Off- and Online Detection of Dynamical Phases in Time Series

with Applications to Molecular Dynamics

Vom Fachbereich Mathematik und Informatik
der Freien Universität Berlin
zur Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften
genehmigte Dissertation

vorgelegt von

Eike Meerbach

Berlin, Dezember 2008

Betreuer: Prof. Dr. Christof Schütte
Freie Universität Berlin
Fachbereich Mathematik und Informatik
Arnimallee 6
14195 Berlin

Gutachter: Prof. Dr. Christof Schütte
Prof. Dr. John H. Maddocks (Lausanne, Switzerland)

Disputationsdatum: 09.04.2009

Dedicado ao Mar

Contents

1 Introduction

The analysis of complex data is an important but yet challenging problem. Important, as nowadays data is generated with an increasing speed and complexity from experiments, simulations, measurements or data logging. Challenging, due to the complex structure of these data and the need for (fast) automatic reduction of this complexity to aid in decision making, modelling and simulation. A large fraction of these data occurs in the form of time series, i.e. data measured or obtained at subsequent points in time. An obvious task when dealing with time series data is the construction of "easy" models that

- can be used to approximate and explain the dynamics of the observed data.
- are able to predict the (unobserved) future of the time series.

Such models should be "easy" on two counts, first, it should be possible to parameterise them with respect to a given time series without too much computational effort, second, they should reproduce the "important" characteristics of the observed dynamic while suppressing unimportant details. Prominent examples of such models are, e.g., *autoregressive* (AR), *vector autoregressive* (VAR) and *moving average* models (MA) [73].

However, faced with time series data from complex dynamical systems, the application of standard models will yield only poor results, since challenges are:

- Complex systems may be *high dimensional* and therefore some kind of dimension reduction or another algorithmic strategy that takes the dimensionality into account may be required.
- It may be hard to obtain time series data which sufficiently reflects the dynamical properties of the observed system (in some contexts this is referred to as the *sampling problem*).
- The dynamical behaviour may change over time, i.e. there are different phases in the time series which should be approximated by different models.

An example for such system are time series obtained from climate observations, they are often very high dimensional, change their dynamical properties over the year and it should be obvious that a description of a typical temperature curve over a year should not only be based on data obtained during summer months [51, 56].

The biocomputing group at the Free University Berlin started to address these questions of modelling, complexity reduction and sampling of complex systems about ten years ago emerging from a specific context: the dynamical analysis of biomolecules. The dynamic of biomolecules is important, since their geometric properties are essential for their function [22, 44, 67, 91, 101]. Therefore, a dynamical description of the geometry is needed to understand their role in biological processes. The dynamic of the geometrical structure of molecules is most often studied on the basis of time series of atomic coordinates obtained from *Molecular Dynamics* [2]. These time series are subject to all the challenges mentioned above, since:

- The number of atoms in a biomolecular system might be very large and, therefore, the time series are often high dimensional.
- Due to the existence of multiple time scales, simulation of such systems is time consuming.
- Typically biomolecules switch between different (meta)stable geometric structures, the observed time series will switch between different dynamical regimes.

The existence of different dynamical regimes means that a dynamical description of the observed system will be highly dependend on the time scale under consideration. On larger time scales the dynamics is characterised by changes of the global geometric structure, while on shorter time scales local flexibility around a globally stable geometric state will be observed, i.e. fluctuations of the system around some mean configuration. We call such a global geometrical state of the system *together* with its flexibility a *conformation*. If there is a time scale separation between local flexibility and changing of the global conformations, the dynamics of the jump process between different conformations will approximately be Markovian.

Although the dimensionality of a biomolecular system can often be reduced in terms of a small number of essential degrees of freedom [3], e.g. the torsion or backbone angles of the molecule under consideration, the problem of efficient algorithmic identification of persistent conformations from a given time series is still a challenging problem. Based upon a solid theoretical foundation Deuflhard, Schütte et al. developed a set-orientated approach for the identification of conformations [19, 28–30, 88, 107, 108], which relies on the analysis of a transition matrix obtained from a careful discretisation of the observable space into discrete states. The transition matrix is set up by counting the transitions between these discrete states in the time series. Normalisation yields a stochastic matrix whose spectral properties can be used to identify the conformations as metastable sets of states of the obtained Markov chain. This procedure is called *Perron cluster cluster analysis (PCCA)*. If the metastable sets are identified, a *discrete reduced model* describing only the conformational changes can be set up by estimating a transition matrix from the observed transitions between the identified metastable sets in the time series at hand.

More recently, approaches based on *hidden Markov models (HMM)* were put forward [36, 52–54, 106]. The underlying idea is that a discrete and hidden, i.e. non-observable, switch process governs the hopping between different conformations and the dynamics of the observed is dependent on the conformations, i.e. the state of the hidden switch process. Again the hidden process is assumed to be Markovian, while dynamic changes of the observable due to conformational changes are supposed to correspond to changes in the parameterisation of an assumed (local) dynamical model. The assumed form of the local dynamic can range from independent Gaussians [36], stochastic differential equations [106] to even non-Markovian dynamics [76].

One of the advantages of the HMM approach is that it does not rely on geometrical separation of the conformations within the observation space, which is important since dimension reduction from the full positional coordinate set to a low dimensional manifold may let conformations overlap. Another advantage is that it provides not only a model for the conformational switching but also a model for the dynamics of the observable within a conformation.

However, the computational effort for fitting the model in this approach is increased, as HMM approaches rely on optimisation of a high dimensional likelihood function, and a clear criteria for the number of metastable sets, which in the PCCA approach can be deduced from the spectral properties of the transition matrix, is lost. Therefore, in applications it turned out to be fruitful to combine PCCA with the HMM approaches [48, 75, 78, 109].

A good deal of this thesis is concerned with the question if such an analysis can also be done *on-line*. An on-line algorithm is an algorithm which can solve a problem while receiving the data package-wise, opposed to an *off-line* algorithm which requires the whole data set at once to solve the problem at hand. The interrelation between on- and off-line analysis with respect to time series with different dynamical phases is revealed by taking a closer look at the functioning of the listed off-line algorithms, i.e. PCCA and the HMM-variants. Since their aim is to estimate a *data-based* model which reflects the existence of different dynamical phases, they synchronously partition a given time series into segments, according to different dynamical phases, and estimate local models based on data contained in segments belonging to the same dynamical phase. The segmentation of a time series can be done on-line as long as it is possible to apply an algorithm that detects dynamical changes in the time series while scanning it sequently. If a time series segment $Z = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_T\}$ is given and the data point $\boldsymbol{z}_t$ is assumed to be generated by

$$\boldsymbol{z}_t = f(\boldsymbol{z}_{t-p}, \ldots, \boldsymbol{z}_{t-1}, \boldsymbol{\theta}),$$

where f, a (possible stochastic) function dependent on the past p values and some parameter $\boldsymbol{\theta}$, represents the dynamical model. The problem can be formulated as deciding if there is a time point $c \in \{p+1, \ldots, T-1\}$ such that

$$\boldsymbol{z}_t = \begin{cases} f(\boldsymbol{z}_{t-p}, \ldots, \boldsymbol{z}_{t-1}, \boldsymbol{\theta}_1), & \text{for } t \leq c \\ f(\boldsymbol{z}_{t-p}, \ldots, \boldsymbol{z}_{t-1}, \boldsymbol{\theta}_2), & \text{for } t > c, \end{cases}$$

with different parameters $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$. This is called a *change point detection* problem. Although the matter of change point detection received considerable attention in the last years, e.g. [5, 17, 20, 32, 40, 64, 70, 93, 94, 99], there is no standard solution to the problem and a number of competitive approaches exists. In this work we adopt a Bayesian approach as it provides a natural way to include parameter uncertainty. However, boon and bane of Bayesian approaches is the need for specification of prior distributions for all parameters. A resort is given by *objective Bayesian strategies* [8,17,25,26,40,63,89,97,113] which essentially use some information content of the data to specify prior distributions. We will show that the *fractional Bayes* [89] and the *imaginary minimal experiment* approach [113] are suitable to the on-line segmentation problem with respect to a wide class of local models.

The advantages having an on-line algorithm for the detection of dynamical phases in time series are two-fold. First, they sometimes produce cheaper and more reliable results as HMM approaches, since the difficult optimisation problem that comes along with HMM approaches is avoided. Second, it can be used in algorithmic settings where some action has to be performed after a dynamical change occurred in the observed time series. An example is given by an application of *distributed computing* for exit rate estimation in biomolecular system, proposed by Art Voter [122, 124]. As said, the off-line algorithms described above rely on having a time series which contains all relevant information about the dynamics. However, the switch from a biomolecular conformation to another is a rare event and therefore a time series which contains sufficient transition events to get a good estimate of the rate might be hard to obtain, since the computational effort for an extensive sampling of such event by molecular dynamics is too high. Art Voter suggested to speed up the simulation by simulating uncorrelated replicas of the system on many processors. Under the assumption of an exponentially distributed exit time from one conformation to another, i.e. assuming that the switching process is essentially Markovian, it can be shown that the time obtained by summing up the time elapsed on all processors until a change occurred on one of them reflects the same statistics as the exit rate in question. Thus, by using many processors it is possible to speed up the simulation and thereby enable the generation of a sample of exit times to estimate the rate more accurately, as long as an on-line detection of the exit event from some conformation is possible. We will show how to solve this problem by the application of the developed change-point algorithm.

The outline of this thesis is the following: In Chapter 2 we start with a short introduction in conformation dynamics of biomolecules with a focus on peptides, since they emerge in many examples throughout this work, additionally, molecular dynamics as a way to simulate the dynamics of biomolecules is introduced and the computational obstacles in the analysis of the so obtained time series are outlined.

An overview over PCCA and HMM approaches is given in Chapter 3, which, even if developed within the applicational scope of conformational analysis of biomolecules, are in fact data analysis algorithms in principle applicable to any given time series and especially suited to handle with time series exhibiting different dynamical phases. In particular we show how to unify the different existing HMM approaches with the help of the class of *vector autoregressive processes (VAR)* and how to apply these algorithms to obtain reduced models for complex systems. Based upon the obtained unified representation via VAR processes we will derive an algorithm for on-line change point detection in Chapter 4. We will demonstrate how to apply it on a series of test examples and on time series obtained from molecular dynamics. Thereby, we show how to cluster identified time series segments efficiently in a post processing step, without the effort of an off-line analysis.

Finally, in Chapter 5, we illustrate how to use the obtained on-line algorithm to employ the distributed computing approach of Art Voter for the estimation of exit rates in biomolecular systems.

Note that most of the molecular examples given in this work were published before in one of the authors publications [36, 48, 75–78, 95, 105, 109], but all were revised and adopted to the HMM-VAR framework developed here.

Acknowledgements

The author would like to thank Tomaso Frigato for performing the MD-simulation of the peptide used in the example in § 4.4.3, and is indebted to Gerhard Stock (Frankfurt) for the courtesy of making the simulation data of penta-alanine analysed in § 3.4.4 available to us. Most of the visualisation of molecular systems is done with the freeware software package Visual Molecular Dynamics (VMD, [58]), for which the developers deserve a special thanks. With the exception of the simulation done by Tomaso Frigato who used NWChem [13, 66] all molecular simulations where done by the free GROMACS package [71, 114], whose developers deserve a super-special thanks.

Last but not least the author would like to thank all members of the Biocomputing Group at the Free University Berlin for a good time. Christof Schütte and Juan Latorre Christobal deserve another special thanks, the first for keeping the science business running and therefore making this thesis possible, the second for proofreading and fruitful discussions about the English language.

This work was funded by the DFG Research Centre MATHEON "Mathematics for Key Technologies" (FZT86) in Berlin.

2 Conformation Dynamics of Biomolecules

2.1 Biomolecules

The notion of biomolecules is a collective term for all biological active molecules. There is a large variety of biomolecules differing considerably in chemical compound, size, function and form. There are small biomolecules like vitamines, hormones and sugar, medium sized like peptides and proteins and large ones like the deoxyribonucleic acid (DNA), ranging from a few atoms to hundreds of million atoms. Biomolecules consist primarily of carbon (C) and hydrogen (H), along with nitrogen (N), oxygen (O), phosphorus (P) and sulfur (S).

This thesis is not primarily concerned with biomolecules and its different functions, it is concerned with the analysis of data coming from time resolved observations/simulations of biomolecules. The examples to demonstrate the algorithms proposed in this thesis are peptides, small sized amino acid chains which are themselves the building blocks of proteins. Therefore, we are going to make a quick introduction to the peptide world in the next section.

2.2 Proteins and Peptides

Proteins and peptides are chains built from so-called proteinogenic amino acids, which are twenty different amino acids found in biomolecules. The distinction between peptides and proteins is merely a non-sharp distinction in amino acid chain length. As a rule of thumb one can call a chain consisting of less than 50 amino acids a peptide and a longer chain a protein.

Amino acids themselves consist of an amino group connected to a carboxyl group via a central carbon atom, called the α-carbon (C_α). Attached to the C_α there is a side group, whose size ranges from a single hydrogen atom to more complex structures with over a dozen atoms, that determines the type of the amino acid, e.g. if the α-carbon is bound to a methyl group (CH_3) the amino acid is called alanine, while if the side group is a single hydrogen atom it is called glycine. If two amino acids fuse to a peptide fragment a hydrogen atom splits off the amino group of one amino acid, while the carboxyl-group of the other one releases an oxygen and a hydrogen atom. This enables the amino and the carboxyl group to form a so-called peptide bond while a water

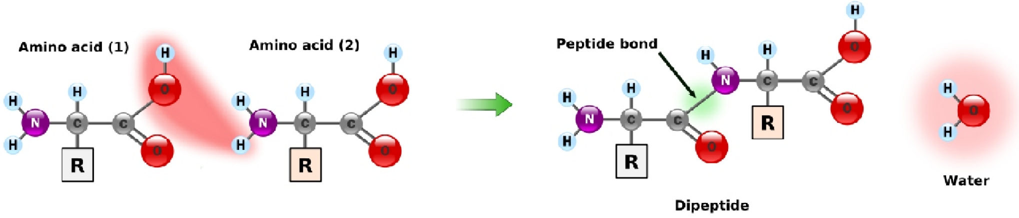


Figure 2.1: The formation of a peptide bond between two amino acids.

molecule is dispensed. A graphical sketch[1]of this process is shown in Fig. 2.1.

An important property of the peptide bond is its planarity, i.e. the atom-chain $C_\alpha - C - N - C_\alpha$ connecting the two amino acids lies on a plain. If on this plain the C_α atoms are both lying on the same side of the C-N-axis, i.e. the dihedral bond ω along $C_\alpha - C - N - C_\alpha$ is $\omega = 0°$, the bond is said to be in cis-conformation, or otherwise in trans-conformation ($\omega = 180°$). In (folded) proteins and peptides the trans-conformation is by far the prevalent conformation.

The repeated $C_\alpha - C - N$ chain connecting the amino acids in a peptide is called the backbone of a peptide. Due to the planarity and stability of the peptide bond, there are only two flexible degrees of freedom for each adjacent pair of amino acids along the backbone, the Φ-angle, which is the dihedral angle specified by $C - N - C_\alpha - C$, and the Ψ-angle along $N - C_\alpha - C - N$. Therefore, disregarding the side chains, the global structure of a peptide can be characterised by the sequence of Φ/Ψ pairs along the backbone. An illustration is given in Fig. 2.2.

2.2.1 Secondary and Tertiary Structure

Even if flexible, the Φ/Ψ angles of a peptide chain are not freely rotating. Steric hindrance and the possibility of H-bond bridging between different peptide bond units restricts them to specific regions in the $(-180, 180]^2$ plane, the so-called Ramachandran plane [98], which are similar for most of the amino acids. In larger peptide chains or proteins this picture is even more restricted. Subsequent amino acids stabilise via H-bond bridges to motifs like helices and so-called β-sheets, characteristic to these motifs are certain Φ/Ψ combinations common to the involved amino acids. These global motifs are called the secondary structures of a peptide chain (while the primary structure is defined through the sequence of amino acids). Finally, the geometry of proteins can be specified by the sequence of secondary structure motifs, which is called the tertiary structure. The relation between Φ/Ψ angle combinations, secondary

[1] The illustration is a modification of an image on Wikipedia which can be accessed via *http://en.wikipedia.org/wiki/Amino_acids.*

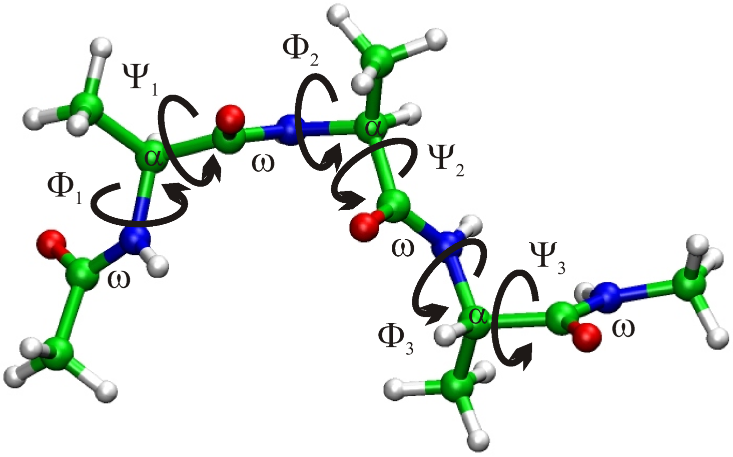


Figure 2.2: The global configuration of a peptide chain is largely fixed by the sequence of Φ/Ψ angles along the backbone. Here an 3-Alanine, a peptide of three alanine amino acids (specified by a methyl side chain $-CH_3$), is depicted. The α-carbons are marked by α and the flexible Φ/Ψ angles are sketched by arrows. Labelled with ω are the, under normal conditions very stable, peptide bond angles.

and tertiary structure is depicted in Fig. 2.3. We just remark, that the mechanism which drives a specific primary structure to a secondary, resp. tertiary, structure, the so-called folding process, is still not fully understood and subject to intensive research.

The importance of understanding the folding process lies in the fact that the geometrical structure of a biomolecule is essential for its function. A misfolding or refolding can correspond to malfunction or diseases, e.g. prion diseases are believed to be caused by a change from a predominantly α-helical tertiary structure of a protein to a tertiary structure containing predominantly β-sheets, triggered by misfolded proteins [22, 101]. But important dynamical events are not restricted to folding and misfolding, folded proteins often show a (localised) flexibility and geometrical alteration of parts which acts as a switch between different functionalities [44, 91]. In drug design flexibility is not only an important question from the functional perspective but also from a geometric perspective. In the *key-lock* approach one tries to find small ligands which binds to a specific site of a protein to either disable the function of the protein or to inhibit the binding site so that other harmful molecules can not bind. Besides the difficulties of finding an appropriate ligand which fits to the binding site and can somehow be transported to it and, of course, without disproportional adverse effects, one has to be concerned with dynamical properties, since both the form of the ligand and the form the protein may change over time [15, 117].

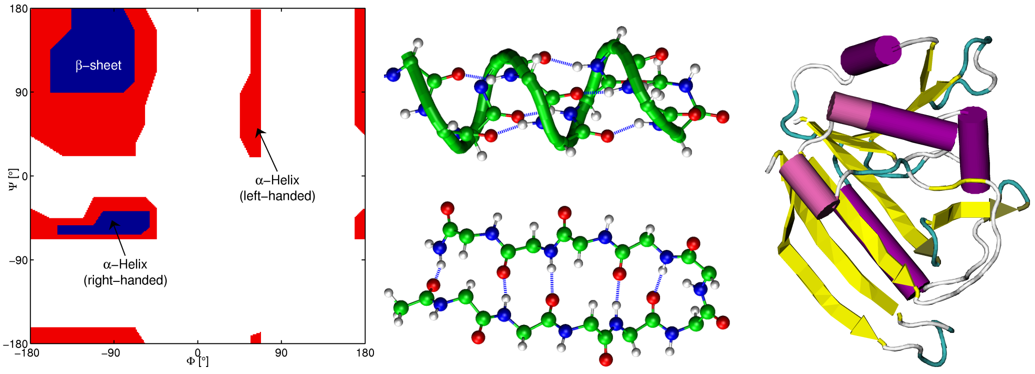


Figure 2.3: *Left*: The Ramachandran plot shows the most favourable energetic regions in the Ramachandran plane and the associated secondary structures. *Middle (top)*: A peptide fragment exhibiting the α-helical secondary structure. For better illustration side chains are omitted and the backbone form is depicted by a tube. The dotted blue lines indicate the H-bond bridges between subsequent amino acids which stabilise the structure. *Middle (down)*: A peptide fragment exhibiting β-sheet secondary structure, again side chains are not shown. *Right*: The tertiary structure consists of a sequence of secondary structures. Shown here is the tertiary structure of Escherichia coli dihydrofolate reductase [14] (PDB entry: 7DFR). The tubes indicate α-helices, while the arrows indicate β-sheets.

2.2.2 Conformations

Since the function of biomolecules, i.e. their interaction with the environment, depends upon their geometrical structure, it is important to identify stable geometrical structures, i.e. geometrical structures which are persistent over some time span of interest. We will call these (meta)stable structures conformations. The definition of a conformation does not only depend upon the time scale of interest but also upon the geometric property of interest, i.e. there might be a part of the molecule which remains stable over a certain time while other parts of the molecule are flexible.

Our notion of a conformation is sometimes used differently in the biomolecular context, where it refers to an energetically favourable structure of a molecule which is often understood as a the structure corresponding to a local minimum of some (potential) energy function of the biomolecular system. In this conception, conformations are rigid structures which are in some sense (locally) optimal. Opposed to that, we have a dynamic understanding of the notion conformation. For example, an α-helical structure of a peptide is a stable structure in the sense that the global structure, the helix, persists over a comparatively long time scale but still there are flexible degrees of freedom, like rotating side chains or end groups, moving on a faster time scale. That is, in our conception a conformation is not a fixed structure but a structure belonging to some dynamical regime, where certain invariant structural properties do not change over time. The appropriate picture would be a low energy region wrt. to the (potential) energy function and not a minimal point.

2.3 Molecular Dynamics

As it should have become clear in the previous section, an understanding of dynamical properties of biochemical systems is an important issue. A major challenge lies in the fact that it is not possible to monitor a molecular system time resolved on an atomic scale. Even though there was considerably progress in experimental techniques over the last years, allowing to observe conformation dynamics in some special systems, e.g. with mid-infrared spectroscopy [6] or with Förster resonance energy transfer (FRET) [41], it is still not clear how to match data obtained from (spectroscopic) experiments with geometrical molecular structures in general. Therefore molecular dynamic (MD) simulations are often used to analyse dynamical properties of biomolecular systems. In classical MD, a molecular system with a fixed number of N atoms is characterised by a state vector $(\boldsymbol{q}, \boldsymbol{v}) \in \mathbb{R}^{3N} \times \mathbb{R}^{3N}$, where $\boldsymbol{q} \in \mathbb{R}^{3N}$ denotes the position vector and $\boldsymbol{v} \in \mathbb{R}^{3N}$ the velocity vector. The dynamical behaviour, given a specified potential energy function $V : \mathbb{R}^{3N} \mapsto \mathbb{R}$, a (diagonal) positive definite mass matrix $M \in \mathbb{R}^{3N \times 3N}$ and initial conditions $(\boldsymbol{q}_0, \boldsymbol{v}_0)$, is obtained by integrating Newton's equations of motion

$$\begin{aligned} \frac{\partial q_i}{\partial t} &= v_i \\ m_i \frac{\partial v_i}{\partial t} &= -\frac{\partial V}{\partial q_i}(\boldsymbol{q}), \ i = 1, \dots, N, \end{aligned}$$

which generates a trajectory $(\boldsymbol{q}(t), \boldsymbol{v}(t))_{t \geq 0}$ with $(\boldsymbol{q}(0), \boldsymbol{v}(0)) := (\boldsymbol{q}_0, \boldsymbol{v}_0)$.

With the introduction of mass weighted velocities (momenta) $\boldsymbol{p} = M\boldsymbol{v}$ the dynamics can equivalently specified by the Hamiltonian equations of motion, which read in vector notation

$$\begin{aligned} \dot{\boldsymbol{q}} &= M^{-1}\boldsymbol{p} \\ \dot{\boldsymbol{p}} &= -\nabla_{\boldsymbol{q}} V(\boldsymbol{q}). \end{aligned} \tag{2.1}$$

The potential function V is in practice approximated by an empirical force function which includes terms for electrostatic interactions, e.g. Lennard-Jones and Coulomb interactions, for bonded interactions like bond stretching and dihedral angles, and for restraints, like frozen distances and angles in water molecules.

Corresponding to the form given in (2.1) the Hamiltonian which specifies the total energy of the system is given by

$$H(\boldsymbol{q}, \boldsymbol{p}) = \frac{1}{2}\boldsymbol{p}' M^{-1} \boldsymbol{p} + V(\boldsymbol{q}). \tag{2.2}$$

Remarkably, the propagation of the system given by (2.1) conserves the total energy along the trajectory. Therefore, a trajectory of a system in a box of constant volume which is obtained by (2.1) is said to sample an NVE-ensemble, since N the number of particles, V the volume and E the total energy are conserved over the whole trajectory.

Although the molecular dynamics approach has some principle limitations as it is based on the classical equations of motions and do not treat electron movement, it is often the only feasible way to obtain a time resolved picture of dynamical processes in molecular systems. A major drawback of using MD simulations instead of experiments is that they rely on empirical force fields which have to be parameterised, therefore leading to inconsistencies between different force fields [82].

An alternative to the NVE ensemble obtained by propagation of a constant volume and particle system according to the Hamiltonian equations of motion is the NVT ensemble, where instead of the total energy the temperature is kept constant. The physical interpretation of such a system is a simulation of the system in an unresolved heat bath which allows energy exchange. In such systems molecules can occupy configurations of arbitrary total energy, as the surrounding heat bath triggers energy fluctuations, but configurations with high total energy are less probable than configurations with lower total energy (with a ratio becoming more pronounced with higher temperature). Therefore, the probability to find the (equilibrated) system in some specific volume of the phase space is specified by the Boltzmann-Gibbs probability distribution

$$\rho_s(\boldsymbol{q},\boldsymbol{p}) \propto \exp\left(-\beta\left(\frac{1}{2}\boldsymbol{p}' M^{-1}\boldsymbol{p} + V(\boldsymbol{q})\right)\right). \tag{2.3}$$

The parameter β is obtained from the temperature T, measured in Kelvin, by

$$\beta = \frac{1}{k_B T},$$

where $k_B \approx 1.38065 \times 10^{-23} JK^{-1}$ is the Boltzmann constant.

A way to generate such NVT ensemble is by usage of the Langevin equation [21, 47]

$$\begin{aligned} \dot{\boldsymbol{q}} &= M^{-1}\boldsymbol{p} \\ \dot{\boldsymbol{p}} &= -\nabla_{\boldsymbol{q}} V(\boldsymbol{q}) - \gamma M^{-1}\boldsymbol{p} + \sigma\dot{\boldsymbol{W}}, \end{aligned} \tag{2.4}$$

with $\gamma > 0$ the friction term and $\sigma >$ the noise intensity and $\boldsymbol{W}(t)$ a Brownian motion. The Langevin equation is stochastic differential equation, cf. 3.3.1 and A.1, where the stochastic term mimics stimulation from the environment. If the fluctuation-dissipation relation

$$\beta = \frac{2\gamma}{\sigma^2}$$

holds, the Langevin dynamics are ergodic with respect to (2.3) (which can be seen by inserting it in the Fokker-Planck equation as described in A.1), meaning that

$$\lim_{t\to\infty}\int_0^t f(\boldsymbol{q}(s),\boldsymbol{p}(s))ds = \iint\limits_{\mathbb{R}^{3N}\times\mathbb{R}^{3N}} f(\boldsymbol{p},\boldsymbol{q})\rho_s(\boldsymbol{p},\boldsymbol{q})d\boldsymbol{p}d\boldsymbol{q}, \tag{2.5}$$

for a function $f:\mathbb{R}^{3N}\times\mathbb{R}^{3N}\mapsto\mathbb{R}$ such that the left hand side exists and any realisation of the process (2.4) putted in the right hand side. Due to the ergodic properties of the Langevin dynamic, it can be used to calculate NVT ensemble averages, while it has an interpretation in its own right as a stochastically excited dynamical system, where the excitation might come from an unresolved surrounding. Other methods to sample the NVT ensemble include the Nosé Hoover thermostat [50] or (hybrid) Monte Carlo techniques [79].

Note however, that the computation of statistical averages using the ergodic property (2.5) is not as straightforward as it seems at the first sight as numerical requirements impose a very fine time discretisation of (2.4), such that the full exploration of the phase space might become very challenging from a computational viewpoint. This holds especially if the potential function imposes energetic barriers which are high compared to the intensity of the stochastic excitation, i.e. the temperature, as then the system becomes trapped for long times in low energy regions before crossing the barrier. This so-called trapping problem requires the application of quite complicated algorithmic solutions to be overcome [33].

2.4 Computational Obstacles in Data Analysis of Biomolecules

Even though the generation or simulation of data which reflects sufficiently accurate the dynamical properties of an investigated molecular system is in general a non-trivial task, in most parts of this thesis it is assumed that such a data set is at hand, with the exception of § 5. Even if the problem of obtaining or generating an accurate time series is solved, there still remains the difficult task of analysing the data. If one wants to analyse data of biomolecular systems, one is confronted with a variety of severe challenges. These challenges do not arise from the specific biomolecular origin of the data but from the complexity of such systems. In other words, since biomolecular systems are complex systems, we need algorithms to analyse them which are capable of dealing with time series exhibiting complex dynamical behaviour. In this sense we want to emphasise that even if the algorithms presented here are developed to analyse trajectories from molecular systems, they are general in the sense that they are designed to handle the following listed obstacles which typically arise in complex time series.

2.4.1 Many Degrees of Freedom

Biomolecular systems are in general high dimensional systems. If such a system consists of N atoms, then there are $3N$ position observables, a number which is doubled if velocities, resp. momenta, are also taken into account. Even if N is a small number, most algorithms will quickly face difficulties just due to the dimensionality of the system. Therefore, algorithms are needed which can cope somehow with high dimensional time series.

Even if such algorithms are at hand, in most cases some sort of dimension reduction is needed. We understand as dimension reduction a mapping

$$R : \mathbb{R}^{3N} \mapsto \mathbb{R}^{d}$$

with $d \ll 3N$, such that the "important" dynamical properties of the system can still be retrieved from the reduced system. To find such a mapping either one has to resort to insight about the specific investigated system or use some automatic procedure. An example for the first is the usage of dihedral angles for peptides as described in § 2.2. An example for the latter is the well-known principal component analysis (PCA) which, given a fixed dimension d of the projection space, projects the time series onto a linear subspace of the original space, such that the variance of the projected time series is maximised [60].

2.4.2 Multiple Time Scales

Symptomatic for complex time series is the existence of multiple time scales. This term refers to a separation of time scales, i.e. the system exhibits characteristic dynamical properties on one time scale and others on a different time scale. In a biomolecular context this could be the femto- to picosecond time scale in which the dynamic is governed by bond-angle vibrations of the atoms around a stable global geometric structure, while on the nanosecond to second time scale the relevant dynamics is the change of the global geometric structure.

The existence of time scale separation poses a twofold problem. The first is the simulation of such systems, while the second arises if data generated by such system needs to be analysed. As mentioned above, this means that one can not omit the small time scales in simulation, as otherwise numerical integration schemes fail, i.e. one has to choose an integration step of a few femtoseconds to simulate dynamical processes on the nanosecond scale which makes the simulation process very time consuming. On the other hand, if one has a time series of such system and wants to extract information about the large scale process one has to find a way to separate the large scale motion information from the short scale motion information.

To cover larger time scales of big systems in MD simulations, coarse grained models are frequently used, i.e. one builds a reduced model from the original atomic description where atom groups are represented as a single particle [118].

The major difficulty with the approach is finding an appropriate force field description for the reduced model.

The perspective put forward in this thesis is somewhat different from this approach, as a *data based* model reduction is proposed. That is, given a time series, we want to fit a model that separates the time scales via the usage of submodels for the small time scales and a global model for the large time scales.

2.4.3 Phase Dependency

Closely related to the problem described in the last section, phase dependency means that statistical properties of the time series under consideration may change over time, i.e. there are phases in time to which different dynamical regimes belong. Phase dependency especially means that fitting a single model is not appropriate for describing the system. Even worse, if a model that takes phase dependency into account is fitted to a time series one has to assure that parts of the time series belonging to different phases are not mixed up when they are used to parameterise the different phase models. In other words, model fitting and phase separation has to be done simultaneously.

2.4.4 Amount of Data

Besides the more structural difficulties in the analysis of complex time series, there is the more basic problem of just handling with the amount of data. Even if dimension or model reduction is applied, one has to cope in general with a large amount of data. Thus, algorithms have to be designed such that the data can handled in an effective way. By an effective way we mean a linear scaling of the execution time with the amount of data and a handling of data structures that prevents working memory overflow. An immediate consequence of the last requirement is the usage of linear algebra packages that support sparse matrix operations.

3 Data Analysis

In the following we are concerned with some given time series

$$Z = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_T\},$$

with $\boldsymbol{z}_t \in \mathbb{R}^d$, which is supposed to be generated by some (stochastic) process, i.e. by Langevin dynamic as given in (2.4). There are two basic assumptions made on the dynamics of this time series: first, that there are different time scales, i.e. there are phase changes on larger time scales than other fluctuations on smaller time scales; second, these phase changes can effectively modeled as a discrete Markovian process. In the context of (positional) molecular time series such phase changes would correspond to global geometrical changes, while on smaller time scales the dynamic is mainly characterised by fluctuations around these metastable large scale structures. The aim is to identify these phases and to set up a Markovian model for the changes between them.

While an employed model for time-series analysis should be able to reproduce complex dynamical behaviour on one hand, it is crucial that the model is simple enough to be parameterised on the other hand. We will start in Sec. 3.1.1 with the assumption that the data is generated from a homogeneous Markov process and then proceed to more complex models.

The first approach presented in § 3.1, the Perron Cluster Cluster Analysis (PCCA) developed by Christof Schütte, Peter Deuflhard et al. [29, 30, 104, 107], is a *set-oriented* approach. It is based upon a the construction of a transition matrix that describes transition probabilities between sets in the state space of the system. The identification of metastable sets is then based on analysis of this transition matrix, i.e. the phases of the system are identified with metastable sets in state space which the process rarely leaves while it is fast-mixing within them.

PCCA relies upon the possibility to discriminate different dynamical phases spatially, i.e. by a decomposition of the observation space. However, this assumption might not hold, especially if a projection of the original observation space is treated, i.e. some torsion angle instead of the phase space trajectory. In such situations a more appropriate conception would be a process which switches in a Markovian manner between different dynamical regimes, i.e. geometrical structures, while only some generated output, i.e. a torsion angles, is observable. A model class for such conception are the Hidden Markov Models (HMM), presented in § 3.2, which can be efficiently employed in the analysis of molecular dynamic trajectory [36].

Based upon this idea, Illia Horenko et al. [52, 53] extended the HMM approach by introducing dependency between the observed data points via a model with a stochastic differential equation whose parameters change with switches of the hidden process, depicted in § 3.3. It turns out that by a subtle reformulation of this approach, we can not only make the estimation process more stable but we can also generalise it to higher memory, i.e. allowing not only dependency in a Markovian way between data points but also include dependency of more than one precedent data point [76].

3.1 Perron Cluster Cluster Analysis

Before describing the PCCA approach we need to collect a few basic terms from the theory of Markov processes and chains on the fly. The reader interested in a complete introduction is referred to existing monographs, e.g. we recommend [31] for stochastic processes in general and [10, 12] for Markov chains and processes.

3.1.1 Homogeneous Markov processes

Denote by $X = \{X_t, t \in I\}$ a stochastic process, i.e. a family of random variables Z on some appropriate probability space with probability measure $\mathbb{P}$ indexed by a parameter t, which is an element of some ordered index set I. More specifically, we interpret the index t as a time specification and set $I = \mathbb{R}^+$ or $I = \mathbb{N}$, in the latter case X is entitled as a time-discrete process, otherwise as a time-continuous process. The state space S, i.e. the union of all possible values of the random variables $\{X_t, t \in I\}$, is assumed to either be $\mathbb{R}^d$, in which case we call X continuous, or a subset of $\mathbb{N}$ in which case we call X discrete.

Definition 3.1.1. We call a time-continuous stochastic process X a *Markov process*, if for all $n \in \mathbb{N}$ and every $t \in I$ it holds that for every n-tuple $t_1, t_2, \ldots, t_n < t \in I$ with $t_1 < t_2 < \ldots < t_n$ and every Borel measurable event A we have

$$\mathbb{P}[X_t \in A | X_{t_1}, \ldots, X_{t_n}] = \mathbb{P}[X_t \in A | X_{t_n}].$$

If $S \subset \mathbb{N}$, we call such process a *Markov jump process*, since it jumps between discrete events, but note that Markov jump processes can also be defined on a continuous state space.

In simple words, a Markov process is a process whose probability distribution of future events only depends upon the last known state. Assume X is a continuous Markov process, then the probability measure $\mathbb{P}$ induces a *transition probability function* $p(s, \boldsymbol{x}, t, B)$ defined as

$$p(s, \boldsymbol{x}, t, B) := \mathbb{P}[X_t \in B | X_s = \boldsymbol{x}],$$

with $s \leq t \in I$, $\boldsymbol{x} \in S$, B Borel measurable.

In the following, we restrict ourselves to homogeneous Markov processes where the transition probability function depends upon time only through the increment $t - s$. Therefore we can write

$$p(t-s, \boldsymbol{x}, B) := p(s, \boldsymbol{x}, t, B).$$

A homogeneous Markov process is called a stationary process if

$$\mathbb{P}[X_0 \in A] = \mathbb{P}[X_t \in A],$$

for every measurable event A and $t \in I$.

We write $X_0 \sim \rho_0$ if the Markov process X is initially distributed according to the probability measure ρ_0 and denote the corresponding probability measure of the process by $\mathbb{P}_{\rho_0}$. We call ρ_s an *invariant probability measure* wrt. X, or ρ_s is invariant wrt. X, if

$$\int_S p(t, \boldsymbol{x}, A) \rho_s(\mathrm{d}\boldsymbol{x}) \;=\; \rho(A)$$

for all $t \in I$. In the following we always assume that the invariant measure of the process under investigation exists and is unique. If $\rho_0 = \rho_s$ the corresponding Markov process is stationary. A Markov process is called *reversible* wrt. an invariant probability measure ρ_s if

$$\int_A p(t, \boldsymbol{x}, B) \rho_s(\mathrm{d}\boldsymbol{x}) \;=\; \int_B p(t, \boldsymbol{x}, A) \rho_s(\mathrm{d}\boldsymbol{x})$$

for every $t \in I$ and $A, B \subset S$.

Assuming a stationary Markov process with invariant density ρ_s, we can define a *stationary transition probability* $p_s = p_{\rho_s}$ which quantifies the dynamical fluctuations of the process within the stationary regime. Given two measurable subsets $A, B \in S$ and a time span τ we define

$$p_s(\tau, A, B) \;=\; \mathbb{P}_{\rho_s}[X_\tau \in B \;|X_0 \in A] \;=\; \frac{\mathbb{P}_{\rho_s}[X_\tau \in B \text{ and } X_0 \in A]}{\mathbb{P}_{\rho_s}[X_0 \in A]}. \tag{3.1}$$

This can be rewritten as

$$p_{\rho_s}(\tau, A, B) \;=\; \frac{1}{\rho_s(A)} \int_A p(\tau, \boldsymbol{x}, B)\, \rho_s(\mathrm{d}\boldsymbol{x}). \tag{3.2}$$

Since for a Markov process any statistical description of future events solely depends on the present state of the system, any partition of an observed process in phases must be ascribable to a partition of the state space if conclusions about the process and not only about a single realisation should be drawn. This motivates the following definition of metastability.

Definition 3.1.2. Let p_s be the stationary transition probability function of a stationary Markov process on state space S. A measurable subset $B \subset S$ is called *metastable* on the time scale $\tau > 0$ if

$$p_s(\tau, B, B^c) \approx 0, \quad \text{or equivalently,} \quad p_s(\tau, B, B) \approx 1, \tag{3.3}$$

where $B^c = S \setminus B$ denotes the complement of B.

Note that the above definition of metastability crucially depends on a chosen time scale τ, as the transition function $p_s(\tau, B, B^c)$ converges to $\rho(B^c)$ for $\tau \to \infty$, and to 0 for $\tau \to 0$.

The aim of the PCCA approach is the identification of a maximal decomposition of the state space into metastable subsets wrt. to a specified time scale. That is, we are looking for a collection of subsets $S_k \subset S,\ k = 1, \dots, m$, with m chosen maximal, with the following properties:

1) Positivity, i.e. $\rho_s(S_k) > 0$ for every k.

2) Disjointness up to null sets, i.e. $\rho_s(S_j \cap S_k) = 0$ for $j \neq k$.

3) The covering property $\cup_{k=1}^m \overline{S_k} = S$.

4) Metastability wrt. to a fixed τ: $\sum_{k=1}^m p_s(\tau, S_k, S_k^c) > m - \epsilon$.

Having identified such a decomposition we can define a meaningful global dynamics of the process as the “flipping dynamics” between the sub-states S_k. The identification of such a decomposition can be done by the *transfer operator approach* as developed in [57, 107, 108] and outlined below.

3.1.2 The Transfer Operator

In the following we always assume a stationary and reversible Markov process, which turns out to be a crucial assumption for the mathematical justification of the transfer operator approach.

The (forward) transfer operator T^τ of a Markov process, with $\tau \in I$, is an operator which maps functions from the Lebesgue spaces $L^r(\rho_s) := L^r_{\rho_s}(\mathbb{R}^d)$, $1 \leq r < \infty$, to itself and is characterised as follows:

$$\int_A T^\tau \nu(\boldsymbol{y})\ \rho_s(\mathrm{d}\boldsymbol{y}) \ = \ \int_S \nu(\boldsymbol{x}) p(\tau, \boldsymbol{x}, A) \rho_s(\mathrm{d}\boldsymbol{x}) \tag{3.4}$$

for any measurable $A \subset S$ and $\nu \in L^r(\rho_s)$. The transfer operator is related to the usual Markov transition operator $T^{*,\tau}$, defined by

$$T^{*,\tau}\nu(\boldsymbol{x}) = \int_S \nu(\boldsymbol{y}) p(\tau, \boldsymbol{x}, \boldsymbol{y}) \rho_s(\mathrm{d}\boldsymbol{y}),$$

as this is the adjoint operator of T^τ. Both operators form a semigroup with respect to τ, i.e. $T^{\tau_1+\tau_2} = T^{\tau_1} \circ T^{\tau_2}$, and both operators are Markov operators, i.e. they are norm conserving and positive [57].

If the underlying Markov process is reversible, the transfer operator is self-adjoint in $L^2(\rho_s)$ and, as proved in [57], the spectral properties of this operator can be used for the identification of metastable states:

Theorem 3.1.3. *Let $T^\tau : L^2(\rho_s) \mapsto L^2(\rho_s)$ denote the forward transfer operator corresponding to a reversible Markov process X (wrt. ρ_s). Then T^τ is self–adjoint in $L^2(\rho_s)$ wrt. the scalar product $< \cdot, \cdot >_{\rho_s}$.*
Furthermore, if the essential spectral radius of T^τ is less than one and the eigenvalue $\lambda = 1$ of T^τ is simple and dominant, i.e.

$$\sigma(T^\tau) \subset [a, b] \cup \{\lambda_m\} \cup \ldots \cup \{\lambda_2\} \cup \{1\}$$

with $-1 < a \leq b < \lambda_m \leq \ldots \leq \lambda_2 < \lambda_1 = 1$, where the isolated eigenvalues λ_i are counted according to their finite multiplicities, then the following can be stated:
Denote by $v_1, \ldots, v_m$ the corresponding eigenfunctions to $\lambda_1, \ldots, \lambda_m$, normalised such that $\|v_k\|_2 = 1$, let Q be the orthogonal projection of $L^2(\rho_s)$ onto the span of $\{\mathbf{1}_{S_1}, \ldots, \mathbf{1}_{S_m}\}$, where $\{S_1, \ldots, S_m\}$ is an arbitrary *partition of the state space. Then the metastability of the partition, can be bounded from above by*

$$p_s(\tau, S_1, S_1) + \ldots + p_s(\tau, S_m, S_m) \leq 1 + \lambda_2 + \ldots + \lambda_m, \tag{3.5}$$

while it is bounded from below according to

$$1 + \kappa_2\lambda_2 + \ldots + \kappa_m\lambda_m + c \leq p_s(\tau, S_1, S_1) + \ldots + p_s(\tau, S_m, S_m), \tag{3.6}$$

where $\kappa_j = \|Qv_j\|^2_{L^2(\rho_s)}$ and $c = a\ ((1 - \kappa_2) + \ldots + (1 - \kappa_n))$.

Eq. (3.5) states that the metastability of an *arbitrary* partition of S into m subsets is bounded from above by the sum of the m largest eigenvalues.

> Therefore a meaningful choice for the number of metastable sets is the number of eigenvalues near one, as long as there is a (spectral) gap between these and the rest of the spectrum.

Furthermore the lower bound Eq. (3.6) is close to the upper bound if the eigenfunctions are almost constant on the metastable subsets $S_1, \ldots, S_m$, which implies $\kappa_2, \ldots, \kappa_m$ close to one and c close to zero.

> In other words, the structure of the eigenfunctions can be used to identify a metastable decomposition.

However, to compute the spectrum of the transfer operator we have to discretise it and, in our setting, estimate the discretised object from a given time series. As outlined in the next section, the discretisation of the transfer operator gives rise to a stochastic matrix and the theory from the continuous case can be transfered to the discretised case.

3.1.3 Discretisation of the Transfer Operator

The discretisation of the transfer operator can be done by discretisation of the state space of the underlying Markov process. Therefore a decomposition of the state space $B_1, \dots, B_n \in S$ is used to define ansatz functions $\chi_i \in L^2(\rho_s), i = 1, \dots, n$, by

$$\chi_i(\boldsymbol{x}) = \begin{cases} 1 \text{ , if } \boldsymbol{x} \in B_i, \\ 0 \text{ , else.} \end{cases}$$

A Galerkin projection Π_n from $L^2(\rho_s)$ to the finite dimensional ansatz space $\mathcal{S}_n = \text{span}\{\chi_1, \dots, \chi_n\}$ is defined by

$$\Pi_n \nu \;=\; \sum_{i=1}^{n} \frac{\langle \nu, \chi_i \rangle_{\rho_s}}{\langle \chi_i, \chi_i \rangle_{\rho_s}} \chi_k.$$

Using the coordinate representation wrt. the basis $\{\chi_1, \dots, \chi_n\}$ of $\mathcal{S}_n$, i.e. $\boldsymbol{v} = (v_1, \dots, v_n)$ notes the vector $\sum_{i=1}^{n} v_i \chi_i$ wrt. to the standard basis, an easy calculation shows that for a self-adjoint operator T^τ the eigenvalue problem $T^\tau \nu = \lambda \nu$ transforms to $P^\tau \boldsymbol{v} = \lambda \boldsymbol{v}$, where P^τ is a matrix with entries

$$p_{ij} := \frac{\langle \chi_j, T^\tau \chi_i \rangle_{\rho_s}}{\langle \chi_i, \chi_i \rangle_{\rho_s}} = \int_{B_k} p(\tau, \boldsymbol{x}, B_l) \rho_s(\mathrm{d}\boldsymbol{x}).$$

The matrix P, note that we omit the superscript τ in the following to simplify notation, inherits important spectral properties of the operator T^τ [57, 108]:

(i) P is an irreducible and aperiodic stochastic matrix, with a unique dominant eigenvalue $\lambda = 1$. For the (normalised) left eigenvector $\boldsymbol{\pi} = (\pi, \dots, \pi_n)$ corresponding to the dominant eigenvalue we have $\pi_i = \rho(B_i)$.

(ii) P is selfadjoint wrt. to the $\boldsymbol{\pi}$-weighted scalar product $\langle \cdot, \cdot \rangle_{\boldsymbol{\pi}}$, i.e. it is similar to a symmetric matrix and all eigenvalues are contained in the real interval $]-1, 1]$.

(iii) For a fine enough Galerkin discretisation, the discrete spectrum of the operator T^τ will be approximated by eigenvalues of P.

To summarise, via a Galerkin discretisation of the transfer operator a stochastic matrix is obtained, which inherits important spectral properties of the continuous object. As shown in the next section, the concept of metastability can be transfered smoothly to the discrete case and, again, the structure of the spectrum can be used to determine metastable sets.

3.1.4 Perron Cluster Cluster Analysis

In this section we transfer the concept of metastability to a discrete space setting, therefore we assume a given stochastic matrix $P = (p_{ij})$ which could be obtained from the discretisation of a transfer operator or just be given as the transition matrix of a (time-)discrete Markov chain. That is we assume a homogeneous (time-)discrete Markov process $X = \{x_1, x_2, \ldots\}$ upon a state space $S = \{1, \ldots, n\}$ for which

$$\mathbb{P}[x_{t+1} = j | x_t = i] = p_{ij}$$

holds. If the distribution of a Markov chain at a point t is given by a vector $\boldsymbol{v}$, i.e. $\mathbb{P}[x_t = i] = v_i$ then the distribution to the time $t + k$ is given by $(\boldsymbol{v}'P^k)'$. We assume the existence and uniqueness of a positive stationary distribution, i.e. a vector $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_n)$ with $\pi_i > 0$ for $1 \leq i \leq n$ and $\sum_{i=1}^n \pi_i = 1$ which fulfils

$$\boldsymbol{\pi}^{'} P = \boldsymbol{\pi}^{'},$$

and that $\lambda = 1$ is the only eigenvalue of P on the unit circle. Such matrix is called primitive [111, Ch. 1], which corresponds to the assumption that the underlying Markov chain is aperiodic and irreducible. Furthermore P is supposed to be reversible wrt. to $\boldsymbol{\pi}$, i.e.

$$\pi_i p_{ij} = \pi_j p_{ji}, \ 1 \leq i, j \leq n.$$

A simple consequence of reversibility is that P can be symmetrised with the diagonal matrix $D = \mathrm{diag}(\pi_1, \ldots, \pi_n)$ and consequently the eigenvalues of P are all real valued and contained in $]-1, 1]$.

Equivalent to the definition of metastability given in (3.3), a subset $B \subset S$ is called metastable if

$$\mathbb{P}_{\boldsymbol{\pi}}[x_t \in B, x_{t+1} \in B] = \frac{\sum_{i,j \in B} \pi_i p_{ij}}{\sum_{i \in B} \pi_i} \approx 1.$$

Correspondingly, a partition $\{S_1, \ldots, S_m\}$ of the state space S is called metastable if

$$w(S_k, S_l) := \frac{\sum_{\substack{i \in S_i \\ j \in S_j}} \pi_i p_{ij}}{\sum_{i \in S_i} \pi_i} \approx \delta_{kl}$$

holds for all k, l with $1 \leq k, l \leq m$.

There is an intuitive illustration why such decomposition can be found using spectral properties of P. Consider a Markov chain with a decoupled state space, i.e. there is a partition in so-called invariant sets $\{S_1, \ldots, S_m\}$ such that

$$w(S_k, S_l) = \delta_{kl}, \ 1 \leq k, l \leq n.$$

Then the state space can be permuted such that the corresponding transition matrix is block-diagonal and the blocks are itself stochastic matrices. Provided that these stochastic submatrices are themselves primitive we would have an m-fold dominant eigenvalue 1 and a corresponding (right-)eigenspace which is spanned by the characteristic vectors of the invariant sets, as any stochastic matrix has at least the unit vector as right eigenvector to the eigenvalue 1. By noting that every other basis of this eigenspace is a linear transformation of this basis, i.e. any eigenvector of $\lambda = 1$ is a linear combination of the characteristic vectors, we immediately see that any eigenvector belonging to the dominant eigenvalue is constant upon all invariant sets. Even more, Deuflhard et al. [29] proved that the invariant sets can be uniquely identified by the sign structure of the eigenvectors. More precisely, since each eigenvector is constant on an invariant subset it can be used to assign a sign, i.e. $+, -, 0$, to an invariant set. For an arbitrary orthogonal basis of the eigenspace $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m$ a sign pattern can be assigned to each state via the map

$$f : S \mapsto \{+, -, 0\}^m, \ f(k) = (\mathrm{sign}(\boldsymbol{v}_{1,k}), \ldots, \mathrm{sign}(\boldsymbol{v}_{m,k}))\,.$$

Defining sets by states with the same sign pattern one obtains the invariant sets.

Assuming that the transition matrix of a Markov chain with metastable sets is obtained by a small perturbation of a transition matrix belonging to a Markov chain with decoupled sets and that the spectrum of the unperturbed transition matrix is bounded away from the unit circle, with exception of the m-fold dominant eigenvalue, it is clear that the above described identification strategy can also be used to identify metastable sets, since the sign pattern of the eigenvectors does not change for small perturbations of the matrix. In fact this assumption can be justified rigorously by perturbation theory up to second order [29,126]. Therefore one obtains the following algorithmic strategy

1.) identify the number of metastable sets by the number of eigenvalues close to one, the so-called Perron cluster (obviously this requires the existence of some spectral gap between the Perron cluster and the rest of the spectrum).

2.) identify the metastable sets by exploiting the sign structure of the corresponding eigenvectors.

This procedure has been established as Perron Cluster Cluster Analysis (PCCA).

Problematic with this approach is that zero as a sign is obviously not conserved even for arbitrary small perturbations. A proposed remedy is to define a threshold, so that every value whose absolute value is smaller than the threshold is set to zero. In our experience a similar approach based on PCCA which circumvents the problem with zero values turned out to be most satisfactory [30,125,126]. This approach is based upon the observation that for the uncoupled case the orthonormal basis of characteristic vectors $\{\boldsymbol{\chi}_1, \ldots, \boldsymbol{\chi}_m\}$, wrt. to the partition in invariant sets, of the eigenspace belonging to the dominant eigenvalue gives an allocation of each state to an edge of an m-dimensional simplex via the mapping

$$f : S \mapsto \mathbb{R}^m, \; f(k) = \left(\boldsymbol{\chi}_{1,k}, \ldots, \boldsymbol{\chi}_{m,k}\right).$$

Choosing another basis, i.e. another set of vectors

$$\boldsymbol{v}_i = \sum_{j=1}^{m} \alpha_{ij} \boldsymbol{\chi_j}, \; 1 \leq i \leq m,$$

corresponds to a linear transformation of this (standard) simplex to another simplex, whose edges are given by the rows of $A = (\alpha_{ij})$. By inverting this matrix one obtains a map from a given eigenvector basis to the unit simplex. Again, having not a decoupled system but metastable sets the simplex structure is expected to be nearly conserved. This leads to the following algorithmic idea, sometimes referred as PCCA+:
Given a stochastic primitive matrix P,

1.) identify the number of metastable sets m via the size of the Perron cluster.

2.) write an arbitrary orthogonal basis of the eigenspace to the dominant eigenvalue as columns into a matrix

$$V = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m),$$

and interpret the rows of V as data points $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n \in \mathbb{R}^m$.

3.) choose iteratively m of these data points to define a simplex in $\mathbb{R}^m$, by choosing first the two points with the largest (Euclidean) distance and then subsequently add a point which is farthest away from the hyperplane spanned by the already chosen points. Define A by putting the obtained simplex edges as columns in a matrix.

4.) invert A and transform the data points to $A^{-1}\boldsymbol{u}_1, \ldots, A^{-1}\boldsymbol{u}_n$. The result should be an approximated standard simplex.

5.) assign $i \in S$ to metastable set S_j if data point $A^{-1}\boldsymbol{u}_i$ is closest to edge j of the standard simplex (up to permutations of the edge numbering).

This heuristic approach can be made more sophisticated and rigorous by defining an optimisation problem, i.e. finding a simplex which fits optimally wrt. to some score function in the data points [30, 125]. However, in practice it turns out that the results of a sophisticated optimisation strategy are not better, but the effort needed is drastically increased.

3.1.5 Estimation of the transition matrix

Above we outlined how to determine metastable sets from a given stochastic matrix. In general we have to estimate this matrix from discrete or discretised observations. This can be done by maximum likelihood estimation. Assume a time series $Z = \{z_1, z_2, \dots, z_T\}$ on a discrete and finite state space $S = \{1, \dots, n\}$, possibly obtained by some discretisation of a former continuous time series. Denote by N_i the sum of data points observed in state i and by N_{ij} the number of transitions from i to j within one time step found in the time series. A likelihood function for the transition matrix $P = (p_{ij})$ is given by

$$L(P|Z) = \prod_{k=1}^{T-1} p_{z_k z_{k+1}}$$

which leads to the log-likelihood function

$$l(P|Z) = \sum_{k=1}^{T-1} \log(p_{z_k z_{k+1}}) = \sum_{i,j=1}^{n} N_{ij} \log(p_{ij}).$$

To obtain an maximum likelihood estimator (MLE) we need to maximise the log-likelihood function, i.e., regarding the constraint $\sum_{k=1}^{n} p_{ik} = 1$ for $i = 1, \dots, n$ and therefore introducing the Lagrange factors $\alpha_1, \dots, \alpha_n$, we have to solve

$$\frac{\partial l(P|Z)}{\partial p_{ij}} + \alpha_i \frac{\partial}{\partial p_{ij}} \left(\sum_{k=1}^{n} p_{ik} = 1 \right) = 0,$$

for $i = 1, \dots, n$. This yields

$$\hat{p}_{ij} = \frac{N_{ij}}{N_i},$$

as an MLE for p_{ij}. Therefore, an estimator of the transition matrix P is obtained by simply counting the observed transitions between states. But in order to employ the above outlined PCCA we need a reversible transition matrix, which leads to the question, under which circumstances the estimated transition matrix is indeed reversible. First, we note that the stationary distribution $\boldsymbol{\pi} = (\pi_1, \dots \pi_n)$ for the above estimated transition matrix is given by

$$\pi_i = \sum_{k=1}^{n} \frac{N_i}{N_k},$$

as this implies

$$\pi_i = \sum_{k=1}^{n} \frac{N_i}{N_k} = \sum_{j,k=1}^{n} \frac{N_j N_{ji}}{N_k N_j} = \sum_{j=1}^{n} \pi_j p_{ji}$$

which can be compactly written as $\boldsymbol{\pi} P = \boldsymbol{\pi}$. Therefore, the reversibility condition $\pi_i p_{ij} = \pi_j p_{ij}$ is equivalent to

$$\sum_{k=1}^{n} \frac{N_i N_{ij}}{N_k N_i} = \sum_{k=1}^{n} \frac{N_j N_{ji}}{N_k N_j} \Leftrightarrow N_{ij} = N_{ji},$$

i.e. the estimated transition matrix is reversible iff the same number of transitions from i to j as from j to i is observed, which meets the intuitive interpretation of reversibility that the direction of time does not change the statistical properties of the process. Of course, with a given (finite) observation this will rarely be the case, even if it is in fact generated by a reversible process. A remedy is to set $\tilde{N}_{ij} = N_{ij} + N_{ji}$ and, to preserve the normalisation, set $\tilde{N}_i = 2N_i$ and estimate P by

$$p_{ij} = \frac{\tilde{N}_{ij}}{\tilde{N}_i}, \tag{3.7}$$

which delivers a reversible transition matrix.

3.1.6 Example

We close the section on PCCA with an easy example illustrated in Fig. 3.1. For demonstration purposes we cut out a short piece of a two-dimensional time series obtained from some angular observable of a molecular dynamics simulation. As seen in the figure, each dimension is discretised in 7 boxes which gives a discrete state space of $7 \cdot 7 = 49$ boxes. Since not all of the possible boxes are occupied in the observed time series, the state space reduces to 36 boxes with non-zero probability. Upon these a stochastic transition matrix is set up according to (3.7). The five largest eigenvalues of this matrix are

k	1	2	3	4	5
$\lambda_k(P)$	1	0.9995	0.9909	0.7339	0.7062

$\ldots$.

As there is a significant gap between the third and the fourth eigenvalue, we assume the existence of three metastable sets and the eigenvectors corresponding to the three largest eigenvalues are used to identify them. Both methods described above, PCCA and PCCA+, are illustrated in Fig. 3.1 and provide the same result. Note that in this example PCCA does not need to use information from the critical “zero-sign” region of the eigenvalues, since just the $+, -$ pattern of values far off zero gives a unique allocation of states to metastable sets, which is quite typical. Finally, permutation of the transition matrix, such that boxes belonging to the same metastable set are neighboured reveals a block diagonal dominant structure as expected.

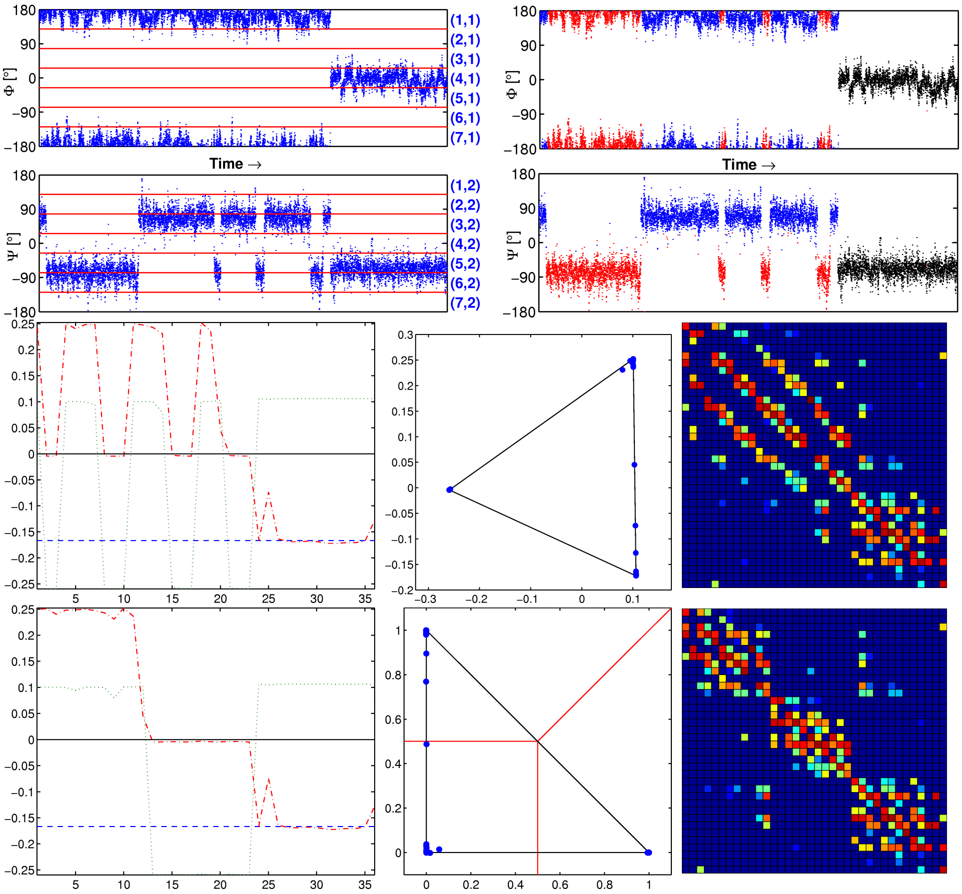


Figure 3.1: *Top row:* Left: A two dimensional periodic time series of two torsion angles $\Phi, \Psi \in]-180°, 180°]$. A box discretisation in 7 boxes along each dimension is indicated by red lines. Right: The outcome of PCCA is a clustering in 3 metastable sets coded here in a colouring of black, blue and red. *Middle row:* Left: Orthogonal eigenvectors belonging to the Perron cluster of the resulting transition matrix are plotted against the index of the 36 occupied states. Middle: Plotting the eigenvector matrix row-wise (omitting the first component as it is constant) reveals a simplex structure. Right: A schematic plot of the transition matrix. The colouring of the boxes indicates the amplitude of the entries, dark blue corresponds to zero entries and dark red to entries close to one.
Bottom row: Left: the eigenvectors are depicted against the state space permuted such that states with the same sign structure of the eigenvectors are together. Middle: Defining a linear map by transforming the edges of the original simplex to a standard simplex conserves the simplex structure of the data points. Allocation to metastable states is been made by allocation of each data point to the closest standard simplex edge. Right: Both methods deliver the same permutation of the state space, the correspondingly permuted matrix is shown.

3.2 Hidden Markov Models

Assume a time series $\boldsymbol{z}_t \in \mathbb{R}^d, t = 1, 2, \ldots, T$ if given from, e.g., an MD-simulation of a molecule, which do not completely specify the state of the molecule in phase space, but rather some low-dimensional observable, for example, some or all torsion angles or a set of essential degrees of freedom. As the Markov property does not hold for projections of Markov processes in general, we have to be aware that the process on the (torsion angles) subspace might no longer be Markovian. Nevertheless, we assume that there is an unknown metastable decomposition into m sets $S_1, \ldots, S_m$, in the full dimensional system.

We can then premise that, at any time t, the system is in one of the metastable sets $S_{h_t}, h_t \in \{1, \ldots, m\}$ to which we simply refer by h_t. However, in contrast to the *observed* time series $\boldsymbol{z}_t$, the time series h_t is *hidden*, i.e., neither known in advance nor observed.

Such a scenario can be represented by a Hidden Markov Model (HMM). An HMM abstractly consists of two related stochastic processes: a hidden process h_t that fulfils the Markov property, and an observed process $\boldsymbol{z}_t$ that depends on the state of the hidden process h_t at time t. For example, within the molecular context, the state of the hidden process could represent the actual conformation of a molecule system and the observed process some torsion angles, which of course are dependent on the global geometry. Note, however, that the concept of HMM's is in general independent of the concept of metastability, i.e. the Markov chain representing the hidden process can in principle be fast mixing. Therefore the assumption of metastability is an additional assumption motivated by the specific application background and is not required in the following.

An HMM is fully specified by an initial distribution $\boldsymbol{\pi}$ and a transition matrix P of the hidden Markov process $H = (h_t)$, and the probability distributions that govern the observable $\boldsymbol{z}_t$ depending on the respective hidden state h_t, so it can be formally is defined as a tuple $\theta = (S, V, P, \boldsymbol{f}, \boldsymbol{\pi})$ where

- $S = \{1, 2, ..., n\}$ is a finite state space,
- $V \subset \mathbb{R}^d$ is the observation space,
- $P = (p_{ij})$ is the transition matrix, with $p_{ij} = \mathbb{P}[h_{t+1} = j | h_t = i]$,
- $\boldsymbol{f} = (f_1, f_2, \ldots, f_n)$ is a vector of probability density functions (pdf) in the observation space,
- $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_n)$ is a stochastic vector, that describes the initial state distribution, $\pi_i = \mathbb{P}[\boldsymbol{z}_1 = i]$.

In the following we use the short notation $\theta = (P, \boldsymbol{f}, \boldsymbol{\pi})$ since S and V are implicitly included, resp. are not estimated but specified. Of course, HMMs can also be defined with discrete observation process $\boldsymbol{z}_t$, in this case $\boldsymbol{f}$ specifies the corresponding probabilities instead of density functions, however, if not otherwise noted, we assume $\boldsymbol{z}_t$ to be continuous in the following. The most popular choice in the continuous case is to use (multivariate) normal distributions for the output distributions f_k, see § 3.2.2. We will substantially generalise the HMM-models in § 3.3.5 by allowing the output distributions to be dependent on former observations.

Example: Consider the following simple example of an HMM: a two element state space $S = \{A, B\}$ with

$$\mathbb{P}[h_{t+1} = A|h_t = A] = \mathbb{P}[h_{t+1} = B|h_t = B] = 0.9,$$

i.e., the transition matrix is given by

$$P = \begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{pmatrix}.$$

The observation space and the output probabilities are given by $V = \{1, 2, 3\}$ and the vectors $\boldsymbol{f}_A = (0.5, 0.5, 0)$ and $\boldsymbol{f}_B = (0, 0.5, 0.5)$. The construction is such that in hidden state A the observations 1 and 2 can be made with equal probability, while 2 and 3 are observable in hidden state B. The hidden process is assumed to be in stationary state, i.e. $\mathbb{P}[h_t = A] = \mathbb{P}[h_t = B] = 0.5$ as with $\boldsymbol{\pi} = (0.5, 0.5)$ we have $\boldsymbol{\pi} P = \boldsymbol{\pi}$. What can we say about the probability of observing $z_t = 1$, when $z_{t-1} = 2$ has been observed previously? A simple calculation yields

$$\begin{aligned}
&\mathbb{P}[z_t = 1|z_{t-1} = 2] \\
&\quad = \sum_{s\in\{A,B\}} \mathbb{P}[z_t = 1|z_{t-1} = 2, h_{t-1} = s]\, \mathbb{P}[h_{t-1} = s|z_{t-1} = 2] \\
&\quad = \sum_{s\in\{A,B\}} \mathbb{P}[z_t = 1|z_{t-1} = 2, h_{t-1} = s] \frac{\mathbb{P}[z_{t-1} = 2|h_{t-1} = s]\, \mathbb{P}[h_{t-1} = s]}{\sum_{s'} \mathbb{P}[z_{t-1} = 2|h_{t-1} = s']\, \mathbb{P}[h_{t-1} = s']} \\
&\quad = 0.9\frac{1}{4} + 0.1\frac{1}{4} = 0.25.
\end{aligned}$$

How is this probability affected if knowledge about another previous observation, i.e. $z_{t-2} = 1$, is taken into account? We have, regarding the output distributions,

$$\mathbb{P}[z_t = 1|z_{t-1} = 2, z_{t-2} = 1] = \mathbb{P}[z_t = 1|z_{t-1} = 2, h_{t-2} = A],$$

and a calculation similar to the one above yields

$$\mathbb{P}[z_t = 1|z_{t-1} = 2, z_{t-2} = 1] = 0.41.$$

This simple example demonstrates two important properties of HMMs. First of all, opposed to the hidden process H, the resulting observed process is obviously not Markov anymore, and second, the same observation can be generated by different hidden states. In reverse this means that an HMM does not rely on spatial separation of the observations as it takes the dynamical behaviour into account, which is a conceptual difference from the PCCA context.

In the context of HMMs, there are three standard tasks to solve [96], these are

(T1) Calculation of the probability $\mathbb{P}[Z|\theta]$, resp. evaluation of a density function $p(Z|\theta)$ in the case of a continuous observation space, for a certain observed sequence Z and a given model $\theta = (P, \boldsymbol{f}, \boldsymbol{\pi})$.

(T2) Estimation of the best model parameters for a given observation sequence, i.e. maximisation of the likelihood function $L(\theta|z) = \mathbb{P}[z|\theta]$ wrt. θ.

(T3) Given the model θ and an observation sequence Z, find the most probable hidden state sequence $\hat{\boldsymbol{h}} = (\hat{h}_1, \hat{h}_2, \ldots, \hat{h}_T)$.

Task $(T1)$ and $(T3)$ can be solved explicitly in an efficient way with Dynamic Programming (DP) approaches. The optimisation problem in task $(T2)$ is more challenging, as it is in general not analytically solvable and non-convex, i.e. local optima of the likelihood function do exist. In the next section we shortly introduce the Expectation-Maximisation (EM) algorithm, used to locally solve $(T2)$, and show how to apply it to the standard case, i.e. independent Gaussian output distribution. In § 3.2.3 it is shown how to use the DP techniques to compute the necessary quantities for the EM algorithm and solve the problems (T1)-(T3). Afterwards in § 3.3 we define a broader class of possible output distribution. As noted in $(T1)$, probabilities in a continuous observation space are specified by a probability density function which is noted by p instead of a probability measure $\mathbb{P}$, in order to avoid notational confusion and for notation transferability, we will solely use the notation p in what follows for both cases.

3.2.1 The Expectation-Maximisation Algorithm

Although previously used, e.g. in the context of HMMs [7], the EM algorithm as a general method to compute Maximum Likelihood estimates in the presence of missing or hidden data was proposed 1977 in an article by Dempster et al. [27]. Assume a vector of observations $\boldsymbol{z}$ and a probability density $p(\boldsymbol{z}|\theta)$ which is parameterised by a parameter vector θ. The aim is to find a $\hat{\boldsymbol{\theta}}$ which maximises the likelihood function $L(\theta) := p(\boldsymbol{z}|\theta)$ or equivalently, as the logarithm is a monotone function, the log-likelihood function

$$l(\theta) = \log p(\boldsymbol{z}|\theta).$$

The strategy employed in the EM algorithm is the following, instead of maximising $l(\theta)$ directly, which is infeasible in many cases, a function $l(\theta|\theta_k)$ is constructed which depends on a current parameter guess θ_k and has the following properties

1.) The log-likelihood function is bounded from below by $l(\theta|\theta_k)$,
i.e. $l(\theta) \geq l(\theta|\theta_k)$ for all θ.

2.) The bound is sharp for the current guess, i.e. $l(\theta_k) = l(\theta_k|\theta_k)$.

From the two given conditions, it follows that any increase in $l(\theta|\theta_k)$ will increase $l(\theta)$. Therefore, the EM algorithm aims at constructing and maximising $l(\theta|\theta_k)$ in each iteration and take the corresponding argument as the next parameter guess, i.e.

$$\theta_{k+1} = \operatorname*{argmax}_{\theta} l(\theta|\theta_k).$$

A variation of the EM algorithm is the Generalised Expectation-Maximisation algorithm (GEM), which just aims at increasing $l(\theta|\theta_k)$ in each iteration instead of maximising it, cf. [84].
So far there is no hidden or missing data in the problem formulation, in some applications the inclusion of hidden variables arises from the problem itself, like in the HMM context, in others it is just a technical dodge in that *assuming* hidden variables may make the maximum likelihood estimation tractable. However, given a vector of hidden variables $\boldsymbol{h}$ the log-likelihood function to be maximised can be written as

$$l(\theta) = \log \int_{\boldsymbol{h}} p(\boldsymbol{z}|\boldsymbol{h}, \theta) p(\boldsymbol{h}|\theta) d\boldsymbol{h}.$$

A function $l(\theta|\theta_k)$ that fulfils the above stated conditions wrt. to the form of the log-likelihood function just given is

$$l(\theta|\theta_k) := \int_{\boldsymbol{h}} p(\boldsymbol{h}|\theta_k, \boldsymbol{z}) \log \left(\frac{p(\boldsymbol{z}, \boldsymbol{h}|\theta)}{p(\boldsymbol{h}|\theta_k, \boldsymbol{z})} \right) d\boldsymbol{h}.$$

It can be easily seen that this function defines a lower by applying Jensen's inequality:

$$\begin{aligned} l(\theta|\theta_k) &\leq \log \left(\int_{\boldsymbol{h}} p(\boldsymbol{h}|\theta_k, \boldsymbol{z}) \frac{p(\boldsymbol{z}, \boldsymbol{h}|\theta)}{p(\boldsymbol{h}|\theta_k, \boldsymbol{z})} d\boldsymbol{h} \right) \\ &= \log \left(\int_{\boldsymbol{h}} p(\boldsymbol{z}|\boldsymbol{h}, \theta) p(\boldsymbol{h}|\theta) \frac{p(\boldsymbol{h}|\theta_k, \boldsymbol{z})}{p(\boldsymbol{h}|\theta_k, \boldsymbol{z})} d\boldsymbol{h} \right) \\ &= \log \left(\int_{\boldsymbol{h}} p(\boldsymbol{z}|\boldsymbol{h}, \theta) p(\boldsymbol{h}|\theta) d\boldsymbol{h} \right) = l(\theta). \end{aligned}$$

Also the inequality for the function $l(\theta|\theta_k)$ gets sharp at the current estimate θ_k, since

$$\begin{aligned} l(\theta_k|\theta_k) &= \int_{\boldsymbol{h}} p(\boldsymbol{h}|\theta_k, \boldsymbol{z}) \log \left(\frac{p(\boldsymbol{z}, \boldsymbol{h}|\theta_k)}{p(\boldsymbol{h}|\theta_k, \boldsymbol{z})} \right) d\boldsymbol{h} \\ &= \int_{\boldsymbol{h}} p(\boldsymbol{h}|\theta_k, \boldsymbol{z}) \log \left(\frac{p(\boldsymbol{z}, \boldsymbol{h}, \theta_k) p(\boldsymbol{z}, \theta_k)}{p(\theta_k) p(\boldsymbol{z}, \boldsymbol{h}, \theta_k)} \right) d\boldsymbol{h} \\ &= \int_{\boldsymbol{h}} p(\boldsymbol{h}|\theta_k, \boldsymbol{z}) d\boldsymbol{h} \log(p(\boldsymbol{z}|\theta_k)) = \log(p(\boldsymbol{z}|\theta_k)) l(\theta_k). \end{aligned}$$

Finding the new estimate θ_{k+1} which maximises $l(\theta|\theta_k)$ is done by

$$\begin{aligned} \theta_{k+1} &= \operatorname*{argmax}_{\theta} l(\theta|\theta_k) \\ &= \operatorname*{argmax}_{\theta} \int_{\boldsymbol{h}} p(\boldsymbol{h}|\theta_k, \boldsymbol{z}) \log \left(\frac{p(\boldsymbol{z}, \boldsymbol{h}|\theta)}{p(\boldsymbol{h}|\theta_k, \boldsymbol{z})} \right) d\boldsymbol{h} \\ &= \operatorname*{argmax}_{\theta} \int_{\boldsymbol{h}} p(\boldsymbol{h}|\theta_k, \boldsymbol{z}) \log \left(p(\boldsymbol{z}, \boldsymbol{h}|\theta) \right) d\boldsymbol{h} \\ &= \mathbb{E}_{\boldsymbol{h}|\theta_k, \boldsymbol{z}}[\log \left(p(\boldsymbol{z}, \boldsymbol{h}|\theta) \right)). \end{aligned}$$

Note that as $p(\boldsymbol{h}|\theta_k, \boldsymbol{z}) = cp(\boldsymbol{z}, \boldsymbol{h}|\theta_k)$ and the constant c is independent of θ and $\boldsymbol{h}$, θ_{k+1} can also be obtained by maximising

$$\int_{\boldsymbol{h}} p(\boldsymbol{z}, \boldsymbol{h}|\theta_k) \log \left(p(\boldsymbol{z}, \boldsymbol{h}|\theta) \right) d\boldsymbol{h}. \tag{3.8}$$

Therefore the name Expectation-Maximisation algorithm, as first the expectation of the full log-likelihood function is calculated, which needs the determination of $p(\boldsymbol{z}, \boldsymbol{h}|\theta_k)$, and afterward it is maximised wrt. to θ. We have shown that these procedure indeed generates a sequence of non-decreasing likelihood estimates, i.e. $l(\theta_{k+1}) \geq l(\theta_k), k \geq 1$. Under quite general conditions it can be shown that this series will converge to a maximum or saddle point of l (for an elegant and easy proof we refer to [84]). In practice the iteration is stopped if the increase in the log-likelihood falls below a predefined threshold. The drawback of this elegant method is that in general a local and not a global maximum of the log-likelihood function is obtained. Therefore one has to provide a good initial guess for the parameters, try different initial parameters or has to couple the EM algorithm with some global optimisation method, e.g. deterministic annealing [119], but of course non of these can guarantee the finding of the global maximum, if it exists.

3.2.2 HMM's with Gaussian Output

In an HMM with continuous output distributions often Gaussian output distributions are assumed, i.e. the output distributions $f_1, \ldots, f_n$ are specified by

$$f_k(\boldsymbol{z}_t) = |2\pi R_k|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\operatorname{tr}\left((\boldsymbol{z}_t-\boldsymbol{\mu}_k)(\boldsymbol{z}_t-\boldsymbol{\mu}_k)'R_k^{-1}\right)\right),\ 1 \leq k \leq n.$$

These are parameterised by a mean vectors $\boldsymbol{\mu}_k$ and positive definite covariance matrices R_k. Therefore the whole parameter set of an HMM with Gaussian output is given by

$$\theta = (\boldsymbol{\pi}, P, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_m, R_1, \ldots, R_m).$$

We now sketch how to adapt the EM algorithm to this setting, for details we refer to [11]. According to Eq. (3.8), given a parameter set θ_k, an improved parameter set is obtained by

$$\begin{aligned}\theta_{k+1} = \operatorname*{argmax}_\theta l(\theta|\theta_k) &= \operatorname*{argmax}_\theta \sum_{\boldsymbol{h}} p(\boldsymbol{z},\boldsymbol{h}|\theta_k)\log\left(p(\boldsymbol{z},\boldsymbol{h}|\theta)\right)\\ &= \operatorname*{argmax}_\theta \sum_{\boldsymbol{h}} p(\boldsymbol{h}|\theta_k,\boldsymbol{z})\log\left(p(\boldsymbol{z},\boldsymbol{h}|\theta)\right),\end{aligned} \tag{3.9}$$

where the sum is over all possible hidden trajectories.

The complete probability density $p(\boldsymbol{z},\boldsymbol{h}|\theta)$ is easily evaluated as

$$p(\boldsymbol{z},\boldsymbol{h}|\theta) = \pi_{h_1} f_{h_1}(\boldsymbol{z}_1)\prod_{t=1}^{T-1} p_{h_t,h_{t+1}} f_{h_{t+1}}(\boldsymbol{z}_{t+1}).$$

Conveniently the function to be maximised in (3.9) splits into three parts,

$$\begin{aligned}\sum_{\boldsymbol{h}}\Bigg(\log(\pi_{h_1})p(\boldsymbol{h}|\boldsymbol{z},\theta_k) &+ \sum_{t=1}^{T-1}\log(p_{h_t,h_{t+1}})p(\boldsymbol{h}|\boldsymbol{z},\theta_k)\\ &+ \sum_{t=1}^{T}\log(f_{h_t}(\boldsymbol{z}_t))p(\boldsymbol{h}|\boldsymbol{z},\theta_k)\Bigg),\end{aligned} \tag{3.10}$$

which means that the optimisation problem can be solved independently for $\boldsymbol{\pi}$, P and $\{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_m, R_1, \ldots, R_m)\}$. Solving these, with the use of Langrangian multiplies to include the constraints $\sum_i \pi_i = 1$ and $\sum_j p_{ij} = 1$ for $1 \leq i \leq m$, gives the solutions

$$\pi_i^{(k+1)} = p(h_1 = i|\theta_k,\boldsymbol{z}), \tag{3.11}$$

$$p_{ij}^{(k+1)} = \frac{\sum_{t=1}^{T-1} p(h_t = i, h_{t+1} = j|\theta_k,\boldsymbol{z})}{\sum_{t=1}^{T-1} p(h_t = i|\theta_k,\boldsymbol{z})}, \tag{3.12}$$

$$\boldsymbol{\mu}_l^{(k+1)} = \frac{\sum_{t=1}^{T} p(h_t = i|\theta_k,\boldsymbol{z})\boldsymbol{z}_t}{\sum_{t=1}^{T} p(h_t = i|\theta_k,\boldsymbol{z})}, \tag{3.13}$$

$$R_l^{(k+1)} = \frac{\sum_{t=1}^{T} p(h_t = i|\theta_k,\boldsymbol{z})(\boldsymbol{z}_t-\boldsymbol{\mu}_l^{(k+1)})'(\boldsymbol{z}_t-\boldsymbol{\mu}_l^{(k+1)})}{\sum_{t=1}^{T} p(h_t = i|\theta_k,\boldsymbol{z})}. \tag{3.14}$$

However, at this point it is not clear if these solutions can be computed efficiently. In fact they can, via the introduction of the so-called backward and forward variables, which is shown in the next section.

3.2.3 Dynamical Programming Approaches for Efficient Implementation of the EM algorithm

In this section we outline how to compute the desired quantities in HMM estimation in an efficient way using dynamical programming approaches. The presentation closely follows [36]. Besides the below cited literature the reader is referred to [96] for a more detailed presentation with examples.

Backward-Forward Variables

In this section the so-called backward and forward variables are introduced. These can be used to update the EM parameters in Eq. (3.11)-(3.14) efficiently. Note that this concept of dynamical programming does not take into account the specific form of the output distributions, and therefore is not restricted to Gaussian output distributions. The backward and forward variables divide an observation sequence $\boldsymbol{z}$ recursively in partial subsequences: those from time 1 to time t and those from time $t+1$ up to T. Given a particular parameter set θ, the forward variables are defined as

$$\alpha_t(i) = p(\boldsymbol{z}_1, \boldsymbol{z}_2, \dots, \boldsymbol{z}_t, h_t = i|\theta),$$

which denotes the probability of the observation sequence up to time t together with the probability that the system is in hidden state i at time t conditioned wrt. the given model θ. The backward variables are defined by

$$\beta_t(i) = p(\boldsymbol{z}_{t+1}, \boldsymbol{z}_{t+2}, \dots, \boldsymbol{z}_T|h_t = i, \theta),$$

which denotes the probability of the observation sequence from time $t+1$ to T, under the condition that the hidden process is in state i at time t and on the model θ. The computation of the probability $\alpha_T(i) = p(\boldsymbol{z}, h_T = i|\theta)$ is possible with $m^2 T$ operations, as recursive formulas can be used:

$$\begin{aligned} \alpha_1(i) &= \pi_i f_i(\boldsymbol{z}_1), && 1 \le i \le n, \\ \alpha_{t+1}(j) &= \left[\sum_{i=1}^{n} \alpha_t(i) p_{ij}\right] f_j(\boldsymbol{z}_{t+1}), && 1 \le j \le n,\, 1 \le t \le T-1 \end{aligned} \tag{3.15}$$

The backward variables $\beta_t(i)$ can be computed with an analogous formula:

$$\begin{aligned} \beta_T(i) &= 1, && 1 \le i \le n, \\ \beta_t(i) &= \sum_{j=1}^{n} p_{ij} f_j(\boldsymbol{z}_{t+1}) \beta_{t+1}(j), && 1 \le i \le n,\, T-1 \ge t \ge 1. \end{aligned} \tag{3.16}$$

In order to avoid obscure notations, we have omitted subscripts indicating the dependence of all the variables above, except those of the observations, on the given model parameters θ. With the introduced forward and backward variables one can easily compute

$$p(\boldsymbol{z}|\theta) = \sum_{i=1}^{n} \alpha_t(i)\beta_t(i), \tag{3.17}$$

and therefore solve the standard problem ($T1$), which would otherwise require a, in general non feasible, summation over all possible hidden paths.

Furthermore, the forward and backward variables, computed wrt. to θ_k, can be used to obtain the required quantities in Eq. (3.11)-(3.14) for every EM step, i.e. $p(h_1 = i|\theta_k, \boldsymbol{z})$ and $p(h_t = i, h_{t+1} = j|\theta_k, \boldsymbol{z})$ for $1 \leq i, j \leq m$ and $1 \leq t \leq T$, and thereby solve (T2), as

$$\begin{aligned} p(h_t = i, h_{t+1} = j|\theta_k, \boldsymbol{z}) &= \frac{p(h_t = i, h_{t+1} = j, \theta_k, \boldsymbol{z})}{p(\theta_k, \boldsymbol{z})} \\ &= \frac{p(h_t = i, h_{t+1} = j, \boldsymbol{z}|\theta_k)}{p(\boldsymbol{z}|\theta_k)} \\ &= \frac{\alpha_t(i)p_{ij}f_j(\boldsymbol{z}_{t+1})\mathrm{B}_{t+1}(j)}{p(\boldsymbol{z}|\theta_k)}, \end{aligned} \tag{3.18}$$

and

$$p(h_t = i|\theta_k, \boldsymbol{z}) = \sum_{j=1}^{n} p(h_t = i, h_{t+1} = j|\theta_k, \boldsymbol{z}). \tag{3.19}$$

The Viterbi algorithm

Having computed the maximum likelihood estimate for the parameters of an HMM model with the EM algorithm, one can face problem ($T3$): conditional on the MLE $\hat{\boldsymbol{\theta}}$ for the model parameters, compute the most likely hidden path $\hat{\boldsymbol{h}} = (\hat{h}_1, \hat{h}_2, \ldots, \hat{h}_T)$. The computation can be done efficiently by the Viterbi Algorithm, a dynamical programming algorithm proposed 1967 by Andrew Viterbi [121], see also [37]. The optimal hidden path, also called Viterbi path, is done by recursive computation of

$$\delta_t(i) = \max_{h_1,\ldots,h_{t-1}} p(h_1, \ldots, h_{t-1}, h_t = i, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_t|\hat{\boldsymbol{\theta}}),\ 1 \leq t \leq T, 1 \leq i \leq n$$

From $\boldsymbol{\delta_T} = (\delta_T(1), \ldots, \delta_T(n))$ one can read off the optimal hidden state $\hat{h}_T$, if one has kept track of the state sequence that led to this state, the optimal path can be determined via backtracking, as shown in the full algorithm:

1) Initialisation:

$$\begin{aligned} \delta_1(i) &= \pi_i f_i(\boldsymbol{z}_1),\ 1 \leq i \leq n \\ \psi_1(i) &= 0 \end{aligned}$$

2) Recursion:

$$\begin{aligned}\delta_t(i) &= \max_{1\leq j\leq n}[\delta_{t-1}(j)p_{ji}]f_i(\boldsymbol{z}_t),\\ \psi_t(i) &= \operatorname*{argmax}_{1\leq j\leq n}[\delta_{t-1}(j)p_{ji}],\\ &\quad 2\leq t\leq T, 1\leq i\leq n\end{aligned}$$

3) Backtracking:

$$\begin{aligned}\hat{h}_T &= \operatorname*{argmax}_{1\leq j\leq n}[\delta_T(j)]\\ \hat{h}_t &= \psi_{t+1}(\hat{h}_{t+1}),\quad t=T-1,T-2,\dots,1.\end{aligned}$$

Note that this algorithm is based on the Markov property of the HMM model, i.e. probabilities about future events are only dependent on the preceding event.

Example

We are going to close the section on HMMs with Gaussian output distributions with an example, where we also illustrate the difference between HMM and PCCA analysis. Therefore an HMM model is set up, the hidden process is a two state process with transition matrix

$$P=\begin{pmatrix}\frac{999}{1000} & \frac{1}{1000}\\ \frac{1}{1000} & \frac{999}{1000}\end{pmatrix},$$

and the output distributions are two Gaussian distribution with means and covariance matrices

$$\boldsymbol{\mu}_1=\begin{pmatrix}0\\ \frac{1}{2}\end{pmatrix},\ \boldsymbol{\mu}_2=\begin{pmatrix}1\\ -1\end{pmatrix},\ R_1=\begin{pmatrix}3 & 1\\ 1 & 1\end{pmatrix},\ R_2=\begin{pmatrix}3 & -0.2\\ -0.2 & 1\end{pmatrix}.$$

As shown in Fig. 3.2, the thereby defined Gaussian output distributions are overlapping in the sense that their 0.95 confidence regions do. Starting the hidden process in state 1 a random trajectory of 10000 data points is generated from the HMM. The described EM algorithm was used to estimate an HMM based upon the observations. An initial hidden path was generated using a transition matrix $\frac{1}{100}\left(\begin{smallmatrix}99 & 1\\ 1 & 99\end{smallmatrix}\right)$, which in turn was used to generate an initial parameter guess by usage of Eq. (3.11)-(3.14). The EM algorithm, started with these randomly generated parameters, converged after 10 iterations returning parameter estimates which are accurate at least to the first decimal place. The estimated parameter set was used to compute a Viterbi path as described in § 3.2.3. The computed Viterbi path is barely distinguishable from the true hidden path as seen in Fig. 3.2, closer inspections yields 7 wrong allocations.

Trying to analyse the obtained trajectory with PCCA does not yield such a smooth looking result. The transition matrix obtained by discretisation of the two dimensional observation space has the following largest eigenvalues:

j	1	2	3	4	5
λ_j	1	0.5830	0.2546	0.2254	0.2045

$\dots$.

Obviously there is no Perron cluster, i.e. a group of eigenvalues close to one, but at least a significant bigger gap between the second and the third eigenvalue than any gap afterwards. However, clustering in 2 metastable sets does not yield a satisfactory solution as the algorithm can not handle the overlapping of the two distributions, cf. Fig. 3.2.

The better performance of the EM algorithm compared to a PCCA analysis is not a big surprise, as the data was actually generated by an HMM. But there are two more general messages to take away from this example. First, the HMM approach is obviously able to cope with overlapping distributions, while PCCA is not. Second, a drawback of the HMM approach is that we have to choose the (expected) number of metastable sets initially as otherwise the EM iterations are not defined, whereas the PCCA approach gives a clear metastability criteria (which, as we have seen, might fail in situations like just constructed). In § 3.4.3 we propose how to overcome this inherent weakness of the HMM approach by combining both approaches.

3.3 Reduced Modelling of Internal Dynamics

In § 3.1 and § 3.2 we introduced methods to identify metastable sets in the observation space of some observed time series. The outcome is a Markov chain which models the transitions between these sets. A drawback of both of the presented approaches is that they rely on ergodicity or stationarity of the observed time series, since on one hand the transfer operator approach depends on estimation of the transfer operator with respect to the invariant measure and on the other hand the HMM approach assumes, within a hidden state, independent and identically distributed observations. To overcome this limitation Illia Horenko et al. [52, 55, 75, 106] proposed to model the dynamics within a hidden state and to include this into the time series analysis. This was done by extending the standard HMM approach in two aspects, first by adding dependency in the observed data, i.e. the observed process is itself not only dependent upon the hidden process but also on previous data, second by assuming a certain kind of dependency, namely, that the observed process is governed by a stochastic differential equation (SDE). In the following we present this approach with some extensions, namely, the employment of different parameter sets, which will turn out useful in Chapter 4, and the inclusion of higher order memory in the HMM.

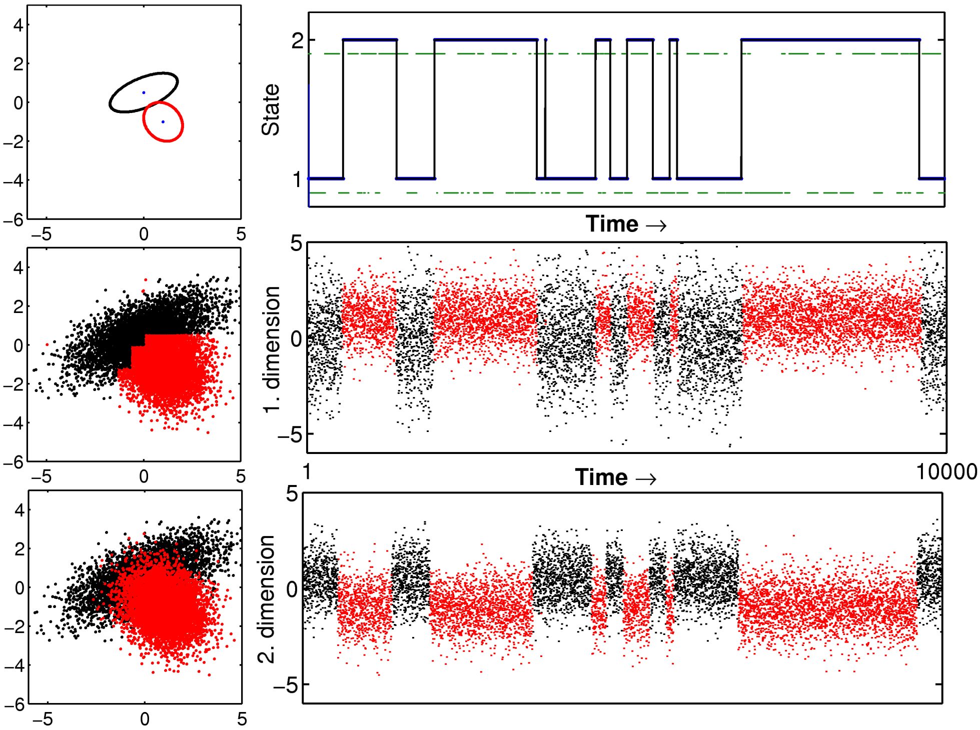


Figure 3.2: *Left column:* Top: The (overlapping) 95% confidence region ellipses of the two Gaussian output distributions. Middle: The result of the PCCA analysis depicted via colour coding of the data points, obviously the spatial separation is not satisfactory. Bottom: The result of the HMM analysis gives a perfect result.
Right column: Top: Shown is the initial Viterbi path to initialise the EM algorithm (green, shifted for better visualisation), the obtained Viterbi path after convergence of the EM algorithm (black) and the indistinguishable true hidden path (blue). Bottom: The two dimensions of the generated time series colour-coded according to the Viterbi path of the HMM analysis.

3.3.1 Langevin Dynamics and Stochastic Differential Equations

A common model to approximate the time evolution of an observation restricted to some potential that is excerpted to noise is by means of a first order SDE

$$\dot{\boldsymbol{z}}(t) = -\nabla_{\boldsymbol{z}} V(\boldsymbol{z}(t)) + \Sigma \dot{\boldsymbol{W}}(t), \tag{3.20}$$

where $\boldsymbol{z} \in \mathbb{R}^d$ is the observed quantity, V is some potential function, $\boldsymbol{W}(t)$ a d-dimensional Brownian motion coupled to the dynamics via the $\Sigma \in \mathbb{R}^{d\times d}$ noise intensity matrix. In fact, such models arise often in the context of model reduction of dynamical systems. If there is a time scale separation within the system , i.e. there is a subset of variables that evolves fast wrt. to the others, then the projected dynamics of the slow variables can, under regularity assumptions, be approximated by such stochastic models, where the slow d.o.f.'s are driven by some *effective potential* function, i.e. V, while the influence of the fast d.o.f.'s is modeled by a noise term [62, 68]. Opposed to the model given in (3.20), such a model would in general contain a memory term. We stick for the moment to models without memory and comment later in § 3.3.3 on generalised models obtained by adding a memory kernel to the noise.

A further model is the Langevin equation as introduced in § 2.3, which can be stated in a somewhat generalised way as

$$\begin{aligned}\dot{\boldsymbol{q}}(t) &= M^{-1}\boldsymbol{p}(t)\\ \dot{\boldsymbol{p}}(t) &= -\nabla_{\boldsymbol{q}} U(\boldsymbol{q}(t)) - \gamma M^{-1}\boldsymbol{p}(t) + \sigma \dot{\boldsymbol{W}}(t).\end{aligned}$$

Here M and γ and σ are each elements of $\mathbb{R}^{d\times d}$, i.e. friction and noise intensity are defined by matrices. Formally the Langevin equation can be put in the form of Eq. (3.20) by defining

$$\boldsymbol{z} := \begin{pmatrix}\boldsymbol{q}\\ \boldsymbol{p}\end{pmatrix}, \ \nabla_{\boldsymbol{z}} V := \begin{pmatrix} 0 & -M^{-1}\\ \nabla_{\boldsymbol{q}} U & \gamma M^{-1}\end{pmatrix}, \ \Sigma := \begin{pmatrix}0 & 0\\ 0 & \sigma\end{pmatrix}.$$

Note that this formal expression does only make sense if the assumed potential is in fact quadratic (which we are going to assume below). If the friction matrix defined by γ is sufficiently large[1] compared to the masses defined in M the dynamics of the positional variable of the Langevin equation can be approximated by the so-called Smoluchowski, or overdamped Langevin, dynamics:

$$\gamma \dot{\boldsymbol{q}}(t) = -\nabla_{\boldsymbol{q}} U(\boldsymbol{q}(t)) + \sigma \dot{\boldsymbol{W}}(t), \tag{3.21}$$

which already is in the form stated above, so that both dynamics can be represented by Eq. (3.20). Both the Smoluchowski and the Langevin dynamics

[1] The sketchy term "sufficiently large" refers to the construction of a Smoluchowski equation from a Langevin equation with a friction $\beta\gamma$ where the limit of β to infinity is considered, for more details we refer to [46, Ch. 2] and [85, Ch. 10].

exhibit an invariant measure of an easy form, the Gibbs distribution, which reads for the Langevin dynamics

$$p_s(\boldsymbol{q},\boldsymbol{p}) \propto \exp\left(-\beta\left(\frac{1}{2}\boldsymbol{p}'M^{-1}\boldsymbol{p}+U(\boldsymbol{q})\right)\right)$$

and for the Smoluchowski dynamics

$$p_s(\boldsymbol{q}) \propto \exp(-\beta U(\boldsymbol{q})),$$

as long as M is symmetric and the so-called multivariate fluctuation-dissipation relation

$$\beta\sigma\sigma' = \gamma + \gamma' \tag{3.22}$$

holds. This can be seen easily by inserting the given measures in the Fokker-Planck equation given in (A.2).

Since estimation of the effective potential of a non-linear stochastic differential equation, like the one given in (3.20), is difficult, the approach followed here is to piece-wise linearise this equation, i.e. (3.20) is approximated by a set of linear SDE's, each of them representing some local dynamics, which are coupled by a Markovian switching process. That is, we assume the following dynamical system

$$\begin{aligned}\dot{\boldsymbol{z}}(t) &= F_{h(t)}\left(\boldsymbol{z}(t)-\boldsymbol{\mu}_{h(t)}\right)+\Sigma_{h(t)}\dot{\boldsymbol{W}}(t)\\ h(t) &\in \{1,\dots,s\},\end{aligned} \tag{3.23}$$

with $(F_1,\dots,F_s)$ a set of $(d\times d)$ dimensional force matrices, $(\Sigma_1,\dots,\Sigma_s)$ a set $(d\times d)$ dimensional noise intensity matrices, a set of d-dimensional mean vectors $(\boldsymbol{\mu}_1,\dots,\boldsymbol{\mu}_s)$ and a switching process h which is supposed to be Markovian. This is equivalent to the assumption of locally quadratic potential functions

$$U_i(\boldsymbol{z}) = -\frac{1}{2}(\boldsymbol{z}-\boldsymbol{\mu}_i)'F_i(\boldsymbol{z}-\boldsymbol{\mu}_i)$$

in Eq. (3.20). If F_i is assumed to be positive definite and symmetric or if F_i is positive definite and commutes with $\Sigma\Sigma'$ it can be shown, via inserting in the Fokker-Planck Equation again, that the invariant density of such local model is given by

$$p_s(\boldsymbol{z}) \propto \exp\left(-(\boldsymbol{z}-\boldsymbol{\mu}_i)'F_i\Sigma\Sigma'(\boldsymbol{z}-\boldsymbol{\mu}_i)\right).$$

This can also be seen by a transformation

$$\gamma_i := 2\Sigma_i\Sigma_i' \quad \tilde{F}_i := \gamma_i F_i \quad \sigma_i := \gamma_i\Sigma_i,$$

which leads to

$$\gamma_i\dot{\boldsymbol{z}} = -\nabla_{\boldsymbol{z}}U_i(\boldsymbol{z})+\sigma_i\dot{\boldsymbol{W}}(t),$$

i.e. the form of the Smoluchowski equation (3.21), and fulfils the fluctuation-dissipation relationship (3.22) by definition of γ_i. Note that even if F_i is not of this form, but have all eigenvalues with negative real parts, a Gaussian invariant measure still can be computed according to Cor. A.1.4, but the covariance matrix does not have the easy form $F_i \Sigma\Sigma'$ anymore.

In the molecular context each of the linear SDE's would represent fluctuations around some global conformation, while the switching process simulates the transitions between them [42, 75, 106]. Before we start to discuss the parameterisation of the piece-wise model given in Eq. (3.23) we discuss how to estimate the parameters for a single linear SDE, i.e. for

$$\dot{\boldsymbol{z}}(t) = F\left(\boldsymbol{z} - \boldsymbol{\mu}\right) + \Sigma \dot{\boldsymbol{W}}(t) \tag{3.24}$$

in the next section.

3.3.2 Parameter Sets and Estimation for Linear SDE's

The theory of linear SDE's, like the one given in Eq. (3.24), is well understood and the most important results for our purposes can be found in Appendix A.1. Its solutions are Markov processes and furthermore, under the assumption of fixed or Gaussian distributed initial conditions, Gaussian processes. Assume for the moment that a time series $\boldsymbol{z}_t := \boldsymbol{z}((t-1)\tau), t = 1, \ldots, T$, i.e. an observation at discrete and equidistant time points, is generated by a single linear SDE as given in Eq. (3.24). According to Cor. A.1.4 the density of $\boldsymbol{z}_{t+1}$ conditional on $\boldsymbol{z}_t$ is given by

$$p(\boldsymbol{z}_{t+1}|\boldsymbol{z}_t) = |2\pi R(\tau)|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\operatorname{tr}\left((\boldsymbol{z}_t - \boldsymbol{\mu}_t)(\boldsymbol{z}_t - \boldsymbol{\mu}_t)' R(\tau)^{-1}\right)\right), \tag{3.25}$$

with mean and covariance

$$\begin{aligned} \boldsymbol{\mu}_t &= \boldsymbol{\mu} + \exp(\tau F)(\boldsymbol{z}_{t-1} - \boldsymbol{\mu}), \\ R(\tau) &= \int_0^\tau \exp(-F(\tau - s))\Sigma\Sigma' \exp(-F'(\tau - s))ds. \end{aligned} \tag{3.26}$$

Therefore a likelihood function conditional on the hole time series $z = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_T\}$ can be written as

$$\begin{aligned} L(F, \Sigma, \boldsymbol{\mu}|z) &= \prod_{t=2}^{T} |2\pi R(\tau)|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\operatorname{tr}\left((\boldsymbol{z}_t - \boldsymbol{\mu}_t)(\boldsymbol{z}_t - \boldsymbol{\mu}_t)' R(\tau)^{-1}\right)\right) \\ &= |2\pi R(\tau)|^{-\frac{T-1}{2}} \exp\left(-\frac{1}{2}\operatorname{tr}\left(\left(\sum_{t=2}^{T}(\boldsymbol{z}_t - \boldsymbol{\mu}_t)(\boldsymbol{z}_t - \boldsymbol{\mu}_t)'\right) R(\tau)^{-1}\right)\right), \end{aligned}$$

where $\boldsymbol{z}_1$ is used as initial value. Unfortunately, it turns out that (analytical) maximisation of the so obtained likelihood function wrt. the parameters F, Σ and $\boldsymbol{\mu}$ is not possible. But Horenko et al. obtained analytical estimators for a transformed parameter set, namely $\theta = (\exp(\tau F), \Sigma\Sigma', \boldsymbol{\mu})$, which are restated in the following theorem [55].

Theorem 3.3.1. *Given a $\tau > 0$ and a time series $z = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_T\}$, with $\boldsymbol{z}_t := \boldsymbol{z}((t-1)\tau)$, which is generated by*

$$d\boldsymbol{z} = F(\boldsymbol{z} - \boldsymbol{\mu}) + \Sigma \dot{\boldsymbol{W}}(t).$$

Define the empirical mean, the empirical covariance matrix and the empirical (normalised) autocorrelation matrix of the time series by

$$\begin{aligned} \bar{\boldsymbol{z}} &= \frac{1}{T-1} \sum_{t=2}^{T} \boldsymbol{z}_t, \\ Cov(\boldsymbol{z}) &= \frac{1}{T-1} \sum_{t=2}^{T} (\boldsymbol{z}_t - \bar{\boldsymbol{z}})(\boldsymbol{z}_t - \bar{\boldsymbol{z}})' \\ Cor(\boldsymbol{z}) &= \frac{1}{T-1} \left(\sum_{t=1}^{T-1} (\boldsymbol{z}_{t+1} - \bar{\boldsymbol{z}})(\boldsymbol{z}_t - \bar{\boldsymbol{z}}) \right) Cov(\boldsymbol{z})^{-1}. \end{aligned} \tag{3.27}$$

Suppose that $Cov(\boldsymbol{z})$ is positive definite. Then, the MLEs of $\exp(\tau F)$ *and $\boldsymbol{\mu}$ are given by*

$$\begin{aligned} \exp(\tau \hat{F}) &= Cor(\boldsymbol{z}) \\ \hat{\boldsymbol{\mu}} &= \bar{\boldsymbol{z}} - \frac{\boldsymbol{z}_T - \boldsymbol{z}_1}{T-1} (I - Cor(\boldsymbol{z}))^{-1}, \end{aligned} \tag{3.28}$$

where I is an identity matrix of appropriate size. From these quantities the optimal MLE of the noise intensity matrix estimator $\hat{\Sigma}\hat{\Sigma}'$ can be computed by

$$\hat{\Sigma}\hat{\Sigma}' = -\Big((Cov(\boldsymbol{z}) + E)\hat{F} + \hat{F}(Cov(\boldsymbol{z}) + E) \Big),$$

where E is a symmetric matrix that satisfies the Sylvester equation

$$\begin{aligned} &Cor(\boldsymbol{z}) E Cor(\boldsymbol{z})' - E = \\ &\quad \frac{(\boldsymbol{z}_T - \boldsymbol{z}_1)(\boldsymbol{z}_T - \boldsymbol{z}_1)'}{(T-1)^2} + \frac{1}{T-1} \left((\boldsymbol{z}_T - \bar{\boldsymbol{z}})(\boldsymbol{z}_T - \bar{\boldsymbol{z}})' - (\boldsymbol{z}_1 - \bar{\boldsymbol{z}})(\boldsymbol{z}_1 - \bar{\boldsymbol{z}})' \right), \end{aligned}$$

which yields a unique solution, whenever $\sigma(F) \in \mathbb{C}^-$.

Although Theorem 3.3.1 provides an analytical expression for the transformed parameter set, we propose yet another parameter set for two reasons. First, it will turn out that the new parameter set, which we are going to introduce now, will allow straightforwardly significant extensions to our original model. Second, the parameter set of Horenko has the drawback that the likelihood function is not integrable wrt. to it. This can be seen by keeping $\exp(\tau F) \equiv I$ fixed, I is an identity matrix of appropriate size, then, cf. Eq. (3.26), the parameter $\boldsymbol{\mu}$ disappears from the likelihood function making the integration impossible. Therefore the likelihood function can not be used to induce a density in parameter space from the observations, which we will need later to carry out the change point analysis in Ch. 4.

Another parameter set can be found by using the density of $\boldsymbol{z}_{t+1}$ conditional on $\boldsymbol{z}_t$ to obtain a discrete time process which has exactly the same distribution as a discrete observation of the continuous SDE. Interpreting $\boldsymbol{z}_{t+1}$ as a random variable dependent on the observation $\boldsymbol{z}_t$ we have

$$\begin{aligned} \boldsymbol{z}_{t+1} &= \mathcal{N}(\boldsymbol{\mu} + \exp(\tau F)(\boldsymbol{z}_t - \boldsymbol{\mu}), R) \\ &= (I - \exp(\tau F))\boldsymbol{\mu} + \exp(\tau F)\boldsymbol{z}_t + \mathcal{N}(\boldsymbol{0}, R), \end{aligned} \tag{3.29}$$

where $\mathcal{N}$ is a multivariate normal distributed random variable and I again an identity matrix of appropriate size. Eq. (3.29) reveals the autoregressive structure of order one, abbreviated by VAR(1), of the time series of discrete observations. Defining

$$\begin{aligned} \varPhi &:= \big((I - \exp(\tau F))\boldsymbol{\mu} \quad \exp(\tau F)\big) && \in \mathbb{R}^{d\times(d+1)} \\ X &:= \begin{pmatrix} 1 & \dots & 1 \\ \boldsymbol{z}_1 & \dots & \boldsymbol{z}_{T-1} \end{pmatrix} && \in \mathbb{R}^{(d+1)\times(T-1)} \\ Y &:= \big(\boldsymbol{z}_2, \dots, \boldsymbol{z}_T\big) && \in \mathbb{R}^{d\times(T-1)} \\ \epsilon &:= \big(\mathcal{N}(\boldsymbol{0}, R), \dots, \mathcal{N}(\boldsymbol{0}, R)\big) && \in \mathbb{R}^{d\times(T-1)}, \end{aligned}$$

allows to write Eq. (3.29) in a compact form

$$Y = \varPhi X + \epsilon.$$

Transforming the parameter set θ to $\tilde{\theta} = (\varPhi, R)$, leads to a reformulated likelihood function

$$L(\tilde{\theta}|\boldsymbol{z}) = \left(\frac{1}{\sqrt{|2\pi R|}}\right)^{(T-1)} \exp\left(-\frac{1}{2}\operatorname{tr}((Y - \varPhi X)(Y - \varPhi X)' R^{-1})\right), \tag{3.30}$$

for which there are analytic MLE's $\hat{\varPhi}$ and $\hat{R}$, easily obtained by matrix calculus[2] [73, 87],

$$\hat{\varPhi} = YX'(XX')^{-1} \text{ and } \hat{R} = (Y - \hat{\varPhi}X)(Y - \hat{\varPhi}X)'/(T - 1). \tag{3.31}$$

Therefore, transforming the parameter set to $\tilde{\theta}$ has the advantages that (i) the distribution of the discrete observations is fully characterised by $\tilde{\theta}$, (ii) analytical MLE's are available and (iii) the likelihood function, as it is shown in App. A.2, is integrable over the parameter space. Note, however, that the estimators given in (3.31) are not computed in the way stated, as the matrix inversion leads to numerical instabilities, we show in § 3.3.4 how to compute them properly.

[2] An excellent collection of matrix calculus rules can be found in The Matrix Cockbook, accessible under http://matrixcookbook.com/.

3.3.3 Higher Order Models

Considering the discrete observations of a linear SDE as realisations of a VAR(1) process naturally establishes a way to include memory in our model by just using a higher order model, i.e. VAR(p) with $p \geq 0$, which is of the form

$$\boldsymbol{z}_{t+1} = A_0\boldsymbol{\mu} + \sum_{i=1}^{p} A_i \boldsymbol{z}_{t-i+1} + \mathcal{N}(\mathbf{0}, R) \tag{3.32}$$

and parameterised by the vector $A_0\boldsymbol{\mu}$ and the matrices $A_1, \dots, A_p, R$. This process is obviously not a Markov process anymore but exhibits a memory lag of p steps into the past. In fact it is shown in [55] that Eq. (3.32) can be interpreted as a time discretisation of a generalised Langevin process

$$\dot{\boldsymbol{z}}(t) = -\nabla_{\boldsymbol{z}} V(\boldsymbol{z}(t)) - \int_0^t \gamma(t-s)\boldsymbol{z}(s)ds + \varSigma\dot{\boldsymbol{W}}(t), \tag{3.33}$$

under the assumption of a quadratic potential function V, as above, and a piecewise constant memory kernel γ with finite support.

If a fixed order parameter p is assumed, estimation of the parameters of a VAR(p) is analogue to that of the VAR(1) process, only the definitions of the data matrices X and Y have to be extended to

$$X := \begin{pmatrix} 1 & \dots & 1 \\ \boldsymbol{z}_1 & \dots & \boldsymbol{z}_{T-p} \\ \vdots & & \vdots \\ \boldsymbol{z}_p & \dots & \boldsymbol{z}_{T-1} \end{pmatrix} \qquad \in \mathbb{R}^{(dp+1)\times(T-p)}$$

$$Y := \left(\boldsymbol{z}_{p+1}, \dots, \boldsymbol{z}_T\right) \qquad \in \mathbb{R}^{d\times(T-p)}.$$

The estimator $\hat{\varPhi}$ in (3.31) now estimates

$$\varPhi = \begin{pmatrix} A_0\boldsymbol{\mu} & A_1 & A_2 & \dots & A_p \end{pmatrix} \in \mathbb{R}^{d\times(dp+1)}.$$

The estimator of $\hat{R}$ has to be adjusted to the growing number of initial points needed for higher order and becomes

$$\hat{R} = (Y - \hat{\varPhi}X)(Y - \hat{\varPhi}X)'/(T-p).$$

Generalising (3.24) by introducing more memory, i.e. using a VAR(p) model, $p \geq 1$, can be quite essential as even a reduced model of the form (3.23) used to describe the effective dynamics of a molecular system can be a high dimensional model. Since the number of parameters needed to estimate a VAR process grows quadratically with the system dimensionality, a further reduction of the dimensionality by restriction to linear subspace, e.g. via principal component analysis, might be advisable. Unfortunately, the class of VAR processes is not invariant w.r.t. linear transformations, instead a VAR process is in general transformed to a VARMA process, a VAR process where the noise has a moving average representation. But, since VARMA processes have a representation as infinite VAR processes, we can still approximate the transformed process via a truncated, i.e. finite, VAR process by choosing a high enough order p, cf. [73, Ch. 9].

3.3.4 Computational Aspects for VAR Parameter Estimation

The analytic estimators given in Eq. (3.31) are in general not used for computation of the parameters, as the matrix inversion can be unstable. Instead one can use the moment matrix

$$\begin{aligned} M = M(Z) :&= \sum_{t=p+1}^{T} \begin{pmatrix} 1 \\ \boldsymbol{z}_{t-p} \\ \vdots \\ \boldsymbol{z}_t \end{pmatrix} \begin{pmatrix} 1 & \boldsymbol{z}'_{t-p} & \dots & \boldsymbol{z}'_t \end{pmatrix} \\ &= \begin{pmatrix} XX' & XY' \\ YX' & YY' \end{pmatrix} =: \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} . \end{aligned} \tag{3.34}$$

The moment matrix is an important object as it contains all statistical relevant information about the observed process (under the assumption of a VAR(p) process). This can be seen by rewriting the likelihood function in terms of the blocks in M:

$$\begin{aligned} L(\varPhi, R|Z) =& L(\varPhi, R|M) = \left(\frac{1}{\sqrt{|2\pi R|}} \right)^m \\ & \cdot \exp\left(-\frac{1}{2} \operatorname{tr}((M_{22} - M_{21}\varPhi' - \varPhi M_{12} + \varPhi M_{22}\varPhi')R^{-1}) \right) , \end{aligned} \tag{3.35}$$

where m denotes the upper left scalar entry of M which equals $T - p$, i.e. the length of the observed time series minus p initial values. We will employ this notation $m = m(Z)$ below to avoid the indices for the length of different time series. Also, we will employ subsequently the notation $f(Z|\varPhi, R) \equiv L(\varPhi, R|Z)$ if we want to highlight Eq. (3.35) as a density in data space. The MLE's can be obtained from the moment matrix M in a stable way via a Cholesky factorisation which gives an upper triangular matrix

$$U = \begin{pmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{pmatrix} ,$$

such that

$$M = \begin{pmatrix} XX' & XY' \\ YX' & YY' \end{pmatrix} = \begin{pmatrix} U'_{11}U_{11} & U'_{11}U_{12} \\ U'_{12}U_{11} & U'_{12}U_{12} + U'_{22}U_{22} \end{pmatrix} = U'U.$$

Plugging the Cholesky factorisation into the estimators (3.31) one obtains

$$\hat{\Phi} = (U_{11}^{-1} U_{12})' \text{ and } \hat{R} = \frac{1}{m} U_{22}' U_{22}. \tag{3.36}$$

In the case of an ill-conditioned moment matrix M one can add a regularisation matrix to ensure a well-posed Cholesky factorisation. A possible choice is to use $M + \delta \mathrm{diag}(M)$ instead of M, with a small parameter δ depending on the dimensionality of the problem and the machine precision. Setting $\delta = (q^2 + q + 1)\epsilon$, with ϵ the machine precision and $q = d(p+1) + 1$ the dimension of M, a successful termination of the Cholesky factorisation can be guaranteed [49, 86].

3.3.5 HMM-VAR

After having transformed the problem of estimating parameters of a linear SDE into a problem of estimating parameters of a VAR(1) model and recapitulation of the standard estimators, we now turn to the question of how to estimate the parameters for a piecewise linear SDE model, as given in (3.23). As we have seen, an easy generalisation of our model is to assume a VAR(p) process with $p \geq 0$, accordingly we generalise the discretised version of (3.23) given in (3.29) to

$$\begin{aligned} \boldsymbol{z}_{t+1} &= \boldsymbol{\nu}_{h_t} + \sum_{j=1}^{p} A_j^{[h_t]} \boldsymbol{z}_{t+1-j} + \mathcal{N}(\mathbf{0}, R_{h_t}) \\ h_t &\in \{1, \dots, n\}, \end{aligned} \tag{3.37}$$

where h_t is assumed to be a Markov process, which we call an HMM-VAR model in the sequel. Setting $p = 0$ in the above formulation we simply get an HMM-Gaussian model as in Sec. 3.2.2, with $\boldsymbol{\mu}_i = \boldsymbol{\nu}_i$. Setting $p = 1$ corresponds to the HMM-SDE model in (3.23), with $\exp(\tau F_i) = A_1^{[i]}$ and $\boldsymbol{\mu}_i = (I - A_1^{[i]})^{-1} \boldsymbol{\nu}_i$. In order to estimate the parameters of the HMM-VAR model we again employ the EM algorithm, which is easily adjusted as the additional dependency is only introduced in the observed process, while the Markov assumption on the hidden process still holds.

Given a parameter set θ_k, an improved parameter set θ_{k+1} is obtained by maximisation of the expected log-likelihood function as in (3.9). In (3.10) we saw that the maximisation splits into three parts which can be solved independently. As the hidden model still has the same the form for the estimators for $\boldsymbol{\pi}$ and P given in (3.11) and (3.12), with the only exception that the summations in (3.12) have to start with $t = p + 1$ since the first p observations are taken as fixed initial values. To obtain reestimation formulas for the parameter of the VAR models we have to maximise the third term in (3.10) which is, using

$\boldsymbol{x_t} := (1\ \boldsymbol{z}'_{t-p} \cdots \boldsymbol{z}'_{t-1}),$

$$\begin{aligned}&\sum_{\boldsymbol{h}} \sum_{t=p+1}^{T} \log(f_{h_t}(\boldsymbol{z}_t|\boldsymbol{z}_{t-1}, \dots \boldsymbol{z}_{t-p}))p(\boldsymbol{h}|\boldsymbol{z}, \theta_k) \\ = -\frac{1}{2} &\sum_{i=1}^{n} \sum_{t=p+1}^{T} \big(\log(|2\pi R_i|) + (\boldsymbol{z}_t - \Phi_i \boldsymbol{x_t}')' \, R_i^{-1} \, (\boldsymbol{z}_t - \Phi_i \boldsymbol{x_t}')\big) \, p(h_t = i|\boldsymbol{z}, \theta_k).\end{aligned}$$

With the help of hidden path weighted moment matrices

$$M^{[i]} = \sum_{t=p+1}^{T} p(h_t = i|\boldsymbol{z}, \theta_k) \begin{pmatrix} 1 \\ \boldsymbol{z}_{t-p} \\ \vdots \\ \boldsymbol{z}_t \end{pmatrix} \begin{pmatrix} 1 & \boldsymbol{z}'_{t-p} & \dots & \boldsymbol{z}'_t \end{pmatrix}, \tag{3.38}$$

with $1 \le i \le n$, and using the notation introduced in § 3.3.4, this can be reformulated as

$$-\frac{1}{2} \sum_{i=1}^{n} \Big(m^{[i]} \log(|2\pi R_i|) + \operatorname{tr}((M_{22}^{[i]} - M_{21}^{[i]} \Phi_i' - \Phi_i M_{12}^{[i]} + \Phi_i M_{22}^{[i]} \Phi_i') R_i^{-1})\Big) .$$

This matches a sum over log-likelihood functions of the VAR(p) model, cf. § 3.35, one for each hidden state, which differ by the weightings of the moment matrices. Therefore maximising this sum term by term yields the already introduced MLEs for VAR(p) models applied to the weighted moment matrices.

Finally, we need the probabilities of the hidden path, i.e. $p(h_t = i|\boldsymbol{z}, \theta_k)$ and $p(h_t = i, h_{t+1} = j|\boldsymbol{z}, \theta_k)$ for $1 \le i, j \le n$ and $p + 1 \le t \le T$, to employ the EM algorithm. Again the forward backward variables introduced in § 3.2.3 can be used. Note that, as the only difference to the case with Gaussian output variables, the recursion for computation of the forward variables starts with initialisation of

$$\alpha_{p+1}(i) = \pi_i f_i(\boldsymbol{z}_{p+1}|\boldsymbol{z}_p, \dots, \boldsymbol{z}_1),$$

while the recursion to compute the backward variables ends with

$$\beta_{p+1}(i) = \sum_{j=1}^{n} p_{ij} f_j(\boldsymbol{z}_{p+2}|\boldsymbol{z}_{p+1}, \dots, \boldsymbol{z}_2)\beta_{p+2}(j),$$

for $1 \le i \le n$. Finally also note that the computation of the Viterbi path does not change except for obvious adjustments like the length of the Viterbi path, which is $T - p$.

To summarise we give an schematic overview over the algorithm in Algorithm 1. Note that implementing the algorithm is more involved than it is shown here, as it requires careful scaling and rescaling of the forward and backward variables to prevent underflow errors caused by quantities close to zero, cf. [96, Sec. V].

Algorithm 1: Optimisation of the HMM-VAR model via the EM-algorithm

```
Parameter: $p$ (assumed order of the VAR models)
           $n$ (number of hidden states)
           $\epsilon$ (a threshold value as stop criterion)
Input    : A time series $\boldsymbol{z} = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \dots \boldsymbol{z}_T\}$ and initial
           parameters $\theta = (\Phi_1, R_1, \dots, \Phi_n, R_n, P, \boldsymbol{\pi})$.
Compute the forward variables $\alpha_t(i)$, $1 \leq i \leq n,\ p+1 \leq t \leq T$
according to (3.15).
Compute the backward variables $\beta_t(i)$, $1 \leq i \leq n,\ p+1 \leq t \leq T$
according to (3.16).
Evaluate the likelihood function $p(\boldsymbol{z}|\theta)$ according to (3.17).
$l_{new} \leftarrow \log(p(\boldsymbol{z}|\theta))$
$l_{old} \leftarrow l_{new} + \epsilon$
while $|l_{old} - l_{new}| \geq \epsilon$ do
    Compute hidden path probabilities according to (3.18) and (3.19).
    Update $P$ and $\boldsymbol{\pi}$ according to (3.11) and (3.12).
    for $i \leftarrow 1$ to $n$ do
        Set up moment matrix $M^{[i]}$ according to (3.38).
        Use (3.36) to obtain updated MLE's $R_i$ and $\Phi_i$.
    Update forward variables according to (3.15).
    Update backward variables according to (3.16).
    $l_{old} \leftarrow l_{new}$
    $l_{new} \leftarrow p(\boldsymbol{z}|\theta)$
Output: The updated parameter vector
        $\theta = (\Phi_1, R_1, \dots, \Phi_n, R_n, P, \boldsymbol{\pi})$.
```

Example

We give a simple illustrative example for the above described procedure. Suppose three VAR(1) parameter sets, cf. (3.37),

$$\begin{aligned}
\boldsymbol{\nu}_1 &= \begin{pmatrix} 0 \\ 0 \end{pmatrix}, & R_1 &= \begin{pmatrix} 0.02 & 0.013 \\ 0.013 & 0.02 \end{pmatrix}, & A_1^{[1]} &= \begin{pmatrix} 0.99 & 0.011 \\ 0.011 & 0.88 \end{pmatrix}, \\
\boldsymbol{\nu}_2 &= \begin{pmatrix} 0.02 \\ 0 \end{pmatrix}, & R_2 &= \begin{pmatrix} 0.01 & 0.005 \\ 0.005 & 0.01 \end{pmatrix}, & A_1^{[2]} &= \begin{pmatrix} 0.99 & 0 \\ -0.022 & 0.44 \end{pmatrix}, \\
\boldsymbol{\nu}_3 &= \begin{pmatrix} 0.02 \\ 0.01 \end{pmatrix}, & R_3 &= \begin{pmatrix} 0.005 & 0.001 \\ 0.001 & 0.005 \end{pmatrix}, & A_1^{[3]} &= \begin{pmatrix} 0.99 & 0.055 \\ -0.055 & 0.99 \end{pmatrix}.
\end{aligned}$$

Note that the specification of parameter sets $(\boldsymbol{\nu}_i, A_1^{[i]}, R_i)$ is equivalent to the specification of parameter sets $(\boldsymbol{\mu}_i, \exp(\tau F_i), R_i)$ with $\exp(\tau F_i) := A_1^{[i]}$ and $\boldsymbol{\mu}_i = (I - \exp(\tau F_i))^{-1}\boldsymbol{\nu}_i$, cf. (3.29), or, as used in the HMM-VAR framework (Φ_i, R_i) with $\Phi_i = (\boldsymbol{\nu}_i \quad A_1^{[i]})$. Furthermore, assume a Markov switching process which is specified by the following transition matrix

$$P = \begin{pmatrix} 0.997 & 0.0015 & 0.0015 \\ 0.0015 & 0.997 & 0.0015 \\ 0.0015 & 0.0015 & 0.997 \end{pmatrix}.$$

First, we obtained a "hidden" path $\boldsymbol{h} = \{h_1, \ldots, h_{3500}\}$ by setting $h_1 = 1$ and get a realisation of a Markov chain according to the given transition matrix. Second, an observation trajectory was generated by setting $\boldsymbol{z}_1 = (0, 0)$ and

$$\boldsymbol{z}_{t+1} = \nu_{h_t} + A_1^{[h_t]} + \mathcal{N}(\mathbf{0}, R_{h_t}), \quad t = 1, \ldots, 3700.$$

This trajectory was analysed within the HMM-VAR framework with memory $p = 0$, i.e. as an HMM-Gauss process, and with memory $p = 1$, i.e. as an HMM-SDE process. In both cases we presumed the correct number of three hidden states. The initial conditions for the EM-algorithm were chosen by a random allocation of the data points to the three sets. After termination of the EM-algorithm, the Viterbi algorithm was employed to get the most likely sequence of hidden states. Fig. 3.3 shows that the Viterbi path obtained from the HMM-VAR(1) procedure nearly perfectly returns the true hidden path, while for $p = 0$ the result looks reasonable at first sight but is totally wrong as no dynamical information between successive data points is included in the model.

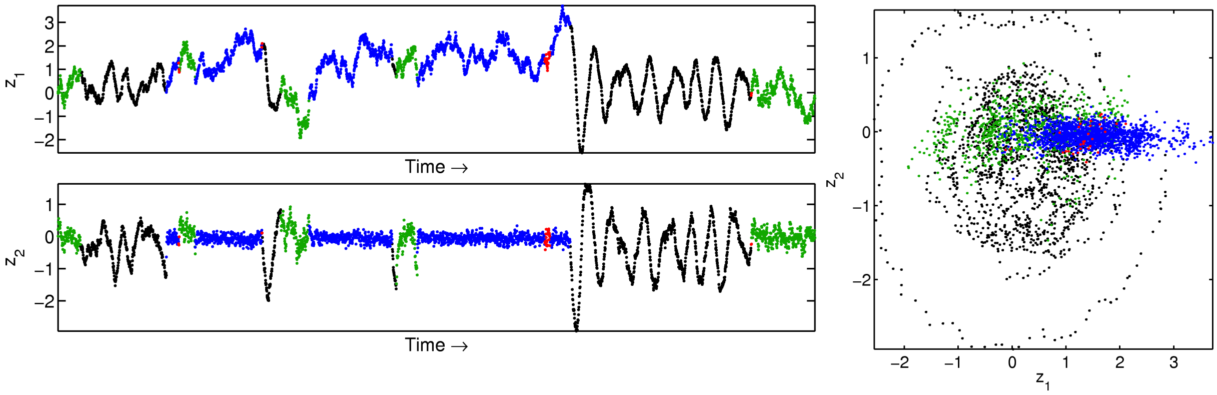
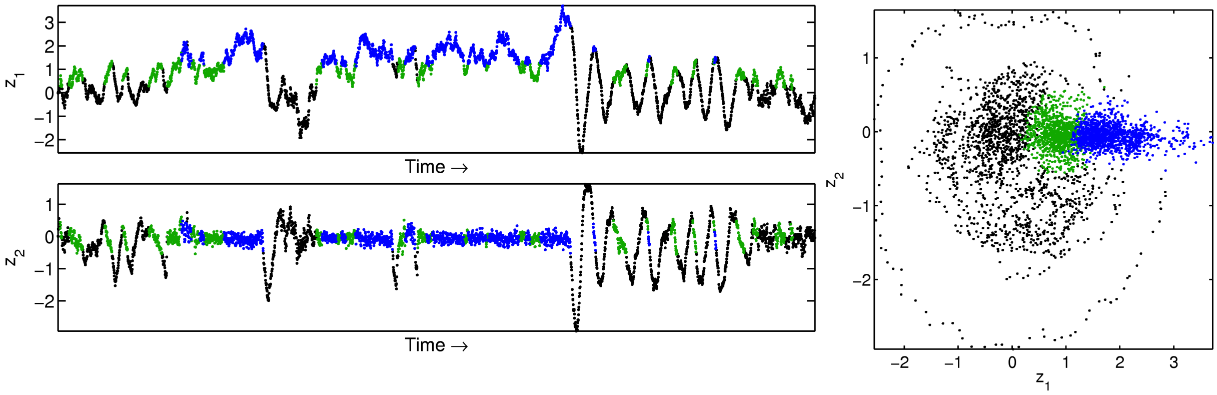


Figure 3.3: *Top:* A two dimensional trajectory (z_1, z_2) generated by a 3-state VAR(1) model plotted against time (left) and in phase plane (right). Colours are chosen according to the Viterbi path obtained from HMM-VAR(1) analysis. Wrong allocations are marked as red dots (32 wrong allocations). *Bottom:* The same trajectory as in the top panel but this time colour-coded according to the Viterbi path obtained from HMM-VAR(0) analysis. Wrong allocations are not marked. Note that even if the result plotted in phase plane looks "more reasonable" as with the VAR(1) analysis it is totally wrong (1701 wrong allocations).



3.4 Practical Considerations - Applications

In this section we show some examples of applications of the introduced methods on “real data” within the context of molecular dynamics. The purpose of these examples is to illustrate the usage of these methods in complex scenarios, where there is no reference solution to the problem anymore. But before passing to the examples, we collect some “recipes” one may have to use if handling with “real” data. More precisely, we will comment on how to choose the right VAR order, how to cope with circular data and how to handle high dimensional data. None of these recipes claims to be the right one, but have turned out to be feasible strategies.

3.4.1 Estimation of the VAR order

In the previous sections we always assumed a fixed and given order of the VAR models to be fitted on a time series. In some applications there might be some knowledge about the memory depth τ_m, i.e. the time span after which the memory effects of the analysed process are negligible. In this case the order of the VAR process can be chosen as the smallest integer p such that $p\tau > \tau_m$, where τ is the time between successive data points. But in general the order of the model, like other model parameters, has to be estimated from the given time series. There are several approaches to this problem, none of them can claim an optimal solution so far.

One approach is to fix a maximal order p_m and determine the order via a sequence of hypothesis tests

$$\begin{array}{lllll} H_0^1 : A_{p_m} & = \mathit{0} \text{ vs. } & H_1^1 : A_{p_m} & \neq \mathit{0} & \\ H_0^2 : A_{p_m-\mathit{1}} & = \mathit{0} \text{ vs. } & H_1^2 : A_{p_m-\mathit{1}} & \neq \mathit{0} | A_{p_m} = \mathit{0} & \\ \quad\vdots & & \quad\vdots & \quad\vdots & \\ H_0^m : A_{\mathit{1}} & = \mathit{0} \text{ vs. } & H_1^m : A_{\mathit{1}} & \neq \mathit{0} | A_{p_m} = \cdots = A_{\mathit{2}} = \mathit{0}. & \end{array}$$

In fact the so-called likelihood ratio statistic can be used as a test statistic for these hypothesis test. If $f^{(k)}$, $1 \leq k \leq m$, denotes the likelihood function for an $p_m - k + 1$ order VAR model, $\hat{\theta}$ the unrestricted MLE and $\hat{\theta}_r$ the MLE with the restriction that $A_{p_m-k+\mathit{1}} = \mathit{0}$, then the likelihood ratio statistic is defined as

$$\lambda_{LR}^{(k)} = 2(\log(f^{(k)}(\hat{\theta}|\boldsymbol{z})) - \log(f^{(k)}(\hat{\theta}_r|\boldsymbol{z}))).$$

As a consequence of the asymptotic normality of maximum likelihood estimators under very general conditions [23,24], the asymptotic distribution, i.e. the limit distribution for a growing number of data points, of $\lambda_{LR}^{(k)}$ under the H_0 hypothesis can be derived and is, in fact, a χ^2-distribution, cf. the section about likelihood ratio tests in § 4.1.2.

Nevertheless, such a sequence of hypothesis tests is difficult to handle as, besides the fact that distribution is only known asymptotically, it is not clear which significance levels one should choose to obtain an appropriate overall significance level [73, Sec. 4.2.3]. A more fundamental criticism is raised by Akaike [1] who points out that in an applied context such models like VAR(p) are only approximative and therefore one should not rely on the H_0 assumption but should introduce some loss function which actually becomes the basis for decision making and is more founded than a significance level based on rarely true assumptions.

An example of such loss function is the expected mean squared forecast error leading to the so-called FPE criteria [73, Sec. 4.3], which consist in choosing the order p which minimises

$$\mathrm{FPE}(p) := \left(\frac{T + pd + 1}{T - pd - 1}\right)^d |\hat{R}(p)|,$$

where $\hat{R}(p)$ denotes the MLE of R in a VAR(p) model. Another prominent example is the expected Kullback-Leibler distance between $p(\boldsymbol{z}|\theta^*)$ and $p(\boldsymbol{z}|\hat{\theta})$ where θ^* is the true parameter and $\hat{\theta}$ the MLE leading to Akaikes criterion [1]

$$\mathrm{AIC}(p) := \log(|\hat{R}(p)|) + \frac{2pd^2}{T}.$$

However, both criteria turn out to be not consistent. A consistent order estimator was derived by Schwarz [110]

$$\mathrm{SC}(p) := \log(|\hat{R}(p)|) + \frac{\log(T)2pd^2}{T}. \tag{3.39}$$

Note that the consistency of the Schwarz criterion does not automatically make it superior to the FPE or AIC criterion in finite sample situations. But our experience indicates that it works better in application as the larger penalty term for the number of parameters prevents choosing an arbitrary large p if the data does not fit well to the VAR model. Therefore we always use the Schwarz criterion in the following examples whenever the order of the model needs to be estimated.

3.4.2 Circular Data

In § 2.2.1 we have seen that for the analysis of peptide simulations the dihedral angles of the backbone are reasonable observables since conformational changes are likely to show up here and problems with rotational and translational degrees of freedom are avoided. On the other hand, analysing angle time series introduces a new problem since the assumed output distributions in the HMM are not periodic and therefore these models are not suitable for periodic data.

A possible resort would be to explicitly formulate a periodic model by replacing the normal distribution with its periodic counterpart: the von Mises distribution [74]. The von Mises distribution $M(\mu, \kappa)$ in one dimension is given by the following probability distribution function depending on the two parameters $\mu \in [0, 2\pi]$, the mean direction, and $\kappa > 0$, the concentration parameter:

$$f(z|\mu,\kappa) = \frac{e^{\kappa \cos(z-\mu)}}{2\pi I_0(\kappa)}, \qquad 0 \leq z < 2\pi,$$

where $I_0(\kappa)$ is the modified Bessel function of the first kind and order zero. The von Mises distribution is unimodal i.e., single-peaked, and symmetrical about the mean direction. The maximum of the pdf, the so-called mode, is at the mean, while the minimum of the pdf (anti-mode) is located at $\mu + \pi$. The larger the value of the concentration parameter κ, the more pronounced the concentration of the distributed data around the mode. In contrast, for $\kappa \to 0$, the von Mises distribution becomes uniformly distributed. Indeed it is shown in [36] that the von Mises distribution can be employed successfully in the analysis of backbone dihedral time series of peptides. However, there is no closed form estimator of the concentration parameter κ and neither an obvious generalisation to higher dimensions, except of regarding all dimensions as uncorrelated, nor to more memory in the system. Therefore we choose a different strategy to cope with periodic data, namely to transform it to essentially non-periodic data.

A very simple but also very effective strategy is to shift the data to remove periodicity, which will work in most cases as the torsion angles are in general not freely rotating. The shifting of the data can be automatised by discretising the angle domain in boxes and determine a borderline with minimal number of transitions across. Additionally, we have to exclude from the statistics transitions of data points that cross the periodic boundary, cf. Fig. 3.4. This can easily done by marking large distances between subsequent (shifted) data points, i.e.

$$\begin{aligned} a_1 &= 1 \\ a_t &= \begin{cases} 1 & \text{if } \|\boldsymbol{z}_{t-1} - \boldsymbol{z}_t\|_\infty > c \; , \\ 0 & \text{else,} \end{cases} \quad 2 \leq t \leq T. \end{aligned}$$

and adjust the statistics by modifying the moment matrices

$$M = \sum_{t=p+1}^{T} \left(\prod_{j=t-p+1}^{t} a_j \right) \begin{pmatrix} 1 \\ \boldsymbol{z}_{t-p} \\ \vdots \\ \boldsymbol{z}_t \end{pmatrix} \begin{pmatrix} 1 & \boldsymbol{z}'_{t-p} & \dots & \boldsymbol{z}'_t \end{pmatrix}.$$

3.4.3 Viterbi Clustering

In large biomolecular systems the dimension d of the observation sequence may be large, which constitutes not only a computational burden but also a statistical difficulty as the number of free parameters of the output distributions

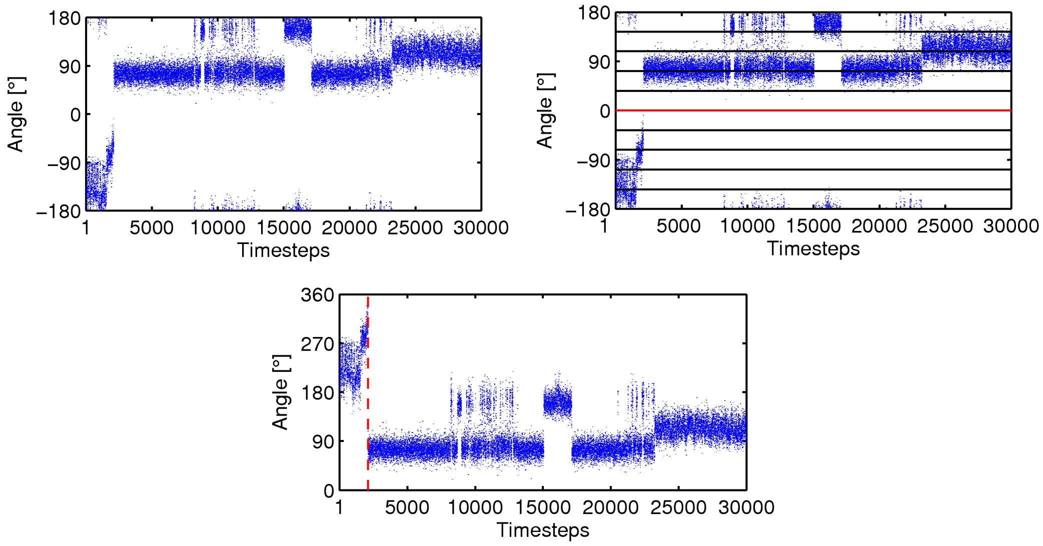


Figure 3.4: *Top left:* a time series exhibiting periodicity. *Top right:* the angular domain is discretised and the borderline with the fewest transitions across is determined. *Bottom:* shifting the data, so that the determined borderline becomes the boundary, makes the time series effectively non-periodic. Single transitions over the boundary (dotted line), i.e. large jumps, are just excluded.

will typically increase with d faster than linearly, e.g. a full d-dimensional Gaussian distribution has $\frac{d(d+3)}{2}$ free parameters. Besides the increase of the statistical error for increasing dimension of the parameter space, the likelihood maximisation via the EM algorithm will converge more and more slowly if the dimension becomes too large.

A resort is the decomposition of a high dimensional observation space, say V, into low-dimensional subspaces $V = V^{(1)} \cup \cdots \cup V^{(k)}$, i.e. V could be the state space of all torsion angles of the system under consideration, and $V^{(j)}$ the subspace of a single torsion angle. By choice of $V^{(j)}$, $j = 1, \ldots, k$, and projection onto each one of them, k low-dimensional time series $\boldsymbol{z}^{(j)} = (\boldsymbol{z}_1^{(j)}, \ldots, \boldsymbol{z}_T^{(j)})$ are obtained. Each of these low-dimensional time series can be separately analysed by means of the above HMM-VAR procedure, presuming a not too small number of hidden states, and then a Viterbi path can be computed. Each of these Viterbi paths can be aggregated by the PCCA method. This results in k aggregated Viterbi paths $\boldsymbol{h}^{(j)} = (h_1^{(j)}, h_2^{(j)}, \ldots, h_T^{(j)})$ that represent the conformational dynamics as detected from the information contained in a single projection of the full observated time series. Note that by using HMMs we are able to discretise the low-dimensional projections even if the metastable sets are overlapping in the projected space, as the output distributions of the HMM are allowed to overlap. All these single aggregated Viterbi paths can be combined into a *global* Viterbi path $\boldsymbol{h}$ via superposition. The combined global

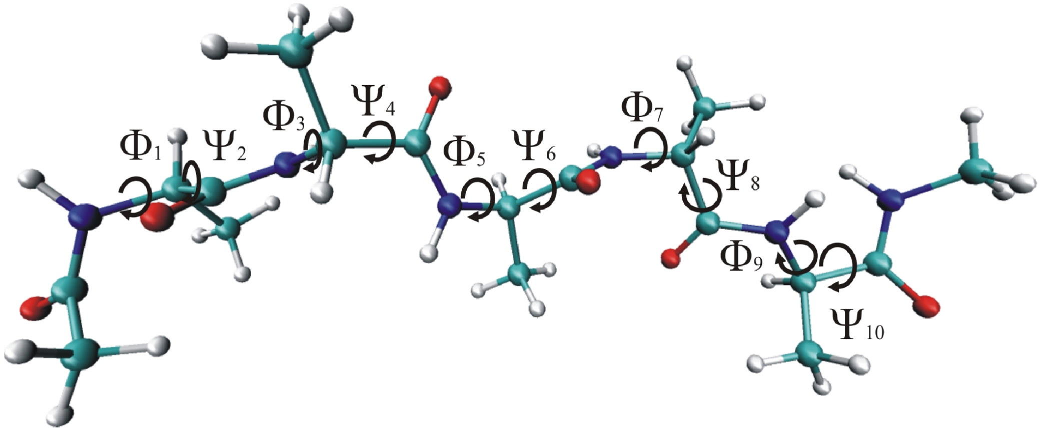


Figure 3.5: The penta-alanine peptide in ball-and-stick representation. The ten peptide angles determining the secondary structure are marked by $\Phi_1, \Psi_2, \ldots, \Phi_9, \Psi_{10}$.

Viterbi path has a finite number of states such that we can directly compute the associated transition matrix and again identify metastable sets by means of PCCA. Based on these metastable sets, the global Viterbi path is aggregated into a clustered global Viterbi path whose resulting discrete states are finally interpreted as the global conformation states of the original full-dimensional time series. An example of this strategy will be given in the next section. Note that the combination of HMM and PCCA approaches resolves one of the major problems of the HMM approach, which is that the number of hidden states is an input parameter.

3.4.4 Example: Analysis of Penta-alanine

In the following we demonstrate the proposed analysis with an application on a data set obtained from an MD simulation of a penta-alanine, i.e. a small peptide consisting of five alanine units (residuals). The analysis we show here is a variation of the analysis published in [75], note that although the algorithmic procedure presented here differs slightly ,the overall results stay the same.

Our analysis is based on a time series of the 10 backbone torsion angles of penta-alanine, see Fig. 3.5, extracted from a long time simulation which is courtesy of Gerhard Stock (Frankfurt) and has been discussed in [83]. The simulation was done in explicit water using a thermostat of 300 Kelvin over an interval of 100 nanoseconds, while the coordinates were written out every 0.1 picosecond, resulting in a 10 dimensional time series of 1000000 data points.

The first step of a typical analysis consists in the identification of meaningful subunits to avoid a blow up in parameter space by trying to fit a model which is too large. In this case the Φ/Ψ angle combinations belonging to the different residuals is a natural choice. Therefore we conducted the HMM-VAR algorithm

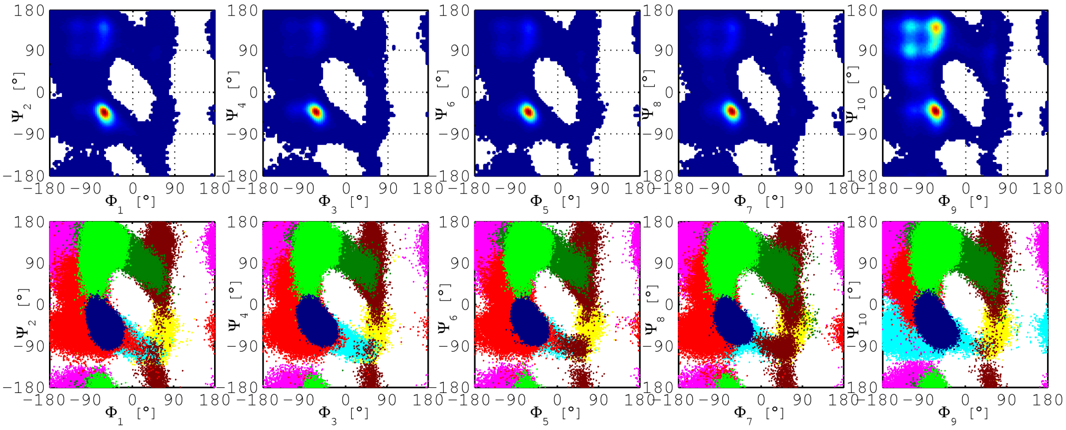


Figure 3.6: *Top:* Plot of the empirical probability densities for each of the Φ/Ψ pairs extracted by projection of the time series onto the Ramachandran planes (red/blue corresponds to high/low probability). Note that there is a distinct peak region in all five plots. *Bottom:* The results of the HMM-VAR analysis applied to each of the Φ/Ψ pairs separately, the data points are coloured according to their allocation to the 8 hidden states.

on each angle pair independently. As preparation, the time series was shifted to minimise periodicity and large distances between subsequent data points were marked as described in § 3.4.2. Afterwards, the memory for each angle pair was estimated on short trajectory pieces with the Schwarz criterion (3.39), resulting in all cases in a VAR order of $p = 8$. The number of hidden states was initially guessed with 8 hidden states per pair. An important step is the choice of initial parameters for the EM algorithm, since it is a local optimiser We have computed an initial assignment of the data points to the hidden states by employing the PCCA approach, i.e. we discretised the two dimensional space for each torsion angle pair in 900 boxes, set up the transition matrix via counting and computed a clustering of these boxes in 8 sets. The resulting assignment of the data points to one of the 8 sets was used to compute an initial estimate for the parameters of the HMM. Of course, one could specify (several) initial conditions at random, but using PCCA turned out to be quite effective in our applications. The result of the HMM-VAR algorithm on each of the Φ/Ψ-pairs is shown in Fig. 3.6.

It can be seen from the figure that the resulting allocation of data points to the hidden states is remarkably similar for each angle pair. Furthermore, we see that even if an inspection by eye would divide each Ramachandran plane in a preferred region and a diffusive rest, the diffusive part can still be subdivided meaningfully.

By means of PCCA we can try to cluster the resulting sets in each Ramachandran plane as described in § 3.4.3, i.e. by setting up transition matrices from the computed (already discrete) Viterbi paths for each projection. Again, the different projections behave similar in that there is a reasonable

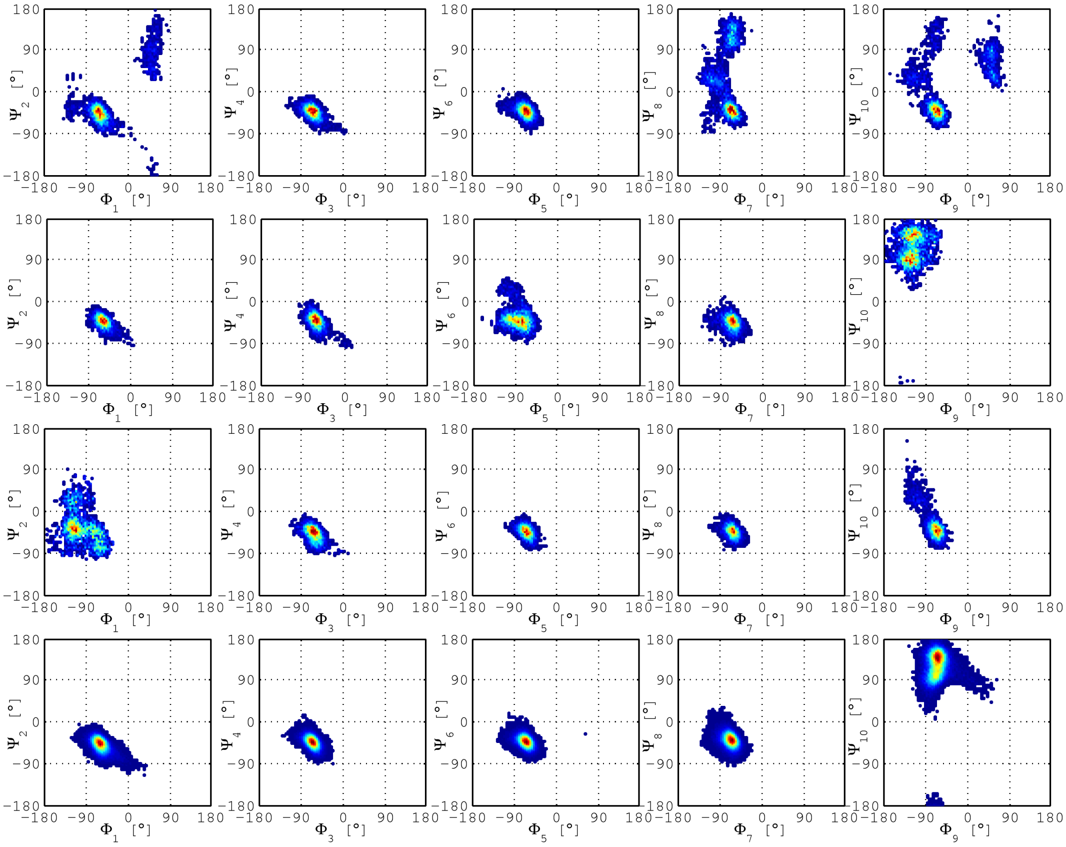


Figure 3.7: From top to bottom row the projections of the empirical densities of the four most populated global-states of penta-alanine on the Ramachandran planes are depicted (Populations are: 12 %, 7 %, 5% and 2% of all data points). Note that they differ mostly in the Φ_1, Ψ_2 and Φ_9, Ψ_{10} planes.

gap within the spectrum of the transition matrices between the sixth and the seventh eigenvalue for each of them. Therefore the eigenvectors are used to reduce each Viterbi path from 8 to 6 states. A global Viterbi path is obtained by superposition of all locally clustered Viterbi paths resulting in 3108 occupied global states (out of $6^5 = 7776$ possible states). It is possible, though in this example not really necessary as the number of global states is feasible, to reduce the number of states further by merging states which are very seldom visited with (dynamically) neighboring states. Merging all states with less than 0.1% of the data points assigned leads to 279 states. In Fig. 3.7 the projected densities of the four most populated global states are depicted.

Again we can use the metastability analysis to reduce the number of global states even further. Therefore it is instructive to compare the eigenvalues of the transition matrices obtained from the global Viterbi path for different lag times τ. That is, we do not count transitions on the basis of a time lag of 0.1 picosecond which means to count transitions from one instance of the time series to the next, but count transitions with respect to a time lag of,

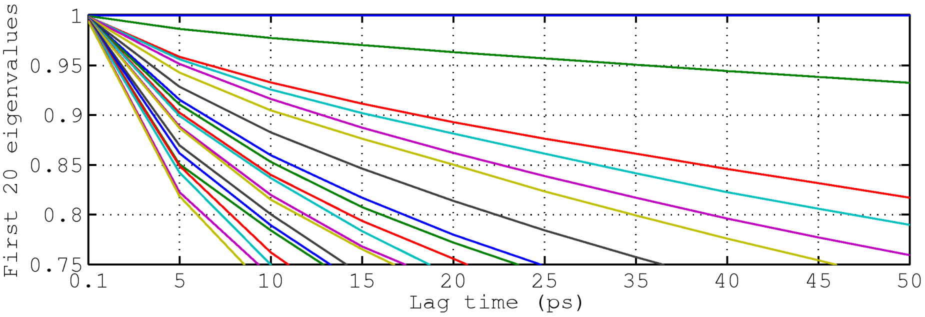


Figure 3.8: The 20 dominant eigenvalues of the transitions matrix obtained from the global Viterbi path wrt. different time lags are shown.

say, 1 picosecond which is from one instance to the instance 10 steps further, i.e. between every tenth step, in the global Viterbi path. In Fig. 3.8 the dominant eigenvalues of the transition matrices with respect to different lag times are shown, we clearly see that across all different time lags there are two dominant eigenvalues after which we find a gap.

Using PCCA once more to cluster the global states in two metastable states and plotting the projected densities of these two dominant global states reveals an intriguing observation. In Fig. 3.9 it can be seen, that one of these two states is fixed to a specific region in the Ramachandran planes while the other is not clearly localised. The specific region belongs in fact to the α-helix secondary structure of peptides, i.e. our global identified conformation is an α-helical one, while the other one has no clear secondary structure, i.e. it is unfolded, which is no surprise since the peptide is too short to exhibit other stable secondary motifs. Notice that besides the identification of a global secondary motif from the data we also have a dynamical (Markov) model which allows us to state transition probabilities between the folded and the unfolded state.

We close this example by remaining that, instead of doing this sort of bottom to top analysis, we could have started with the identification of the two global conformations right from the beginning by fitting a VAR model to the whole 10-dimensional time series. In this case, our order estimation suggest a VAR(1) model for the full dimensional time series. Assuming 2 hidden states and taking an initial random allocation of the data points to the two hidden states the HMM-VAR procedure yields very much the same result as we obtained at last in our previous analysis, see Fig. 3.10. Still the bottom to top analysis has its own right as it allows us to obtain a detailed (dynamical) picture of what happens on the subunits and allows to control the complexity reduction step by step.

Further examples of the proposed analysis can be found in [78] where it is shown that metastability analysis can even identify micro solvation patterns for a small solvated molecule and in [36] where a circular output distribution for the HMM is used.

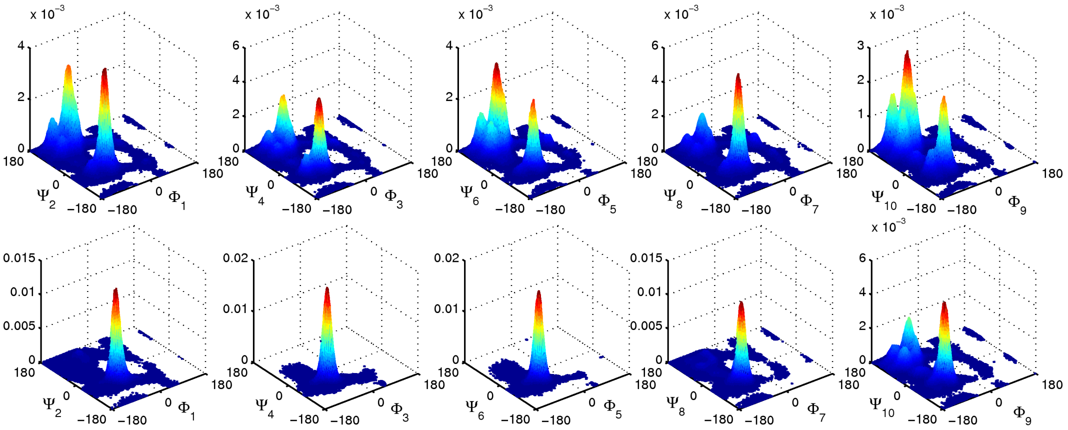


Figure 3.9: Clustering the global Viterbi path into 2 conformations reveals an unfolded conformation (top) and a folded one (bottom), namely an α-helical one.

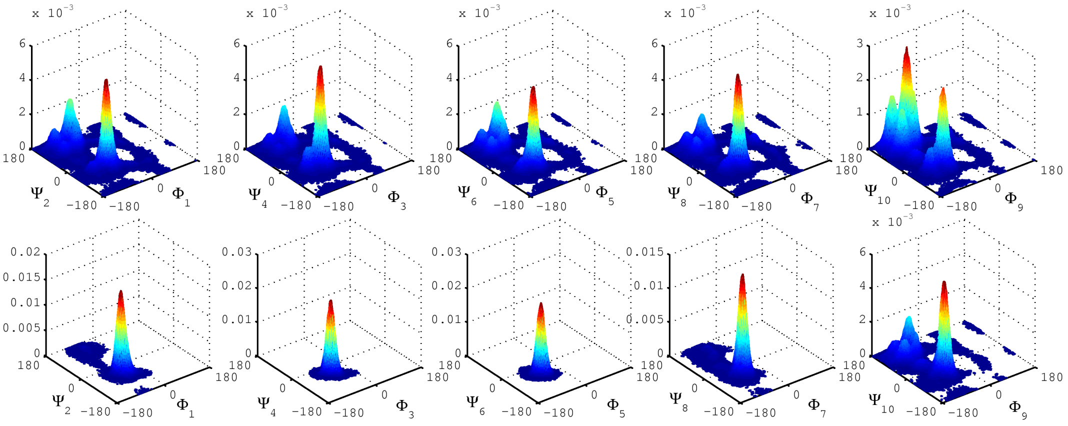


Figure 3.10: Fitting a 10 dimensional HMM-VAR(1) model with two hidden states and projecting the data points belonging to each of these states according to the Viterbi path on the Ramachandran planes reveals the same structure as our bottom to top analysis presented before.

4 On-line change point detection

So far the proposed approaches to analyse time series data are post processing algorithms, i.e. it is assumed that a time series is given, which contains all relevant information to set up a meaningful reduced model. This time series is analysed en bloc and the outcome is a fragmentation of the time series in different dynamical phases, each approximated by a linear model. In this chapter we discuss a different setting. Assume we observe a process in time which can be approximated by some (unknown) linear model, can we detect a change in the dynamical regime, i.e. a switch to another (unknown) linear model as fast as possible, i.e. on-line? A solution to this question could be used in an algorithmic setting where an action has to be taken when the dynamical phase of an observed system changes. We will give an example for such situation in § 5, where we will use the results of this chapter to determine switching rates between molecular conformations from parallel simulated MD trajectories. Of course the employment of such algorithm is not restricted to on-line applications, such algorithm could also be used to parameterise models as introduced in § 3 in a linear fashion, i.e. by scanning a given time series once from beginning to end, while avoiding the complex likelihood optimisation problem, which otherwise has to be tackled via usage of the EM algorithm. Most of the results of this chapter were published in [76].

4.1 Change Point Detection

The problem to be considered in the subsequent sections is the following. Assume a given sequence of observations

$$Z = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \dots \boldsymbol{z}_T\}, \boldsymbol{z}_i \in \mathbb{R}^d$$

for which a VAR(p) model is presumed as the generating mechanism. Our aim is to decide if Z was generated by a single VAR(p) model or if there was a parameterisation change at some time t, $t_1 \leq t \leq t_2$, i.e.

$$Z_1 = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \dots \boldsymbol{z}_t\}$$

was generated with parameters Φ_1, R_1 and

$$Z_2 = \{\boldsymbol{z}_{t+1}, \boldsymbol{z}_{t+2} \dots \boldsymbol{z}_T\}$$

with parameters Φ_1 and R_1. We will call such time t a *change point* in the following. The function of the window defined by t_1 and t_2 will become apparent later. Note that by solving the stated problem an on-line algorithm can be constructed easily which works on incoming data packages.

4.1.1 The Model Selection Problem

Change point detection problems are more challenging than estimation problems as they are essentially model selection problems. To clarify the underlying problem we simplify the task formulated above for the moment and assume that we have two arbitrary time series fragments Z_1 and Z_2. The question is: are they both generated from the same VAR(p) model, or not? Obviously, the maximum likelihood approach does not work anymore as we have for the log-likelihood function

$$l(\Phi, R|Z_1, Z_2) = l(\Phi, R|Z_1) + l(\Phi, R|Z_2)$$

and therefore

$$\max_{\Phi,R} l(\Phi, R|Z_1, Z_2) \leq \max_{\Phi_1,R_1} l(\Phi_1, R_1|Z_1) + \max_{\Phi_2,R_2} l(\Phi_2, R_2|Z_2)$$

always holds, i.e. making a model more complex will always increase the likelihood function. In fact we have encountered the same problem in the order estimation of linear models in § 3.4.1 since choosing a higher order, i.e. introducing more parameters, always increases the maximum of the likelihood function. A common resort is the introduction of a penalty term on the number of parameters, as in the Schwarz estimator (3.39) for the model order. However, the choice of the penalty term is somewhat arbitrary and can, at the best, be justified in an asymptotic sense. A natural alternative approach is the formulation of a hypothesis test, cf. the next section, but the drawback is that distributions under the H_0 hypothesis are in general not known, respectively only asymptotically known.

A different perspective would be to ask how well Z_1 and Z_2 fit together, i.e. to ask if the model estimated by Z_1 can be used to explain the dynamical behaviour in Z_2, or to ask if the induced parameter distributions are similar enough. Obviously one then has to answer what close or similar enough means. In the next section we give an overview of some of the approaches to the change point problem before we adopt a Bayesian approach to our problem.

4.1.2 Approaches to the Change Point Problem

The subsequent approaches are not restricted to change point detection in VAR(p) models and most of them can be formulated in a more general way. However, to avoid the introduction of too much new notation, and as the philosophy behind these approaches can still be illustrated if restricted to VAR(p) models, we stick mostly to a more restricted presentation.

Likelihood partition approaches. As pointed out above one can not employ a maximum likelihood formalism to decide if there is a change point or not. But if the existence of a change point is known, a natural choice would be to choose the change point $\hat{\tau}$ such that it maximises

$$\max_{\Phi_1, R_1} l(\Phi_1, R_1 | Z_1(\hat{\tau})) + \max_{\Phi_2, R_2} l(\Phi_2, R_2 | Z_2(\hat{\tau})),$$

with $Z_1(\hat{\tau}) = \{\boldsymbol{z}_1, \dots \boldsymbol{z}_{\hat{\tau}}\}$ and $Z_2(\hat{\tau}) = \{\boldsymbol{z}_{\hat{\tau}+1}, \dots \boldsymbol{z}_T\}$. Appealingly this idea can be generalised to the identification of $(N-1)$ change points by choosing $\hat{\tau}_0 = 0 < \hat{\tau}_1 < \dots < \hat{\tau}_{N-1} < T = \hat{\tau}_N$ such that

$$\sum_{k=1}^{N} \max_{\Phi_k, R_k} l(\Phi_k, R_k | Z(\hat{\tau}_{k-1}, \hat{\tau}_k)), \tag{4.1}$$

with $Z(\hat{\tau}_{k-1}, \hat{\tau}_k)) := \{\boldsymbol{z}_{\hat{\tau}_{k-1}+1}, \dots \boldsymbol{z}_{\hat{\tau}_k}\}$, is minimised [64]. As mentioned above, to use the likelihood function to decide on the existence of change points one has to add to (4.1) a penalty term $c(N)$ which increases with the number of change points, i.e. with the number of newly introduced parameters, and maximise over both, the number of change points and the locations of the change points. In fact, Lavielle [69,70] proved under quite general assumptions and for a family of penalty terms that asymptotically the correct number and location of change points is identified by the likelihood partition approach and that the convergence rate of the estimated change points to the true change points is optimal. However, "asymptotical" in this case means not only for the limit of an increasing length of the time series but also *under the condition that the distance between any two change points scales with the increasing length of the time series.* Which makes the asymptotic justification, in practice of not much use anyway, questionable for multiple change point scenarios.

Likelihood ratio test. Assume a likelihood function $L(\theta|Z)$ dependent on some observations Z for an r-dimensional parameter vector $\theta = (\theta_1, \theta_2)$, which can be partitioned into θ_1 and θ_2 which are s and t-dimensional respectively such that $r = s + t$. Furthermore, suppose we want to test the hypotheses

$$H_0 : \theta_1 = \theta_c \text{ against } H_1 : \theta_1 \neq \theta_c,$$

for a given $\theta_c \in \mathbb{R}^s$. An often employed test statistic in this rather general setting is the likelihood ratio statistic, which we have already encountered in the context of VAR order selection in § 3.4.1, which reads in general

$$\lambda_{LR} = -2 \log \left(\frac{\max\limits_{\theta} L(\theta|Z)}{\max\limits_{\theta:\theta_1=\theta_c} L(\theta|Z)} \right).$$

As the fraction is in $[0,1]$, we clearly have $\lambda_{LR} \in [0, \infty]$ and would accept H_0 if λ_{LR} is close to 0. However, what makes this test statistic so useful is that even if its distribution can not obtained analytically, it can be shown under quite general assumptions that under the H_0 hypothesis $-2\log(\lambda)$ is asymptotically, i.e. for $T \to \infty$, χ^2 distributed with s, the number of constraints, degrees of freedom [65, 102, 127]. This is a consequence of the fact that, under some regularity assumptions, MLE's are asymptotically normal distributed (even for non i.i.d. data) [24] and that for a θ of length r which is $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ distributed the quadratic form $(\theta - \boldsymbol{\mu})'\Sigma^{-1}(\theta - \boldsymbol{\mu})$ is χ^2 distributed with r degrees of freedom.

Adopted to the change point setting a likelihood ratio test can be used by assuming Z_1 to be generated by a VAR(p) model with parameters (Φ_1, R_1) and Z_2 with parameters (Φ_2, R_2) and test

$$H_0 : (\Phi_1, R_1) - (\Phi_2, R_2) = (0, 0) \text{ against } H_1 : (\Phi_1, R_1) - (\Phi_2, R_2) \neq (0, 0).$$

The likelihood ratio is easily computed as

$$\lambda_{LR} = -2\log\left(\frac{|\hat{R}|^{\frac{T_1+T_2-2p}{2}}}{|\hat{R}_1|^{\frac{T_1-p}{2}}|\hat{R}_2|^{\frac{T_2-p}{2}}}\right),$$

where $\hat{R}$ denotes the MLE of R using all data points, i.e. Z_1 *and* Z_2, while $\hat{R}_1$, resp. $\hat{R}_2$ is the MLE obtained just upon the basis of Z_1, resp. Z_2. As the number of degrees of freedom in a single VAR(p) model equals $d\left(\frac{d+1}{2} + dp + 1\right)$, the distribution of $-2\log(\lambda_{LR})$ will converge against a χ^2 distribution with $d\left(\frac{d+1}{2} + dp + 1\right)$ degrees of freedom as T_1 and T_2 go to infinity. Therefore the quantiles of the χ^2 distribution can be used to obtain a decision criterion with respect to a chosen significance level.

The CUMSUM approach. Another change point detection approach is based upon observation of the residuals, i.e.

$$\boldsymbol{r}_t := \boldsymbol{z}_t - A_0\boldsymbol{\mu} - \sum_{k=1}^{p} A_k \boldsymbol{z}_{t-k}.$$

In our setting we assumed the residuals to be Gaussian distributed, that is $\boldsymbol{r}_t \sim \mathcal{N}(\mathbf{0}, R)$. Under this assumption the distribution of the quadratic form $Q_t := \boldsymbol{r}_t' R^{-1} \boldsymbol{r}_t$ can be used, which is χ^2 distributed with d degrees of freedom, where d is the dimension of $\boldsymbol{r}_t$. An increase in variance or a shift in the mean will give large values of Q_t while a decrease in variance will produce small values. Therefore a two-tailed test can be employed to detect changes [61]. To detect changes which are persistent over a given time interval one can use the statistic

$$\sum_{k=t}^{t+m} Q_k,$$

which is χ^2 distributed with md degrees of freedom. Using more sophisticated change point detection schemes the assumption of Gaussian distributed residuals can be weakened somewhat [5]. The drawback of the CUMSUM approach is that, to detect departures from a given model, obviously the parameters of the actual valid model have to be known. In other words, a given error probability for the detection procedure can only be asymptotically guaranteed as an infinite number of data points is needed to estimate the actual model consistently, i.e. $T_1 \to \infty$. It is remarkable however, that having estimated the model correctly, the validity of the derived residual distribution does not depend on T_2.

All of the above approaches are asymptotically justified as they all rely on the convergence of the MLE's to the true parameters. But if there is only a finite number of data points, the estimated parameters will be flawed with uncertainty, an uncertainty which is not accounted for in the so far reviewed approaches. Taking this uncertainty into account becomes especially important with increasing dimension of the observed time series. As a measure of uncertainty the likelihood induced parameter density

$$p(\theta) \propto L(\theta|Z),$$

is a natural candidate, as long as the likelihood function is normalisable. Using the parameter density introduces in general another kind of asymptotic reasoning, as the density needs to be sampled in most cases. However, we will see in the subsequent chapter that in the case of the linear model this can be done analytically. If one wants to incorporate parameter densities into the analysis a Bayesian approach is a natural choice. Before describing Bayesian model selection and the application to our setting in detail, we close this section with the presentation of an approach to the change point problem which is based just upon comparison of likelihood induced densities.

Density distance measures. Assume a given time series Z_1 and a corresponding induced *prior* parameter density

$$p_1(\theta) \propto L(\theta|Z_1).$$

Furthermore assume another observed time series Z_2 and transform the prior parameter density to a posterior parameter density according to

$$p_2(\theta) \propto L(\theta|Z_1, Z_2).$$

If Z_1 and Z_2 would have been generated by the same model, one would expect the prior and the posterior density to be similar in some sense. Note that they still should be different as the inclusion of more data points should make the parameter density p_2 more focused than p_1. Obviously one needs to define a distance measure between densities and might have to sample the corresponding statistics to employ this approach. A possible choice [99] is the Kullback-Leibler divergence, defined by

$$KL(p_1, p_2) := \int \log\left(\frac{p_1(\theta)}{p_2(\theta)}\right) p_1(\theta) d\theta,$$

as a distance measure. Evaluation of this distance can be done analytically in some cases but still one has to sample its statistics wrt. p_1 to evaluate a qualitative decision criterion.

4.2 The Bayesian Approach

In the previous section we reviewed a variety of different approaches to the change point problem. In this section we start to develop our own approach based on Bayesian techniques of model selection. Our goal is an algorithm which takes parameter uncertainty into account, as we want to apply it to high dimensional systems, and which does not rely on sampling, since we want to use it for an on-line change point detection procedure. The Bayesian approach relies on the fact that the probabilities of different models $H_0, H_1, \ldots, H_n$ given a set of observations Z can be computed by the Bayesian formula

$$\mathbb{P}[H_i|Z] = \frac{\mathbb{P}[Z|H_i]\,\mathbb{P}[H_i]}{\sum_{l=1}^{n} \mathbb{P}[Z|H_l]\,\mathbb{P}[H_l]}.$$

Note that the probabilities of observations given the model are in general easy to compute. However, boon and bane of Bayesian methods is the need for the specification of prior distributions for the parameters. In the best case such prior distributions can be specified by prior knowledge, see [93,94] for an example where environmental studies are used to obtain prior distribution for the parameters of a water level model. But in general we have to code somehow ignorance in these prior distribution. As we will see, the situation in model selection is considerably more complicated than in parameter estimation, since it turns out that the prior distributions have to be proper. Obtaining meaningful proper prior distributions which code ignorance is a matter of intensive research e.g. [8, 17, 25, 26, 39, 40, 63, 89, 97, 113]. We have chosen the fractional Bayes approach of O'Hagan [89] for our setting as it can be employed in an elegant way even for high dimensional systems and leads to expressions which can be calculated analytically such that no sampling procedures need to be involved. However, opposed to sampling based algorithms [20, 32], the algorithm we derive can not handle multiple change points, but in a sequential form, which is not a strong drawback in our applications as we suppose change points to be rare events.

4.2.1 Bayesian Model Selection

To avoid confusion, we restate our change point problem again: given a sequence of observations

$$Z = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \dots \boldsymbol{z}_T\}, \boldsymbol{z}_i \in \mathbb{R}^d$$

for which we presume a VAR(p) model as the data generating mechanism, we want to decide if there is a change in the parameterisation, a so-called change point, within some window from t_1 to t_2. As it will become apparent later, t_1 and t_2 have to be in the range from $(d+1)p+d+2$ to $T-(d+1)p-d-2$.

Thus we have $n := t_2 - t_1 + 1$ candidate change points giving rise to $n+1$ models $H_i, 0 \le i \le n$, where

$$H_i := \begin{cases} Z \text{ is generated by only one VAR}(p) \text{ process,} & \text{for } i = 0. \\ Z_1 = \{\boldsymbol{z}_1, \dots, \boldsymbol{z}_{t_1+i-1}\} \text{ and } Z_2 = \{\boldsymbol{z}_{t_1+i}, \dots, \boldsymbol{z}_T\} & \\ \text{are generated by distinct VAR}(p) \text{ processes.} & \text{for } 1 \le i \le n. \end{cases}$$

Note that alternatively, one could define the segments Z_1 and Z_2 overlapping, so that the last p points of Z_1 are used as initial conditions for Z_2, as long as Z_1 and Z_2 are directly subsequent. However, this choice does not affect any of the following considerations.

The probability of each model given the observations Z can be computed via the Bayes formula

$$\mathbb{P}[H_i|Z] = \frac{\mathbb{P}[Z|H_i]\,\mathbb{P}[H_i]}{\sum_{j=0}^{n} \mathbb{P}[Z|H_j]\,\mathbb{P}[H_j]}, \tag{4.2}$$

where we have for $i = 0$,

$$\mathbb{P}[Z|H_0] = \int p(Z|\Phi_1, R_1)\pi_1(\Phi_1, R_1) d\Phi_1 dR_1,$$

and for $i \ge 1$,

$$\mathbb{P}[Z|H_i] = \int p(Z_1|\Phi_1, R_1)\pi_1(\Phi_1, R_1) p(Z_2|\Phi_2, R_2)\pi_2(\Phi_2, R_2) d\Phi_1 dR_1 d\Phi_2 dR_2$$

with prior distributions π_1 and π_2 on the parameters.

Having (4.2) and assuming a so-called M-closed perspective [9, Chapter 6], i.e. we believe that the true model is within them and we do not believe in other possible models, we can easily evaluate the probability of a change point as:

$$\mathbb{P}[\text{change}|Z] = \frac{\sum_{i=1}^{n} \mathbb{P}[Z|H_i]\,\mathbb{P}[H_i]}{\sum_{j=0}^{n} \mathbb{P}[Z|H_j]\,\mathbb{P}[H_j]}. \tag{4.3}$$

But to evaluate these probabilities we obviously have to specify the prior probabilities for the models, i.e. $\mathbb{P}[H_i]$, and the parameters, i.e. π_1 and π_2, and, of course, evaluate the above integrals.

A natural choice to code our ignorance on a parameter change before observing data is to assign a prior probability of $\frac{1}{2}$ to the event of a change and distribute the rest probability among the other models, i.e.

$$\mathbb{P}[H_0] = \frac{1}{2},\ \mathbb{P}[H_i] = \frac{1}{2n}, 1 \leq i \leq n.$$

More problematic is the choice of prior distributions for the parameters of the VAR models under ignorance. A common choice is the usage of the *diffusive prior*, which consist of a flat prior on Φ and a Jeffrey's prior on R, so that

$$\pi_D(\Phi, R) \propto |R|^{-\frac{d+1}{2}},$$

a discussion of this prior and other possibilities is given in [87,116]. Although it can be easily shown that under the diffusive prior the posterior distribution

$$\int p(Z|\mathit{\Phi}, R)\pi_D(\mathit{\Phi}, R)d\mathit{\Phi}dR$$

is proper, i.e. normalisable, the choice is problematic for model comparison, as the prior itself is unproper, i.e. we can set

$$\pi_1 \equiv \pi_2 \equiv c\pi_D$$

with an arbitrary chosen constant c. This means that the model probabilities (4.2) as well as probability of change (4.3) are also defined up to a constant, i.e.

$$\mathbb{P}[\text{change}|Z] = c \cdot \frac{\sum_{i=1}^{n} \mathbb{P}[Z|H_i]\,\mathbb{P}[H_i]}{\sum_{j=0}^{n} \mathbb{P}[Z|H_j]\,\mathbb{P}[H_j]}.$$

The constant does not cancel out of the fraction as there are parameters which are not common to all models, i.e. the parameters for the VAR model after a change has occurred.

To emphasise: with the use of an unproper prior we can compare different change point models, as the indeterminate constants do cancel out, but we can not compare the probability between change and no-change.

This general obstacle of Bayesian model selection can be tackled by the usage of so-called "objective" Bayes factors [63], which we are going to introduce in the next section.

Note that on the other hand it is possible to split the change point detection problem into two parts

1. Identify the most likely change point under the assumption that there is one, this requires specifications of parameter priors up to a constant only.

2. Compare the probability of change at the identified possible change point with the probability of no change. Now we have to specify a proper prior for the parameters after the change to avoid arbitrariness.

In fact, this is favourable from an algorithmic viewpoint as it is easier to exclude outliers from the change/no-change decision, see § 4.4, and the M-closed perspective is somewhat arbitrary anyway as the number of models obviously depends upon the defined window $[t_1, t_2]$. However, we still have to compare a change model with a (single) no-change model, i.e. we have to compute

$$\mathbb{P}[\text{change}|Z] = \frac{\mathbb{P}[Z|H_c]\,\mathbb{P}[H_c]}{\mathbb{P}[Z|H_c]\,\mathbb{P}[H_c] + \mathbb{P}[Z|H_0]\,\mathbb{P}[H_0]}, \tag{4.4}$$

where H_c is the model of a change at the pre-computed candidate change point c. Note that, of course, having computed a candidate change point, one could use any other decision criteria, like the approaches presented in § 4.1.2. However, we stick to the Bayesian framework as it naturally allows to handle uncertainty.

4.2.2 Bayes Factors

The Bayes factors are a common way to compare posterior probabilities of two distinct models within a Bayesian setting. Given two models H_i and H_j, the ratio

$$\frac{\mathbb{P}[H_i|Z]}{\mathbb{P}[H_j|Z]} = \frac{\mathbb{P}[Z|H_i]\,\mathbb{P}[H_i]}{\mathbb{P}[Z|H_j]\,\mathbb{P}[H_j]}, \tag{4.5}$$

is called posterior odds. A high ratio means that model H_i is more probable in the light of data the Z than H_j. The Bayes factor B_{ij} is defined as

$$B_{ij} = \frac{\mathbb{P}[Z|H_i]}{\mathbb{P}[Z|H_j]}.$$

Eq. (4.5) reveals the meaning of the Bayes factor: it defines how the data Z transforms the prior odds $\mathbb{P}[H_i]/\,\mathbb{P}[H_j]$ to the posterior odds, i.e. in which direction the data shifts our prior beliefs. The Bayes factor approach is similar to the likelihood ratio statistic, introduced in § 4.1.2, but while the likelihood ratio is obtained via *maximisation* of the likelihood function over the parameter space, the Bayes factor is obtained by *integration* of the likelihood function over the parameter space [16, 63]. Eq. (4.3) can be reformulated in terms of the Bayesian factors, as

$$\mathbb{P}[\text{change}|Z] = \frac{\sum\limits_{i=1}^{n} B_{i0}\,\mathbb{P}[H_i]}{\sum\limits_{j=0}^{n} B_{j0}\,\mathbb{P}[H_j]}. \tag{4.6}$$

This expression can can be interpreted as an assembly of a sequence of tests against the null hypothesis of no change [40].

4.2.3 Non-informative Priors in Model Selection Problems

Unfortunately, the Bayes factors do not resolve the problem with the unproperness of the non-informative standard priors, as they become also arbitrary when used with non-proper priors. Subsequently we present three approaches to tackle this problem by deriving proper priors in a data driven way.

Partial Bayes

A way to obtain a proper prior distribution for some parameter θ despite of ignorance is to split the data Z into two parts Z_p and Z_{-p} and use one part (Z_p) as a training set to specify the prior while the other part (Z_{-p}) is used for testing or analysis, i.e. we set

$$\pi_{PB}(\theta) \propto \pi_D(\theta) L(\theta | Z_p),$$

where $\pi_D(\theta)$ denotes an improper parameter prior. The size of the training set is usually taken as the minimal size to guarantee properness of the resulting prior. A problem is the arbitrariness in the choice of which data points are taken into the training sample. A proposal to overcome this arbitrariness is given by Berger [8], who suggested to average over all possible minimal training sets, the so-called *intrinsic Bayes* approach. The intrinsic Bayes approach can be elegantly expanded if nested models are tested, [17, 40], but has the drawback that computation, even with sampling procedures, of intrinsic Bayes factors is often hard, resp. feasible only for a restricted class of models.

Fractional Bayes

The fractional Bayes approach, put forward by O'Hagan 1995 [89], is based on the idea to use a fraction of the likelihood function, instead of using part of the data, to specify a prior, i.e. to set

$$\pi_{FB}(\theta) \propto \pi_D(\theta) L^b(\theta | Z)$$

with a constant $b \in]0, 1[$. The likelihood function used for decision making is then transformed to $\tilde{L}(\theta|Z) := L^{(1-b)}(\theta|Z)$, thus becoming flatter as a fraction of the information is already used to define the prior distribution. The question of the right choice of a training set is elegantly avoided, as a fraction of *all* data is used. A reasonable choice of b is the minimal value which guarantees properness of the resulting prior, which corresponds to the choice of a maximal spreaded distribution centered by the data.

Imaginary minimal experiment

Another approach presented by Spiegelhalter and Smith [113] is the use of a so-called imaginary minimal experiment. Suppose there are two models to be compared and in at least one of them there is a parameter for which we can only specify an improper non-informative prior. Then the resulting Bayes factor is given by

$$B_{01} = c \cdot \frac{\int f_1(Z|\theta_1)\pi_1(\theta_1)d\theta_1}{\int f_2(Z|\theta_2)\pi_2(\theta_2)d\theta_2},$$

with c an unknown constant. The idea of an imaginary minimal experiment is to fix the undetermined constant c by imagination of a data set Z_I which is just big enough to discriminate between the two models, therefore minimal, but gives maximal support for one of the two models. The reasoning then is that the Bayes factor should favour the supported model but only minimally, due to the smallness of the data set, so that

$$B_{01} \approx 1 \Rightarrow c \approx \frac{\int f_2(Z_I|\theta_2)\pi_2(\theta_2)d\theta_2}{\int f_1(Z_I|\theta_1)\pi_1(\theta_1)d\theta_1}.$$

It has been argued that the definition of an imaginary minimal experiment is sufficient only in rather special cases [89]. Furthermore, it is not clear that the claim $B_{01} \approx 1$ is an appropriate choice in all cases. But, as we will show, in the change point detection framework as presented, the imaginary minimal approach seems to be sensible.

4.2.4 Implementation of the Objective Bayesian Strategies

In the previous sections we collected all necessary ingredients for a Bayesian change point detection, so now we make more precise how to do this in our given scenario. As mentioned, we are going to split the change point detection in two parts, first identify a possible candidate change point, second decide whether it is a change point.

The key ingredient to employ the approaches stated above is that our model allows analytical integration of the likelihood function over parameter space. Assume for the moment an arbitrary time series Z of length T, and the corresponding moment matrix $M = M(Z)$. Since M contains all statistical relevant information of the data we can write $p(M|\Phi, R)$ instead of $p(Z|\Phi, R) = L(\Phi, R|Z)$, as given in Eq. (3.35). Following the notation introduced in § 3.3.4 we denote by U_{11} and U_{22} the corresponding diagonal blocks of the triangular matrix U obtained from the Cholesky factorisation of M, and by $m := M_{11} = T - p$ the upper left scalar entry of M. Then, see Appendix A.2,

$$\begin{aligned} I[M] :=& \int p(M|\Phi, R)\pi_D(\Phi, R)d\Phi dR = \int L(\Phi, R|M)\pi_D(\Phi, R)d\Phi dR \\ =& \pi^{\frac{d(d-1)}{4}}|U_{11}|^{-d}|\sqrt{\pi}U_{22}|^{-(m-dp-1)}\prod_{j=1}^{d}\Gamma\left(\frac{m-dp-j}{2}\right), \end{aligned} \tag{4.7}$$

where Γ denotes the Gamma function and $|\cdot|$ the matrix determinant. Note that the integral exist only if $m > d(p+1) \Leftrightarrow T > d(p+1)+p$, therefore at least $(d+1)(p+1)$ subsequent points before and after a change point are needed for evaluation. From Eq. (3.35) another property of the M-matrices can be deduced, namely that information coming from different time series (parts), e.g. Z_1 and Z_2 can be combined by just adding the moment matrices, since

$$L(\Phi, R|M(Z_1))L(\Phi, R|M(Z_2)) = L(\Phi, R|M(Z_1) + M(Z_2)). \tag{4.8}$$

Identification of a change point assuming its existence Using the notation introduced in § 4.2.1, the aim is to calculate the probabilities of potential positions of a candidate change point, i.e. to calculate

$$\mathbb{P}[H_i|Z] \propto \int p(Z_1|\Phi, R)\pi_1(\Phi, R)\ d\Phi dR \int p(Z_2|\Phi, R)\pi_2(\Phi, R)d\Phi dR, \tag{4.9}$$

with $1 \leq i \leq n = t_2 - t_1$ and

$$\begin{aligned} Z_1 =& Z_1(i) := \{\boldsymbol{z}_1, \boldsymbol{z}_2 \ldots, \boldsymbol{z}_{t_1+i-1}\}, \\ Z_2 =& Z_2(i) := \{\boldsymbol{z}_{t_1+i}, \boldsymbol{z}_{t_1+i+1} \ldots, \boldsymbol{z}_T\}. \end{aligned}$$

Note that t_1 must be larger or equal than $(d+1)(p+1)$ and t_2 smaller than $T - 2 - (d+1)(p+1)$, so that each segment contains at least $(d+1)(p+1)$ data points, since otherwise the integrals can not be evaluated. We can include information which might be already obtained from a previous observation Z_0 into the prior distribution π_1 by setting

$$\pi_1(\Phi, R) \propto \pi_D(\Phi, R)L(\Phi, R|Z_0),$$

which is formally a partial Bayes approach, however, the motivation is not make the prior distributions proper, as at this stage proper priors are not essential, but to include prior information from previous observations. Otherwise we take the diffuse prior for both parameter sets, i.e.

$$\pi_1(\Phi, R) = \pi_2(\Phi, R) \propto \pi_D(\Phi, R).$$

Using (4.7) and (4.8) we have

$$\mathbb{P}[H_i|Z] \propto I[M(Z_0) + M(Z_1)]I[M(Z_2)], \tag{4.10}$$

with prior observations Z_0. If there are no prior observations we set $M(Z_0)$ to $\mathit{0}$. Thus it is possible to determine the most probable change point $\hat{c}$ analytically, by choosing

$$\hat{c} = \operatorname*{argmax}_{1 \leq i \leq n} \mathbb{P}[H_i|Z],$$

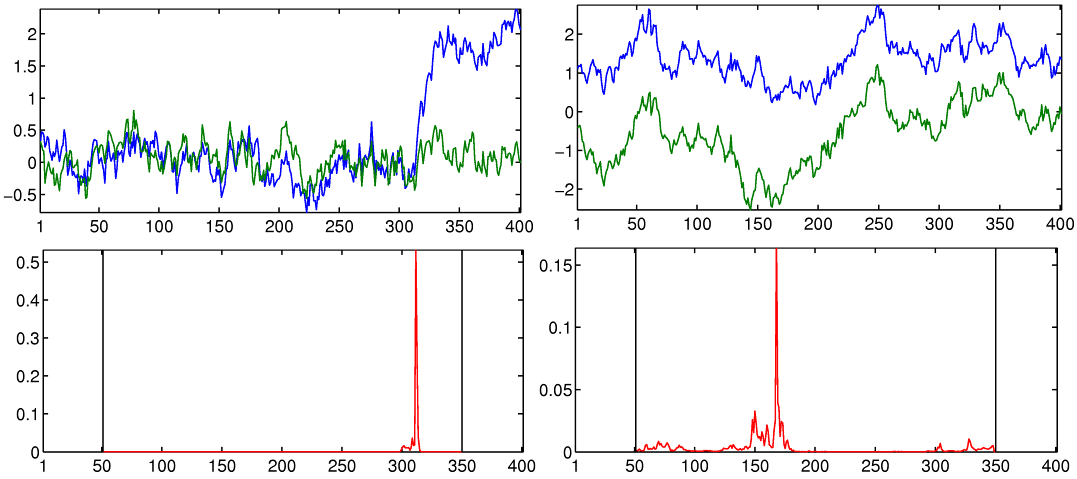


Figure 4.1: *Left:* the top panel displays a 2-dimensional trajectory against time generated from a VAR(1) process with a switch in the mean occurring at $t = 311$. Below is the probability density of a change point conditional to its existence. The margin lines left and right of the panel mark the test interval $[t_1, t_2]$. *Right:* An example where no change point occurs in the trajectory. Still we obtain a candidate change point.

even if we do not know if this is a real change point. An example is depicted in Fig. 4.1, where it can be seen how a change point can be identified by locating the maximum of the conditional density, but that one still has to decide if this maximum really belongs to a change point. This can be done by Fractional Bayes or the Imaginary Minimal Experiment as exemplified below. In general, however, we could use any method for our decision which seems to be appropriate. Therefore splitting the change point analysis has the big advantage that the hard problem, i.e. the model decision problem, is now separated from the easy problem, i.e. locating the most probable change point.

Fractional Bayes. The fractional Bayes approach can be easily implemented by noting from (3.35) that

$$\begin{aligned} &L^b(\Phi, R|M) \\ &= \left(\frac{1}{\sqrt{|2\pi R|}}\right)^{bm} \exp\left(-\frac{1}{2}\operatorname{tr}((bM_{22} - bM_{21}\Phi' - \Phi bM_{12} + \Phi bM_{22}\Phi')R^{-1})\right) \\ &\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad = L(\Phi, R|bM), \qquad (4.11) \end{aligned}$$

so that, using the notation introduced above, we have

$$\int L^b(\Phi, R|Z)\pi_D(\Phi, R)d\Phi dR = I[bM(Z)],$$

and, using (4.8),

$$\int L(\Phi, R|Z_1)L^{(1-b)}(\Phi, R|Z_2)\pi_D(\Phi, R)d\Phi dR = I[M(Z_1) + (1-b)M(Z_2)].$$

Setting the prior probabilities for change and no change equally to $\frac{1}{2}$ the quantity to compute reads

$$\begin{aligned}\mathbb{P}[\text{change}|Z] &= \frac{\mathbb{P}[Z|H_{\hat{c}}]}{\mathbb{P}[Z|H_{\hat{c}}] + \mathbb{P}[Z|H_0]} \\ &= \frac{\prod\limits_{i=1,2} \int p_i(Z_i|\Phi, R)\pi_i(\Phi, R)d\Phi dR}{\prod\limits_{i=1,2} \int p_i(Z_i|\Phi, R)\pi_i(\Phi, R)d\Phi dR + \int p_0(Z|\Phi, R)\pi_1(\Phi, R)d\Phi dR}.\end{aligned}$$

We leave $\pi_1 \propto \pi_D$ unproper, since the normalisation constant cancels out anyway, or with prior observations $\pi_1(\Phi, R) \propto \pi_D(\Phi, R)L(\Phi, R|Z_0)$, but for π_2 we use the fractional Bayes approach

$$\pi_2(\Phi, R) := \frac{\pi_D(\Phi, R)L^b(\Phi, R|Z_2)}{\int \pi_D(\Phi, R)L^b(\Phi, R|Z_2)d\Phi dR}.$$

Since some data is used to specify the prior distribution, we can not use all of it for calculation of the probability, i.e. we set

$$\begin{aligned} p_1(Z_1|\Phi, R) &= L(\Phi, R|Z_1), \\ p_2(Z_2|\Phi, R) &= L^{(1-b)}(\Phi, R|Z_2), \\ p_0(Z|\Phi, R) &= L^{(1-b)}(\Phi, R|Z_2)L(\Phi, R|Z_1). \end{aligned}$$

Assembling all the pieces we obtain, in compact notation, the probability

$$\mathbb{P}[\text{change}|Z] = \frac{I[M(Z_1)]I[bM(Z_2)]}{I[M(Z_1)]I[bM(Z_2)] + I[bM(Z_1) + (1-b)M(Z_2)]} \tag{4.12}$$

The minimal value of b is determined by the minimal value for which

$$I[bM(Z_2)]$$

is defined (cf. § A.2). Therefore the minimal value of b is given by

$$b_{min} = \frac{d(p+1)+1}{m(Z_2)},$$

which means that the upper left entry of $bM(Z_2)$ just meets the threshold of $d(p+1)+1$.

Imaginary minimal experiment. To employ the Spiegelhalter/Smith approach we have to define an adequate imaginary minimal experiment Z_I. If we want to decide if Z_2 is generated by the same VAR model as Z_1 we need, as stated above, a minimum of $(d+1)(p+1)$ observations, otherwise integration over the parameter space is not defined anymore. Maximal support for the "no change"-model would be the same observed statistic in both observed time series, i.e.

$$\frac{M(Z_1)}{m(Z_1)} = \frac{M(Z_I)}{m(Z_I)} \Leftrightarrow M(Z_I) = \frac{dp+d+1}{m(Z_1)} M(Z_1).$$

With this definition of $M(Z_I)$ we can fix the undetermined constant in the Bayes factor as

$$c_I = \frac{I[M(Z_1) + M(Z_I)]}{I[M(Z_1))]I[M(Z_I)]},$$

and obtain the Bayes factor

$$B_I^{(i)} = c_I \cdot \frac{I[M(Z_1)]I[M(Z_2)]}{I[M(Z_1) + M(Z_2)]}.$$

Substituting the obtained Bayes factor in (4.6) gives an expression for the change probability:

$$\mathbb{P}[\text{change}|Z] = \frac{I[M(Z_1) + M(Z_I)]I[M(Z_2)]}{I[M(Z_1) + M(Z_2)]I[M(Z_I)] + I[M(Z_1) + M(Z_I)]I[M(Z_2)]}. \tag{4.13}$$

Coming back to the example given in Fig. 4.1, we can now compute the probability of a change for the identified candidate change point in both time series. Then Z_1 becomes the part of the analysed time series before the candidate change point, i.e. where the conditional change point probability is maximised, and Z_2 the part after the candidate change point. Computation of the change probabilities (4.12) and (4.13) corresponding to these segments for both time series gives

$\mathbb{P}[\text{change}\|Z]$	Time series 1 (left in Fig. 4.1)	Time series 2 (right in Fig. 4.1)
Fractional Bayes (4.12)	1	0.0217
Imaginary Experiment (4.13)	1	0.0226

We see that both procedures yield the right result, and reject a change point where no change occurred (time series 2) while accepting the true change point (time series 1). Of course besides the both suggested procedures any other of the approaches mentioned in § 4.1.2 can be used. However we choose the fractional Bayes approach as it worked out satisfactory in various test cases, is computational cheap, includes parameter uncertainty and is less speculative than the imaginary minimal experiment approach.

4.2.5 Asymptotics

Assume, that we have identified a candidate change point and want to test/ decide if it is a real change point, i.e. we have to decide between two hypotheses

$$H_0 : (\mathit{\Phi}_1, R_1) - (\mathit{\Phi}_2, R_2) = (\mathit{0}, \mathit{0}) \text{ against } H_1 : (\mathit{\Phi}_1, R_1) - (\mathit{\Phi}_2, R_2) \neq (\mathit{0}, \mathit{0}).$$

A common procedure in this setting is to perform a likelihood ratio test, cf. § 4.1.2, since the distribution of the likelihood ratio statistic under H_0 is asymptotically known. But hypothesis testing as a decision rule is not consistent. In fact, the concept of consistency is very much opposed to that of hypothesis testing, since the essential point in hypothesis testing is to define a decision rule with a predefined (small) level of probability to reject H_0 even if it is true. In other words, even in the asymptotic case there is a (known) probability of making a wrong decision. Surprisingly, this does not hold if decisions are based on Bayes factors or Bayesian probabilities, like in (4.6). Then, under quite general assumptions, it can be shown [39, 89], that under the alternative H_0 or H_1 the decision rule is consistent, i.e.

$$\lim_{T\to\infty} \mathbb{P}[H_0|Z] = \begin{cases} 1 \text{ , if } H_0 \text{ is true.} \\ 0 \text{ , if } H_1 \text{ is true.} \end{cases}$$

However, consistency is achieved only if the increase of data does increase the accuracy of all parameters under both hypotheses. That is, consistency breaks down if the amount of information in the likelihood concerning some parameters increases significantly slower with T than information on other parameters, for examples see [25, Sec. 7]. With respect to the change point problem this means that consistency is only guaranteed if both segments of the time series grow with T at the same rate. There are proposed correction terms if information on different parameters grows at different speed [26], but they are hard to evaluate.

4.3 Algorithmic Procedure

In this section we are going to state the proposed algorithmic procedure derived by the considerations above. Before we do this, we will comment on how to cope with effects due to the finiteness of the time series the change point analysis is applied to. After stating the core algorithm, we will also comment on post processing possibilities, followed by showing two examples in the next section.

To make the following sections more readable we introduce the following notation: Given some time series (segment) $\{\boldsymbol{z}_{t_0}, \boldsymbol{z}_{t_0+1}, \ldots, \boldsymbol{z}_{t_1}\}$ we define by

$M(t_0, t_1)$ the corresponding moment matrix

$$M(t_0, t_1) := \sum_{t=t_0+p}^{t_1} \begin{pmatrix} 1 \\ \boldsymbol{z}_{t-p} \\ \vdots \\ \boldsymbol{z}_t \end{pmatrix} \begin{pmatrix} 1 & \boldsymbol{z}'_{t-p} & \dots & \boldsymbol{z}'_t. \end{pmatrix}$$

Obviously the definition depends on VAR order p, but we omit an appropriate index if the p is evident from the context. Note that even though the notation does not capture possible modifications of the statistics by leaving out certain summands, as done in § 3.4.2 to cope with circular data, all of the following holds also for modified moment matrices as such modifications only correspond to the ignorance of some information contained in the data.

4.3.1 Margin effects

A systematic problem that occurs if change point detection based upon parameter estimation or parameter densities, as our approach, is applied to finite a time series is that, if the segments of the time series are too short, the information about the parameters in these segments can be very misleading. An illustration of this effect is given in the left panel of Fig. 4.2. Therefore, if a time series with no change point is analysed, the change point algorithm will tend to detect change points close to the ends of the time series. The key point is that this effect can not be overcome just by regarding parameter uncertainty, as information contained in a short time series segment is not just insufficient but misleading. An example is given in the right panel of Fig. 4.2. Here a trajectory generated by a VAR model is shown together with the function

$$\begin{gathered} f(i) = I[M_1(i)]I[M_2(i)], \\ M_1(i) = M(1, i), \qquad M_2(i) = M(i+1, T), \end{gathered}$$

which determines the candidate change points when no prior information is available, cf. § 4.2.3. It should be of no surprise that the candidate change points are located at the margins of the time series. But if we look at the probability that a given candidate change point is a real change point, according to (4.6), we see that it is, as a function of candidate change points, close to one at the margins. To prevent this effect one could add a penalty function for change points close to the border. We implement this strategy by testing for change points only within a window which leaves the margins large enough. Note that we do not need a left margin if prior information about the parameters is included from prior observations, i.e. if another moment matrix M_p is at hand such that we can set

$$M_1(i) = M_p + M(1, i).$$

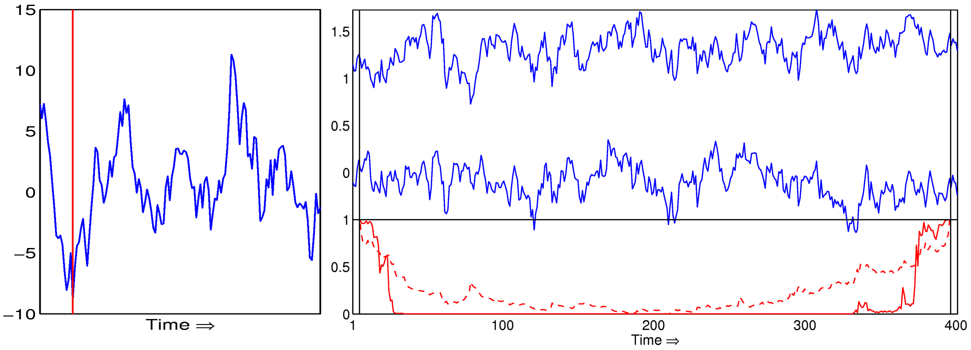


Figure 4.2: *Left:* A realisation of an one dimensional VAR(1) process is plotted. If parameters are estimated only from data points left to the red line they will significantly differ from parameters obtained using the whole time series. *Right:* In the upper part of the figure a two dimensional trajectory from a VAR(1) process is plotted (blue lines). Below the black line the logarithm of f (see text) is shown (dashed red line, arbitrary scale). Also shown is the probability of a change point for each possible change point that would be obtained by using the fractional Bayes approach (red line). The vertical lines border the minimal length of a time series segment, here $(d+1)(p+1) = 6$. It can be seen that in the margin regions the procedure would always detect a change point.

4.3.2 Short Time Deviations - Recrossings

When applying change point detection to real data one naturally has to handle with outliers, i.e. single points whose dynamical behaviour is different than the others, or short time deviations, i.e. for a short period of time the dynamical behaviour of the time series is different from that before and after. In the computation of reaction rates as done in Chapter 5 such effects are encountered quite systematically and called recrossings. Often one does not want to detect such short time deviations as one is interested only in persistent changes of the dynamical behaviour. One can avoid detection of deviations shorter than some predefined time $t_r \in \mathbb{N}$ in the following way:

Having identified a candidate change point c, one calculates the probability of a change occurring at that position based upon the matrices M_1, M_2, M_3 which contain the sufficient statistics of the time series before, after and without a change point, i.e.

$$M_1 = M(1, c-1), \quad M_2 = M(c, T), \quad M_3 = M_1 + M_2.$$

Instead of using these matrices one can exclude the information contained in the trajectory for t_b steps after the candidate change point by using, instead of M_2 and M_3,

$$\widetilde{M}_2 = M(c + t_b, T), \quad \widetilde{M}_3 = M_1 + \widetilde{M}_2.$$

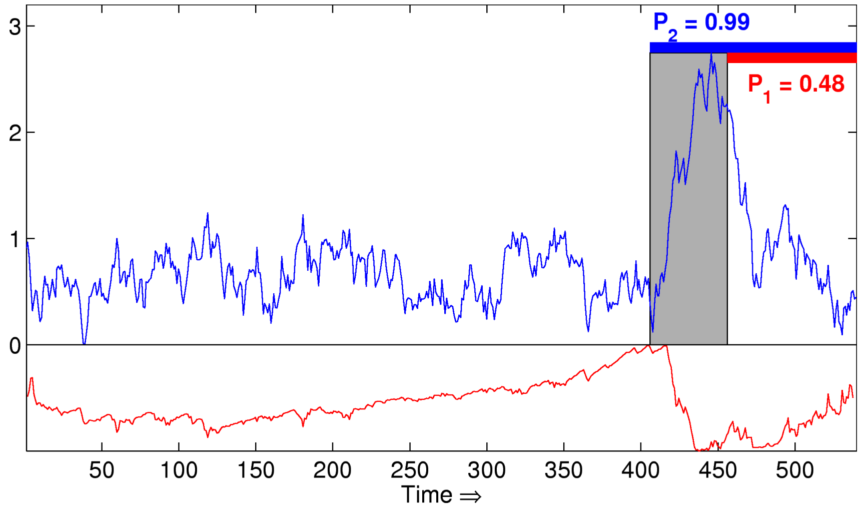


Figure 4.3: A one dimensional time series is shown (blue) which fluctuates mainly in $[0, 1.5]$, at around $t = 400$ there is an obvious rise but after approx. 50 steps it seems to regain the beforehand behaviour. The (logarithm of the) conditional change point distribution (red line) clearly identifies a candidate change point at the beginning of the rise. The change probability at this candidate change point is $P_2 = 0.99$. If a window as described in the text is used to mask out a part of the trajectory after the candidate change point (gray shaded) the change probability drops to $P_1 = 0.48$, as after the window the behaviour of the trajectory is similar to that before.

The rational behind this strategy is, that only if the dynamical behaviour after the potential change point stays different longer than the predefined time t_b, it will affect the calculated probabilities because the dynamical information between c and t_b is not used. For an example see Fig. 4.3.

4.3.3 Algorithm

We summarise the results obtained so far in the basic algorithmical procedure stated in Alg. 2.

```
Algorithm 2: Sequential change point detection
Parameter: $t_m \in \mathbb{N}$ (minimal segment size)
           $t_u \in \mathbb{N}$ (length of update window)
           $t_b \in \mathbb{N}$ (length of buffer zone)
           $\alpha \in ]0, 1[$ (threshold value for detection of a change)
           $p$ (VAR order) or $p_{max}$ (maximal VAR order)
Input    : A $d$-dimensional time series $\boldsymbol{Z} = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots\}$ with se-
           quential access.
If $p$ is not given estimate $p \in \{0, 1, \ldots, p_{\max}\}$ based on $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{t_m}$.
$M_I \leftarrow M(1, t_m)$
$t_E \leftarrow 2t_m + t_u$
$\mathbb{P}[\text{change}] \leftarrow 0$
while $\mathbb{P}[change] < \alpha$ do
2.1    Determine candidate change point between $t_m$ and $t_E - t_m$:
       for $k \leftarrow p + 2$ to $t_E - 2t_m$ do
           $M_1 \leftarrow M(t_m + 1, t_m + k) + M_I$
           $M_2 \leftarrow M(t_m + k + 1, t_E)$
           $l_{k-p-1} \leftarrow I[M_1]I[M_2]$                          (cf. (4.10))
       $\hat{c} = t_m + p + \operatorname{argmax}_k l_k$
       If the candidate change point is not too close to the margin compute its
       probability:
2.2    if $t_E - \hat{c} > t_b + t_m$ then
           $M_1 \leftarrow M(t_m, \hat{c}) + M_I$
2.3        $M_2 \leftarrow M(\hat{c} + 1 + t_b : t_E)$
           $b \leftarrow (dp + d + 1)/(M_2(1, 1))$
           $\mathbb{P}[\text{change}] = \frac{I[M(Z_1)]I[bM(Z_2)]}{I[M(Z_1)]I[bM(Z_2)] + I[bM_1 + (1-b)M_2]}$    (cf. (4.12))
       $t_E \leftarrow t_E + t_u$
Output: The change point $\hat{c}$ and a corresponding moment ma-
        trix $M_1$ for the identified segment.
```

Note that in the implementation of this algorithmic scheme we substituted quantities by their logarithmic values where suited to avoid numerical problems. A few comments on the parameters:

- Note that the most important parameter is t_m, as it determines the resolution of the change point algorithm, examples are given in the § 4.4.1. Defining a minimal segment size of course requires that there is no change point within the first t_m data points.

- The parameter t_b has a double role, first it is used as described in § 4.3.2 to exclude short deviations from the change point detection and, second, it excludes (2.2) candidate change points which are close to the right margin of the test window, because the change might have had happen at the very end of the time series (where it was not tested due to margin effects). This is unproblematic as the change point will be detected in the next circle again.
- In practice, it turns out that the threshold parameter α should be chosen rather large to avoid false alarms, fundamental changes will reflect in a change probability close to one anyway.

For a long time series the stated algorithm can become quite ineffective as, after every new received data package, the loop to determine a candidate change point (2.1) is executed over all data points received so far. In practice this can be overcome by testing for a candidate change point only over the last w_{max} received data points, and add the information content of the beforehand received data points to the moment matrix M_I.

In order to detect multiple change points the algorithm can be used multiple times, starting each time from the last detected change point again. In applications it is often advisable to start the algorithm again with a certain lag to the last detected change point, in order to prevent that the transition phase between different dynamical phases spoils the statistics.

Note that after the information of a part of the time series is stored in a moment matrix M this part can be completely discarded as all statistical relevant information is now stored in M. Therefore the whole approach is suited to handle with large data sets as e.g. occur in molecular dynamics simulations, see § 5.3 for an example.

4.3.4 Post processing

If the change point algorithm is applied repeatedly on a time series to obtain multiple change points, the procedure will finally generate a sequence of change points $c_0 := 1, c_1, \ldots, c_{s-1}, c_s := T + 1$ and therefore a segmentation of a given time series Z, whose segments are given by

$$Z_i = \boldsymbol{z}_{c_{i-1}}, \ldots, \boldsymbol{z}_{c_i - 1}, \quad 1 \leq i \leq s,$$

and the corresponding moment matrices $M_1, \ldots, M_s$ which contain the statistically relevant information. These matrices can be used for post processing purposes, e.g. to drop falsely detected change points or to group the data globally.

For this purpose we define a distance matrix D, measuring the distance between all identified segments $Z_1, \dots, Z_s$ of the time series, according to the probability that the segments are generated by the same VAR model:

$$\begin{aligned} D_{ij} &= \begin{cases} \dfrac{I[M_i]I[bM_j]}{I[M_i]I[bM_j] + I[bM_i + (1-b)M_j]}, & \text{if } m_i \geq m_j \\ \dfrac{I[bM_i]I[M_j]}{I[bM_i]I[M_j] + I[(1-b)M_i + bM_j]}, & \text{if } m_i < m_j, \end{cases} \\ b &= \frac{dp + d + 1}{\min(m_i, m_j)}, \end{aligned} \tag{4.14}$$

with $1 \leq i, j \leq s$ and m_i the upper left entry of M_i. So the distance is just the probability of a change point, where the change point has to be between the two segments. To make the distance matrix symmetric and to avoid waste of information from the shorter segment we always use the longer segment to extract prior information about the parameters.

In order to exclude falsely detected change points one should test again for a change point between adjacent segments, i.e. generate a new set of change points $\tilde{c}_0, \dots, \tilde{c}_k$ and a corresponding set of moment matrices $(\tilde{M}_1, \dots, \tilde{M}_k)$ using the distance defined in (4.14) by Algorithm 3.

Algorithm 3: Exclusion of falsely detected change points

```
$\tilde{c}_0 = 1$
$j = 1$
for $i = 1$ to $s - 1$ do
  if $D_{i,i+1} < \alpha$ then
    $\tilde{M}_j = M_i + M_{i+1}$
  else
    $\tilde{c}_j = \tilde{c}_i$
    $j = j + 1$
    $\tilde{M}_j = M_i$
  end if
end for
$\tilde{c}_{j+1} = T$
```

Segments may be merged again, even if the same criteria is used as in the change point Algorithm 2, due to the fact that the decision is now based on more data points.

Furthermore, the obtained distance matrix can be used to cluster the data, i.e. to merge different time series segments (in fact one would first exclude the falsely detected change points and then set up a full distance matrix with the set of merged moment matrices), e.g. by an hierarchical clustering algorithm [59]. Therefore, the distance between two clusters C_1 and C_2 is given by the maximal distance between any member of one cluster to any member of the other cluster:

$$d(C_1, C_2) = \max_{\substack{M_i \in C_1 \\ M_j \in C_2}} D_{ij},$$

alternatively one can define the distance between clusters by the minimal distance between any two members. The hierarchical structure appears by gradually raising the maximal distance $d_{\max}$ allowed for objects within a cluster. If $d_{\max} = 0$ all moment matrices $M_1, \dots, M_s$ define their own cluster. By raising $d_{\max}$ eventually two segments are allowed to form a cluster, further on other segments may join the cluster or define their own cluster or two clusters may merge to a single cluster. After merging the moment matrices belonging to the same cluster wrt. $d_{\max}$ one would iterate the process until there is no more merging of moment matrices, an example is given in § 4.4.3.

Of course, one can think of other clustering strategies, e.g a clustering on parameter space instead of clustering the moment matrices.

4.4 Application to Time Series

4.4.1 1D Test Potentials

As mentioned before, the parameter for the minimal segment size t_m in Alg. 2 is of special importance as it controls the sensitivity of the change point detection. To demonstrate this we define three simple one dimensional test potentials. The first

$$V_s(x) = -2\left(\exp(-0.3(x+3)^2) + \exp(-0.3(x-3)^2)\right) + 0.001x^4 \tag{4.15}$$

is a smooth double well potential with two minima at $x = 3$ and $x = -3$. The added fourth order term embeds this structure in a basin with unbounded walls. The second one,

$$\begin{aligned} V_p(x) = &- 2\left(\exp(-0.3(x+3)^2) + \exp(-0.3(x-3)^2)\right) + 0.001x^4 \\ &+ \sum_{i=1}^{5} a_i \sin(b_i x + c_i), \end{aligned} \tag{4.16}$$

is obtained by small perturbations of the first one with sinusoidal terms. The parameter $a_i \in [0, 0.3]$, $b_i \in [0, 10]$ and $c_i \in [0, \pi]$ are randomly drawn[1]. Finally

[1] In this example the parameters were, in vector notation $a = (0.096, 0.160, 0.027, 0.034, 0.041)$, $b = (9.899, 5.144, 8.843, 5.880, 1.548)$ and $c = (0.628, 1.279, 2.352, 2.594, 2.482)$.

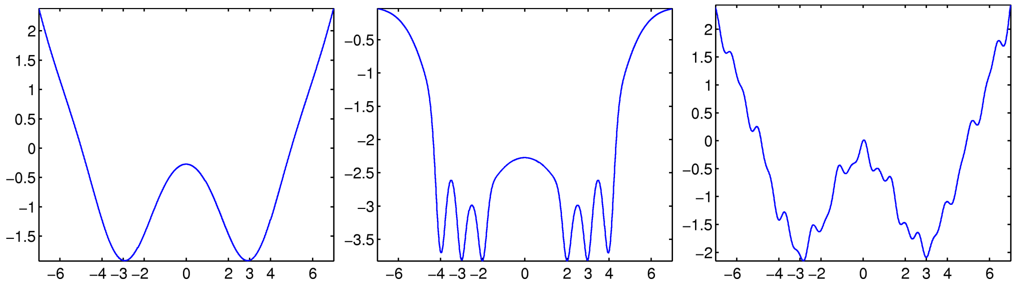


Figure 4.4: *Left:* The smooth potential V_s consisting of two local wells. *Middle:* The potential V_l exhibits three local wells within each major well. *Right:* V_p is based on V_s with minor irregular perturbations.

we have

$$\begin{aligned} V_l(x) = &-2\left(\exp(-0.3(x+3)^2)+\exp(-0.3(x-3)^2)\right) \\ &-1.8\left(\exp(-10(x+4)^2)+\exp(-10(x-4)^2)\right) \\ &-\left(\exp(-15(x+3)^2)+\exp(-15(x-3)^2)\right) \\ &-\left(\exp(-13(x+2)^2)+\exp(-13(x-2)^2)\right)+B(x), \end{aligned} \tag{4.17}$$

which has three pronounced local minima added to each of the major wells. The fourth order term which appears in V_s and V_p is replaced a switching potential

$$B(x) := \begin{cases} 0.892x & \text{, if } x \geq 5, \\ -0.892x & \text{, if } x \leq 5, \\ 0 & \text{, else.} \end{cases}$$

The three test potentials, which are depicted in Fig. 4.4, are used to define a diffusion process via

$$\dot{x}(t) = -\nabla_x V(x(t)) + \sqrt{\beta}\dot{W}(t),$$

where V is replaced by V_s, V_p or V_l, while the temperature parameter β is set to $\beta = 1.21$ in all three cases.

We expect that the dynamical behaviour of the so defined diffusions can be characterised by a switching process between the two dominant wells of V_s on a large time scale, while the local structure in V_l should induce another switching regime on a shorter time scale. The perturbed potential V_p was chosen to check if the change point algorithm can handle (weak) deviations from the model assumptions as it destroys the harmonic structure of the global wells. From the so defined diffusions we obtain a trajectory by a simple Euler-Maryuama discretisation

$$x_{t+1} = x_t - \tau\nabla_x V(x_t) + \sqrt{\beta\tau}\mathcal{N}(0,1),$$

with the integration time step τ set to $\tau = 0.01$ and a start value $x_0 = 2$. Integrating until $T = 199.99$ yields three trajectories with 20000 data points for each of the three models, which we denote by X_s, X_l and X_p subsequently. These trajectories where analysed with the change point algorithm applied repeatedly so that multiple change points were detected while scanning the time series from beginning to end. Several runs were conducted. The buffer parameter t_b, the probability threshold α and the order parameter p were kept fixed all the times, i.e. $t_b = 20, \alpha = 0.7$ and $p = 1$, while the minimal segment size t_m was varied and the update window t_u was set equal to t_m. Four runs where conducted for each trajectory with t_m set to $50, 100, 200$ and 1000. The results, after deletion of false alarms as described in Alg. 3, are shown in Fig. 4.5.

A first glance at the results confirms that the change point detection algorithm is able to detect transitions between the two major wells in all three test cases. There are no other detected change points than these transitions if the minimal segment size t_m is chosen large enough. Decreasing t_m corresponds to an increase of detected change points, this is more pronounced if the potential used for generation of the time series has a more pronounced local structure, i.e. for $t_m = 50$ the number of detected change points drastically decreases from X_s to X_p to X_l. The reason for this can be seen in Fig. 4.6 where we zoomed in the first 2100 data point of the three time series and marked the change points detected with $t_m = 50$. Within this interval only one change point is detected in X_s, the trajectory coming from the smooth potential V_s, and this one corresponds to a transition between the two major wells. For the same interval there are 4 detected change point X_p and none of them corresponds to a change to another major well. Instead these change points mark two areas where the time series is trapped for a short time in one of the local wells created by the perturbation. Finally, in the time series piece of X_l there are 13 detected change points and one can nicely see how these resolve the local structure of the potential, i.e. how metastabilities on a faster time scale than the transitions between the major wells are resolved.

In the first two figures of Fig. 4.7 it can be seen how the increase of t_m reduces the number of detected change points. While with $t_m = 50$ the local structure of X_l is nicely resolved, setting the $t_m = 100$ prevents the detection of the first change point in the interval $[900, 2000]$ of X_l as the resulting segment would be too small. This results into a large estimate for the variance of the current time series segment such that excursions from one local well to another one are not seen as jumps anymore but as excursions due to the high variance. But if the time series stays longer than t_m in one of these local wells again, another change point is detected (at $t \approx 1800$), since then a segment with much lower variance can be identified. The third figure in Fig. 4.7 gives an example of a falsely detected change point in X_s, looking at the MLEs for the local models obtained from the corresponding moment matrices M_1 and M_2 (see picture) illuminates the reason. The MLE model for the first segment is given by

$$\dot{x} = -1.21(x(t) - 2.64) + 1.14\dot{W}(t),$$

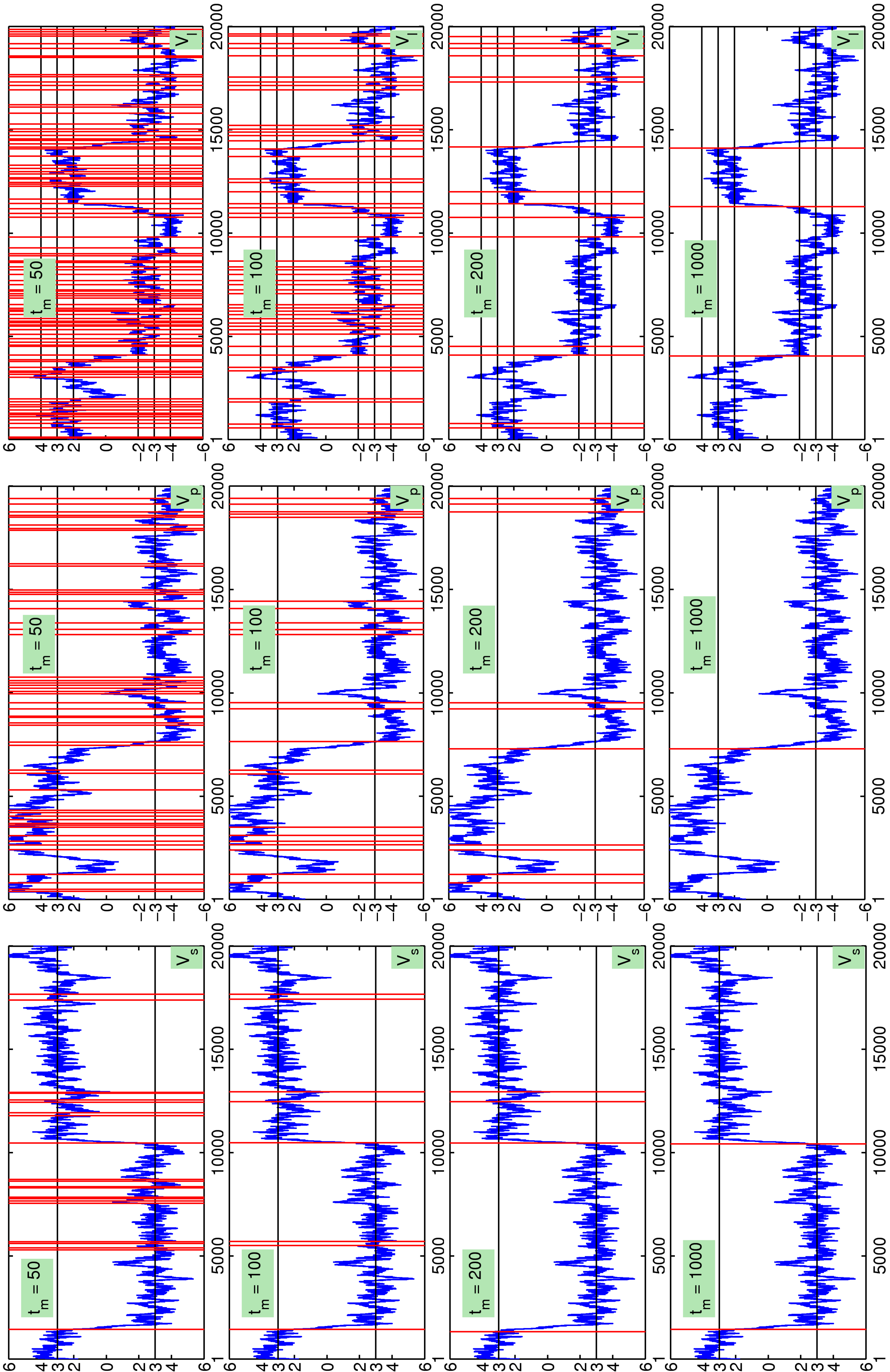


Figure 4.5: Results of the change point analysis for the 3 generated time series with minimal segment size set to $t_m = 50, 100, 200$ and 1000 (see text).

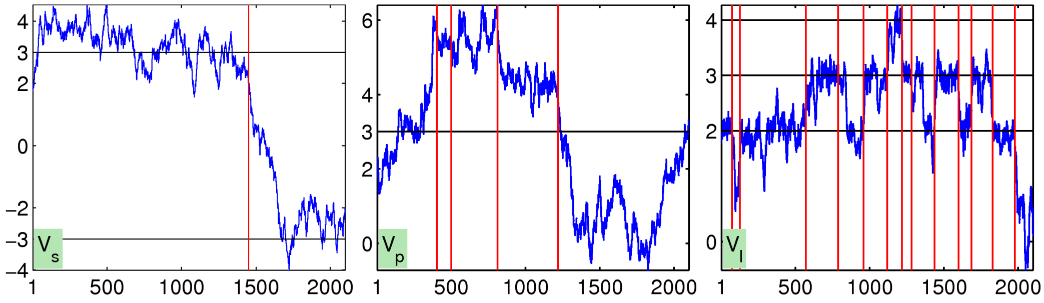


Figure 4.6: The first 2100 data points of the time series X_s, X_p and X_l (from left to right) are shown. Red lines mark detected change points (with $t_m = 50$).

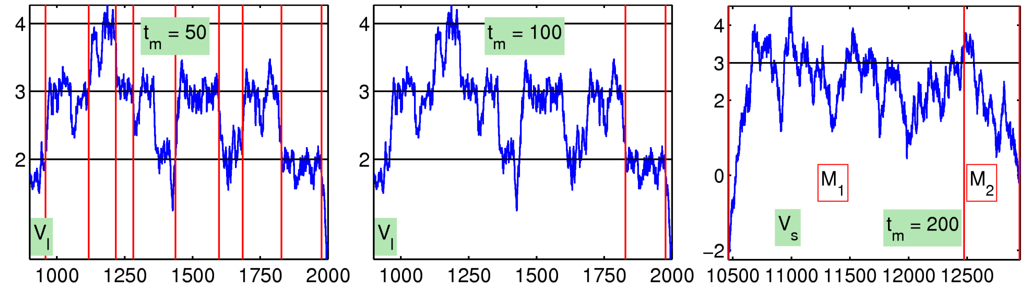


Figure 4.7: A close up of X_l to the interval $[900, 2000]$. Detected change points, with $t_m = 50$ (left) and $t_m = 100$ (middle), are marked by red lines. *Right:* A falsely detected change point in X_s ($t_m = 200$), cf. text.

while for the second segment we have the local model

$$\dot{x} = -0.31(x(t) + 0.58) + \dot{W}(t).$$

This is due to the fact that the MLE setting always tries to reduce the variance in the estimated model, i.e. if there is a random excursion away from the area where the mean is assumed then, if possible, it is always tried to be interpreted this as a drift to a new mean. A larger minimal segment size eliminates this problem as with high probability the trajectory will return to the old area and therefore making the new model implausible.

Note that the detection of change points stemming from rough potentials or randomly generated patterns can be prevented by thinning out the time series which destroys such “local” effects. In Fig. 4.8 we demonstrate this by analysing the time series obtained by taking only every 20th time step in X_s, X_p and X_l, which corresponds to set the time discretisation step τ from 0.01 to 0.2, yielding three time series of length 1000. Analysing them using $t_w = 50$ for change points gives, with a single exception in X_l, only jumps between the major wells.

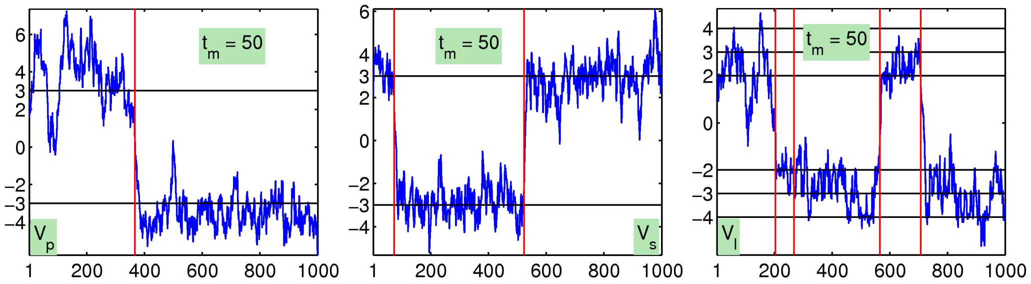


Figure 4.8: The results of the change point analysis with $t_m = 50$ for the three thinned out time series, i.e. only every 20th point was taken, is shown.

4.4.2 The Threehole Potential

The next example is diffusion in a two-dimensional potential, which for obvious reasons it is called the Three-hole potential and is defined as

$$\begin{aligned} V(x,y) =& -3\exp(-x^2-(y-\tfrac{5}{3})^2) - 5\exp(-(x-1)^2-y^2) \\ & - 5\exp(-(x+1)^2-y^2) + 3\exp(-x^2-(y-\tfrac{1}{3})^2) \\ & + 0.2x^4 + 0.2(y-\tfrac{1}{3})^4. \end{aligned} \tag{4.18}$$

It exhibits three minima, a shallow one at approximately $(0, 1.7)$, two deep ones at approximately $(\pm 1, 0)$, and a maximum at approximately $(0, 0.3)$. The fourth order term in (4.18) again embeds the structure in a basin with unbounded walls. This potential has been studied in [80, 92] to analyse the dynamics of diffusion processes within it, which are given by

$$\dot{\boldsymbol{z}}(t) = -\nabla_{\boldsymbol{z}} V(\boldsymbol{z}(t)) + \sqrt{\beta}\dot{\boldsymbol{W}}(t), \tag{4.19}$$

with $\boldsymbol{z} = (x, y)$. The invariant measure of (4.19) is the Boltzmann-Gibbs distribution, i.e. proportional to $\exp(-\beta V)$. It can be seen in Fig. 4.9 that at lower temperatures the invariant measure concentrates in the minimal potential energy basins, while at higher temperatures it is more spreaded.

A linear SDE is expected to be a good approximation of the diffusion process (4.19) as long as it moves in the vicinity of any of the potential energy basins, since the shape of the potential energy surface is approximately quadratic there. This approximation definitely breaks down if the process switches from one basin to another basin, which it (rarely) does due to the random force. But in the other basin the dynamical behaviour should be well approximated by a linear SDE as well. Therefore, the Threehole potential should be a good test system of our change point detection algorithm. In fact, in all our trials it worked very satisfactory. As an illustration a segment of a trajectory, obtained via an Euler-Maryuama integration of (4.19), with a time discretisation step $\tau = 0.01$ and the temperature parameter set to $\beta = 2$, is depicted in Fig. 4.10. The change point detection was done with the parameters set to $t_m = t_b =$

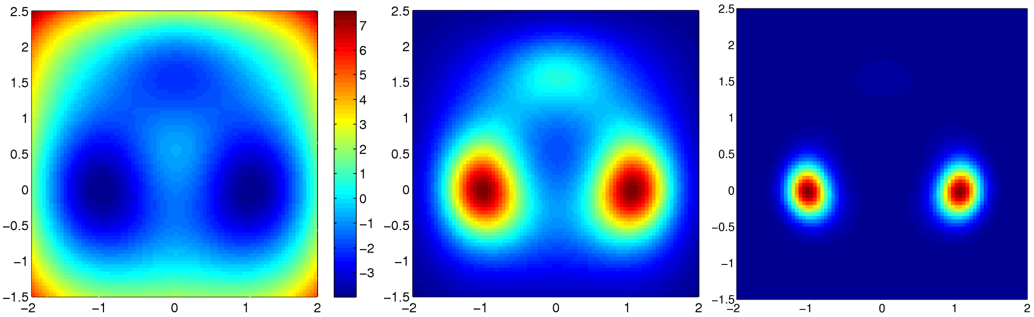


Figure 4.9: *Left:* A surface plot of the threehole potential $V(x, y)$ as given in (4.18). *Middle:* The invariant measure is proportional to $\exp(-\beta V)$, which is plotted here for $\beta = 0.5$ (red marks larger values of the measure while blue corresponds to nearly zero values). *Right:* The invariant measure for a lower temperature ($\beta = 2$).

$t_u = 50, p = 1$ and $a = 0.7$. Note that as hops between potential are very rare events, the testing, as described in § 4.3.3, was restricted to the last 750 data points of the time series for each cycle. Also note that after detection of a change point $\hat{c}$ the detection of a subsequent change point starts at $\hat{c} + t_b$ to allow the trajectory to relax to a new potential well after leaving one.

In order to test if the change point algorithm can be used to obtain meaningful data-based reduced models from time series, a long time simulation for two different temperatures $\beta_1 = 2.4$ (low temperature) and $\beta_2 = 1.2$ (high temperature) is performed until 499 change points are detected in each one of the trajectories. Note that, as β_1 corresponds to a lower temperature than β_2, approximately 10 times more simulation time is needed with β_1 to detect 499 change points than with β_2. The output of the algorithm is, besides the change points, 500 moment matrices $M = (M^{(1)}, \ldots, M^{(500)})$, which we keep instead of the discarded time series segments.

The moment matrix set $\tilde{M}$, obtained by sorting out falsely detected change points with Algorithm 3, is then clustered with the hierarchical clustering approach described in § 4.3.4. After clustering, moment matrices within a cluster are summed together yielding again a reduced set of moment matrices. This clustering and summing of moment matrices is repeated until no pair of moment matrices has a distance smaller than α (which was set to 0.7). Finally, we obtain 20 moment matrices for the low temperature β_1 where over 99% of the overall time series information is contained in three of them. Using the moment matrices to estimate parameters of the corresponding VAR(1) model reveals that the estimated means of the three local SDE's exactly correspond to the location of the three wells in the potential function, cf. Fig. 4.11.

For the higher temperature model with β_2 we end up with 23 moment matrices with over 99% of the overall time series information contained in 12 of them. The fact that more local models are needed to describe the dynamics is of no surprise, since with higher temperature the local quadratic approxima-

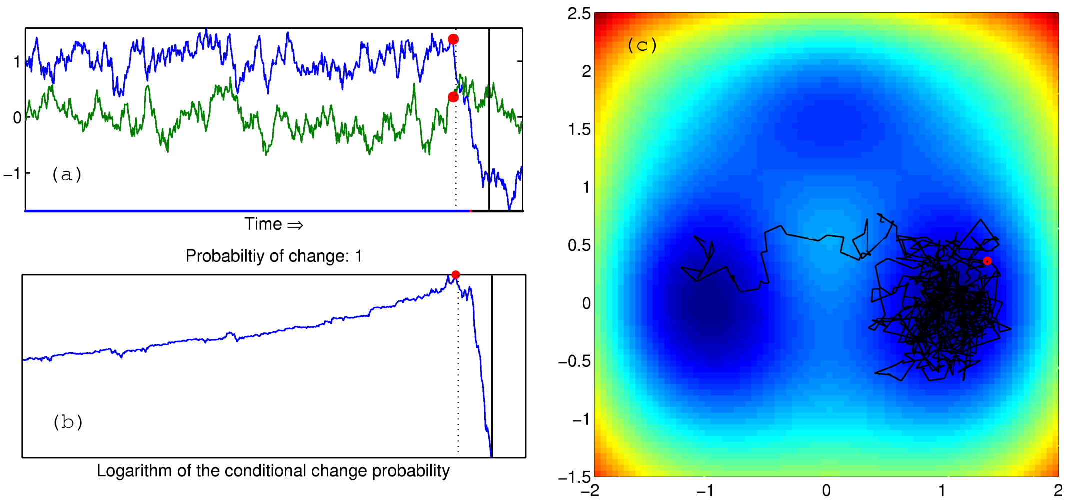
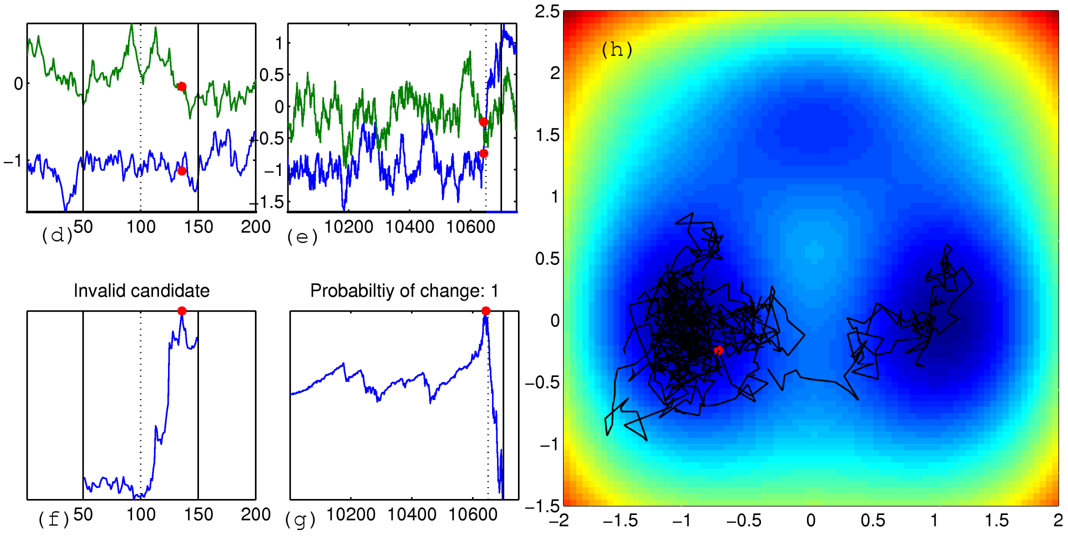


Figure 4.10: *(a)* After moving a long time in the left basin of the potential the trajectory finally hops to the right basin. *(b)* A candidate change point $\hat{c}$ is easily identified by locating the maximum of the conditional probability (red circle = detected change point, black line = right margin defined by t_m, dotted line = puffer zone defined by t_b). *(c)* The diffusion process seen with bird's eye view in the two dimensional potential, $\hat{c}$ is marked with the red circle again. *(d)* The procedure starts again from $\hat{c} + t_b$. At first there is a left and a right margin to our test window. *(e)* The first candidate change point is invalid as it is too close to the right margin. *(f)+(g)* After iterating the algorithm many times a subsequent change point occurs and is detected. *(h)* From the bird's eye view we see that the new change point corresponds to a jump back to the vicinity of the start basin.



tion of the potential function becomes more and more inaccurate and therefore more local models are needed for a reasonable approximation. In both cases it is now straightforward to obtain a local linear model for the dynamical description of the process in the sense of (3.37) as the parameters can be obtained from the moment matrices and the rates of the switching process can be estimated from the detected change points. For comparison we define the following approximate invariant densities of these switching models

$$\pi_i(\boldsymbol{z}) = \sum_{j=1}^{k_i} w_i^{(j)} |2\pi \Sigma_i^{(j)}| \exp\left(\left(\boldsymbol{z} - \boldsymbol{\mu}_i^{(j)} \right)' \left(\Sigma_i^{(j)} \right)^{-1} \left(\boldsymbol{z} - \boldsymbol{\mu}_i^{(j)} \right) \right), \qquad (4.20)$$

with $i = \{1, 2\}$ corresponding to β_1 and β_2, which is a weighted sum of invariant densities of the local VAR models. By $\Sigma_i^{(j)}$ and $\boldsymbol{\mu}_i^{(j)}$ we denote the stationary covariance matrices and means of the local linear models. As shown in § 3.3.2 the mean can be extracted from the estimator of Φ. The stationary covariance matrix can be obtained from the solution[2] of (A.10):

$$\Sigma_i{}^{(j)} \exp(\tau F_i{}^{(j)})' - \exp(\tau F_i{}^{(j)})^{-1} \Sigma_i{}^{(j)} = \exp(\tau F_i{}^{(j)})^{-1} R_i{}^{(j)},$$

where $R_i{}^{(j)}$ is directly estimated and $\exp(\tau F_i{}^{(j)})$ is obtained from $\Phi_i{}^{(j)}$ of the corresponding local model. By $w^{(j)}$ the weights of each model, defined by

$$w_i^{(j)} = \frac{m_i^{(j)}}{\sum_{l=1}^{k_i} m_l^{(l)}},$$

are denoted, where $m_i^{(j)}$ is the upper left entry of the corresponding moment matrix and k_i the number of local models for the corresponding temperature. These densities are approximate as they rely on the assumption that the invariant density of a local model is sampled before the process switches to another local model, but since there is a time scale separation between the jumps in different basins and the diffusive movement, this assumption is justified. These obtained approximate invariant densities are compared to the Boltzmann-Gibbs distributions of the original dynamic in Fig. 4.11. It can be seen that they resemble the major characteristics of the invariant densities at both temperature levels, which can not be done by using a single linear model from all the simulated data points.

[2] An analytical solution is possible in most cases as shown in [106, Appendix A].

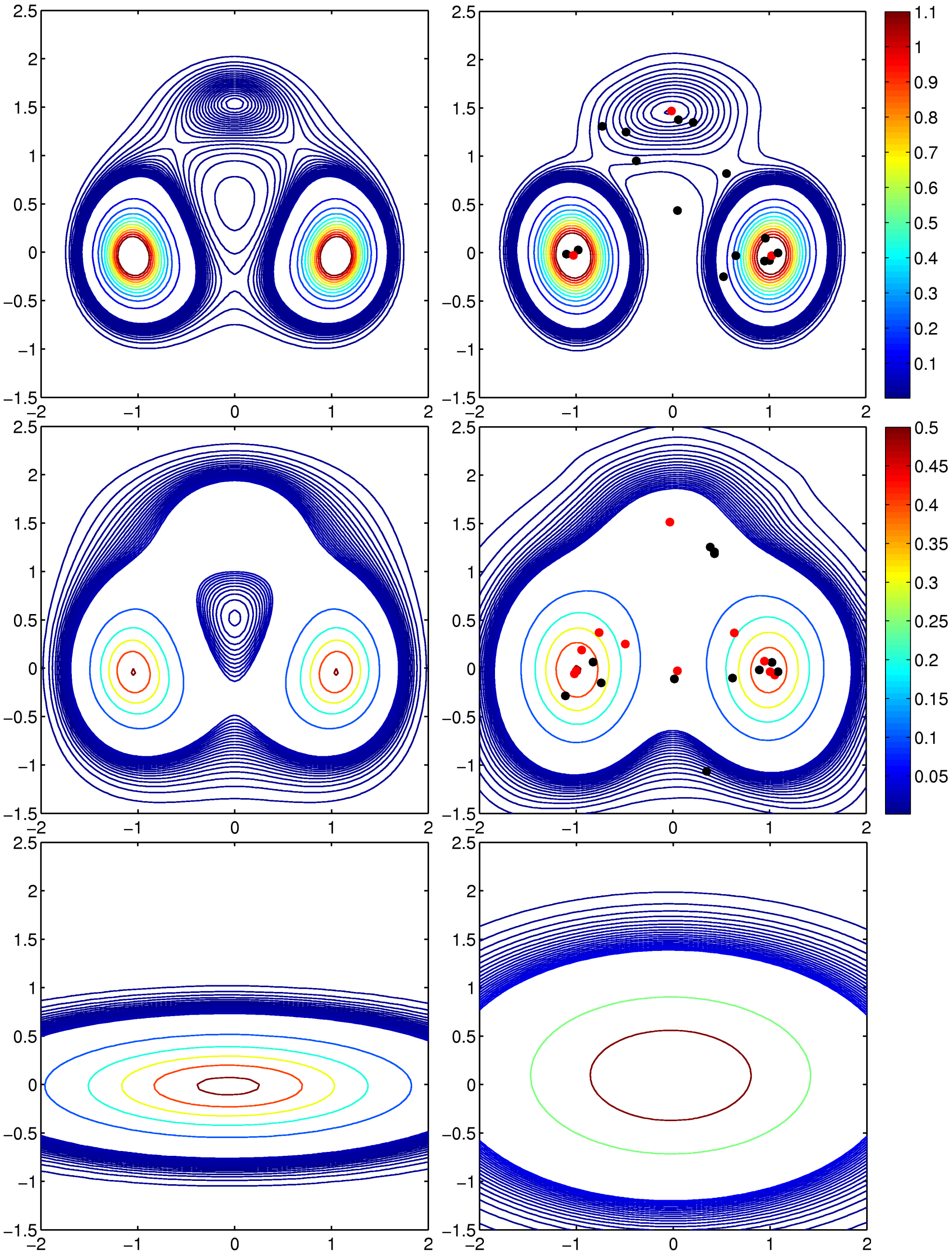


Figure 4.11: *Top (middle) left*: The Boltzmann-Gibbs density of the Threehole potential with $\beta_1 = 2.4$ ($\beta_2 = 1.2$). *Top (middle) right*: The approximated invariant density of the switching model as given in (4.20) for β_1 (β_2). The centrepoints of the used harmonic potentials are plotted as red and black circles, the weights of the linear models corresponding to the red circles would sums up to more than 99%. *Bottom left (right)*: The invariant density obtained by fitting a single linear model to the whole simulated trajectory for β_1 (β_2) with 2898704 (307843) data points.

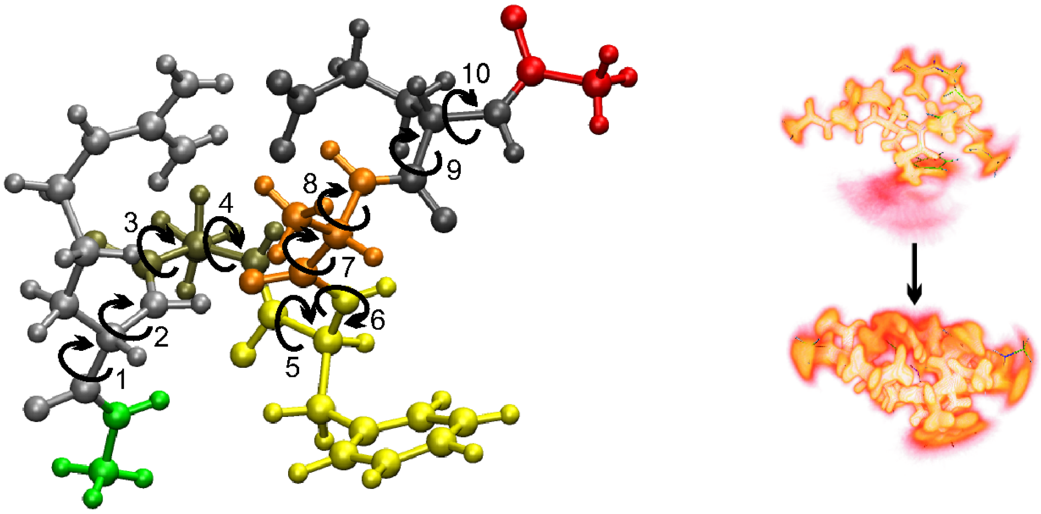


Figure 4.12: *Left:* The simulated penta-peptide with the 10 observed torsional backbone angles marked. *Right:* During the simulation the molecule transforms from a structure where mainly the side chains interact to a more compact and stable structure via several metastable intermediates. The metastable structures at the beginning and at the end of the trajectory are visualised by density plots showing the flexibility within a conformation (Visualisation by AMIRA, [115]).

4.4.3 Penta-alanine

In order to demonstrate the applicability of the precedingly presented algorithm to segment time series in a similar way as the HMM-VAR algorithm does, we present an example from molecular dynamics (MD). We will use simulation data of an artificial penta-peptide, consisting of a capped chain of five amino-acids: glutamine-alanine-phenylalanine-alanine-argenine, shown in Fig. 4.12. The peptide is itself an interesting object to study, as it is a small molecule which is able to form salt bridges, an important and still not well understood matter. We will not concern with this subject but rather use a trajectory of the peptide for demonstration purposes of our algorithm only. The trajectory was obtained from an MD-simulation in vacuum using the NWChem software package [13, 66]. The integration time step was set to 1 femtosecond, while the coordinates were written out every 200 femtoseconds. The trajectory we use consists of 100000 points thus covering a time span of 20 nanoseconds in total. What can be seen in the trajectory is the folding of the peptide from a spread out structure where only the two long side chains interact (the salt bridge) to a more compact and very stable structure, see Fig. 4.12.

Since the dimension of the time series is higher than in our two dimensional example before we choose more conservative parameters, i.e. $t_m = t_u = t_b = 100$, and as before $p = 1$ and $\alpha = 0.7$. Choosing t_m and t_b in a range from 100

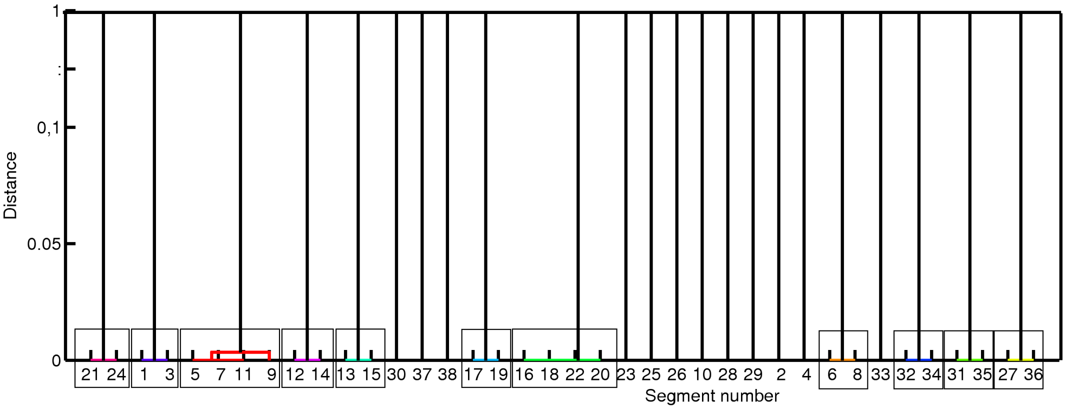


Figure 4.13: In this dendrogram the allocation of the 38 identified time series segments to 23 clusters via hierarchical clustering is shown (marked by colours and boxes). The tree represents the hierarchical cluster distances using the distance measure given in (4.14), however as the inter cluster distances are very close to zero while the intra cluster distances are almost one, it is not very structured.

to 500 does not significantly alter the results, if they are chosen smaller, resp. larger, more, resp. less, change points will be detected, as these determine the resolution of the algorithm, cf. § 4.4.1. Note that since we now deal with circular data the algorithm has to be adjusted such that the actual tested time series segment is shifted to make it quasi non-circular, cf. § 3.4.2. Unfortunately this means that we can not discard the time series data and instead use the moment matrices for post processing, as shifting the time series will alter the moment matrices in a non-reversible way (one could think of various work-arounds, like imposing restrictions on the shifting, i.e. shift the whole time series the same way), but this is no obstacle here as the time series is short enough. The change point algorithm terminates with 37 detected change points. Doing the post processing as described above (with recomputation of the moment matrices), these 38 segments are clustered in 23 clusters, cf. Fig. 4.13. The outcome is depicted in Fig. 4.14 and 4.15 and seem to be quite reasonable. Note that the same analysis with the HMM-VAR algorithm would require much more computational effort and is sensitive to initial conditions.

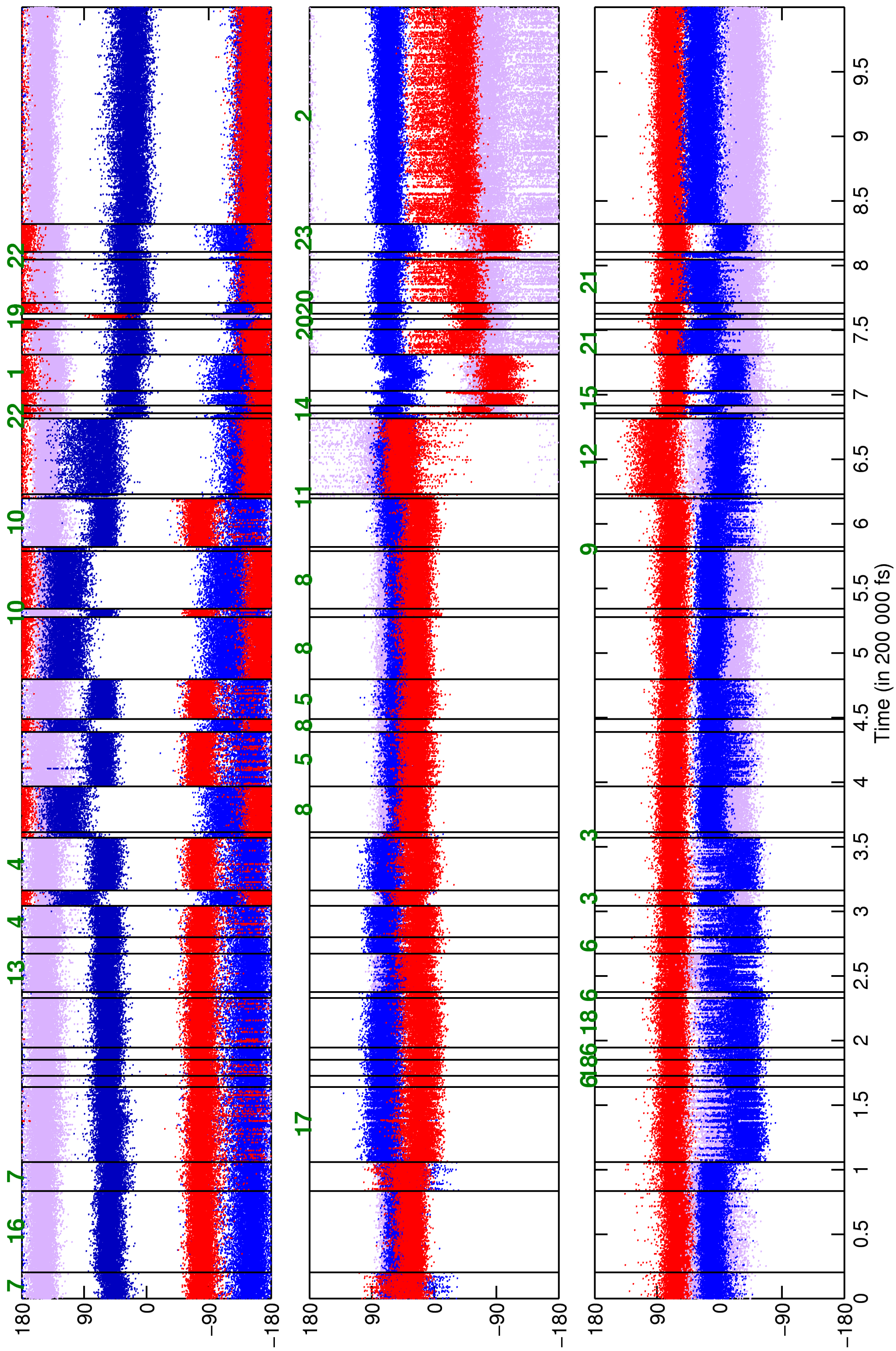


Figure 4.14: The 10-dimensional backbone torsion angle time series of the peptide (splitted in 3 sub panels, Top: dimension 1-4, Middle: 5-7: Bottom: 8-10). The vertical lines mark the detected change points. The digits 1 to 23 over the panels indicate the membership of the segments to the 23 clusters obtained from hierarchical clustering as explained in the text (the digits are distributed over different panels only for reasons of readability).

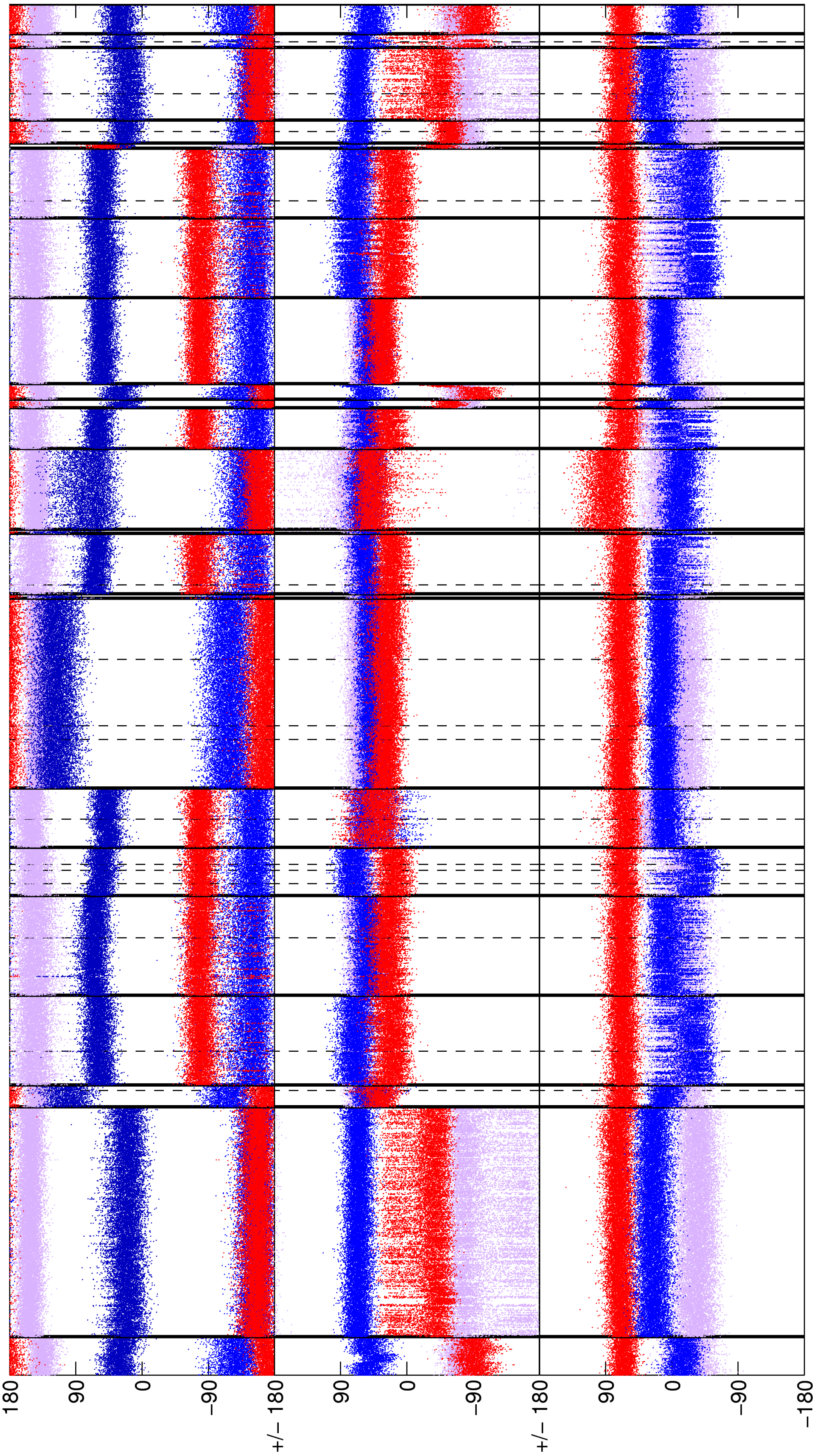


Figure 4.15: Here the obtained time series segments, bordered by dashed lines, are plotted in a permutation such that the ones allocated to the same cluster, bordered by thick lines, are side by side.

5 Computation of Rate Constants

In this chapter we show how to employ the techniques previously described to compute rate constants in molecular systems following some ideas of Art Voter [122, 124]. We begin by describing briefly the very basic idea of transition state theory and the estimation of rate constants, for a more complete presentation the reader is referred to [45, 120]. Afterwards, the approach of Art Voter is presented, who tackles the problem of rate constant computation via direct molecular dynamics simulation, which is only possible by speeding up the transition events, e.g. via distributed computing. This approach depends crucially on the detection of transitions between different potential energy basins *on-line*, which can be done by the on-line change point detection developed above. At last we show how to implement the obtained procedure for a small molecular example.

5.1 Transition State Theory

In principle Transition State Theory (TST) provides a way to compute the (approximate) transition rate of a rare events system without simulating the system under consideration. Assume a dynamical system which can be described by some Langevin or Smoluchowski dynamics, cf. (3.20), i.e.

$$\begin{aligned}\dot{\boldsymbol{q}}(t) &= M^{-1}\boldsymbol{p}(t)\\ \dot{\boldsymbol{p}}(t) &= -\nabla_{\boldsymbol{q}}U(\boldsymbol{q}(t)) - \gamma M^{-1}\boldsymbol{p}(t) + \sigma \boldsymbol{W}(t),\end{aligned}$$

or

$$\gamma\dot{\boldsymbol{q}} = -\nabla_{\boldsymbol{q}}U(\boldsymbol{q}(t)) + \Sigma\dot{\boldsymbol{W}}(t),$$

with $\boldsymbol{q}, \boldsymbol{p} \in \mathbb{R}^d$, such that the fluctuation-dissipation relation $\gamma + \gamma' = \beta\sigma\sigma'$ holds for an arbitrary constant β. As seen in § 3.3.1 the invariant densities of these dynamical systems are given by the Gibbs densities

$$f(\boldsymbol{q}, \boldsymbol{p}) \propto \exp\left(-\beta\left(\frac{1}{2}\boldsymbol{p}'M^{-1}\boldsymbol{p} + U(\boldsymbol{q})\right)\right), \text{ resp. } f(\boldsymbol{q}) \propto \exp\left(-\beta U(\boldsymbol{q})\right).$$

Assume that there are m disjoint subsets $S_1, \ldots, S_m \in \mathbb{R}^d$ of the position space such that the following two conditions hold

(1) With $N_i := \int_{S_i} Z^{-1}\exp(-\beta U(\boldsymbol{q}))d\boldsymbol{q}$, $1 \leq i \leq m$, defined as the invariant weight of each subset, where $Z = \int_{\mathbb{R}^d}\exp(-\beta U(\boldsymbol{q}))d\boldsymbol{q}$ is the normalisation constant,

$$N_1 + N_2 + \cdots N_m \approx 1.$$

(2) If a realisation of the observed process stays within one subset S_i, the waiting time T_i till it reaches another subset can be approximately modeled by an exponential distribution (independently of previous waiting times), i.e.
$$\mathbb{P}[T_i > t] = \exp(-\lambda_i t), \quad \lambda_i > 0.$$

Note that the second condition holds if there are potential barriers between the subsets $S_1, \ldots, S_m$ which are high compared to the stochastic excitation [100, Ch. 5.10], as then the system will stay with high probability long enough in one subset to loose its memory, i.e. to equilibrate locally within some time τ_{corr} before changing to other potential basins at a time larger than τ_{xc} with $\tau_{corr} \ll \tau_{xc}$. This in turn induces a metastable decomposition with respect to the fastest local relaxation time τ_{rel}, as one can embed the subsets $\tilde{S}_1 \supset S_1, \ldots, \tilde{S}_m \supset S_m$ such that $\tilde{S}_1, \ldots, \tilde{S}_m$ define a metastable decomposition of the state space.

For the moment assume that there are only two subsets $a := S_1$ and $b := S_2$, or $a := S_i$ and $b := \cup_{j \neq i} S_j$. Since the time between hops from one set to the other is supposed to be an independent exponential processes the dynamic of the jump process
$$h(t) = \begin{cases} 1 & \text{, if } \boldsymbol{q}(t) \in S_1, \\ 2 & \text{, if } \boldsymbol{q}(t) \in S_2, \end{cases}$$

must be ruled by a Markov jump process [12, Ch. 8]. Therefore the population densities in a and b over time, denoted by

$$n_a(t) := \frac{\mathbb{E}[\chi_a(\boldsymbol{q}(t)]}{\mathbb{E}[\chi_a(\boldsymbol{q}(t))] + \mathbb{E}[\chi_b(\boldsymbol{q}(t))]}, \quad n_b(t) := \frac{\mathbb{E}[\chi_b(\boldsymbol{q}(t))]}{\mathbb{E}[\chi_a(\boldsymbol{q}(t))] + \mathbb{E}[\chi_b(\boldsymbol{q}(t))]}$$

where χ is the characteristic function and the expectation is taken with respect to the propagated initial distribution, are governed by a master equation

$$\begin{aligned} \dot{n}_a(t) &= -\lambda_{a\to b} n_a(t) + \lambda_{b\to a} n_b(t) \\ \dot{n}_b(t) &= -\lambda_{b\to a} n_b(t) + \lambda_{a\to b} n_a(t). \end{aligned} \tag{5.1}$$

To estimate the rates $\lambda_{a\to b}$ and $\lambda_{b\to a}$ one can use that the expectation value of an exponential distributions equals the inverse rate, i.e.

$$k_{a\to b}^{-1} = \mathbb{E}[T_a], \quad k_{b\to a}^{-1} = \mathbb{E}[T_b],$$

where T_a, resp. T_b, is a random variable denoting the exit time from set a, resp. b. Due to ergodicity these expectation values can be estimated from a trajectory, i.e. a realisation of the stochastic process. If N_T^{ab} counts the number of jumps between a and b up to time T and N_a, resp. N_b, are the invariant weights of the sets a, resp. b, we have

$$\mathbb{E}[T_a] = \lim_{T\to\infty} 2\frac{N_a T}{N_T^{ab}} = \frac{2N_a}{\nu}, \quad \mathbb{E}[T_b] = \lim_{T\to\infty} 2\frac{N_b T}{N_T^{ab}} = \frac{2N_b}{\nu},$$

with the jump frequency ν defined as $\lim\limits_{T\to\infty} \frac{N_T^{ab}}{T}$. If N_a and N_b are not known, they can be estimated from the trajectory as the fraction of time spend in set a, resp. b. Therefore, the exit rates and therefore the first order model given in (5.1) can in principle be estimated from a single trajectory. In practice, however, this is infeasible in a direct way for most cases since one needs very long trajectories to estimate the jump frequency between a and b as these jumps are rare events.

A resort is provided by the classical transition state theory which is based upon the insight that the escape rate constants can be approximated by the equilibrium flux through a dividing surface. Therefore, we embed $a \subset A$ and $b \subset B$ and define a *dividing surface* S between A and B such that $A \,\dot{\cup}\, S \,\dot{\cup}\, B = \mathbb{R}^d$. For convenience assume that the dividing surface can be parametrised as the level set of some scalar function s, such that $S = \{\boldsymbol{q} : s(\boldsymbol{q}) = 0\}$ and that $s(\boldsymbol{q}) < 0$ if $\boldsymbol{q} \in A$ and $s(\boldsymbol{q}) > 0$ if $\boldsymbol{q} \in B$. With analogous notation as above we have

$$\lambda_{A\to B}^{-1} = \frac{2N_A}{\nu^{TST}}, \quad \lambda_{B\to A}^{-1} = \frac{2N_B}{\nu^{TST}}, \quad \nu^{TST} = \lim_{T\to\infty} \frac{N_T^{AB}}{T},$$

where $N_A \approx N_a$ and $N_B \approx N_b$ as $N_a + N_b \approx 1$. To evaluate the frequency ν^{TST} note that with the use of the heavyside function

$$H(x) = \begin{cases} 1, \text{ if } x > 0, \\ 0, \text{ if } x \leq 0, \end{cases},$$

the characteristic function of the sets A and B can be written as

$$\chi_A = H(s(\boldsymbol{q})), \quad \chi_B = H(-s(\boldsymbol{q})).$$

This allows to express ν^{TST} as

$$\begin{aligned} \nu^{TST} &= \lim_{T\to\infty} \frac{N_T^{AB}}{T} \\ &= \lim_{T\to\infty} \frac{1}{T} \int_0^T \left| \frac{d}{dt} H(s(\boldsymbol{q}(t))) \right| dt \\ &= \lim_{T\to\infty} \frac{1}{T} \int_0^T \left| \dot{\boldsymbol{q}}(t) \cdot \nabla_{\boldsymbol{q}} s(\boldsymbol{q}(t)) \right| \delta(s(\boldsymbol{q}(t))) dt \\ &= \int_{\mathbb{R}^d \times \mathbb{R}^d} |M^{-1} \boldsymbol{p} \nabla_{\boldsymbol{q}} s(\boldsymbol{q}(t))| \delta(s(\boldsymbol{q})) f(\boldsymbol{q}, \boldsymbol{p}) d\boldsymbol{q} d\boldsymbol{p}, \end{aligned}$$

where ergodicity was used in the last step. This integral can be evaluated by sampling techniques, but unfortunately the obtained transition state frequency ν^{TST} might overestimate the true frequency ν significantly due to recrossings, i.e. due to the fact that not every trajectory crossing the dividing surface S will reach the other basin before recrossing it, which might happen many times. Therefore we have

$$\nu^{TST} \geq \nu.$$

An obvious way to improve the estimate for the jump frequency ν is to choose the dividing surface that minimises ν^{TST}, the so called *variational TST* approach [120]. But obviously optimisation is only feasible within specific classes of dividing surfaces like planar surfaces. Furthermore there is no straightforward extension to evaluate an exit rate to a particular state, e.g. $\lambda_{i\to j}$ if there are multiple states.

Another strategy to improve ν^{TST} is to find a *dynamical correction factor c* such that $c\nu^{TST}$ is closer to ν. There are different strategies to compute this correction factor, either one estimates a time dependent exit rate and identifies a quasi constant plateau value which it takes on a timescale between τ_{corr} and τ_{rx} [18], or one starts trajectories from the dividing surface and computes the correction factor from the fraction that reaches another (specified) set after τ_{corr} [123]. But also the computation of the correction factor needs specification of an appropriate dividing surface, i.e. enclosing a potential energy basin in a dynamically meaningful way, as otherwise its sampling has to be very extensive.

5.2 Estimation from Time Series

We have seen in the preceding section that the need for extensive simulations to compute exit rates can be circumvented by evaluation of integrals in phase space. However, specification of a suitable dividing surface is in general not a trivial task and integration in phase space can be limited in high dimensional systems. Therefore Art Voter propagated an approach to speed up simulations via distributed computing such that rare events can be observed in a reasonable time [122,124]. The approach is based upon the observation that if $T_1, T_2, \dots T_N$ are independent exponential distributed random variables with rates $\lambda_1, \lambda_2, \dots, \lambda_N$, i.e.

$$\mathbb{P}[T_i > t] = \exp(-\lambda_i t),$$

then $T_{min} := \min(T_1, T_2, \dots, T_N)$ is exponential distributed as well with rate $\lambda_1 + \lambda_2 + \cdots + \lambda_N$. Therefore, if all rates are identical, i.e. $\lambda_1 = \lambda_2 = \cdots = \lambda_N = \lambda$, T_{min} is exponential distributed with rate $N\lambda$. Adapted to the problem of reaction rate estimation of a molecular system this means that if one starts N uncorrelated trajectories within the same potential energy basin the expected time to observe an exit in any one of these trajectories is equal to

$$\mathbb{E}[T_{min}] = \frac{1}{N\lambda}. \tag{5.2}$$

The crucial point is that uncorrelated trajectories can be simulated on different processors, e.g. via distributed computing, such that even if the computational effort is not lowered, the expected wall clock time to observe a rare event, i.e. the exit from a start basin, is speeded up by a factor of N, while the dynamical properties of the system are exactly retained. Due to immense increase of computer power and the use of global networks like Folding at Home [112] the factor N can lead to an enormous increase. It makes no difference if the computer clock on the different processors is not equal, i.e. if there are faster and slower processors, as we have

$$\mathbb{P}[T_1 > t_1, T_2 > t_2, \dots, T_N > t_N] = \prod_{i=1}^{N} \exp(-\lambda t_i) = \exp(-\lambda(t_1 + \cdots + t_N)),$$

as long as $T_1, T_2, \dots, T_N$ are uncorrelated. That means that the effect of simulating uncorrelated trajectories can be interpreted in two ways, as an increasing of the exit rate (5.2), or as a (virtual) increase of the time elapsed, since the computed time on all processors sums up. Therefore the exit rate estimate from a single observed exit time would be

$$\hat{\lambda} = \frac{1}{T_x}, \tag{5.3}$$

where T_x is the simulation time elapsed on all processors until the first exit event is detected on one of them. This naturally leads to an estimator based on k observed exit events $T_x^{(1)}, \dots, T_x^{(k)}$ and corresponding rate estimators $\hat{\lambda}_1, \dots, \hat{\lambda}_k$:

$$\hat{\lambda} = \frac{1}{k} \sum_{i=1}^{k} \hat{\lambda}_i. \tag{5.4}$$

Based on this the procedure advocated by Art Voter is the following

- Start simulating N trajectories of the system within the same potential basin, i.e. metastable state, but with uncorrelated initial conditions, on N different processors.

- After some predefined time t_u check if one of the trajectories left the potential basin. If not proceed with simulation and repeat this step until such exit has been detected.

- Propagate the trajectory in which the exit occurred another time span τ_{corr} to detect recrossings. If there is a recrossing continue simulating all trajectories as before, otherwise proceed to the next step.

- Compute $\hat{\lambda}_i$ as in (5.3) and use the end configuration of the trajectory which left the basin to generate N uncorrelated copies of the system within the new potential basin and start the procedure from the beginning.

After some iterations of this procedure the transition rates from one metastable state to another can be estimated by (5.4).

In the next paragraphs we are going to comment on three obvious hurdles one has to overcome to employ the suggested procedure:

a) How to detect an exit from a potential basin?

b) How to create uncorrelated copies from a configuration?

c) How to detect in which basin the process currently is, resp. in which basin it jumps?

A more detailed description of a possible implementation of the herein suggested solutions (wrt. to a molecular example) is done in § 5.3.

How to detect an exit from a potential basin? In the easiest case the reaction coordinate as well as its mapping to different metastable states is known. For example, if one is interested in folding and unfolding of proteins this could be some distance between the end groups of the protein such that a shorter distance means a folded and a larger distance an unfolded state, like it is also used in experimental techniques [103]. Another feasible case is a smooth potential energy surface, in this case a simple gradient descent procedure can be used to test if the basin has been left. This is exploited in the work of Art Voter where diffusion on metal surfaces is investigated [122, 124].

However, energy surfaces in the biomolecular context are in general very rough, such that a simple gradient descent algorithm would end up in different local minima even if the overall basin is the same. A different approach applied for molecular systems in MD simulations by Zagrovic et al. [128] is to detect conformational changes by monitoring the variance of the potential energy and defining a change if the variance grows rapidly. This approach is based upon the reasoning that to jump from one basin to another a high energy barriers must be crossed, while within the basins the fluctuations in the potential energy will be much lower. However, there are three potential pitfalls with this approach, (1) it is not always the case that potential energy excitation corresponds to a jump to another basin, (2) the problem of recrossings is not treated, (3) since transitions between different basins are, in general, fast events configurations with high potential energy may not show up in the trajectory if the time lag between successive data points is too large.

An alternative way to detect changes is provided by the on-line change point detection algorithm developed in § 4. This is tempting as stochastic differential equations, e.g the Langevin dynamic, are a quite natural dynamical description for reduced (reaction) coordinates [68]. Furthermore as shown in § 4.4.1, the roughness of the potential energy surface can be approximated in a dynamical sense by harmonic potentials, cf. [54]. The change point detection algorithm can be implemented efficiently in a distributed computing setting, as only moment matrices, instead of trajectories, need to be exchanged between processors to exchange information about the models.

How to create uncorrelated copies from the system? An easy way to get uncorrelated copies of a system within a basin is to create a number of copies of the coordinates, randomly draw momenta from the Boltzmann distribution

$$f_B(\boldsymbol{p}) \propto \exp\left(-\beta\left(\frac{1}{2}\boldsymbol{p}'M^{-1}\boldsymbol{p}\right)\right), \tag{5.5}$$

where β is chosen according to the given temperature, for each of the copies and propagate them in phase space for some time which is approximately τ_{corr}. Finally the end points of the obtained trajectories can be used as uncorrelated copies of the system. This can be refined by more sophisticated techniques like short hybrid Monte Carlo schemes [34,35,79], see the example in § 5.3 for further explanations, to get a (locally) representative sample of points within a basin. However, one has to check if all obtained copies are still within the same basin, since during sampling it is possible, even if unlikely, that an exit event took place. A way to do this is to generate short trajectories from the obtained points in phase space and check with the clustering techniques suggested in § 4.3.4 if they belong to the same basin and discard all points for which this seems not to be the case (again further explanations are given in the next section).

How to identify different basins? The identification of different or allocation to identical basins of trajectory pieces before and after a change point can be done by the post processing techniques described in § 4.3.4, i.e. by clustering the moment matrices obtained from trajectory pieces between detected change points with, e.g., the hierarchical clustering algorithm.

5.3 The Alanine-Dipeptide

As an illustration how to employ the change point detection algorithm to compute exit rates we use a small molecular system, the alanine-dipeptide, depicted in Fig. 5.1. Its essential dynamics can be described by the two backbone torsion angles Φ and Ψ. Stabilised by the dipole of the peptide bonds, the alanine-dipeptide will take a stable conformation in vacuum with the central atoms $O - C - N - C_\alpha - C - N - H$ forming a ring, a so called C_7 conformation (as 7 atoms are forming the ring). By rotation of the dihedral angles Φ, Ψ the formation of this ring is possible in two ways. We call this two ring conformation A and B, corresponding approximately to the two regions $[-100, -45] \times [70, 120]$ and $[45, 90] \times [-90, 20]$ in (Φ, Ψ) dihedral angle space. These two conformations, however, are not equivalently favourable due to interference of the ring with the side chain carbon atom in conformation

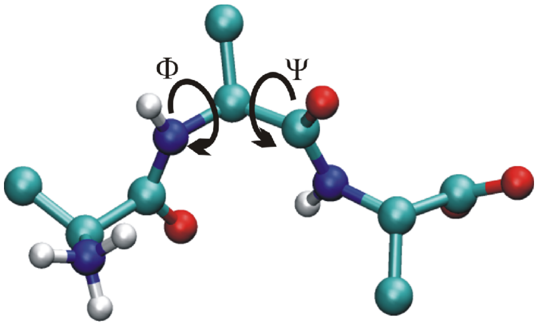


Figure 5.1: The Alanine-Dipeptide in united atoms representation, i.e. some hydrogen atoms are not shown as they are not treated explicitly in the used force field. The conformational dynamic can be described by the two torsion angles shown.

B. The conformations A and B are metastable, since at room temperature the molecule will flip between them on the nanosecond scale, which is very slow compared to the other molecular motions like bond vibrations on the femtosecond scale.

To get a detailed picture we performed a long time simulation over 600 nanoseconds of the molecule coupled the Nosé Hoover thermostat at 300 Kelvin and extracted the torsion angles every 2 picoseconds. From these we obtained a histogram in torsion angle space approximates the projected equilibrium density, see Fig. 5.2. First, it can be clearly seen that indeed conformation A is preferred to B, second, we see that the projected invariant density is not uni-modal in conformation A, as there is a second peak. This peak belongs to a so-called C_5 conformation, a ring formed of the five central atoms $H-N-C_\alpha-C-O$, which, however, is not (meta)stable at a temperature of 300 Kelvin, but flips rapidly back to the C_7 conformation again (note that exchanging the central alanine residuum against glycine would make this conformation stable at room temperature [78]).

Our aim is the computation of the exit rates $\lambda_{A\to B}$ and $\lambda_{B\to A}$ using distributed computing as outlined above. In the next paragraphs the procedure is explained stepwise. As demonstrated in § 4.4.1 the existence of a subconformation in state A should not be a problem as long as on the time scale we perform our analysis it is not metastable. The implementation of the molecular dynamic simulation needed is done in GROMACS, we do not explicate how to set up the simulations in detail, but just remark that the communication between GROMACS and the control and analysing programs is done file-based, i.e. saying that we have a molecular configuration to start with means that we can create a file with the positional coordinates that can be read in by GROMACS to specify the initial conditions. Note also that we have shifted the dihedral angle space from $[-180, 180[^2$ to $[-225, 135[\times[-145, 215[$ to circumvent problems with periodicity.

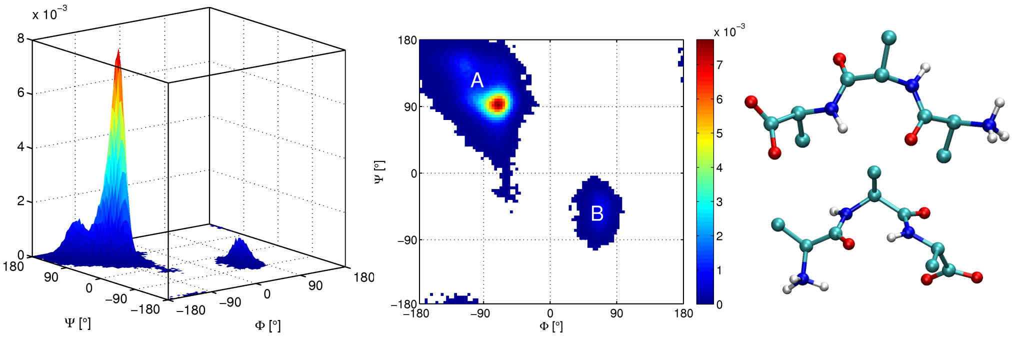


Figure 5.2: *Left:* A histogram of the sampled distribution in torsion angles space of alanine-dipeptide in vacuum obtained from a long time simulation. *Middle:* Bird's eye view, the two metastable conformations are marked with A and B. *Right:* Two configurations belonging to conformation A (top) and B (bottom).

Step 1: Generation of uncorrelated copies. Having a start configuration $\boldsymbol{q}_0$ located in a local basin of the potential energy surface, a set of uncorrelated copies is created by a hybrid Monte Carlo (HMC) scheme. Therefore, N copies of the start configuration $\boldsymbol{q}_0^{(1)}, \dots, \boldsymbol{q}_0^{(N)}$ are created and sent to N different processors. On each of the processors a short HMC simulation is performed, that is, for each start configuration $\boldsymbol{q}_0^{(i)}$ momentas $\boldsymbol{p}_0^{(i)}$ are drawn from the Boltzmann distribution (5.5) with a temperature parameter β corresponding to 300 Kelvin. The initial conditions $(\boldsymbol{q}_0^{(i)}, \boldsymbol{p}_0^{(i)})$ are then propagated according to the Hamiltonian equations of motion for a time span T_{HMC} yielding candidate configurations $(\boldsymbol{q}_*^{(i)}, \boldsymbol{p}_*^{(i)})$. This candidate configuration is accepted with probability

$$p_{ac} = \min\{1, \exp(-\beta(H((\boldsymbol{q}_*^{(i)}, \boldsymbol{q}_*^{(i)})) - H(\boldsymbol{q}_0^{(i)}, \boldsymbol{q}_0^{(i)})))\},$$

where H denotes the total energy as in (2.2), otherwise $(\boldsymbol{q}_0^{(i)}, \boldsymbol{p}_0^{(i)})$ are kept. If the integrator used to generate candidate configurations is volume preserving [72,79] the ensemble generated by iteration of this scheme will approximate the canonical ensemble. Since we do not want to sample the canonical ensemble but only to obtain uncorrelated copies in the vicinity of the starting configuration we do this iteration only a few times and finally keep the last accepted configuration on each processor. In the given example we used $N = 15$ copies, a propagation time for the HMC scheme of $T_{HMC} = 1$ picosecond and 10 iterations on every processor.

Step 2: Selection of start points from the uncorrelated copies. Even though only a short HMC-sampling was performed to obtain start configurations $(\boldsymbol{q}_0^{(i)}, \boldsymbol{p}_0^{(i)})$, $1 \leq i \leq N$, it is possible that either the local potential energy well has been left during sampling and therefore an invalid starting configuration has been generated. As this is a sensitive point of the algorithm, i.e. one has to make sure all uncorrelated copies are within the same metastable conformation, we first sort out configurations which may have left the corresponding potential energy basin.

This is done automatically by independently propagating the obtained starting configurations according to the equations of motions coupled to a stochastic thermostat for some time which yields a collection of two dimensional time series (here the thermostat was set to 300 Kelvin, with a discretisation time step of 1 femtosecond and a total integration time of 90 picoseconds; extraction of the torsion angles every 300 steps gives 300 data points per time series). After shifting the angles as specified above to remove periodicity, large distances between subsequent data points are marked and the moment matrices, assuming a VAR(1) dynamic, are computed, excluding the marked transitions as described in § 3.4.2. Note that again these steps can all be done independently on different processors. The obtained moment matrices are collected to one of the processors and clustered as described in § 4.3.4 by hierarchical clustering with complete linkage and a (very conservative) cut-off criterion of 0.3 (which means that if the probability of a "change" between two moment matrices is higher than 30% they do not belong to the same cluster). Now we proceed only with the start configurations which belong to the biggest obtained cluster.

The described procedure will ensure that only start configurations within the same potential well are taken. Before proceeding to the next step all moment matrices in the biggest cluster are added up yielding a moment matrix M_I holding our (prior) knowledge of the dynamical behaviour in the actual potential well. This moment matrix and the end configurations of the simulations are handled over to the next step.

Step 3: Sequential change point detection. Next, simulation of the selected start points proceeds on the different processors again, and Alg. 2 is used to detect change points. That is, iteratively $t_u = 300$ (update window) new data points on each processor are obtained by proceeding the simulation and, after shifting and exclusion of large transitions, tested for a change point with $t_b = 40$ (buffer zone) and $t_m = 100$ (right margin, cf. § 4.3.1) and a threshold value of $\alpha = 0.8$ (setting it down to 0.6 does not significantly change the behaviour of the algorithm). Note that we do not need to specify a left margin and therefore we do not have a minimal segment size since prior information is already given in matrix M_I. If a change point is detected the simulations on all processors are stopped, a moment matrix is obtained by adding up all information generated on the different processors up to the time of the detected change point to M_I, and the simulation time till the change point is stored. If no change point is detected 300 new data points are generated on each processor and the test is repeated.

Note that we set the maximal length of the test window to detect a change point to 1500 data points to control the computational effort. If one of the simulated time series exceeds this length, a part from the beginning of the time series is cut out and the information content in this part is added to the moment matrix M_I which is communicated to all processors.

If a change point has been detected, the configuration in the corresponding trajectory (which is 50 steps after the detected change points to allow for relaxation into a new well) is extracted and taken as the new start configuration to use in Step 1 again.

Step 4: Post processing. The steps 1-3 were repeated until 400 change points were detected. This corresponds to a simulation time, distributed upon 15 processors, of over 2.2 microseconds. The outcome of the procedure was 400 moment matrices and samples of 400 exit times. First it was tried to exclude falsely detected change points, which may correspond to recrossing events, by usage of Alg. 3, reducing the number of detected change points to 260. Then the merged moment matrices are clustered with hierarchical clustering (with single linkage and a cutoff criteria of 0.8), yielding 4 clusters. Two of the clusters are marginal as they contain only a single moment matrix and less than 0.1 % of the total generated information, therefore they were discarded. Since we knew for this example which regions in Ramachandran space correspond to the expected two metastable states A and B, we accumulated a statistics during the simulations, storing from which region the information compressed in the moment matrices came. So it was possible to check afterwards that each of the two remaining clusters indeed consists exclusively of moment matrices containing only information from the same state A, resp. B.

Using the obtained samples of exit times from A to B and B to A the rates $\lambda_{A\to B}$ and $\lambda_{B\to A}$ can be estimated by (5.4), which yields

$$\lambda_{A\to B} = 0.078 \times 10^{-3}, \text{ and } \lambda_{B\to A} = 1.127 \times 10^{-3},$$

with respect to a picosecond scale. In Fig. 5.3 we compared the cumulative distribution function of the exponential distribution with the estimated rates, i.e.

$$F(t) := \mathbb{P}[T \leq t] = 1 - \exp(-\lambda t)$$

for a exponential distributed random variable T with rate λ, to the empirical distribution function obtained from our statistic of exit times for each state, i.e.

$$F_n(t) = \frac{\text{number of exit times} \leq t}{n},$$

where n is the total number of exit times. The figure shows a very good agreement between these two distribution functions.

With these rates a reduced model can be set up where the rates specify the switching between the two local models obtained. In Fig. 5.4 we see the (approximate) invariant density, based upon the parameter estimates from the two moment-matrices, obtained by summing up all moment matrices belonging to the same cluster. Comparison with Fig. 5.2 reveals that this approximates the invariant density of the original system well and that the multi-modal local invariant density of state A is approximated by a single (wider) Gaussian shape.

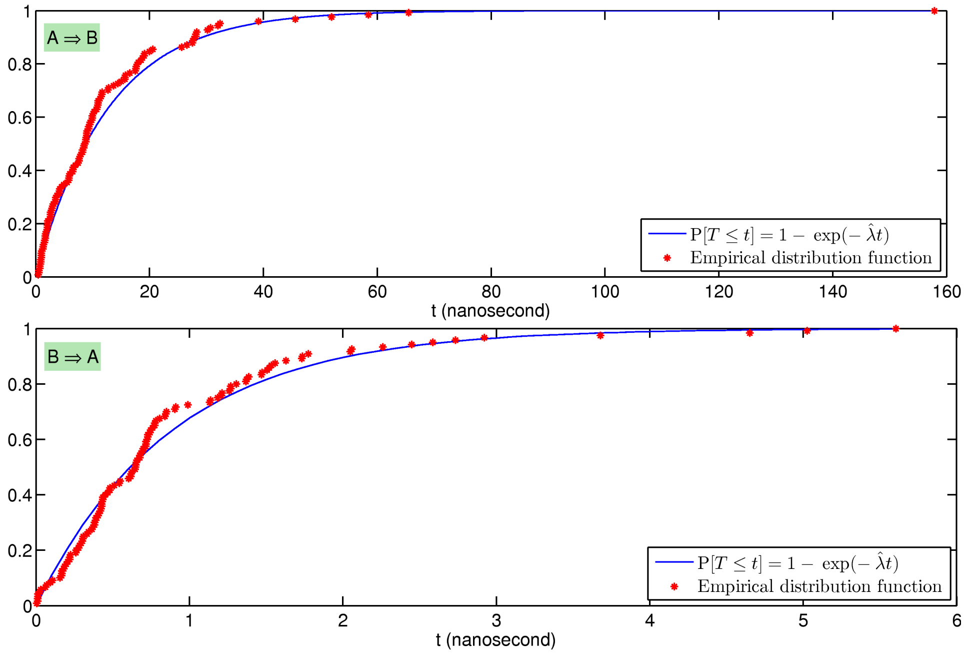


Figure 5.3: The distribution functions of the exit time T, i.e. $\mathbb{P}[T \le t]$, from metastable states A (*top*) and B (*bottom*). The blue line depicts the distribution function of the exponential distribution F corresponding to the estimated rate, the red stars mark the empirical distribution function F_n.

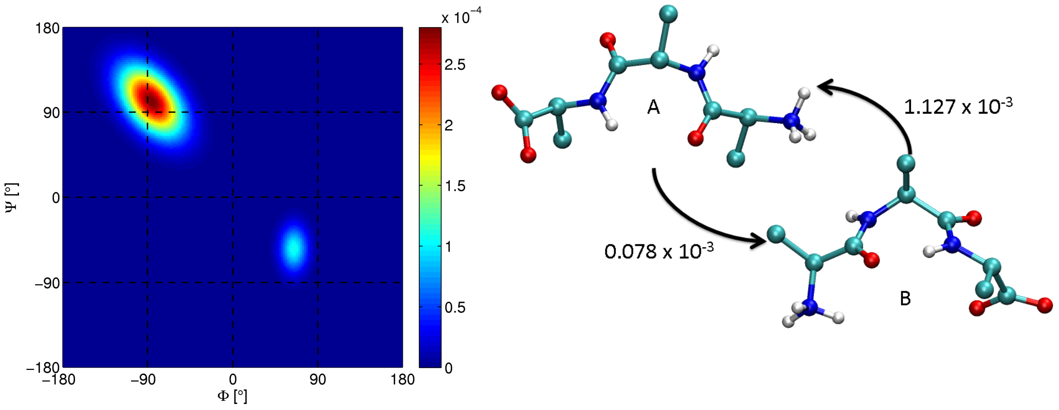


Figure 5.4: *Left:* The (approximate) invariant density of the reduced model, where A and B are approximated by harmonic potentials. *Right:* With the estimated rates (given beside the arrows) a Markov switching model between A and B can be set up.

6 Conclusions

In this thesis we provided a consistent framework for the data analysis of time series exhibiting a complex dynamical behaviour. We showed that, while Markov chains are a natural choice to model the change of dynamical phases, vector autoregressive (VAR) processes provide a convincing model for the flexibility within a dynamical phase. They arise naturally from the discretisation of stochastic differential equations, allow to include non-Markovian effects and can be used to unify several hidden Markov model (HMM) variants. A combination of a Markov model for the change of dynamical phases with VAR processes for the modelling of internal flexibility yielded into a procedure which we named HMM-VAR. We demonstrated how to combine HMM-VAR with Perron cluster cluster analysis (PCCA) to analyse time series from complex systems.

Furthermore, we developed an algorithm to detect dynamical changes in a time series on-line, i.e. reading the data sequently. Application of so-called objective Bayes techniques provided a change point detection procedure which is (i) sampling free, as all needed integrals can be solved analytically, (ii) applicable to high-dimensional time series and (iii) computationally cheap.

It turned out, that the central object of our analysis is the so-called moment matrix, since it does not only allow a stable computation of the estimators for parameters of a VAR process, the compression of information contained in a time series, i.e. assuming a VAR(p) process the information of a $T \times d$ time series is coded in a matrix of dimension $d(p+1)+1$ only, and combination of information belonging to different time series by summing up their moment matrices, and therefore allowing efficient implementation of all algorithms presented here, but also a way to cluster obtained time series segments to the same dynamical phases without the computational effort of an HMM procedure.

Finally, we demonstrated how to apply the change point detection algorithm within a rather complex computational setting to compute rate constants for a small biomolecular system with the help of distributed computing.

In this work, we very much tried to avoid the usage of application specific knowledge in the suggested procedures, which is in fact somewhat opposed to the philosophy of the Bayesian approaches employed here. Therefore, with respect to the conformational analysis of biomolecules, an interesting question for future investigations could be how to include knowledge about the statistical distribution of backbone torsion angles in peptides and proteins, see e.g. [81], via suitable prior distributions.

Appendices

A.1 Linear Stochastic Differential Equations

In this appendix the most important results about stochastic differential equations (SDE) with emphasis on linear stochastic differential equations, as stated e.g. in [4, 38, 90, 100], are collected. The construction of stochastic integrals is not treated the interested reader is referred to the excellent introduction of Ludwig Arnold [4]. An SDE is a stochastic process $\boldsymbol{z}(t) \in \mathbb{R}^d$, $t \in [t_0, T]$, which obeys the following *symbolic* equation

$$d\boldsymbol{z}(t) \quad = A(\boldsymbol{z}(t), t)dt + \Sigma(\boldsymbol{z}(t), t)d\boldsymbol{W}(t), \tag{A.1}$$

or the corresponding integral form

$$\boldsymbol{z}(t) = \boldsymbol{z}(t_0) + \int_{t_0}^{t} A(\boldsymbol{z}(s), s)ds + \int_{t_0}^{t} \Sigma(\boldsymbol{z}(s), s)d\boldsymbol{W}(s),$$

where $A : \mathbb{R}^d \times [t_0, T] \mapsto \mathbb{R}^d$ is called the drift coefficient and $\Sigma : \mathbb{R}^d \times [t_0, T] \mapsto \mathbb{R}^{d \times m}$ the diffusion coefficient. By $\boldsymbol{W}(t)$ an m-dimensional Wiener process is denoted. The term

$$\int_{t_0}^{t} \Sigma(\boldsymbol{z}(s), s)d\boldsymbol{W}(s)$$

is a *stochastic integral* which can be defined rigorously by means of Itô's theory. A stochastic process $\boldsymbol{z}(t)$ is called a solution of the SDE (A.1) on an interval $[t_0, T]$ if it is measurable wrt. to the sigma algebra generated by the Wiener process and the initial value $\boldsymbol{c} := \boldsymbol{z}(t_0)$ and if Eq. (A.1) is fulfilled with probability 1 for every $t \in [t_0, T]$. A solution $\boldsymbol{z}(t)$ of Eq. (A.1) is called unique, if it is continuous and if for any other continuous solution $\boldsymbol{y}(t)$ with the same initial value $\boldsymbol{c}$

$$\mathbb{P}\left[\sup_{t_0 \le t \le T} |\boldsymbol{z}(t) - \boldsymbol{y}(t)| > 0\right] = 0$$

holds. The existence and uniqueness of solutions of a given SDE is stated in the following fundamental theorem:

Theorem A.1.1. *If, with the notations introduced above, A and Σ are measurable functions and there exist two constants $K_1, K_2 \in \mathbb{R}^+$ such that the following conditions hold:*

*a) (*Lipschitz condition*) For all $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$ and $t \in [t_0, T]$*

$$|A(\boldsymbol{x}, t) - A(\boldsymbol{y}, t)| + |\Sigma(\boldsymbol{x}, t) - \Sigma(\boldsymbol{y}, t)| \le K_1|\boldsymbol{x} - \boldsymbol{y}| \le K_1|\boldsymbol{x} - \boldsymbol{y}|.$$

*b) (*Growth condition*) For all $\boldsymbol{x} \in \mathbb{R}^d$ and $t \in [t_0, T]$*

$$|A(\boldsymbol{x}, t)|^2 + |\Sigma(\boldsymbol{x}, t)|^2 \le K_2^2(1 + |\boldsymbol{x}|).$$

If additionally $\boldsymbol{z}(t_0)$ is independent of the Wiener process $\boldsymbol{W}(t)$ and has a finite second moment, then there exist a unique and continuous solution of (A.1) *with initial condition $\boldsymbol{c} := \boldsymbol{z}(t_0)$.*

For the probability density of the solution of an SDE $p(\boldsymbol{z}, t) \equiv p(\boldsymbol{z}(t), t|\boldsymbol{z}(t_0), t_0)$, which is conditional on the initial conditions, a most important relation, the *Fokker-Planck equation*, holds:

$$\partial_t p = -\sum_i \partial_i [A_i(\boldsymbol{z}, t)p] + \frac{1}{2}\sum_{i,j} \partial_i \partial_j ([\Sigma(\boldsymbol{z}, t)\Sigma(\boldsymbol{z}, t)']_{i,j} p). \tag{A.2}$$

In other words, the progression of the (realisation independent) probability density function in time can be described by a known partial differential equation. This relationship can be exploited for e.g. the computation of an invariant measure, i.e. a stationary distribution, by equating the left hand side of (A.2) to zero.

Under a linear SDE in a narrow sense we understand a stochastic process $\boldsymbol{z}(t) \in \mathbb{R}^d$ which is specified by an SDE with

$$\begin{aligned} A(\boldsymbol{z}(t), t) &\equiv A(t)\boldsymbol{z}(t) + \boldsymbol{a}(t) \\ \Sigma(\boldsymbol{z}(t), t) &\equiv \mathit{\Sigma}(t), \end{aligned}$$

where A and $\mathit{\Sigma}$ are matrices and $\boldsymbol{a}$ a vector which only depend on time, so that we have

$$d\boldsymbol{z}(t) = A(t)\boldsymbol{z}(t) + \boldsymbol{a}(t)dt + \mathit{\Sigma}(t)d\boldsymbol{W}(t). \tag{A.3}$$

For linear SDE's the application of the so-called *Itô formula* reveals the nature of its solutions more precisely:

Theorem A.1.2. *A linear stochastic differential equation* (A.3) *has, for every initial value $\boldsymbol{z}(t_0) = \boldsymbol{c}$ which is independent of $\boldsymbol{W}(t) - \boldsymbol{W}(t_0)$, $t \geq t_0$, a unique solution in the interval $[t_0, T]$ provided that the functions $A(t), \boldsymbol{a}(t), \mathit{\Sigma}(t)$ are measurable and bounded on that interval. The solution is a Markov process and is given by*

$$\boldsymbol{z}(t) = \mathit{\Phi}(t)\left(\boldsymbol{c} + \int_{t_0}^t \mathit{\Phi}(s)^{-1}\boldsymbol{a}(s)ds + \int_{t_0}^t \mathit{\Phi}(s)^{-1}\mathit{\Sigma}(s)d\boldsymbol{W}(s)\right), \tag{A.4}$$

where $\mathit{\Phi}(t)$ is the (fundamental) solution of $\frac{d}{dt}\mathit{\Phi}(t) = A(t)\mathit{\Phi}(t)$ with $\mathit{\Phi}(t_0) = I$ (the identity matrix).

Note that if $A(t) \equiv A$, we have $\mathit{\Phi}(t) = \exp((t - t_0)A)$ and therefore

$$\boldsymbol{z}(t) = \exp((t - t_0)A)\boldsymbol{c} + \int_{t_0}^t \exp(-(s - t_0)A)\left(\boldsymbol{a}(s)ds + \mathit{\Sigma}(s)d\boldsymbol{W}_s\right).$$

The first two moments of the solution of (A.3) are easily obtained by Eq. (A.4). If $\mathbb{E}[|\boldsymbol{c}^2|] < \infty$ we have for the mean $\boldsymbol{m}(t) = \mathbb{E}[\boldsymbol{z}(t)]$, using that the expectation of a stochastic integral is zero,

$$\boldsymbol{m}(t) = \Phi(t)\left(\mathbb{E}[\boldsymbol{c}] + \int_{t_0}^{t} \Phi(s)^{-1}\boldsymbol{a}(s)ds\right). \tag{A.5}$$

From Eq. (A.5) also a differential equation can be derived

$$\dot{\boldsymbol{m}}(t) = A(t)\boldsymbol{m}(t) + \boldsymbol{a}(t), \quad \boldsymbol{m}(t_0) = \mathbb{E}[\boldsymbol{c}].$$

The second moment $K(s,t)$ is given by

$$\begin{aligned} K(s,t) &= \mathbb{E}[(\boldsymbol{z}(s)-\mathbb{E}[\boldsymbol{z}(s)])(\boldsymbol{z}(t)-\mathbb{E}[\boldsymbol{z}(t)])'] = \mathbb{E}[\boldsymbol{z}(s)\boldsymbol{z}(t)'] - \mathbb{E}[\boldsymbol{z}(s)]\,\mathbb{E}[\boldsymbol{z}(t)] \\ &= \Phi(s)\left(\int\limits_{t_0}^{\min(t,s)} \Phi^{-1}(u)\Sigma(u)\Sigma'(u)(\Phi^{-1}(u))'du + \mathbb{E}[(\boldsymbol{c}-\mathbb{E}[\boldsymbol{c}])(\boldsymbol{c}-\mathbb{E}[\boldsymbol{c}])']\right)\Phi'(t). \end{aligned} \tag{A.6}$$

Above we used the independence of $\boldsymbol{z}$ from $\boldsymbol{c}$ and the stochastic integral and that

$$\mathbb{E}\left[\int_{t_0}^{s} G(u)d\boldsymbol{W}(u)\int_{t_0}^{t} H(u)d\boldsymbol{W}(u)\right] = \int\limits_{t_0}^{\min(t,s)} \mathbb{E}[G(u)H(u)]du$$

for bounded matrix functions G and H. Differentiating (A.6) yields for the covariance matrix $C(t) := K(t,t)$ of $\boldsymbol{z}(t)$ the relation

$$\dot{C}(t) = \Phi(t)C(t) + C(t)\Phi'(t) + \Sigma(t)\Sigma'(t). \tag{A.7}$$

Further examination of Eq. (A.4) reveals that the stochasticity of the solution $\boldsymbol{z}(t)$ is due to two components, the stochastic integral $\int_{t_0}^{t}\Phi(s)^{-1}\Sigma(s)d\boldsymbol{W}(s)$ on the one hand and the initial value $\boldsymbol{c}$ on the other hand. As a stochastic integral over deterministic functions is normally distributed, i.e.

$$\int_{t_0}^{t} G(u)d\boldsymbol{W}(u) \sim \mathcal{N}\left(\boldsymbol{0}, \int_{t_0}^{t} G(u)G(u)'du\right), \tag{A.8}$$

and the initial value is assumed to be independent of the stochastic integral, we immediately have

Theorem A.1.3. *The solution of* (A.3) *is a Gaussian stochastic process, if and only if the initial value* $\boldsymbol{c}$ *is constant or normal distributed. The first and second moments are given by* (A.5) *and* (A.6).

In the following corollary we adopt the above stated theory to the situation which occurs most often in this Thesis

Corollary A.1.4. *Assume that in Eq.* (A.3) *the matrix functions are given by*

$$A(t) \equiv F,\ a(t) \equiv -F\boldsymbol{\mu},\ \Sigma(t) \equiv \Sigma,$$

where all eigenvalues of F have negative real parts, and the initial value is given by a fixed $\boldsymbol{z}(t_0)$. Then the solution is a Markov process and $\boldsymbol{z}(t)$ is normal distributed for each $t \in [t_0, T]$ with mean

$$\boldsymbol{m}(t) = \boldsymbol{\mu} + \exp(F(t-t_0))(\boldsymbol{z}(t_0) - \boldsymbol{\mu}),$$

and covariance matrix

$$K(s,t) = \int_{t_0}^{t} \exp(F(t-s))\Sigma\Sigma' \exp(F'(t-s))ds,$$

which is positive definite if the assumptions on the spectrum of F are met. Furthermore, see [38, Ch. 4.4.6], a stationary solution exists, the first moment is given by $\boldsymbol{\mu}$ while the second moment Σ_s is the solution of the Sylvester equation, which can be obtained by using (A.7),

$$\Sigma_s F' + F\Sigma_s = -\Sigma\Sigma'. \tag{A.9}$$

Note that (A.9) can be transformed to

$$\Sigma_s \exp(\tau F)' - \exp(\tau F)^{-1}\Sigma_s = \exp(\tau F)^{-1}R, \tag{A.10}$$

with $\tau > 0$ and

$$R := \int_{t_0}^{t_0+\tau} \exp(sF)\Sigma\Sigma' \exp(sF)ds.$$

As (A.10) can be solved easily [106, Appendix A] for diagonalisable $\exp(\tau F)$, it can be used alternatively to (A.9) to compute the stationary covariance matrix.

A.2 Integration of the likelihood function of a VAR(p) model

Integration of the integral in (4.7) is rather straightforward but for completeness we will derive it in this appendix. The aim is to integrate $f(Z|\Phi, R)\pi_D(\Phi, R)$,with Z a given time series of length T and dimension d, $\pi_D(\Phi, R) \propto |R|^{-\frac{d+1}{2}}$ the diffusive prior and f the density as given in (3.35). The integration shall be done over all $\Phi \in \mathbb{R}^{d\times(dp+1)}$ and over all positive definite matrices $R \in \mathbb{R}^{d\times d}$. With the notation in § 3.3.3 and § 3.3.4 we have

$$\int f(Z|\Phi, R)\pi_D(\Phi, R)d\Phi dR =$$

$$\int |2\pi R|^{-\frac{T-p}{2}} \exp\Big(-\frac{1}{2}\Big(\mathrm{tr}\left((Y-\Phi X)(Y-\Phi X)'R^{-1} \right) \Big)\Big) |R|^{-\frac{d+1}{2}} d\Phi dR.$$

The argument of the trace function can be Taylor expanded around the MLE $\hat{\Phi}$ of Φ yielding

$$\int |2\pi R|^{-\frac{T-p}{2}} \exp\Bigg(-\frac{1}{2}\Big(\operatorname{tr}\big((Y-\hat{\Phi}X)(Y-\hat{\Phi}X)'R^{-1} \\ +(\mathbf{\Phi}-\hat{\mathbf{\Phi}})'[R^{-1}\otimes XX'](\mathbf{\Phi}-\hat{\mathbf{\Phi}})\big)\Big)\Bigg)|R|^{-\frac{d+1}{2}} d\Phi dR,$$

where $\mathbf{\Phi}$, resp. $\hat{\mathbf{\Phi}}$, denote the vectorised notation of Φ, resp. $\hat{\Phi}$, and $\otimes$ the Kronecker product. From this form it can be seen that $\mathbf{\Phi}$ is normal distributed and therefore can be integrated out, giving

$$\int |2\pi R|^{-\frac{T-p}{2}}|R|^{-\frac{d+1}{2}}|2\pi(R^{-1}\otimes XX')^{-1}|^{\frac{1}{2}} \exp\Bigg(-\frac{1}{2}\Big(\operatorname{tr}\big((Y-\hat{\Phi}X)(Y-\hat{\Phi}X)'R^{-1}\big)\Big)\Bigg)dR,$$

which can be simplified to

$$(2\pi)^{-\frac{d(T-(d+1)p-1)}{2}}|XX'|^{-\frac{d}{2}}\int |R|^{-\frac{T-(d+1)p+d}{2}} \exp\Bigg(-\frac{1}{2}\Big(\operatorname{tr}\big((Y-\hat{\Phi}X)(Y-\hat{\Phi}X)'R^{-1}\big)\Big)\Bigg)dR$$

The resulting integrand is proportional to an inverted Wishart distribution with $T-(d+1)p+d$ degrees of freedom, which has a defined density as long as $T > (d+1)p+d$ [43, ch. 3.4]. Therefore R can be integrated out giving rise to

$$\pi^{\frac{d(d-1)}{4}}|XX'|^{-\frac{d}{2}}|\pi(Y-\hat{\Phi}X)(Y-\hat{\Phi}X)'|^{-\frac{T-p-dp-1}{2}}\prod_{j=1}^{d}\Gamma\left(\frac{T-p-dp-j}{2}\right), \tag{A.11}$$

where Γ denotes the Gamma function. The form stated in (4.7) is obtained by simply using the notation introduced in § 3.3.4.

A.3 Deutsche Zusammenfassung

Motiviert durch die Analyse von Daten aus Molekül Dynamik Simulationen, befasst sich diese Arbeit mit der Analyse von Zeitreihen mit komplexen dynamischen Eigenschaften. Typischerweise lässt sich das dynamische Verhalten von Molekülen in verschiedene dynamische Phasen, sogenannte Konformationen, unterteilen. Diese Phasen können sich z.B. aus verschiedenen geometrischen Strukturen, zwischen denen das thermisch angeregte Molekül wechselt, ergeben. Solch ein komplexes Verhalten sollte bei der Erstellung eines (reduzierten) Modells zur Modellierung der ursprünglichen Dynamik berücksichtigt werden. Wir zeigen auf, dass, neben der Modellierung des Wechsels zwischen verschiedenen Konformationen durch eine Markov-Kette, für die dynamische Modellierung innnerhalb einer Konformation die Verwendung vektorwertiger autoregressiver Prozesse (VAR) naheliegend ist. Da der Sprungprozess zwischen verschiedenen Phasen in der Regel nicht beobachtbar ist, koppeln wir diese lokalen VAR-Prozesse mit einem sogenannten Hidden Markov Model und demonstrieren, wie dieses, zusammen mit der sogenannten Perron Cluster Cluster Analyse, zur Analyse von Moleküldaten verwendet werden kann.

Desweiteren wird ein Algorithmus entwickelt, der es erlaubt den Wechsel zwischen verschiedenen dynamischen Phasen *on-line*, d.h. mit sukzessiven Zugriff auf die Daten, zu detektieren. Hierbei stellt sich als zentrales Objekt die sogenannte Moment-Matrix heraus. Zum einen erlaubt diese die numerisch stabile Schätzung der Parameter der VAR-Prozesse, die Kodierung von Information in einem kleinen Objekt und das effiziente Zusammenfassen von, in verschiedenen Zeitreihen enthaltenen, Informationen. Zum anderen ist es, allein auf Grundlage der Moment-Matrizen, möglich, Zeitreihen-Segmente entsprechend ihrer jeweiligen dynamischen Phase zusammenzufassen. Hierdurch kann der online Algorithmus, kombiniert mit einem nachträglichen Clustern der Moment-Matrizen, alternativ zu den HMM basierten Ansätzen eingesetzt werden. Der Vorteil hiervon liegt in der Vermeidung des komplexen Optimierungsproblem, welches beim Einsatz von HMM's gelöst werden muss.

Abschliessend wird das erlangte on-line Verfahren zur Schätzung von Austrittsraten, d.h. die Sprunghäufigkeit zwischen verschiedenen Konformationen, in molekularen Systemen verwendet. Die naive Simulation molekularer Systeme zur Schätzung solcher Raten ist oftmals nicht praktikabel, da der, für die numerische Stabilität notwendige, Integrationszeitschritt zu klein ist um in vertretbarer Zeit Konformationswechsel ausreichend oft zu beobachten. Nach wie vor ist die Suche nach Algorithmen zur Verringerung des Aufwands solcher Simulation, ein wichtiges Forschungsthema. Eine alternative Strategie wäre, statt der Verringerung des Aufwands, eine *Verteilung* des Aufwand auf parallele Prozessoren. Es wurde gezeigt, dass dieses sinnvoll möglich ist, sofern ein Konformationswechsel on-line detektiert werden kann [122,124]. Anhand eines molekularen Beispiel demonstrieren wir, wie die hier entwickelten Algorithmen benutzt werden können, um diese Idee umzusetzen.

Bibliography

[1] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19:716–723, 1974.

[2] M. Allen. *Computational Soft Matter: From Synthetic Polymers to Proteins*, chapter Introduction to Molecular Dynamics Simulation, pages 1–28. John von Neumann Institute for Computing, 2004.

[3] A. Amadei, A. B. Linssen, and H. J. Berendsen. Essential dynamics of proteins. *Proteins*, 17(4):412–425, 1993.

[4] L. Arnold. *Stochastic differential equations.* Wiley, New York, 1974.

[5] A. Aue, L. Horváth, M. Hušková, and P. Kokoszka. Change-point monitoring in linear models. *Econometrics Journal*, 9:373–403, 2006.

[6] J. M. Bakker, L. M. Aleese, G. Meijer, and G. von Helden. Fingerprint IR spectroscopy to probe amino acid conformations in the gas phase. *Physical Review Letters*, 91(20):203003, Nov 2003.

[7] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171, 1970.

[8] J. Berger and L. Pericchi. The intrinsic Bayes factor for linear models. *Bayesian Statistics*, 5:25–44, 1996.

[9] J. M. Bernardo and A. F. Smith. *Bayesian Theory.* John Wiley & Sons, 1994.

[10] P. Billingsley. *Probability and Measure.* John Wiley & Sons, New York, 1979.

[11] J. A. Bilmes. A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian Mixture and Hidden Markov Models. Technical report, International Computer Science Institute, Berkeley, 1998.

[12] P. Brémaud. *Markov Chains.* Springer, 2nd edition edition, 2008.

[13] E. J. Bylaska, W. A. de Jong, K. Kowalski, T. P. Straatsma, M. Valiev, D. Wang, E. Aprá, T. L. Windus, S. Hirata, M. T. Hackler, Y. Zhao, P.-D. Fan, R. J. Harrison, M. Dupuis, D. M. A. Smith, J. Nieplocha, V. Tipparaju, M. Krishnan, A. A. Auer, M. Nooijen, E. Brown, G. Cisneros, G. I. Fann, H. Früchtl, J. Garza, K. Hirao, R. Kendall, J. A. Nichols, K. Tsemekhman, K. Wolinski, J. Anchell, D. Bernholdt, P. Borowski, T. Clark, D. Clerc, H. Dachsel, M. Deegan, K. Dyall, D. Elwood, E. Glendening, M. Gutowski, A. Hess, J. Jaffe, B. Johnson, J. Ju, R. Kobayashi, R. Kutteh, Z. Lin, R. Littlefield, X. Long, B. Meng, T. Nakajima, S. Niu, L. Pollack, M. Rosing, G. Sandrone, M. Stave, H. Taylor, G. Thomas, J. van Lenthe, A. Wong, and Z. Zhang. NWChem, a computational chemistry package for parallel computers, version 5.0. Technical report, Pacific Northwest National Laboratory, Richland, Washington 99352-0999, USA, 2006.

[14] C. Bystroff, S. J. Oatley, and J. Kraut. Crystal structures of Escherichia coli dihydrofolate reductase: the NADP+ holoenzyme and the folate NADP+ ternary complex. Substrate binding and a model for the transition state. *Biochemistry*, 29(13):3263–3277, Apr 1990.

[15] H. A. Carlson. Protein flexibility and drug design: How to hit a moving target. *Current Opinion in Chemical Biology*, 6(4):447–452, Aug 2002.

[16] G. Casella and R. L. Berger. *Statistical Inference*. Wadsworth & Brooks, 1990.

[17] G. Casella and E. Moreno. Objective Bayesian variable selection. *Journal of the American Statistical Association*, 101(101):157–167, 2006.

[18] D. Chandler. Statistical mechanics of isomerization dynamics in liquids and the transition state approximation. *Journal of chemical Physics*, 68:2959–2970, 1978.

[19] J. D. Chodera, N. Singhal, V. S. Pande, K. A. Dill, and W. C. Swope. Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics. *Journal of Computational Chemistry*, 126(15):155101, 2007.

[20] N. Chopin. Dynamic detection of change points in long time series. *Annals of the Institute of Statistical Mathematics*, 59(2):349–366, 2007.

[21] W. T. Coffey, Y. P. Kalmykov, and J. T. W. Waldron. *The Langevin Equation: With Applications in Physics, Chemistry and Electrical Engineering*. World Scientific, 1996.

[22] F. E. Cohen. Protein misfolding and prion diseases. *Journal of Molecular Biology*, 293(2):313–320, Oct 1999.

[23] D. R. Cox and D. V. Hinkley. *Theoretical Statistics.* Chapman & Hall, 1974.

[24] M. J. Crowder. Maximum likelihood estimation for dependent observations. *Journal of the Royal Statistical Society. Series B (Methodological)*, 38:45–53, 1976.

[25] F. De Santis and F. Spezzaferri. Methods for default and robust bayesian model comparison: the fractional bayes factor approach. *International Statistical Review*, 67:267–286, 1999.

[26] F. De Santis and F. Spezzaferri. Consistent fractional bayes factor for nested normal linear models. *Journal of Statistical Planning and Inference*, 97:305–321, 2001.

[27] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39:1–38, 1977.

[28] P. Deuflhard, M. Dellnitz, O. Junge, and C. Schütte. *Computational Molecular Dynamics: Challenges, Methods, Ideas*, volume 4 of *Lecture Notes in Computational Science and Engineering*, chapter Computation of Essential Molecular Dynamics by Subdivision Techniques, pages 98–115. Springer, 1999.

[29] P. Deuflhard, W. Huisinga, A. Fischer, and C. Schütte. Identification of almost invariant aggregates in reversible nearly uncoupled Markov chains. *Linear Algebra and its Applications*, 315:39–59, 2000.

[30] P. Deuflhard and M. Weber. Robust Perron cluster analysis in conformation dynamics. *Linear Algebra and its Applications*, 398:161–184, 2005.

[31] J. Doob. *Stochastic Processes.* Wiley, 1953.

[32] P. Fearnhead and Z. Liu. On-line inference for multiple changepoint problems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 69:589–605, 2007.

[33] A. Fischer. *An Uncoupling-Coupling Method for Markov Chain Monte Carlo Simulations with an Application to Biomolecules.* PhD thesis, Freie Universität Berlin, 2003.

[34] A. Fischer, F. Cordes, and C. Schütte. Hybrid monte carlo with adaptive temperature in mixed-canonical ensemble: Efficient conformational analysis of RNA. *Journal of Computational Chemistry*, 19:1689–1697, 1998.

[35] A. Fischer, F. Cordes, and C. Schütte. Hybrid monte carlo with adaptive temperature choice: efficient conformational analysis of RNA. *Computer Physics Communications*, 121:37–39, 1999.

[36] A. Fischer, S. Waldhausen, I. Horenko, E. Meerbach, and C. Schütte. Identification of biomolecular conformations from incomplete torsion angle observations by hidden Markov models. *Journal of Computational Chemistry*, 28(15):2453–2464, 2007.

[37] G. D. Forney. The Viterbi algorithm. In *Proceedings of the IEEE*, volume 61, pages 268–278, 1973.

[38] C. W. Gardiner. *Handbook of Stochastic Methods*, volume 3. Springer, 2004.

[39] A. E. Gelfand and D. K. Dey. Bayesian model choice: Asymptotics and exact calculations. *Journal of the Royal Statistical Society. Series B (Methodological)*, 56:501–514, 1994.

[40] F. J. Girón, E. Moreno, and G. Casella. Objective bayesian analysis of multiple changepoints for linear models. In J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith, and M. West, editors, *Bayesian Statistics*, volume 8, pages 1–27. Oxford University Press, 2007.

[41] I. V. Gopich and A. Szabo. Single-molecule FRET with diffusion and conformational dynamics. *The Journal of Physical Chemistry B*, 111(44):12925–12932, Nov 2007.

[42] H. Grubmüller and P. Tavan. Molecular dynamics of conformational substates for a simplified protein model. *The Journal of Chemical Physics*, 101:5047–5057, 1994.

[43] A. Gupta and D. Nagar. *Matrix variate distributions.* Chapman & Hall, 2000.

[44] O. Guvench, C.-K. Qu, and A. D. MacKerell. Tyr66 acts as a conformational switch in the closed-to-open transition of the SHP-2 N-SH2-domain phosphotyrosine-peptide binding cleft. *BMC Struct Biol*, 7:14, 2007.

[45] P. Hänggi, P. Talkner, and M. Borkovec. Reaction-rate theory: fifty years after kramers. *Reviews of Modern Physics*, 62:252–336, 1990.

[46] C. Hartmann. *Model reduction in classical molecular dynamics.* PhD thesis, Freie Universität Berlin, 2007.

[47] D. W. Heermann. *Computer Simulation Methods in Theoretical Physics*. Springer, 1990.

[48] M. Held, E. Meerbach, S. Hinderlich, W. Reutter, and C. Schütte. Conformational studies of UDP-GlcNAc in environments of increasing complexity. In U. H. E. Hansmann, J. Meinke, S. Mohanty, and O. Zimmermann, editors, *From Computational Biophysics to Systems Biology*, volume 36 of *NIC Series*, pages 145–148, 2007.

[49] N. J. Higham. *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, second edition, 2002.

[50] W. G. Hoover. Canonical dynamics: Equilibrium phase-space distributions. *Physical Review A*, 31(3):1695–1697, Mar 1985.

[51] I. Horenko. On clustering of non-stationary meteorological time series. *Journal of Climate*, 2008. Submitted.

[52] I. Horenko, E. Dittmer, A. Fischer, and C. Schütte. Automated model reduction for complex systems exhibiting metastability. *Multiscale Modeling and Simulation*, 5(3):802–827, 2006.

[53] I. Horenko, E. Dittmer, F. Lankas, J. Maddocks, P. Metzner, and C. Schütte. Macroscopic dynamics of complex metastable systems: Theory, algorithms, and application to B-DNA. *SIAM Journal on Applied Dynamical Systems*, 7:532–560, 2008.

[54] I. Horenko, E. Dittmer, and C. Schütte. Reduced stochastic models for complex molecular systems. *Computing and Visualization in Science*, 9:89–102, 2006.

[55] I. Horenko, C. Hartmann, C. Schütte, and F. Noé. Data-based parameter estimation of generalized multidimensional Langevin processes. *Physical Review E*, 76(1 Pt 2):016706, 2007.

[56] I. Horenko, R. Klein, S. Dolaptchiev, and C. Schütte. Automated generation of reduced stochastic weather models i: Simultaneous dimension and model reduction for time series analysis. *Multiscale Modeling and Simulation*, 6:1125–1145, 2008.

[57] W. Huisinga. *Metastability of Markovian systems*. PhD thesis, Freie Universität Berlin, 2001.

[58] W. Humphrey, A. Dalke, and K. Schulten. VMD – Visual Molecular Dynamics. *Journal of Molecular Graphics*, 14:33–38, 1996.

[59] S. Johnson. Hierarchical clustering schemes. *Psychometrika*, 2:241–254, 1967.

[60] I. Jolliffe. *Principal Component Analysis*. Springer, 2002.

[61] R. H. Jones, D. H. Crowell, and L. E. Kapuniai. Change detection model for serially correlated multivariate data. *Biometrics*, 26(2):269–280, 1970.

[62] W. Just, H. Kantz, C. Rödenbeck, and M. Helm. Stochastic modelling: Replacing fast degrees of freedom by stochastic processes. *Journal of Physics A: Mathematical and Theoretical*, 34:3199, 2001.

[63] R. E. Kass and A. E. Raftery. Bayes factors. *Journal of the American Statistical Association*, 90(430):773–795, 1995.

[64] A. Kehagias, E. Nidelkou, and V. Petridis. A dynamic programming segmentation procedure for hydrological and environmental time series. *Stochastic Environmental Research and Risk Assessment*, 20:77–94, 2005.

[65] M. G. Kendall and A. Stuart. *The Advanced Theory of Statistics*. Griffin London, 3. edition edition, 1973.

[66] R. Kendall, E. Apra, D. Bernholdt, E. Bylaska, M. Dupuis, G. Fann, R. Harrison, J. Ju, J. Nichols, J. Nieplocha, T. Straatsma, T. Windus, and A. Wong. High performance computational chemistry: An overview of NWChem a distributed parallel application. *Computer Physics Communications*, 128:260–283, 2000.

[67] K. Kuwata. An emerging concept of biomolecular dynamics and function: applications of nmr & mri. *Magn Reson Med Sci*, 1(1):27–31, 2002.

[68] O. Lange and H. Grubmüller. Collective langevin dynamics of conformational motions in proteins. *Journal of Chemical Physics*, 124:214903, 2006.

[69] M. Lavielle. Detection of multiple changes in a sequence of dependent variables. *Stochastic Processes and its Applications*, 83:79–102, 1999.

[70] M. Lavielle and G. Teyssieére. Detection of multiple change-points in multivariate time series. *Lithuanian Mathematical Journal*, 46(3):287–306, 2006.

[71] E. Lindahl, B. Hess, and D. van der Spoel. Gromacs 3.0: A package for molecular simulation and trajectory analysis. *Journal of Molecular Modelling*, 7:306–317, 2001.

[72] J. Liu. *Monte Carlo Strategies in Scientific Computing.* Springer, 2001.

[73] H. Lütkepohl. *Introduction to Multiple Time Series Analysis.* Springer, Berlin - Heidelberg, 1991.

[74] K. V. Mardia. *Statistics of Directional Data.* Academic Press, 1972.

[75] E. Meerbach, E. Dittmer, I. Horenko, and C. Schütte. Multiscale modelling in molecular dynamics: Biomolecular conformations as metastable states. In M. Ferrario, G. Ciccotti, and K. Binder, editors, *Computer Simulations in Condensed Matter: Systems: From Materials to Chemical Biology. Volume I*, number 703 in Lecture Notes in Physics, pages 475–497. Springer, 2006.

[76] E. Meerbach and C. Schütte. Sequential change point detection in molecular dynamics trajectories. *Submitted to Multiscale Modeling and Simulation*, 2008.

[77] E. Meerbach, C. Schütte, and A. Fischer. Eigenvalue bounds on restrictions of reversible nearly uncoupled markov chains. *Linear Algebra and its Applications*, 398:141–160, 2005.

[78] E. Meerbach, C. Schütte, I. Horenko, and B. Schmidt. *Analysis and Control of Ultrafast Photoinduced Reactions*, chapter Metastable Conformational Structure and Dynamics: Peptides between Gas Phase and Aqueous Solution, pages 796–806. Springer, 2007.

[79] B. Mehlig, D. W. Heermann, and B. M. Forrest. Hybrid monte carlo method for condensed-matter systems. *Physical Review B*, 45(2):679–685, Jan 1992.

[80] P. Metzner, C. Schütte, and E. Vanden-Eijnden. Illustration of transition path theory on a collection of simple examples. *The Journal of Chemical Physics*, 125(8):084110, 2006.

[81] A. L. Morris, M. W. MacArthur, E. G. Hutchinson, and J. M. Thornton. Stereochemical quality of protein structure coordinates. *Proteins*, 12(4):345–364, Apr 1992.

[82] Y. Mu, D. S. Kosov, and G. Stock. Conformational dynamics of trialanine in water. 2. Comparison of AMBER, CHARMM, GROMOS, and OPLS force fields to NMR and infrared experiments. *Journal of Physical Chemistry B*, 107:5064–5073, 2003.

[83] Y. Mu, P. H. Nguyen, and G. Stock. Energy landscape of a small peptide revealed by dihedral angle principal component analysis. *Proteins: Structure, Function, and Bioinformatics*, 58(1):45–52, 2004.

[84] R. Neal and G. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In M. I. Jordan, editor, *Learning in Graphical Models*. Kluwer, 1998.

[85] E. Nelson. *Dynamical Theories of Brownian Motion*. Princeton University Press, 2. edition, 2001. Posted on the web at: http://www.math.princeton.edu/∼nelson/books.html.

[86] A. Neumaier and T. Schneider. Estimation of parameters and eigenmodes of multivariate autoregressive models. *ACM Transactions on Mathematical Software*, 27(1):27–57, 2001.

[87] S. Ni and D. Sun. Bayesian estimates for vector autoregressive models. *Journal of Business & Economic Statistics*, 23:105–117, 2005.

[88] F. Noé, I. Horenko, C. Schütte, and J. C. Smith. Hierarchical analysis of conformational dynamics in biomolecules: transition networks of metastable states. *J Chem Phys*, 126(15):155102, Apr 2007.

[89] A. O'Hagan. Fractional bayes factors for model comparision. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):99–138, 1995.

[90] B. Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Springer, December 2005.

[91] A. Ostermann, R. Waschipky, F. G. Parak, and G. U. Nienhaus. Ligand binding and conformational motions in Myoglobin. *Nature*, 404(6774):205–208, Mar 2000.

[92] S. Park, M. K. Sener, D. Lu, and K. Schulten. Reaction paths based on mean first-passage times. *The Journal of Chemical Physics*, 119(3):1313–1319, 2003.

[93] L. Perreault, J. Bernier, B. Bobée, and E. Parent. Bayesian change-point analysis in hydrometeorological time series. part 1. the normal model revisited. *Journal of Hydrology*, 235:221–241, 2000.

[94] L. Perreault, J. Bernier, B. Bobée, and E. Parent. Bayesian change-point analysis in hydrometeorological time series. part2. comparision of change-point models and forecasting. *Journal of Hydrology*, 235:242–263, 2000.

[95] R. Preis, M. Dellnitz, M. Hessel, C. Schütte, and E. Meerbach. Dominant paths between almost invariant sets of dynamical systems. Preprint, 2004.

[96] L. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, pages 257–286, 1989.

[97] A. E. Raftery. Approximate bayes factors and accounting for model uncertainty in generalised linear models. *Biometrika*, 83(2):251–266, 1996.

[98] G. N. Ramachandran and V. Sasiskharan. Conformations of polypeptides and proteins. *Advan. Prot. Chem.*, 23:283–427, 1968.

[99] F. Rigat. Sequential change-point detection for time series models: assessing the functional dynamics of neuronal networks. Technical Report 07-07v2, Center for Research in Statistical Metholodogy, Department of Statistics, University of Warwick;, 2007.

[100] H. Risken. *The Fokker-Planck Equation*. Springer, 1989.

[101] L. Ronga, P. Palladino, S. Costantini, A. Facchiano, M. Ruvo, E. Benedetti, R. Ragone, and F. Rossi. Conformational diseases and structure-toxicity relationships: lessons from prion-derived peptides. *Current Protein and Peptide Science*, 8(1):83–90, 2007.

[102] K. Roy. A note on the asymptotic distribution of likelihood ratio. *Calcutta Statistical Association Bulletin*, 7:73–77, 1957.

[103] B. Schuler, E. A. Lipman, and W. A. Eaton. Probing the free-energy surface for protein folding with single-molecule fluorescence spectroscopy. *Nature*, 419(6908):743–747, 2002.

[104] C. Schütte, A. Fischer, W. Huisinga, and P. Deuflhard. A direct approach to conformational dynamics based on hybrid Monte Carlo. *Journal of Computational Physics*, 151:146–168, 1999.

[105] C. Schütte, R. Forster, E. Meerbach, and A. Fischer. Uncoupling-coupling techniques for metastable dynamical systems. In R. Kornhuber, R. Hoppe, J. Périaux, O. Pironneau, O. Widlund, and J. Xu, editors, *Domain Decomposition Methods in Science and Engineering*, volume 40 of *Lecture Notes in Computational Science and Engineering*, pages 115–129. Springer, 2005.

[106] C. Schütte and I. Horenko. Likelihood-based estimation of multidimensional Langevin models and its application to biomolecular dynamics. *Multiscale Modeling and Simulation*, 2008. Accepted.

[107] C. Schütte and W. Huisinga. *Biomolecular Conformations can be Identified as Metastable Sets of Molecular Dynamics*, pages 669–744. Handbook of Numerical Analysis X. Elsevier, 2003. Special Volume: Computational Chemistry.

[108] C. Schütte, W. Huisinga, and P. Deuflhard. Transfer operator approach to conformational dynamics in biomolecular systems. In B. Fiedler, editor, *Ergodic Theory, Analysis, and Efficient Simulation of Dynamical Systems*, pages 191–223. Springer, 2001.

[109] C. Schütte, F. Noé, E. Meerbach, P. Metzner, and C. Hartmann. Conformation dynamics. In *Proceedings of the ICIAM 2007*, in press.

[110] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.

[111] E. Seneta. *Non-negative matrices and Markov Chains.* Springer, 1981.

[112] M. Shirts and V. S. Pande. Computing: Screen savers of the world unite! *Science*, 290(5498):1903–1904, Dec 2000.

[113] D. Spiegelhalter and A. Smith. Bayes factors for linear and log-linear models with vague prior information. *Journal of the Royal Statistical Society. Series B (Methodological)*, 44(3):377–387, 1982.

[114] D. V. D. Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark, and H. J. C. Berendsen. Gromacs: fast, flexible, and free. *Journal of Computational Chemistry*, 26(16):1701–1718, Dec 2005.

[115] D. Stalling, M. Westerhoff, and H.-C. Hege. *The Visualization Handbook*, chapter Amira: A Highly Interactive System for Visual Data Analysis (Ch. 38), pages 749–767. Elsevier Academic Press, 2004.

[116] D. Sun and S. Ni. Bayesian analysis of vector-autoregressive models with noninformative priors. *Journal of Statistical Planning and Inference*, 121:291–309, 2004.

[117] S. J. Teague. Implications of protein flexibility for drug discovery. *Nature Reviews Drug Discovery*, 2(7):527–541, 2003.

[118] V. Tozzini. Coarse-grained models for proteins. *Current Opinion in Structual Biology*, 15(2):144–150, 2005.

[119] N. Ueda and R. Nakano. Determinisitc annealing em algorithm. *Neural Networks*, 11:271–282, 1998.

[120] E. Vanden-Eijnden and F. A. Tal. Transition state theory: Variational formulation, dynamical corrections, and error estimates. *Journal of Chemical Physics*, 123:184103, 2005.

[121] A. J. Viterbi. Error bound for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory.*, 13(2):260–269, 1967.

[122] A. F. Voter. Parallel replica method for dynamics of infrequent events. *Physical Review B*, 57(22):R13985–R13988, 1998.

[123] A. F. Voter and J. D. Doll. Dynamical corrections to transition state theory for multistate systems: Surface self-diffusion in the rare-event regime. *Journal of Chemical Physics*, 82:80–89, 1985.

[124] A. F. Voter, F. Montalenti, and T. C. Germann. Extending the time scale in atomistic simulation of materials. *Annual Review of Material Research*, 32:321–346, 2002.

[125] M. Weber. Improved Perron cluster analysis. Technical Report 03-04, Zuse Institute Berlin, 2004.

[126] M. Weber. *Meshless Methods in Conformation Dynamics.* PhD thesis, Freie Universit/"at Berlin, 2005.

[127] S. Wilks. The large-sample distribution of the likelihood ratio for testing composite hypotheses. *The Annals of Mathematical Statistics*, 9(1):60–62, 1938.

[128] B. Zagrovic, E. J. Sorin, and V. Pande. Beta-hairpin folding simulations in atomistic detail using an implicit solvent model. *Journal of Molecular Biology*, 313(1):151–169, Oct 2001.