tensors of the spin labels associated with the Markovian macrostates (see Sects. 10.5.1 and 10.5.2). At both sites, the slowest relaxation times ($\tau_1 \gtrsim$ 100 ns) are related to transitions of the disulfide torsion angle between its two stable conformations $\chi_3 \approx -90°$ (m) and $\chi_3 \approx +90°$ (p). The exact numerical values of τ_1, as well as the relative populations of the m and p conformations are not expected to be accurately estimated by the constructed MSM due to the small number of such transitions observed in the free simulations (Table 10.2). To determine accurately the m:p ratio, the free energy difference between two Markovian states on the opposite side of the χ_3 torsion was calculated using umbrella sampling simulations [45]. This resulted in 27 % m—73 % p for 72R1, and 55 % m—45 % p for 131R1 (Table 10.2) [50].

The spin-label conformations of the most populated five states of the MSMs of 72R1 and 131R1 are presented in Fig. 10.7. In spite of the

Table 10.2 Number of transitions between the m and p conformations observed in all the MD trajectories and the number of Markov states assigned to those conformations

site	number of transitions		number of states			m:p[a]
	m → p	p → m	m	p	total	
72R1	2	1	18	19	37	27:**73**
131R1	1	1	19	19	38	**55**:45

[a] The m:p ratio (%) is determined from independent restrained simulations

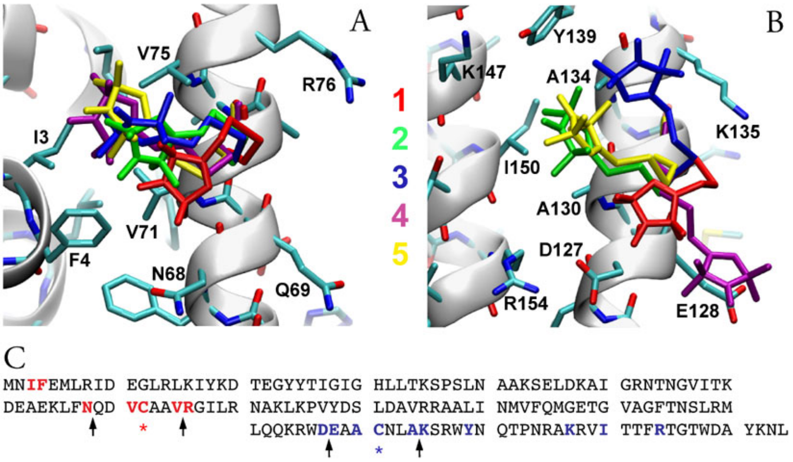

A
V75
R76
I3
V71
F4
N68
Q69
1
2
3
4
5
B
Y139
K147
A134
K135
I150
A130
D127
R154
E128
C
MNIFEMLRID EGLRLKIYKD TEGYYTIGIG HLLTKSPSLN AAKSELDKAI GRNTNGVITK
DEAEKLFNQD VCAAVRGILR NAKLKPVYDS LDAVRRAALI NMVFQMGETG VAGFTNSLRM
LQQKRWDEAA CNLAKSRWYN QTPNRAKRVI TTFRTGTWDA YKNL
*
*

Fig. 10.7 The most populated five conformations of 72R1 (**A**), and 131R1 (**B**) are colored as indicated. The rotameric states of the neighboring side chains when interacting with R1 are not necessarily the same as the ones shown here. (**C**) The amino acid sequence of T4L. Stars indicate the spin labeled sites and arrows point between the $i \pm 3$ and $i \pm 4$ positions. Residues with which R1 interacts are colored red and blue for 72 and 131, respectively

variation of χ_4 and χ_5, as well as χ_1, χ_2, and χ_3, across the most populated states of 72R1, the nitroxide ring appears to be rather well localized (Fig. 10.7A). In contrast, the conformations corresponding to the first five most populated states of 131R1 are dispersed (Fig. 10.7B). In the simulations the spin label adopts a multiplicity of conformations, many of which tend to interconvert on time scales relevant for cw-ESR experiments. When the exchanges are slow, the states appear to be stabilized by a range of polar and nonpolar interactions with neighboring amino acids (Fig. 10.7C) frequently involving the nitroxide ring.

The MSMs of spin-label dynamics at positions 72 and 131 in T4L constructed from the MD simulations can be used in the model (10.8) to simulate cw-ESR spectra. The results of such simulations are compared with experiments in Sect. 10.5.3. In the next section we describe the procedure for simulating cw-ESR spectra from trajectories of the spin-label dynamics.

10.4 Numerical Simulation of cw-ESR Spectra

Cw-ESR experiments are performed under the presence of a strong constant magnetic field and a weak microwave (mw) magnetic field applied continuously in a direction perpendicular to the constant field. Subjecting the sample to these external magnetic fields leads to detectable bulk magnetization. Cw-ESR experiments measure the component of the magnetization perpendicular to the constant magnetic field—transverse component; the longitudinal component along the direction of the constant magnetic field is not detected. For the sufficiently weak mw fields typically employed, the cw spectrum is identical to the Fourier transform of a free induction decay (FID) [1], where FID refers to the decay of the transverse magnetization in the absence of any mw field. In FID the electron spins are first flipped to the transverse plane using a ninety-degree mw pulse after which the oscillating field is switched off. Hence, at the beginning of the decay, the longitudinal magnetization is equal to zero while the transverse magnetization can be taken to be equal to one. During an FID the spins evolve only under the action of the constant magnetic field and the observed decay of the transverse magnetization results from the decoherence of the spins. In magnetic resonance the time scale of this decay is known as T_2. In contrast, the time scale on which the longitudinal magnetization builds up is referred to as T_1. At sufficiently strong magnetic fields and for motionally broadened cw-ESR spectra, which are our main interest, T_2 is much shorter than T_1 (e.g., nanoseconds versus microseconds). Therefore, it is safe to assume that by the time the transverse magnetization has decayed to zero the longitudinal magnetization has remained at its initial value of zero. This constitutes the high-field approximation whose implications will become apparent below (see Sect. 10.4.1).

Because there is no mw field during the FID evolution, in numerical work aiming to calculate cw-ESR spectra it is preferable to simulate the FID and compute its Fourier transform. With $\mathscr{M}_+(t)$ denoting the bulk transverse magnetization after a ninety-degree pulse, the spectrum is the (one-sided) Fourier transform

$$\mathscr{S}(\omega) = \int_0^\infty \mathscr{M}_+(t)\mathrm{e}^{-\mathrm{i}\omega t}\,\mathrm{d}t. \qquad (10.13)$$

Due to the way a cw experiment is actually performed—which is different than recording an FID—the spectrum is in fact the derivative of $\mathscr{S}(\omega)$ with respect to ω. Differentiating (10.13), a derivative-mode cw-ESR spectrum is readily found to be

$$\frac{\mathrm{d}\mathscr{S}(\omega)}{\mathrm{d}\omega} = -\mathrm{i}\int_0^\infty t\mathscr{M}_+(t)\mathrm{e}^{-\mathrm{i}\omega t}\,\mathrm{d}t. \qquad (10.14)$$

Using two channels both the real and imaginary parts of this spectrum can be recorded. In practice, however, only the real part is reported. All the calculated cw-ESR spectra shown in this chapter correspond to the real part of (10.14), obtained by taking the discrete Fourier transform of $t\mathscr{M}_+(t)$ numerically. Because it follows the decay of $\mathscr{M}_+(t)$ in time, the presented approach constitutes a simulation in time domain (in contrast to the frequency-domain methodology based on the SLE).

10.4.1 The Nitroxide Density Matrix

To calculate the macroscopic transverse magnetization appearing in (10.14), the microscopic magnetizations, $M_+(t)$, from all the possible random trajectories that a nitroxide spin label in the solution may undergo need to be averaged. Denoting this ensemble averaging with angular brackets we have

$$\mathscr{M}_+(t) = \langle M_+(t)\rangle. \qquad (10.15)$$

Assuming a dilute solution of non-interacting free radicals, the state of each electron-nuclear spin system on one nitroxide can be described by a density operator. With $\rho(t)$ denoting the density matrix associated with one stochastic trajectory of an individual spin label, microscopic transverse and longitudinal magnetizations can be calculated as follows:[6]

$$\begin{aligned} M_+(t) &= \mathrm{Tr}\{\rho(t)\hat{S}_+\} \quad \text{and} \\ M_z(t) &= \mathrm{Tr}\{\rho(t)\hat{S}_z\}. \end{aligned} \qquad (10.16)$$

[6] Since the absolute value of the cw-ESR measurement depends on instrumental factors and is not relevant for our purposes, proportionality constants relating the magnetizations and the respective spin operators have been neglected in (10.16).

In these expressions, $\hat{S}_+$ and $\hat{S}_z$ are spin-1/2 operators[7] and Tr denotes a trace. The density operator lives in the outer product of the electron and nuclear spin Hilbert spaces. For easier access to the microscopic magnetizations in (10.16) it can be written as a sum of Kronecker products between the spin-1/2 matrices and matrices ρ_κ with the dimensionality of the nuclear-spin Hilbert space:[8]

$$\begin{aligned}\rho(t) &= \rho_+(t)\hat{S}_- + \rho_-(t)\hat{S}_+ + 2\rho_z(t)\hat{S}_z \\ &\quad + \rho_0(t)\hat{S}_0 \\ &= \begin{bmatrix} \rho_0+\rho_z & \rho_- \\ \rho_+ & \rho_0-\rho_z \end{bmatrix}. \end{aligned} \tag{10.17}$$

Thus, ρ is represented by a 6 × 6 matrix (for ^{14}N) or a 4 × 4 matrix (for ^{15}N). Using (10.17) in (10.16) it is straightforward to deduce that

$$\begin{aligned} M_+(t) &= \mathrm{Tr}\{\rho_+(t)\} \quad \text{and} \\ M_z(t) &= \mathrm{Tr}\{\rho_z(t)\}, \end{aligned} \tag{10.18}$$

where the trace in these expressions is only over the nuclear spin degrees of freedom.

To calculate the microscopic magnetization $M_+(t)$ along a given dynamical trajectory we need to be able to follow numerically the evolution of the density matrix starting from the appropriate initial conditions. To this end, we recall that the density matrix evolves according to the Liouville-von Neumann equation

$$\dot{\rho}(t) = -\mathrm{i}\big[\hat{H}(t), \rho(t)\big], \tag{10.19}$$

where $\hat{H}(t)$ is the Hamiltonian (10.1) of the spin system (in units of angular frequency) and $[\cdot,\cdot]$ denotes a commutator. The initial conditions for an FID can be obtained from (10.18). These are the identity matrix for $\rho_+(0)$ and the zero matrix for $\rho_z(0)$.

In analogy to (10.17), the ESR spin Hamiltonian can be written as

$$\hat{H}(t) = 2H_z(t)\hat{S}_z + H_+(t)\hat{S}_- + H_-(t)\hat{S}_+, \tag{10.20}$$

where the matrices H_κ have the dimensionality of the Hilbert space of the nuclear spin. Substituting the expansions (10.17) and (10.20) into the equation of motion (10.19), and equating the coefficients of $\hat{S}_-$ on both sides of the equality leads to the evolution law

$$\begin{aligned}\dot{\rho}_+(t) &= \mathrm{i}\big\{H_z(t), \rho_+(t)\big\} - \mathrm{i}\big\{H_+(t), \rho_z(t)\big\} \\ &\quad - \mathrm{i}\big[H_+(t), \rho_0(t)\big], \end{aligned} \tag{10.21}$$

where $\{\cdot,\cdot\}$ denotes an anticommutator. At this point, it is convenient to invoke the high-field approximation, which amounts to assuming that ρ_z and ρ_0 remain zero throughout the times we follow the evolution of ρ_+.[9] As a result, (10.21) simplifies to

$$\dot{\rho}_+(t) = \mathrm{i}\big\{H_z(t), \rho_+(t)\big\}. \tag{10.22}$$

Hence, $\rho_+(t)$ is the only part of the full density matrix that needs to be considered and $H_z(t)$ is the only part of the Hamiltonian that needs to be calculated at every time step. From (10.1),

$$H_z(t) = |\gamma_\mathrm{e}|\big[BG^\mathrm{L}_{zz}(t) + \mathbf{a}^\mathrm{L}(t)\cdot\hat{\mathbf{I}}\big]/2, \tag{10.23}$$

where $G^\mathrm{L}_{zz}(t)$ is the respective component of the rescaled Zeeman tensor in the laboratory frame, and the components of $\mathbf{a}^\mathrm{L}$ are defined in terms of the components of the hyperfine tensor as

$$a^\mathrm{L}_i(t) \equiv A^\mathrm{L}_{iz}(t). \tag{10.24}$$

[7] These are

$$\hat{S}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad \hat{S}_+ = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix},$$
$$\hat{S}_- = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \qquad \hat{S}_z = \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

[8] In principle, the submatrix ρ_0 in (10.17) contains the 3 × 3 or 2 × 2 identity matrix I_0 along its main diagonal. However, the part proportional to the identity matrix is neither affected by the relaxation or the coherent evolution nor does it affect the evolution of the rest of the density matrix. Hence, ρ_0 can be taken as traceless.

[9] As mentioned above, the justification lies in the fact that the time scale T_1—on which ρ_z and ρ_0 build up—depends on motions at the time scale of the Larmor precession and is much longer than the time scale T_2—on which ρ_+ decays—dominated by slow motions. The high field approximation automatically excludes the possibility to account for the contribution of T_1 processes to T_2 relaxation using Eq. (10.22).

Table 10.3 Magnetic fields, corresponding Larmor frequencies,[a] microwave bands,[b] and time scales of precession[c]

B/T	0.12	0.34	1.21	3.39	6.07	9.2
f_e/GHz	3.4 (S)	9.5 (X)	34 (Q)	95 (W)	170 (G)	260 (J)
τ_0/ps	47	17	4.7	1.7	0.95	0.61

[a] $f_e = \omega_0/2\pi$

[b] In parenthesis

[c] $\tau_0 = 1/\omega_0$

10.4.2 Propagation of the Quantum Spin System

Each electron spin in the ensemble undergoes a precession about the applied constant magnetic field B with average angular frequency, known as the Larmor frequency of the electron spin, equal to [32]

$$\omega_0 \equiv -\gamma_e B G_0, \tag{10.25}$$

where $G_0 = \mathrm{Tr}\{\mathbf{G}\}/3$. Table 10.3 contains the frequencies and time scales of precession for several different magnetic fields of experimental interest. When following the time evolution of the density matrix with the purpose of calculating $M_+(t)$, it proves convenient to work in a coordinate system rotating about the laboratory z-axis with the electron Larmor frequency. In this rotating frame (denoted with a prime) the operators $\hat{S}_0$ and $\hat{S}_z$ remain unchanged whereas $\hat{S}_+$ and $\hat{S}_-$ acquire a phase: $\hat{S}'_\pm = \hat{S}_\pm e^{\pm i\omega_0 t}$. Hence, from (10.17), the density matrix in the rotating frame becomes $\rho'(t) = \rho'_+(t)\hat{S}_- + \rho'_-(t)\hat{S}_+ + 2\rho_z(t)\hat{S}_z + \rho_0(t)\hat{S}_0$, where

$$\rho_\pm(t) = \rho'_\pm(t)e^{\pm i\omega_0 t}. \tag{10.26}$$

Using this last relation in (10.18) yields the transverse magnetization[10]

$$M_+(t) = \mathrm{Tr}\{\rho'_+(t)\}e^{i\omega_0 t} = M'_+(t)e^{i\omega_0 t}. \tag{10.27}$$

Substituting (10.27) in the expression for the spectrum (10.13), we find

$$\mathscr{S}(\omega - \omega_0) = \int_0^\infty \langle M'_+(t)\rangle e^{-i\omega t}\,dt. \tag{10.28}$$

Hence, taking the Fourier transform of the ensemble-average of the slowly varying $M'_+(t)$ produces the desired experimental cw-ESR spectrum but with the origin shifted to the Larmor frequency ω_0.

Since M'_+ is defined as the trace of ρ'_+ the latter needs to be evolved numerically. From (10.22) and (10.26) it is straightforward to conclude that

$$\begin{aligned}\dot{\rho}'_+(t) &= i\{H_z(t), \rho'_+(t)\} - i\omega_0\rho'_+(t) \\ &= i\{V(t), \rho'_+(t)\},\end{aligned} \tag{10.29}$$

where

$$\begin{aligned}V(t) &\equiv H_z(t) - \omega_0/2 \\ &= |\gamma_e|\left[BG'^{L}_{zz}(t) + \mathbf{a}^L(t)\cdot\hat{\mathbf{I}}\right]/2.\end{aligned} \tag{10.30}$$

In the last equality of (10.30) we have introduced the traceless coupling tensor $\mathbf{G}'$ obtained by subtracting G_0 from the diagonal entries of the Zeeman tensor $\mathbf{G}$: $\mathbf{G}' \equiv \mathbf{G} - G_0\mathbf{E}$.

The numerical evolution of $\rho'_+(t)$ according to (10.29) over a short time step Δt can be achieved as[11]

$$\rho'_+(t+\Delta t) = U(t,\Delta t)\rho'_+(t)U(t,\Delta t) \tag{10.31}$$

after introducing the short-time propagator

[10] The numerical advantages associated with working in the rotating frame are apparent from (10.27), where the transverse magnetization $M_+(t)$ consists of a rapidly oscillating "carrier" wave whose amplitude is modulated by the slowly changing "signal" $M'_+(t)$. Thus, following $M_+(t)$ numerically would require an integration time step sufficient to resolve the fast oscillations on the time scale of the Larmor precession (cf. Table 10.3). In contrast, calculating the slowly changing $M'_+(t)$ numerically allows us to take time steps larger by several orders of magnitude.

[11] Note that the same propagator matrix acts on both sides of ρ'_+ in this equation, which is different from the propagation of the density matrix ρ in the full Hilbert space. The source of the difference lies in replacing the commutator in (10.19) by an anticommutator in (10.22).

$$
\begin{aligned}
U(t, \Delta t) &\equiv \mathrm{e}^{\mathrm{i}\Delta t V(t)} \\
&= \mathrm{e}^{\mathrm{i}\Delta t |\gamma_\mathrm{e}| B G'^{\mathrm{L}}_{zz}(t)/2} \mathrm{e}^{\mathrm{i}\Delta t |\gamma_\mathrm{e}| \mathbf{a}^{\mathrm{L}}(t)\cdot\hat{\mathbf{I}}/2},
\end{aligned}
\tag{10.32}
$$

which needs to be calculated from the instantaneous values of the magnetic tensors in the laboratory frame at every integration time step. (How to efficiently compute $U(t, \Delta t)$ was described in Ref. [48].)

The resulting computational framework is summarized by the following steps:

(i) To represent the dynamics of the classical degrees of freedom, a stochastic trajectory combining continuous rotational diffusion together with an MSM according to the model (10.8) is propagated, generating the time series $R^{\mathrm{LN}}(t) = R^{\mathrm{LM}}(t) R^{\mathrm{MN}}(t)$ sampled at the time step Δt.

(ii) At every time step the instantaneous values of the magnetic tensors in the laboratory frame are calculated according to (10.4) and used to obtain the short-time propagator (10.32). (It should be noted that these correspond to the magnetic tensors averaged over the fast librations of the spin label assigned to a given Markovian macrostate, as described in Sects. 10.5.1 and 10.5.2).

(iii) The QM density matrix, $\rho_+(t)$, is evolved along a single stochastic trajectory according to (10.31) and a microscopic magnetization $M'_+(t)$ is calculated from its trace at every time step Δt.

(iv) A large number of such trajectory-specific magnetization time series, calculated by generating different realizations of the stochastic trajectories, are generated and added together to obtain the ensemble-averaged macroscopic magnetization in the rotating frame, $\langle M'_+(t) \rangle$.

(v) The magnetization is Fourier transformed to obtain a shifted version of the desired cw-ESR spectrum centered at the origin instead of the Larmor frequency ω_0, as given by Eq. (10.28).

10.5 MSMs in Service of cw-ESR of Biomolecules

In this section, the classical molecular dynamics described in Sect. 10.3 and the quantum spin dynamics of Sect. 10.4 are integrated with the purpose of calculating cw-ESR spectra from MD simulations. The methodology is applied to T4 Lysozyme in Sect. 10.5.3, for which multifrequency cw-ESR spectra from 72R1 and 131R1 are available. Before presenting this application, however, we start in Sect. 10.5.1 by analyzing the results of a simple analytical model designed to illustrate the influence of motional time scales on the spectral line shape. This analysis helps provide a deeper understanding of the sensitivity and demands of cw-ESR on the time scales and duration of the classical molecular motions, which is discussed in Sect. 10.5.2.

10.5.1 Coupling Between Markov State Dynamics and Spin Relaxation

Rather than treating the problem in full generality, the effect of MSM relaxation rates on the spectrum will first be illustrated through the simplest possible example of coupling between a two-state MSM and a two-level spin system. This situation, known as chemical exchange, does arise naturally in magnetic resonance, especially NMR. In our case, it can be reached after a few simplifying assumptions.

For the sake of simplicity, we consider a spin label system in which the electron spin is not coupled to any nuclear spin (unlike a real nitroxide spin label). This leaves us with an ensemble of independent spin-1/2 systems. In this case, the coherence matrix $\rho_+(t)$ becomes a scalar, which is in fact equal to $M_+(t)$ (cf. (10.18)). Also, the hyperfine contribution to $H_z(t)$ in (10.23) is not present so $H_z(t) = |\gamma_\mathrm{e}| B G^{\mathrm{L}}_{zz}(t)/2$ is also a scalar. With these simplifications the evolution equation (10.22) reduces to

$$
\dot{M}_+(t) = \mathrm{i}|\gamma_\mathrm{e}| B G^{\mathrm{L}}_{zz}(t) M_+(t) = \mathrm{i}\omega(t) M_+(t),
\tag{10.33}
$$

where the last equality defines the precession frequency $\omega(t)$.

To incorporate molecular motion, we assume that the molecules to which the spins are attached can exist in two different conformations that exchange in a random manner via a hopping process. The conformations are taken to be magnetically distinguishable in the sense that the precession frequency $\omega(t)$ in (10.33) is equal to ω_1 in one of the conformations and to ω_2 in the other, with $\omega_1 \neq \omega_2$. If k_+ denotes the probability of transition from state 1 to 2 per unit time and k_- denotes the probability of transition from state 2 to 1 per unit time, the rate matrix for this two-state MSM is

$$K = \begin{bmatrix} -k_+ & k_+ \\ k_- & -k_- \end{bmatrix}. \qquad (10.34)$$

Using matrix notation, the Master equation (10.10) can be written as

$$\langle \dot{p}(t) \rvert = \langle p(t) \rvert K, \qquad (10.35)$$

where $\langle p(t)| = [p_1(t), p_2(t)]$, and p_1 and p_2 are the probabilities for the chain to be in states 1 and 2, respectively. The left eigenvector of K with eigenvalue zero is the equilibrium (row) vector $\langle \pi| = [\pi_1, \pi_2]$. The corresponding right eigenvector is the (column) vector $|1\rangle = [1, 1]^\top$. The only non-zero eigenvalue of K is $-k$, where $k \equiv k_+ + k_-$ is the sole relaxation time scale in this problem. (For comparison, the lifetimes of the two states are $\nu_1 = 1/k_+$ and $\nu_2 = 1/k_-$, according to (10.11).)

In cw-ESR one detects the transverse magnetization $\mathscr{M}_+(t)$ of the whole ensemble and not the probabilities of the two states. Let M_1 and M_2 denote the transverse magnetizations of the molecules in the two conformations *weighted by the respective probabilities* p_1 and p_2. In terms of the weighted magnetization (row) vector $\langle M(t)| = [M_1(t), M_2(t)]$, the magnetization averaged over the ensemble of molecules is $\mathscr{M}_+(t) = M_1(t) + M_2(t) = \langle M(t)|1\rangle$. As has been shown already by Anderson [2] and Kubo [28, 29], the evolution of the probability weighted magnetization vector is given by the joint dynamics of oscillatory motion (10.33) and exchange between the two states (10.35):

$$\langle \dot{M}(t) \rvert = \langle M(t) \rvert (\mathrm{i}\Omega + K). \qquad (10.36)$$

Here, K is the transition rate matrix from (10.34) and

$$\Omega = \begin{bmatrix} \omega_1 & 0 \\ 0 & \omega_2 \end{bmatrix} \qquad (10.37)$$

is a matrix containing the state-dependent precession frequencies along its main diagonal. For an equilibrated ensemble of spins with unit transverse magnetization the appropriate initial condition is $\langle M(0)| = \langle \pi|$. From (10.13), the cw-ESR spectrum is the Laplace transform of the transverse magnetization evaluated at iω. Taking the Laplace transform of both sides of the Kubo-Anderson equation (10.36), using the initial condition of $\langle M(t)|$, and taking an inner product with $|1\rangle$, leads to the following expression for the spectrum: $\mathscr{S}(\omega) = \langle \pi|(\mathrm{i}\omega - \mathrm{i}\Omega - K)^{-1}|1\rangle$. Differentiation with respect to ω, in analogy with (10.14), yields the derivative spectrum

$$\frac{\mathrm{d}\mathscr{S}(\omega)}{\mathrm{d}\omega} = -\mathrm{i}\langle \pi|(\mathrm{i}\omega - \mathrm{i}\Omega - K)^{-2}|1\rangle. \qquad (10.38)$$

As a numerical example, we choose $\omega_1 = 30$, $\omega_2 = 60$ and $k_+ = k/3$, $k_- = 2k/3$, where the parameter k allows us to vary the exchange rates from slow to fast. This choice of k_+ and k_- implies that $\pi_1 = 2/3$ and $\pi_2 = 1/3$ due to detailed balance. The calculated derivative spectra with the relaxation rate k ranging from 10 to 60 are shown in Fig. 10.8. For clarity, only the real parts of the complex spectra are plotted. From the figure, it is seen that for the slow exchange rate ($k = 10$) the spectrum consists of two lines centered at the two precession frequencies. The different intensity of the lines reflects differences in the equilibrium probabilities of the two conformations. When the exchange rate increases ($k = 15$) the centers of the two lines approach each other. At the same time the lines get broader. The approach and broadening of the lines leads to their eventual merger with further increase in the exchange rate ($k = 30$). After that point, the spectrum consists of only one line. Upon further speed up of the exchange ($k = 60$) the center of the single line shifts and its width decreases. For even faster rates of exchange (not shown) there is only one very narrow line centered at the average frequency $\pi_1\omega_1 + \pi_2\omega_2$, which is equal to 40 in our numerical example.

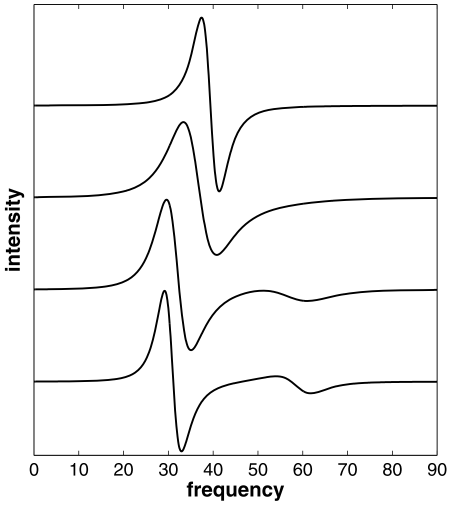


Fig. 10.8 Derivative spectra simulated using Eq. (10.38) with (from bottom to top) $k = 10, 15, 30, 60$. For clarity, the spectra are shifted along the vertical axis, which corresponds to the intensity of the spectrum in arbitrary units

In summary, this simple example illustrates that for motions with rates much faster than the spectral width, $W = |\omega_1 - \omega_2|$, the spectrum reflects the average of the magnetic properties. In contrast, for motions with rates much slower than the spectral width the magnetic properties do not experience averaging due to the dynamics. For intermediate exchange rates the spectral line shape is complex and its detailed structure shows great sensitivity to the rate of exchange. It is exactly in this motional regime that explicit simulations with stochastic models become necessary for the proper interpretation of the experimental situation. All the spectra in Fig. 10.3, for example, fall in this regime.

10.5.2 Time Scales of the cw-ESR Experiment

The width of the cw-ESR spectrum depends on the field at which the experiment is performed. For $B = 0.35$ T the hyperfine contribution to the propagator $U(t, \Delta t)$ in (10.32) dominates over the Zeeman contribution. Thus, at X-band the width is determined by the A_{zz} component of the hyperfine tensor as $W \approx 2A_{zz} \approx 90$ G. When the strength of the magnetic field is increased to $B = 3.4$ T, the anisotropy of the g tensor (multiplied by the field) is comparable to $2A_{zz}$. Hence, the Zeeman and hyperfine interactions contribute almost equally to the spectral width, yielding $W \approx 180$ G. At even higher fields, the spectral width is completely dominated by the anisotropy of the g tensor, which increases linearly with the field strength. Hence, for $B = 6.1$ T the spectral width can be estimated as $W \approx 320$ G.

The relationship between a signal and its Fourier transform implies that the width of a cw-ESR spectrum, W, is inversely proportional to the maximum time step, Δt, with which the classical dynamics should be followed: $\Delta t = 2\pi/W$. A similar inverse proportionality holds between the total duration of a dynamical trajectory, T, and the desired resolution of the cw-ESR spectrum, $\Delta\omega$: $T = 2\pi/\Delta\omega$. When the frequency axis is reported in units of magnetic field, like in Fig. 10.3, the conversion between magnetic field and angular frequency given in (10.25) needs to be employed. With γ_e as given under Eq. (10.1) and $G_0 \approx 1$, from $\mathbf{g}$ in (10.3) and the definition (10.2), we conclude that *1 Gauss corresponds to a time scale of roughly 360 ns*.

Using this conversion factor and the spectral widths estimated above, it can be deduced that for the simulation of cw-ESR spectra at X-band, the magnetization should be known roughly every $\Delta t \lesssim 4$ ns. The maximum allowable time step decreases to $\Delta t \lesssim 2$ ns for W-band and $\Delta t \lesssim 1$ ns for G-band spectra. These estimates of Δt reflect the temporal resolution with which the FID of the transverse magnetization needs to be known. In the simulation of cw-ESR spectra according to the motional model (10.8), the time step of the numerical integration has to be small enough to faithfully follow not only the FID but the rotational and MSM dynamics as well. Thus, the simulated stochastic process may impose additional demands on the temporal resolution, further reducing the values of Δt.

An extreme example of the discrepancy between the Δt required to follow the decay of the magnetization and the time step of the classical dynamics emerges in the approach (10.6), since the trajectories coming from atomistic MD simulations are typically sampled about every $\delta t =$

1 ps. In principle, one could use every snapshot from the trajectories and integrate the spin dynamics with this time step according to (10.31). However, the above estimates of Δt indicate that such an approach is unnecessary and wasteful. One option is to decimate the MD trajectories and use snapshots separated by about a thousand steps [12, 53]. An alternative that we prefer, which follows from the simple example in Sect. 10.5.1 and can be justified rigorously [48, 49], is to average the magnetic tensors over a time window Δt ($\Delta t \gg \delta t$) along each MD trajectory.

For the motional model (10.8), the same logic allows us to pre-average the magnetic tensors over the fast librations of the spin label, visible in the time-traces of its dihedral angles in Fig. 10.5. Using the fragments of the MD trajectories assigned to a given Markovian (macro)state, the magnetic tensors were averaged over the dynamics of N relative to M for each state. Such pre-averaging not only reduces the effective diagonal values of the magnetic tensors in the nitroxide-fixed frame (cf. (10.3)), but also leads to nitroxide frames which have state-dependent orientation relative to the protein frame. In fact, the state-dependent “nitroxide frame” calculated in this way is different for the **g** and **A** tensors since their anisotropies average differently.

The conversion between magnetic field and time implies that a for a spectral resolution of $\Delta\omega \approx 1$ G the FID has to be followed for $T \approx 360$ ns, which gives the necessary duration of a single stochastic trajectory. For spectra broadened by the molecular motion, like the ones shown in Fig. 10.3, a less fine resolution should be sufficient, thus bringing T down by a factor of 3 to 4 ($T \approx 90$–120 ns). The time scale T implied by the spectral resolution should be compared with the relaxation time scale $\tau_1 \gtrsim 100$ ns associated with the rare transition of the disulfide dihedral angle χ_3 between the two stable conformations m and p (Sect. 10.3.2). Since τ_1 falls beyond the time scale relevant for cw-ESR experiments at X-band and especially at higher frequencies, we conclude that its precise value is immaterial for the calculation of such spectra. The example in Sect. 10.5.1 implies that cw-ESR spectra can be simulated as a linear superposition of the separate contributions from the m and p conformations of the spin label R1 weighted by their relative populations. This justifies the use of umbrella sampling to determine the populations of the m and p states (Sect. 10.3.2), without any knowledge about the rate of their interconversion.

Table 10.4 Tumbling time scales, $\tau_D = 1/6D$, associated with diffusion coefficients D

$D/10^6\ \mathrm{s}^{-1}$	10	18	25	40
τ_D/ns	17	9.3	6.7	4.2

The considerations in the previous paragraphs suggest that cw-ESR experiments are very sensitive to motions in the time window of about 2–50 ns, which constitutes the middle of the estimated spectral time scale. Additional dynamics falling outside this time window are expected to have a lesser effect on the spectral line shape and be largely inaccessible on the background of the 2–50 ns motions. The time scales corresponding to the DNA tumbling rates used in the spectral simulations of Fig. 10.3 are compiled in Table 10.4. Clearly, they all fall in the window where the cw-ESR spectra are expected to be strongly affected. The additional diffusion coefficient in Table 10.4, with $\tau_D = 9.3$ ns, corresponds to the tumbling of the protein T4 Lysozyme in solution. On the basis of this time scale, we expect that the rotational diffusion of the protein has to be explicitly taken into account for quantitative simulation of cw-ESR spectra from spin-labeled T4 Lysozyme, to be examined in Sect. 10.5.3.

Let us use the developed intuition to rationalize the qualitative differences between the spectra shown in Fig. 10.3. As already mentioned, at X-band the effect of the g tensor is negligible and the spectral line shape is heavily dominated by the hyperfine tensor. Since **A** is an axial tensor, any differences in the rate of mixing of its components $A^{\mathrm{N}}_{xx} = A^{\mathrm{N}}_{yy}$ by the rotational diffusion of the macromolecule is inconsequential. This is the reason the spectra simulated using $D_\parallel = 40 \times 10^6\ \mathrm{rad}^2/\mathrm{s}$ (black) and $D_\parallel = 10 \times 10^6\ \mathrm{rad}^2/\mathrm{s}$ (blue) appear identical at $B = 0.34$ T (Fig. 10.3A). In contrast, spectra at $B = 0.34$ T are strongly influenced by differences in the rate of mixing between the $A^{\mathrm{N}}_{xx} = A^{\mathrm{N}}_{yy}$

components of the hyperfine tensor and A^{N}_{zz}, as illustrated by the spectra in Fig. 10.3A simulated using $D_{\perp} = 10 \times 10^6$ rad^2/s (black) and $D_{\perp} = 25 \times 10^6$ rad^2/s (red). At W-band, the g-tensor anisotropy influences the spectral line shape as much as the hyperfine tensor. Because the g tensor of the nitroxide distinguishes between all the three directions of the coordinate axes, the high-field spectra are sensitive to the rates of mixing induced by both $D_{\|}$ and $D_{\perp}$, as clearly seen in Fig. 10.3B.

By dwelling further on the spectra of Fig. 10.3 we hope to have convinced the reader that cw-ESR spectra are very sensitive to both the directionality and the magnitude of the molecular motions (reflecting the local structure) when they fall in the spectral time scale of the experiment. When the dynamics is either faster or slower than the spectral time scale, the spectrum still carries information about the average magnetic properties or the populations of the slowly exchanging conformations, as demonstrated by the example in Sect. 10.5.1. Therefore, quantitative comparison with cw-ESR spectra at several different frequencies, from X- to W- to G-band, should provide an unprecedented check on the structural and dynamical aspects of the internal spin-label dynamics captured by the MD simulations of T4 Lysozyme.

10.5.3 Multifrequency cw-ESR Spectra of Spin-Labeled T4 Lysozyme

Although the MSMs of the spin label R1 at positions 72 and 131 on T4L were constructed on the basis of the time-series of the five spin label torsion angles (Sect. 10.3.2), the influence of the environment is implicitly incorporated in two ways. First, the electrostatic and van der Waals interactions with the protein and the solvent molecules dictate which rotameric states of the spin label are populated and to what extent. Due to the internal flexibility and amphiphilic nature of the spin label, the populations of its conformations are sensitive to the subtle balance between various interactions, and are hard to predict on the basis of simplified steric and hydrodynamic arguments [47, 54]. Second, explicit protein and solvent dynamics on time scales up to about 100 ps was used to calculate pre-averaged magnetic tensors for each of the states of the MSMs, as described in Sect. 10.5.2. Therefore, in addition to the exchange between the rotamers, the rattling of the nitroxide in the solvent cage and the local thermal fluctuations of the protein backbone (as opposed to larger scale conformational changes, e.g. partial unfolding of the helices, etc.), are implicitly accounted for in the spectral simulations presented next.

Cw-ESR spectra for 72R1 and 131R1 in T4 Lysozyme were calculated for three different magnetic field strengths—0.33 T, 3.4 T and 6.1 T—according to the motional model (10.8). Isotropic rotational diffusion with diffusion constant $D = 18 \times 10^6$ rad/s (Table 10.4) was used to account for the global tumbling of the protein. In Figs. 10.9 and 10.10, the spectra calculated by using separately the m and p subblocks of the estimated transition probability matrices are shown on the left, and the final spectra obtained by linearly mixing the FID decays of the m and p states are shown on the right. For the three fields, the spectra from the m and p conformations of 131R1 are quite similar to each other and to the experimental spectra, with the difference increasing slightly with the increase of the field (Fig. 10.10). The agreement between the calculated and the experimental spectra is remarkably good over the entire field range. In the case of 72R1, the m and p contributions to the spectra are markedly different, with the latter being consistently more similar to the experimental spectrum for all the three field strengths (Fig. 10.9). At 0.33 T (9 GHz), the p component by itself is basically identical to the experimental spectrum, whereas adding 27 % of the m component is essential for the good agreement at the two higher fields.

By changing the ESR frequency from 9 GHz to 170 GHz the time window of sensitivity of the experiment is changed by about an order of magnitude. Also, whereas at 9 GHz the spectrum is dominated by the hyperfine tensor, at 95 GHz the contribution of the g tensor becomes more significant, and eventually dominates at 170 GHz. Therefore, the quantitative

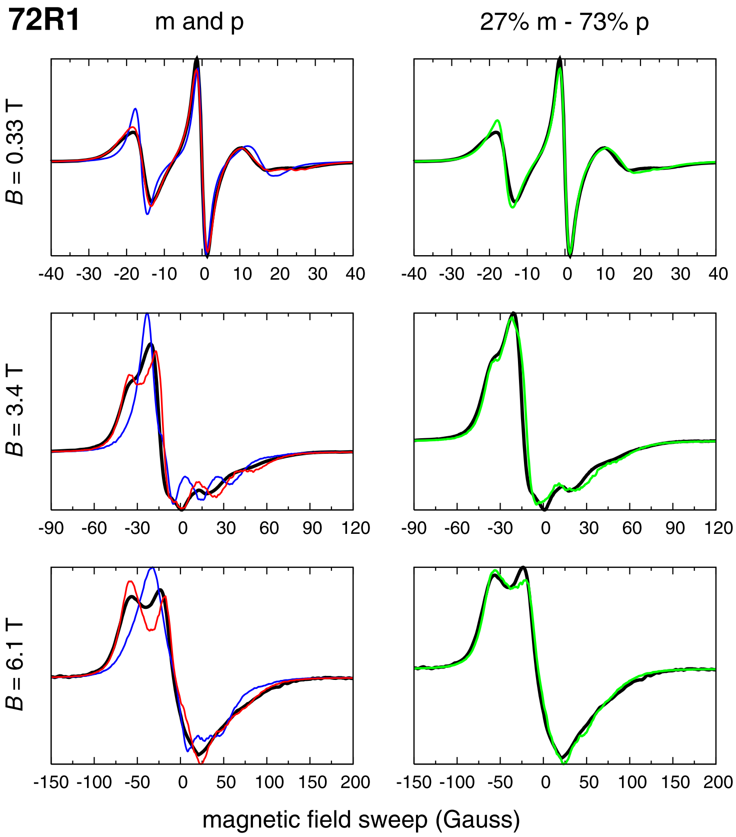


Fig. 10.9 Experimental spectra of 72R1 at 22 °C (*black*). *Left*: Calculated spectra of conformations m (*blue*) and p (*red*); *Right*: Spectra calculated by mixing the m and p conformations in the specified ratio (*green*). Simulation parameters given in Ref. [50]

agreement of the calculated spectra with experiment over the 9–170 GHz range is strongly suggestive that the dynamics of the spin label in the computer simulations is quite similar to the real underlying dynamics.

A unique dynamical model cannot be inferred on the basis of 9 GHz spectra alone. Multifrequency ESR analysis attempts to address this issue by placing additional restrictions on the nature of the microscopic dynamics. The constraints presented by multifrequency spectra are important, e.g., a single diffusional MOMD model is unable to simultaneously fit both the low and high frequency spectra from T4L [3, 33]. Achieving a simultaneous agreement is challenging even for state-of-the-art fitting approaches based on the SRLS model, which also raise the question of uniqueness of fit, even with the additional restrictions imposed by multifrequency cw-ESR [56]. It is notable that the present results, which have yielded spectra in excellent agreement with experiment at three different frequencies, were generated from a single microscopic model without any fitting parameters or *ad hoc* empirical adjustement of the model.

10.6 Summary and Future Outlook

A novel methodological framework was elaborated for the purpose of simulating cw-ESR spectra of spin labeled proteins from all-atom MD trajectories [49]. Within this framework, the information from multiple independent MD trajectories is employed to construct an MSM of the R1 dynamics in the space of its five dihedral angles. Using the transition probability matrix of the MSM determined from the MD simulations, long stochastic trajectories including rotational diffusion are generated to simulate realistic cw-ESR spectra [48]. This framework was used to study the conformations and dynamics of

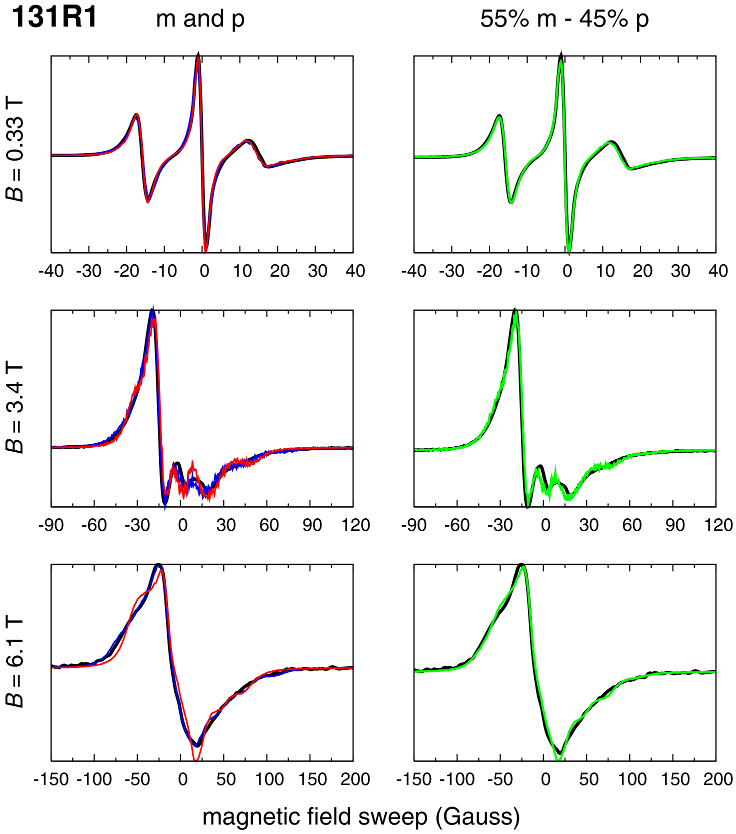


Fig. 10.10 Experimental spectra of 131R1 at 22 °C (*black*). *Left*: Calculated spectra of conformations m (*blue*) and p (*red*); *Right*: Spectra calculated by mixing the m and p conformations in the specified ratio (*green*). Simulation parameters given in Ref. [50]

the spin label R1 at positions 72 and 131 in T4 Lysozyme. For the first time, very good agreement with multifrequency cw-ESR experiments at three different magnetic field strengths was obtained. The atomically-detailed picture of the spin label emerging from the MD simulations helps to unify spectroscopic and crystallographic data and provides useful insight into their molecular origins.

The MSMs constructed from the MD simulations can be viewed as a natural extension of the multicomponent SRLS model, in which the spin label dynamics is modeled as a linear superposition of several independent motional modes, each characterized by a microscopic ordering potential and a rotational diffusion tensor. It is our hope that the overall perspective developed from the MD simulations can help design better motional models tailored to the specific spin label and biomolecule to which it is attached. Going beyond a universal, generic stochastic model is expected to be of crucial importance, given the extensive applications of SDSL to diverse biological systems and the increased availability of high-field ESR.

Accurate calculations of multifrequency cw-ESR spectra by mapping MD trajectories onto MSMs are extremely challenging because they require that a whole host of molecular motions be accounted for correctly, not only in terms of their amplitudes and resulting equilibrium populations, but also in terms of their dynamical timescales. While the observables from many experimental methods (e.g. FRET, NMR, hydrogen exchange) are often dominated by one or a few relaxation modes, multifrequency cw-ESR spectra provides perhaps one of the rare applications of the methodology where a large fraction of the set of eigenvalues and eigenvectors of the MSM rate matrix is truly put to the test. In this context, the present success in quantitatively reproducing experimental multifrequency cw-ESR spectra, achieved without any empirical adjustment, is truly remarkable. From a broader perspective, the ESR/MSM methodology elaborated here offers a powerful route to test and validate the ability of existing force fields to reproduce both structural and dynamical aspects of the molecular motions as reported by the spin label. It will be of great interest to carry out additional ESR/MSM simulations to cover a range of experimental conditions (e.g., temperature, viscosity) to further test their ability to predict cw-ESR spectra under different conditions.

References

1. Abragam A (1961) The principles of nuclear magnetism. Oxford University Press, New York
2. Anderson PW (1954) A mathematical model for the narrowing of spectral lines by exchange or motion. J Phys Soc Jpn 9(3):316–339
3. Barnes JP, Liang Z, Mchaourab HS, Freed JH, Hubbell WL (1999) A multifrequency electron spin resonance study of T4 Lysozyme dynamics. Biophys J 76(23):298–3306
4. Beier C, Steinhoff HJ (2006) A structure-based simulation approach for electron paramagnetic resonance spectra using molecular and stochastic dynamics simulations. Biophys J 91:2647–2664
5. Bennati M, Prisner TF (2005) New developments in high field electron paramagnetic resonance with applications in structural biology. Rep Prog Phys 68(2)
6. Borbat PP, Costa-Filho AJ, Earle KA, Moscicki JK, Freed JH (2001) Electron spin resonance in studies of membranes and proteins. Science 291:266–269
7. Budil DE, Lee S, Saxena S, Freed JH (1996) Nonlinear-least-squares analysis of slow-motion EPR spectra in one and two dimensions using a modified Levenberg-Marquardt algorithm. J Magn Reson, Ser A 120:155–189
8. Budil DE, Sale KL, Khairy KA, Fajer PG (2006) Calculating slow-motional electron paramagnetic resonance spectra from molecular dynamics using a diffusion operator approach. J Phys Chem A 110:3703–3713
9. Cekan P, Sigurdsson ST (2009) Identification of single-base mismatches in duplex DNA by EPR spectroscopy. J Am Chem Soc 131(50):18,054–18,056
10. Columbus L, Hubbell WL (2002) A new spin on protein dynamics. Trends Biochem Sci 27:288–295
11. Columbus L, Kalai T, Jeko J, Hideg K, Hubbell WL (2001) Molecular motion of spin labeled side chains in α-helices: analysis by variation of side chain structure. Biochemistry 40:3828–3846
12. DeSensi SC, Rangel D, Lybrand TP, Hustedt EJ (2008) The calculation of nitroxide cw-EPR spectra from Brownian dynamic trajectories and molecular dynamics simulations. Biophys J 94(10):3798–3809
13. Earle KA, Dzikovski B, Hofbauer W, Moscicki JK, Freed JH (2005) High-frequency ESR an ACERT. Magn Reson Chem 43:S256–S266
14. Eviatar H, van Faassen E, Levine Y, Hoult D (1994) Time-domain simulation of ESR spectra of nitroxide spin probes. Chem Phys 181:369–376
15. Eviatar H, van der Heide U, Levine YK (1995) Computer simulations of the electron spin resonance spectra of steroid and fatty acid nitroxide probes in bilayer systems. J Chem Phys 102:3135–3145
16. Fanucci GE, Cafiso DS (2006) Recent advances and applications of site-directed spin labeling. Curr Opin Struct Biol 16:644–653
17. Fedchenia II, Westlund PO, Cegrell U (1993) Brownian dynamics simulation of restricted molecular diffusion. The symmetric and deformed cone models. Mol Simul 11:373–393
18. Fleissner MR, Cascio D, Hubbell WL (2009) Structural origin of weakly ordered nitroxide motion in spin-labeled proteins. Protein Sci 18(5):893–908
19. Freed JH (1976) Theory of slow motional ESR spectra for nitroxides. In: Berliner LJ (ed) Spin labeling: theory and application. Academic Press, New York, pp 53–132
20. Freed JH (2000) New technologies in electron spin resonance. Annu Rev Phys Chem 51:655–689
21. Guo Z, Cascio D, Hideg K, Kalai T, Hubbell WL (2007) Structural determination of nitroxide motion in spin-labeled proteins: tertiary contact and solvent-inaccessible sites in helix G of T4 Lysozyme. Protein Sci 16:1069–1086
22. Hakansson P, Westlund PO, Lindahl E, Edholm O (2001) A direct simulation of EPR slow-motion spectra of spin labelled phospholipids in liquid crystalline bilayers based on a molecular dynamics simulation of the lipid dynamics. Phys Chem Chem Phys 3:5311–5319
23. Halle B (2009) The physical basis of model-free analysis of NMR relaxation data from proteins and complex fluids. J Chem Phys 131(22):224,507–224,522
24. Hartigan JA (1975) Clustering algorithms. Wiley, New York
25. Jiao D, Barfield M, Combariza JE, Hruby VJ (1992) Ab initio molecular orbital studies of the rotational barriers and the sulfur-33 and carbon-13 chemical shieldings for dimethyl disulfide. J Am Chem Soc 114(10):3639–3643
26. Klare JP, Steinhoff HJ (2009) Spin labeling EPR. Photosynth Res 102(2–3):377–390
27. Krstic I, Endeward B, Margraf D, Marko A, Prisner TF (2012) Structure and dynamics of nucleic acids. Top Curr Chem 321:159–198
28. Kubo R (1954) Note on the stochastic theory of resonance absorption. J Phys Soc Jpn 9(6):935–944
29. Kubo R (1969) A stochastic theory of line shape. Adv Chem Phys 15:101–127

30. Kuprusevicius E, White G, Oganesyan VS (2011) Prediction of nitroxide spin label EPR spectra from MD trajectories: application to myoglobin. Faraday Discuss 148:283–298
31. Langen R, Oh KJ, Cascio D, Hubbell WL (2000) Crystal structures of spin labeled T4 Lysozyme mutants: implications for the interpretation of EPR spectra in terms of structure. Biochemistry 39:8396–8405
32. Levitt MH (2008) Spin dynamics: basics of nuclear magnetic resonance, 2nd edn. Wiley, Chichester
33. Liang Z, Lou Y, Freed JH, Columbus L, Hubbell WL (2004) A multifrequency electron spin resonance study of T4 Lysozyme dynamics using the slowly relaxing local structure model. J Phys Chem B 108:17,649–17,659
34. Maragakis P, Lindorff-Larsen K, Eastwood MP, Dror RO, Klepeis JL, Arkin IT, Jensen MO, Xu H, Trbovic N, Friesner RA, Palmer AG, Shaw DE (2008) Microsecond molecular dynamics simulation shows effect of slow loop dynamics on backbone amide order parameters of proteins. J Phys Chem B 112(19):6155–6158
35. Mchaourab HS, Kalai T, Hideg K, Hubbell WL (1999) Motion of spin-labeled side chains in T4 Lysozyme: effect of side chain structure. Biochemistry 38:2947–2955
36. Mchaourab HS, Lietzow MA, Hideg K, Hubbell WL (1996) Motion of spin-labeled side chains in T4 Lysozyme. Correlation with protein structure and dynamics. Biochemistry 35:7692–7704
37. Mchaourab HS, Steed PR, Kazmier K (2011) Toward the fourth dimension of membrane protein structure: insight into dynamics from spin-labeling EPR spectroscopy. Structure 19(11):1549–1561
38. Meirovitch E, Nayeem A, Freed JH (1984) Analysis of protein-lipid interactions based on model simulations of electron spin resonance spectra. J Phys Chem 88:3454–3465
39. Norris JR (1997) Markov chains. Cambridge University Press, Cambridge
40. Polimeno A, Freed JH (1993) A many-body stochastic approach to rotational motions in liquids. Adv Chem Phys 83:89–210
41. Polimeno A, Freed JH (1995) Slow motional ESR in complex fluids: the slowly relaxing local structure model of solvent cage effects. J Phys Chem 99:10,995–11,006
42. Rabiner LR (1989) A tutorial on hidden Markov models and selected applications in speech recognition. Proc IEEE 77:257–286
43. Redfield AG (1957) On the theory of relaxation processes. IBM J Res Dev 1:19–31
44. Robinson BH, Slutsky LJ, Auteri FP (1992) Direct simulation of continuous wave electron paramagnetic resonance spectra from Brownian dynamics trajectories. J Chem Phys 96:2609–2616
45. Roux B (1994) The calculation of the potential of mean force using computer simulations. Comput Phys Commun 91:275–282
46. Schneider DJ, Freed JH (1989) Spin relaxation and motional dynamics. Adv Chem Phys 73:387–527
47. Sezer D, Freed JH, Roux B (2008) Parametrization, molecular dynamics simulation, and calculation of electron spin resonance spectra of a nitroxide spin label on a polyalanine alpha-helix. J Phys Chem B 112(18):5755–5767
48. Sezer D, Freed JH, Roux B (2008) Simulating electron spin resonance spectra of nitroxide spin labels from molecular dynamics and stochastic trajectories. J Chem Phys 128(16):165,106–165,116
49. Sezer D, Freed JH, Roux B (2008) Using Markov models to simulate electron spin resonance spectra from molecular dynamics trajectories. J Phys Chem B 112(35):11,014–11,027
50. Sezer D, Freed JH, Roux B (2009) Multifrequency electron spin resonance spectra of a spin-labeled protein calculated from molecular dynamics simulations. J Am Chem Soc 131(7):2597–2605
51. Sezer D, Sigurdsson ST (2011) Simulating electron spin resonance spectra of macromolecules labeled with two dipolar-coupled nitroxide spin labels from trajectories. Phys Chem Chem Phys 13(28):12,785–12,797
52. Steinhoff HJ, Hubbell W (1996) Calculation of electron paramagnetic resonance spectra from Brownian dynamics trajectories: application to nitroxide side chains in proteins. Biophys J 71:2201–2212
53. Stoica I (2004) Using molecular dynamics to simulate electronic spin resonance spectra of T4 Lysozyme. J Phys Chem B 108(5):1771–1782
54. Tombolato F, Ferrarini A, Freed JH (2006) Dynamics of nitroxide side chain in spin-labeled proteins. J Phys Chem B 110:26,248–26,259
55. Usova N, Westlund PO, Fedchenia I (1995) Direct simulation of slow-motion electron spin resonance spectra by solving the stochastic Liouville equation in time domain with stochastic dynamics in the form of trajectories. J Chem Phys 103:96–103
56. Zhang Z, Fleissner MR, Tipikin DS, Liang Z, Moscicki JK, Earle KA, Hubbell WL, Freed JH (2010) Multifrequency electron spin resonance study of the dynamics of spin labeled T4 Lysozyme. J Phys Chem B 114(16):5503–5521

Software for Building Markov State Models

11

Gregory R. Bowman and Frank Noé

11.1 MSMBuilder

MSMBuilder is an open source software package for constructing Markov state models. The software is primarily written in python to facilitate rapid integration of new methods and readable implementations of existing methods. Time-sensitive routines are implemented in C with python wrappers to enhance the performance of the software. At the time of this writing, stable public releases of MSMBuilder are available via the SimTk software repository (https://simtk.org/home/msmbuilder; http://msmbuilder.org). The current development build is also available via GitHub (https://github.com/SimTk/msmbuilder).

11.2 EMMA

EMMA is a software library and set of command-line tools for constructing, validating and analyzing Markov state models. The library is written in Java, thus being fast and platform-independent without requiring local compilation or installation. At the time of this writing, stable public releases of EMMA are available via the SimTk software repository (https://simtk.org/home/emma). The Java source code is available on request. As soon as publicly available, it will be linked on the simtk.org website.

G.R. Bowman (✉)
Departments of Molecular & Cell Biology and
Chemistry, University of California, Berkeley, CA 94720,
USA
e-mail: gregoryrbowman@gmail.com

F. Noé
Institut für Mathematik II, Freie Universität Berlin,
Arnimallee 2-6, 14195 Berlin, Germany

G.R. Bowman et al. (eds.), *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology 797,
DOI 10.1007/978-94-007-7606-7_11, © Springer Science+Business Media Dordrecht 2014