Master's thesis

Estimating low dimensional dynamical models for molecules

Submitted by

Max Philipp Holl

December 12, 2018

First examiner
Prof. Dr. Andreas Heuer

Second examiner
Dr. Oliver Kamps

Westfälische Wilhelms-Universität Münster
Fachbereich Physik
Institut für Theoretische Physik

Contents

1 Introduction **1**

2 Methods **5**

2.1 Dimension Reduction 5

2.1.1 Principal Components Analysis 5

2.1.2 ISOMAP 7

2.1.3 Comparison of PCA and ISOMAP 10

2.1.4 Determination of Embedding Dimension - The Plateau Dimension 11

2.2 Reconstruction of Dynamics 13

2.2.1 Langevin Equation 13

2.2.2 Drift-Diffusion Estimation 13

2.2.3 Step-wise Drift-Diffusion Estimation 14

2.3 Kernel Density Estimation 15

3 Single TS Dimension Reduction **17**

3.1 Trialanine 17

3.2 Trp-cage 22

3.3 Aβ42 26

4 Multi TS Dimension Reduction **31**

4.1 Trialanine 31

4.2 Trp-cage 35

4.3 Aβ42 39

5 Reconstruction of Dynamics **43**
5.1 Trialanine 43
5.1.1 Drift fields in the two dimensional representations . . 43
5.1.2 Reproduction of known results 44
5.1.3 Dynamics on the curved embedding 46
5.1.4 Dynamics using a reduced library 47
5.2 Trp-cage 49
5.2.1 Single time series drift fields in the two dimensional representations 49
5.2.2 Single time series step-wise drift-diffusion estimation . 51
5.2.3 Multiple time series drift fields in two dimensions . . . 53
5.2.4 Multiple time series drift-diffusion estimation 55
5.3 Aβ42 56
5.3.1 Drift fields in the two dimensional representations . . 56
5.3.2 Single time series step-wise drift-diffusion estimation . 58
5.3.3 Multiple time series drift fields in two dimensions . . . 60
5.3.4 Multiple time-series drift-diffusion estimation 61

6 Conclusion **63**

Chapter 1

Introduction

Proteins are one of the main building blocks of life as we know it. Within a living cell they perform a wide variety of functions. Enzymes play an important role in catalysing chemical reactions, other proteins provide structural stability, be it inside the cell as part of the cytoskeleton, or outside, e.g. in hair. Yet another function they provide is transport inside the body, such as haemoglobin transporting oxygen from the lungs to the other organs.

Proteins are composed of many amino acid residues, whose sequence is coded into the DNA. This sequence is called the primary structure of the protein. Each amino acid is bounded to its neighbour by a peptide bond. The functions a protein fulfils are however not simply governed by the sequence of amino acids that build it, but by the larger structures these macromolecules build. Of high interest is here the so called tertiary structure, meaning the three-dimensional structure of a protein. This functional structure is assembled by a process called protein folding. Understanding these folding processes is highly relevant in understanding the functions of certain proteins. The misfolding of certain proteins is for example associated with neurodegenerative diseases such as Alzheimer's or Parkinson's [12]. Knowing what is going wrong on the molecular level could provide a path to healing these diseases.

To this end simulations of folding processes are the best option of understanding the molecular details. These simulations are typically performed as Molecular Dynamics (MD) simulations. These MD simulations produce

time series of a nanosecond length. They involve solving Newton's laws of motion for each atom in the molecule simulated. From a physical point of view this forms a vastly high dimensional system. Aβ42, the largest of the molecules investigated in this thesis, is comprised of 627 atoms, each with 3 cartesian coordinates. Typically these large systems are simulated using supercomputers.

An important part of analysing these systems is identifying metastable states that are highly populated by the system. In order to obtain the correct distribution of these states, long time series are necessary since a system can stay in one metastable state for a long time and rarely jumps to another. These long time series are very expensive to calculate using molecular dynamics. However an alternative was proposed by Hegger and Stock [11], based on initial work on the estimation of drift and diffusion coefficients by Friedrich et al. (see [9, 10, 22]). Their Langevin algorithm can model a time series of, e.g. the dynamics of a molecule, from local information about the drift and stochastical driving. Due to only taking into account local information, this algorithm does not need statistically converged input data in order to produce the correct distribution of metastable states. The library data used by the algorithm can be composed from several short MD trajectories that together cover the phase space of the system. These will however not represent a Boltzmann weighted distribution. The production of a long time series is then the task of the Langevin algorithm, which can run on a standard PC. The algorithm involves a nearest neighbour search at each time step, which becomes highly inefficient at higher dimensions. This is one of the problems related to the term *curse of dimensionality*. It is mainly due to data in high dimensional space being inherently sparse. The Langevin algorithm therefore runs much more efficiently in low dimensional space. This algorithm presents a method to bridge the gap between the time scale of MD simulations of a few nanoseconds and that of, e.g. folding processes of proteins, that take place in the order of microseconds. It has been applied to a number of short peptides (see [11, 20, 18]).

The molecules considered in this thesis are, as described above, represented by the $3N$ coordinates of their N atoms. These are however not the degrees of freedom of the system since the atoms are tightly bound to each other by

different intramolecular bonds, such as covalent or hydrogen bonds. Therefore it should be possible to find suitable collective reaction coordinates that enable a low dimensional description of the system. Commonly used empirical coordinates are, e.g. the radius of gyration R_g or the fraction of native contacts Q. Other options to reduce the dimensionality are of a more general nature and do not need any previous knowledge about the system analysed. The best known dimension reduction algorithm in this context is probably the Principal Components Analysis [1]. It is a fast and reliable algorithm, but has limitations due to being a linear method.

Many nonlinear dimension reduction algorithms have been developed in the context of machine learning, where they perform a feature extraction before e.g. pattern recognition is applied. One of these nonlinear methods is the ISOMAP algorithm [23]. Itself and various derivatives have been used in fields ranging from motion recognition [2] and classification of emotional states from EEG data [25] to finding reaction coordinates of a folding protein [5]. Other nonlinear dimension reduction methods are diffusion maps [4], sketch map [3] and locally linear embedding [17].

Dimension reduction not only enables the Langevin algorithm to be used efficiently, it is also an important tool for human understanding of high dimensional systems. To this end a good dimension reduction algorithm should be able to distinguish between different metastable states and capture the reaction pathways leading from one such state to another. It should reveal the internal structure of the system.

The combination of dimension reduction and step-wise drift-diffusion estimation is of course not restricted to molecular systems. In general any high dimensional dynamical system can be studied using these methods. Possible other applications range from systems such as the human body, studying different vitals such as EEG data, heart or respiratory rates etc., to larger systems, such as environmental parameters in climate research.

In this thesis time series of three different peptides, trialanine [11], Trp-cage [14] and Aβ42 [12], are investigated. In a first step both dimension reduction algorithms, PCA and ISOMAP, are performed on a single time series of each of these molecules (section 3) and the results are compared to each other. Leading towards the usage of multiple short trajectory, the trialanine time

series is used as a simple non-trivial toy system (section 4.1). It is split into several short time series which are then reassembled, in order to test the performance of the dimensionality reduction algorithms on data that is not continuously sampled. The dimension of multiple short time series of the Trp-cage and Aβ42 molecules is subsequently reduced by both algorithms and again their performance is compared.

Finally the data-driven Langevin algorithm by Hegger and Stock is used on the obtained time series of reduced dimensionality. In order to validate the code written for this thesis, the results from [11] are reproduced in section 5.1.2. The trialanine time series is then again split into shorter time series and used as a toy system to test the performance of the Langevin algorithm on library data comprised of multiple short time series. In sections 5.2 and 5.3 this is then applied to multiple time series of the larger peptides, Trp-cage and Aβ42. For each of these molecules a 40 μs time series is estimated using the data-driven Langevin algorithm with library time series whose dimensionality was reduced using the ISOMAP algorithm.

In the last chapter, the results of this thesis are summarized and starting points for further investigation of the methods used here are given.

Chapter 2

Methods

2.1 Dimension Reduction

Data from molecular dynamics (MD) simulations is usually presented in cartesian coordinates. A molecule comprised of N atoms is therefore described by $3N$ cartesian coordinates. The underlying problem however is not necessarily a $3N$-dimensional one. The degrees of freedom of this system are constrained by e.g., covalent bonds or hydrogen bonds etc., this leads to a necessary dimensionality reduction. This dimensionality reduction needs to condense the dynamics of the system to as few dimensions as possible, without discarding relevant properties of the dynamics. The methods to achieve dimensionality reduction can be classified into linear and nonlinear methods.
One of the best known linear methods is the Principal Components Analysis (PCA), where a transformation matrix is found that minimizes covariances between the variables. A nonlinear method is ISOMAP, which, based on Multidimensional Scaling (MDS), approximates the low-dimensional manifold by preserving geodesic distances.

2.1.1 Principal Components Analysis

Standard PCA

The goal of the PCA transformation is to find a linear transformation of the data set from the space of observations Y to an uncorrelated orthogonal

basis set. The first principle component is along the direction with the largest variance, i.e. it describes the highest possible variability of the data set. The other basis vectors of the transformation are found along the axis with variance in descending order, with the constraint of being orthogonal to each other. The motion of the $3N$ coordinates $q_i ... q_{3N}$ is stored in the covariance matrix [6]:

$$\sigma_{ij} = \langle q_i q_j \rangle - \langle q_i \rangle \langle q_j \rangle. \tag{2.1}$$

This matrix $\boldsymbol{\sigma}$ is decomposed into a diagonal matrix of eigenvalues $\boldsymbol{\Lambda}$ in descending order and the corresponding matrix of eigenvectors $\boldsymbol{V}$:

$$\boldsymbol{\sigma} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}. \tag{2.2}$$

The principal components x_i are retrieved by the multiplication:

$$x_i = \boldsymbol{v}_i \boldsymbol{q}, \tag{2.3}$$

where v_i is the i-th eigenvector of the covariance matrix $\boldsymbol{\sigma}$. A reduced dimensionality is now achieved by discarding the principal components with the least variance.
The dimensionality P of the reduced data set is chosen by the fraction of explained variance of the principal components $\sum_{i=1}^{P} \lambda_i / \sum_{j=1}^{3N} \lambda_j$.

Dihedral PCA

To avoid artefacts from the overall motion of the molecule it may be beneficial to use the dihedral angles as observational variable instead of the cartesian coordinates. These angles $\{\varphi_n\}$ are converted to a metric coordinate space by the transformation

$$\begin{aligned} q_{2n-1} &= \cos \varphi_n \\ q_{2n} &= \sin \varphi_n \end{aligned} \tag{2.4}$$

to avoid problems arising from the circularity of the angles [11].

2.1.2 ISOMAP

ISOMAP is a nonlinear dimensionality reduction method based on Multidimensional Scaling [23]. MDS embeds a high dimensional data set into a low dimensional space by finding an embedding that best preserves the euclidean distances between the points.

Since the constraint surface on which the points lie, can be folded or otherwise distorted in the high dimensional space, linear methods like PCA or MDS are not able to transform the data set correctly to a low dimensional embedding.

In contrast to MDS, ISOMAP seeks to preserve the geodesic distances on the low dimensional manifold. These distances are approximated by 'hopping' from one point to another through their nearest neighbours and adding this 'hopping' or graph distance. The use of the geodesic distances is done in order to incorporate the structure of the manifold in the nonlinear embedding.

To realise this graph distance at first a network is built, where each point is connected to its k-nearest neighbours. Then the distances on the network are computed using Dijkstra's algorithm [7]. Since MDS uses not the distances but the scalar products, the matrix of distances is squared and converted to a Gramian matrix $\boldsymbol{S}$ of scalar products by double centering it, i.e. removing from each row the row mean and from each column the column mean and adding the grand mean. The eigendecomposition of this matrix is then computed:

$$\boldsymbol{S} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{-1}. \tag{2.5}$$

From here the low dimensional data set $\boldsymbol{X}$ can be retrieved via

$$\boldsymbol{X} = \mathbb{I}_{P\times N}\boldsymbol{\Lambda}^{1/2}\boldsymbol{U}^{T}, \tag{2.6}$$

where $\mathbb{I}_{P\times N}$ is an identity matrix with additional zero columns [23].

The larger the data set, the larger the distance and Gramian matrices get, up until a point where the matrices are no longer computationally tractable. Therefore an alternative has to be employed that is cheaper to compute.

This alternative is called Scalable ISOMAP (ScIMAP). [5] It speeds up computation drastically by calculating the embedding on a specifically chosen set of n landmarks and calculating only the desired number of largest eigenvalues and eigenvectors. This approach works on the assumption, that the system stays in low free energy areas most of the time. Therefore a large portion of data points is redundant for finding the geometry of the low dimensional embedding.

The embedding of the set of landmarks is then found in the same way as the standard ISOMAP by eigendecomposition of the Gramian matrix. The remaining points $\boldsymbol{y}$ of the data set are reinserted into the embedding by computing the squared distance vectors $\boldsymbol{\delta}$ to all landmarks. These are again demeaned to obtain the scalar product vectors $\boldsymbol{s}$. Finally the low dimensional representation is found by [6]:

$$\boldsymbol{x} = \mathbb{I}_{P\times n}\boldsymbol{\Lambda}^{-1/2}\boldsymbol{U}^T\boldsymbol{s}. \tag{2.7}$$

A lower bound for the number of landmarks used is the dimensionality. For a P-dimensional embedding, at least $P + 1$ landmarks are needed.[5] The accuracy of the embedding is of course improved by considering much more landmarks. Upper bounds for the number of landmarks are the time needed to compute the eigenvalues of the Gramian matrix and the memory available to store the matrices involved.

The landmarks used in this method can be chosen in different manners, which are described and discussed in the following sections.

Random Selection [6]

For a random selection of landmarks, the landmarks are simply drawn randomly from the existing points. Since each point is equally likely to be a landmark, the majority of landmarks will be in highly sampled metastable states. The spatial separation between the landmarks is not optimal and many of the landmarks are redundant and do not offer additional information about the manifolds geometry.

Iterative MaxMin Selection [6]

In this landmark selection method, the landmarks are chosen in an iterative manner. The first landmark is chosen randomly from the points of the data set. All subsequent landmark points are chosen to maximize the minimum distance to the existing landmarks. This procedure ensures an optimal spacial separation between the landmarks, that sufficiently samples the sparse regions of the conformational space while avoiding to oversample the high density regions. However it takes a very long time to compute a sufficient number of landmarks. E.g. it took more than a week to calculate 5000 landmarks for the Aβ42 molecule.

Distance Weighted Random Selection

An improvement of this method can be achieved by assigning weights to each point from the data set, and drawing weighted samples. These weights can be distance based. This distance could for example be the distance to the mean of all points. Alternatively one could choose several points, e.g. the centres of previously identified metastable states, to which the distances are taken.
The weights are then assigned, such that points farther away from the reference points are more likely to be selected as a landmark. This method ensures more equally distributed landmarks than the fully random approach. However especially when the distance to metastable states is used, prior knowledge of some features of the system is necessary.

Kernel Density Weighted Random Selection

Similarly to the distance weighted random selection of landmarks, the weights can also be assigned by first calculating the kernel density at each point in the time series (for a detailed explanation of the kernel density estimation see section 2.3). This can be done with different kernels, such as the Epanechnikov kernel or a Gaussian kernel. The bandwidth in the kernel density estimate can be chosen, e.g., by applying Scott's rule. Since the landmarks should be more or less equally spaced in order to efficiently represent the overall structure of the phase space, points from less populated regions in

phase space shall be selected preferably. Therefore the weights are computed as the inverse of the kernel density at each point of the library data.

2.1.3 Comparison of PCA and ISOMAP

As an initial test of the two algorithms, the swiss roll data set is used. It consists of N points sampled from a two dimensional plain. These points (x_1, x_2) are then transformed into three dimensional space by the relation

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} x_1 \cos(x_1) \\ x_2 \\ x_1 \sin(x_1) \end{pmatrix}. \tag{2.8}$$

The aim of any dimensionality reduction algorithm is now to reproduce the original, in this case two dimensional structure of the manifold. The results of both algorithms, as well as the three dimensional representation of the swiss roll data set can be seen in figure 2.1.

As can be seen in these figures, PCA as a linear dimensionality reduction method fails to 'unroll' the swiss roll. As expected it delivers a projection of the data onto the two axes with the highest variance, in this case the plane spanned by the y_1 and y_3 axes in figure 2.1A, apart from rotations.

ISOMAP in contrast is able to recover the two dimensional structure much better, although the recovered structure is not rectangular but distorted along the edges. This is due to the selection method of the neighbouring points. In this case a fixed number of neighbours was chosen. At the edges of the rectangle the neighbours are not isotropically distributed, the network provides shortcuts through the manifolds. As a consequence the path distances along the edges are a bit shorter than in the plane, leading to the distortion seen in figure 2.1C.

Apart from the obvious advantage of being able to develop curved manifolds that PCA is not able to reduce correctly, ISOMAP has a grave disadvantage: the computational time needed to embed the data set into a lower dimension. A PCA embedding of a d-dimensional data set consists only of computing the $d \times d$-covariance matrix and its eigendecomposition and the matrix multiplication of the data set with the transformation matrix.

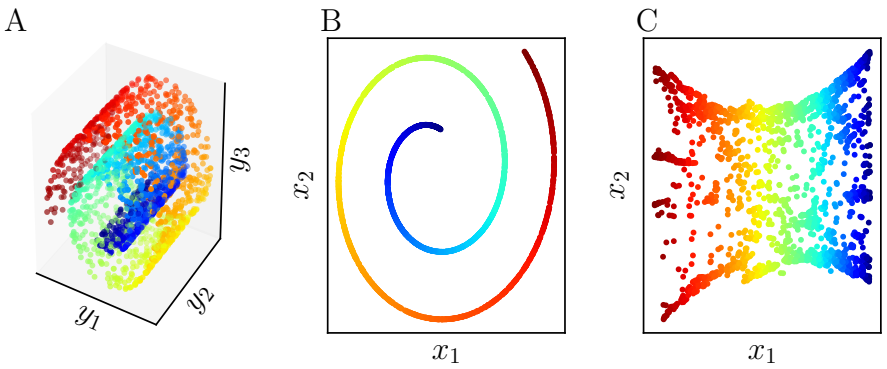

Figure 2.1: Swiss roll data set with $N = 2048$ colour coded points, (A) in three dimensional representation, (B) in two dimensional representation found by the PCA algorithm, (C) in two dimensional representation found by the ISOMAP algorithm

Landmark ISOMAP on the other hand requires the calculation of n landmarks, finding nearest neighbours in a high dimensional data set, calculating distances on the resulting network using Dijkstra's algorithm, the eigendecomposition of the $n \times n$ squared distance matrix, and finally for each point a multiplication of the $P \times n$ transformation matrix with the length n scalar product vector s. All this can easily take more than an hour to calculate, with the computationally most expensive calculations running in parallel on eight cores.

2.1.4 Determination of Embedding Dimension - The Plateau Dimension

A major challenge of reducing the dimensionality of a given data set is the estimation of the intrinsic dimensionality. One approach is to look at the eigenvalues calculated during the calculation of the embedding. In both methods described above, only the eigenvectors corresponding to high magnitude eigenvalues are considered in the embedding. The embedding dimension is chosen to be the dimension, where the magnitude of the eigenvalues no longer changes significantly [13]. A second approach is to calculate the residual variance, i.e., the variance that is not explained in a d-dimensional

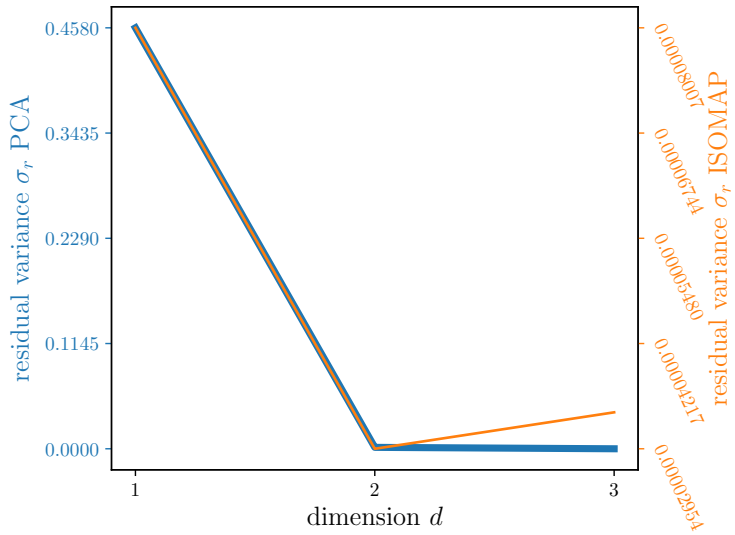

Figure 2.2: Residual variance σ_r against number of dimensions for both PCA and ISOMAP for the swiss roll data set.

embedding. It is defined as [23]:

$$\sigma_r = 1 - R^2(D_M, D_Y). \tag{2.9}$$

Where R denotes the correlation coefficient, taken over all entries of the matrices D_M and D_Y, with D_M being the estimate of the manifold distance of each algorithm and D_Y the euclidean distance in the low dimensional embedding. Again one looks for the dimension where the residual variance no longer changes significantly and the graph of the residual variance enters a plateau, thus the name Plateau Dimension.

The residual variance σ_r additionally offers a means of comparison between the different dimensionality reduction algorithms. A lower residual variances indicates a more accurate embedding. This is reflected in the residual variance for the swiss roll data set in figure 2.2. The residual variance of the ISOMAP embedding is four orders of magnitude lower than that of the PCA embedding. The only exception is the third dimension, where PCA has a zero residual variance, since it is perfectly able to embed a three dimensional data set into three dimensions.

2.2 Reconstruction of Dynamics

2.2.1 Langevin Equation

Apart from finding and understanding a low dimensional representation of a system it is as well instructive to study its dynamics, either in the high or the low dimensional representation. For this the Langevin equation is employed [11]:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{h}(\boldsymbol{x}(t)) + \mathcal{D}(\boldsymbol{x}(t))\boldsymbol{\epsilon}(t). \tag{2.10}$$

This equation is comprised of a drift function $\boldsymbol{h}(\boldsymbol{x}(t))$, describing the deterministic motion of the system in phase space, e.g. the motion of a molecule towards a metastable state. The second part of equation 2.10 is the stochastic driving $\mathcal{D}(\boldsymbol{x}(t))\boldsymbol{\epsilon}(t)$, with the diffusion operator $\mathcal{D}$ and white noise $\boldsymbol{\epsilon}$. It accommodates the influence of low amplitude fluctuations of the molecule and acts as a driving of the system.
Since the library data is discretely sampled, the discrete form of the Langevin equation is used. The differential $\dot{\boldsymbol{x}}$ is approximated by the difference quotient $\Delta\boldsymbol{x}/\Delta t$:

$$\Delta\boldsymbol{x}/\Delta\boldsymbol{t} = \boldsymbol{x}_{n+1} - \boldsymbol{x}_n \tag{2.11}$$

$$\Delta\boldsymbol{x}_n = \boldsymbol{h}(\boldsymbol{x}_n)\Delta t + \mathcal{D}(\boldsymbol{x}_n)\boldsymbol{\epsilon}_n\sqrt{\Delta t} \tag{2.12}$$

2.2.2 Drift-Diffusion Estimation

In order to estimate Drift and Diffusion locally, the local average $\langle f(\boldsymbol{x})\rangle$ is defined [11]:

$$\langle f(\boldsymbol{x})\rangle = \frac{\sum_i f(\boldsymbol{x}_i)\Theta(\delta - ||\boldsymbol{x}_i - \boldsymbol{x}||)}{\sum_i \Theta(\delta - ||\boldsymbol{x}_i - \boldsymbol{x}||)}. \tag{2.13}$$

Where the step function $\Theta(\delta - ||\boldsymbol{x}_i - x||)$ ensures only points inside a sphere of diameter δ contribute to the average. In practice δ is varied, such that a fixed amount of points contributes.
Since the noise term in equation 2.10 is unknown, the terms $\boldsymbol{h}$ and $\mathcal{D}$ cannot be estimated directly, e.g. by a least-squares fit. Rather the statistical properties of the noise have to be exploited. To this end the local average

is applied to equation 2.12:

$$\langle \Delta \boldsymbol{x}_n \rangle = \langle \boldsymbol{h}(\boldsymbol{x}_n) \rangle \Delta t + \langle \mathcal{D}(\boldsymbol{x}_n) \boldsymbol{\epsilon}_n \rangle \sqrt{\Delta t}. \tag{2.14}$$

If the neighbourhood is small enough, and $\boldsymbol{h}$ and $\mathcal{D}$ are smooth, the averages $\langle \boldsymbol{h}(\boldsymbol{x}_n) \rangle$ and $\langle \mathcal{D}(\boldsymbol{x}_n) \boldsymbol{\epsilon}_n \rangle$ can be replaced by $\boldsymbol{h}(\boldsymbol{x}_n)$ and $\mathcal{D}(\boldsymbol{x}_n) \langle \boldsymbol{\epsilon}_n \rangle$ respectively.

Since the average over the gaussian noise is zero, one obtains directly the drift function:

$$\boldsymbol{h}(\boldsymbol{x}) = \langle \Delta \boldsymbol{x}_n \rangle. \tag{2.15}$$

In order to estimate the diffusion operator $\mathcal{D}$ the product $\langle \Delta \boldsymbol{x}_n \Delta \boldsymbol{x}_n^\dagger \rangle$ is regarded:

$$\langle \Delta \boldsymbol{x}_n \Delta \boldsymbol{x}_n^\dagger \rangle = \langle [\boldsymbol{h}(\boldsymbol{x}_n) + \mathcal{D}(\boldsymbol{x}_n) \boldsymbol{\epsilon}_n][\boldsymbol{h}(\boldsymbol{x}_n) + \mathcal{D}(\boldsymbol{x}_n) \boldsymbol{\epsilon}_n]^\dagger \rangle \tag{2.16}$$

$$= \ldots$$

$$= \boldsymbol{h}(\boldsymbol{x}_n) \boldsymbol{h}(\boldsymbol{x}_n)^\dagger + \mathcal{D}(\boldsymbol{x}_n) \langle \boldsymbol{\epsilon}_n \boldsymbol{\epsilon}_n^\dagger \rangle \mathcal{D}(\boldsymbol{x}_n) \tag{2.17}$$

with $(\langle \boldsymbol{\epsilon}_n \boldsymbol{\epsilon}_n^\dagger \rangle)_{ij} = \delta_{ij} \sigma^2$:

$$\sigma^2 \mathcal{D}(\boldsymbol{x}_n) \mathcal{D}(\boldsymbol{x}_n)^\dagger = \langle \Delta \boldsymbol{x}_n \Delta \boldsymbol{x}_n^\dagger \rangle - \boldsymbol{h}(\boldsymbol{x}_n) \boldsymbol{h}(\boldsymbol{x}_n)^\dagger, \tag{2.18}$$

where $\langle \Delta \boldsymbol{x}_n \Delta \boldsymbol{x}_n^\dagger \rangle - \boldsymbol{h}(\boldsymbol{x}_n) \boldsymbol{h}(\boldsymbol{x}_n)^\dagger$ is the local covariance matrix. The diffusion operator $\mathcal{D}(\boldsymbol{x_n})$ can then be estimated by a Cholesky decomposition.

2.2.3 Step-wise Drift-Diffusion Estimation

The beforehand described calculations can now be used to estimate a model time series from experimental or simulational data. To this end a start point is chosen. This might be a random point. To prevent transient motion in the model time series it is however advisable to choose a point from the library, e.g. the last point of the library time series. For this point only, the drift and diffusion fields are calculated. These are then used to estimate the

next point of the time series according to:

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + \boldsymbol{h}(\boldsymbol{x}_n)\Delta t + \mathcal{D}(\boldsymbol{x}_n)\boldsymbol{\epsilon}_n\sqrt{\Delta t}. \tag{2.19}$$

From this point, the next step is calculated in the same manner. Ultimately a time series with similar statistical properties to the library time series should be obtained. This can be used in a first step to validate the method used.

However since the method uses only local data to obtain a time series, from which statistical properties like a probability distribution can be calculated, it is not necessary to supply statistically converged library data. It is sufficient to use short time series, which sample the phase space well. From these short time series a model time series can be calculated, which then provides the statistical properties of interest for this system (see [11, 21]).

The input data needs to contain a sufficient amount of transitions from one metastable state to another for the algorithm described above to work properly. These transitions are usually scarce since the system spends most of the time in one of the metastable states and transitions between them happen rapidly and rarely.

Additionally the high frequency noise of the library data might need to be removed using a low-pass filter. Otherwise the noise might cause the system to be driven away from the populated area, towards possibly unphysical states. This is however not necessary for all systems. In this thesis only the trialanine molecules library data was submitted to a low-pass filter.

2.3 Kernel Density Estimation

The kernel density estimation is a method to estimate the probability density function f of a given variable x. It can be regarded as the smoother brother of a simple histogram and was created by Parzen and Rosenblatt individually [15, 16]. The kernel density estimator of a function f is:

$$f_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K\Big(\frac{x - x_i}{h}\Big). \tag{2.20}$$

As can be seen, it depends not only on the individual data points but also on the bandwidth h, which can be regarded as a smoothing parameter, and on the kernel K. The choice of bandwidth is crucial, since choosing a too small bandwidth would lead to the estimated density function containing artefacts arising from undersmoothing. If the bandwidth is however chosen to large, much of the structure of the density function might be smoothed away. To choose an appropriate bandwidth, several methods have been devised, however in this thesis, one of the simplest of them, Scott's rule [24] is used:

$$h = n^{-1/(d+4)}\sigma, \tag{2.21}$$

with n being the number of data points, d the dimensionality and σ the standard deviation of the data.
Another issue is the choice of an appropriate kernel function K. It may be a simple top-hat function, a gaussian or, as used here, the Epanechnikov kernel [19]:

$$K(u) = \begin{cases} \frac{3}{4}(1-u^2) & , u \leq 1 \\ 0 & , u \geq 1 \end{cases} \quad \text{with} \quad u = \frac{x - x_i}{h} \tag{2.22}$$

The kernel density estimator (equation 2.20) provides an estimate of the local density of a distribution, which can be used as a visualization alternative that is smoother than a simple histogram. On the other hand it can be used, as described above to provide weights for the landmark selection in the landmark variant of the ISOMAP algorithm.

Chapter 3

Dimension Reduction of a single time series

3.1 Trialanine

A 500000 steps long time series of the dihedral angles of trialanine, transformed to a metric coordinate space according to equations 2.4, is embedded into one to four dimensions using the two algorithms explained in sections 2.1.1 and 2.1.2.

The residual variances of the two embedding methods can be seen in figure 3.1. The ISOMAP algorithm outperforms PCA in all but the four dimensional embedding. This is to be expected, since the time series was initially presented in four dimensions and PCA as a linear algorithm performs projections only. Furthermore, the residual variance of the ISOMAP embedding shows a prominent 'knee' at two dimensions and enters a plateau afterwards, indicating the estimated intrinsic dimension of the trialanine system. This plateau is not present in the residual variance of PCA.

The effect of the chosen number of landmarks on the residual variance of the embedding can be seen in figure 3.2. The residual variance of the embedding using 500 landmarks forms a low outlier compared to the other embeddings. It does not change much for an increased number of landmarks. Since an increased number of landmarks means a larger graph on which the distances are calculated and a larger Gramian matrix, whose first P eigenvalues are

calculated for a P - dimensional embedding, the time needed to embed the high dimensional data increases.

In the same way one can determine whether the number of nearest neighbours considered in the graph building has an effect on the residual variance of the embedding (see figure 3.3). The residual variance is always lowest for 15 nearest neighbours used by the ISOMAP algorithm. However the difference to the embeddings using more neighbours becomes less for dimensions higher than one.

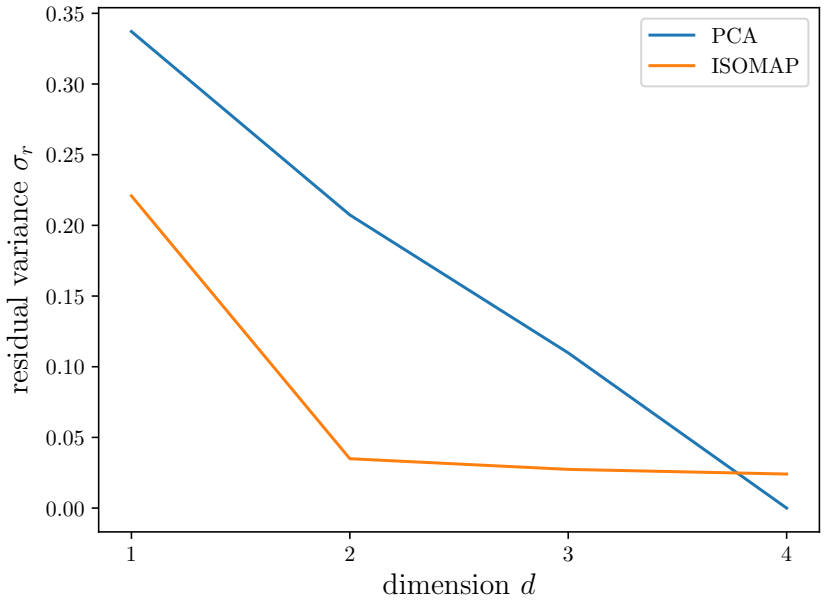

Figure 3.1: Residual variance σ_r against the number of dimensions d for both PCA and ISOMAP for a single time series of the trialanine molecule. The ISOMAP algorithm used $k = 20$ neighbouring points and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

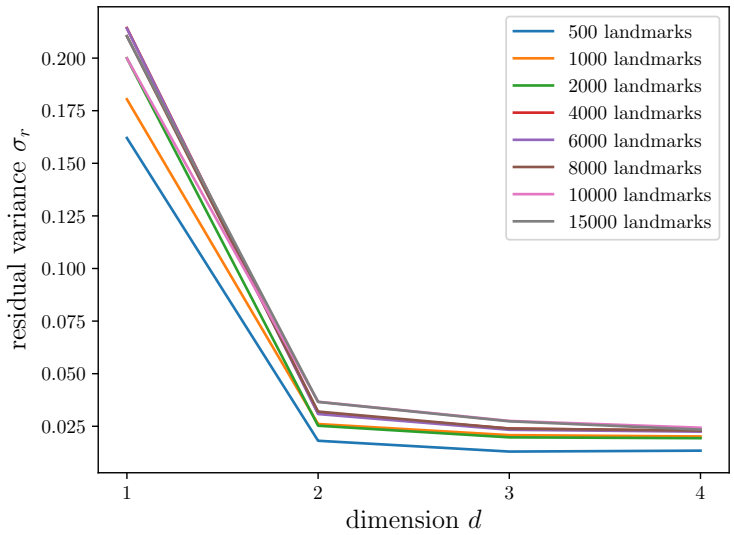

Figure 3.2: Residual variance σ_r against the number of dimensions d for a single time series of the trialanine molecule. The ISOMAP algorithm used $k = 20$ neighbouring points and various numbers of landmarks n_l chosen with the Kernel Density Weighted Random Selection.

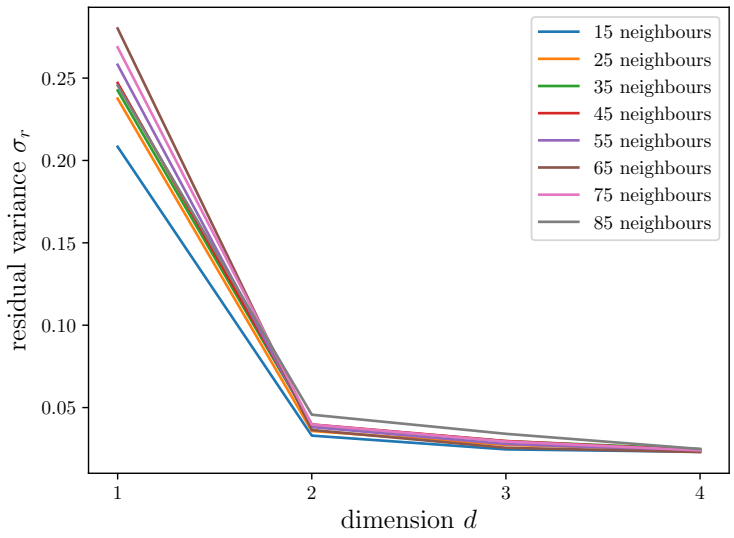

Figure 3.3: Residual variance σ_r against the number of dimensions d for a single time series of the Trialanine molecule. The ISOMAP algorithm used a varying number of neighbouring points k and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

In the kernel density estimations of the two dimensional embeddings in figures 3.4 and 3.5 the three main states of trialanine are clearly visible.The extended conformation β (42%) and the poly-L-proline II (P_{II}, 42%) are considerably more populated than the right-handed helix conformation α_R (15%).[11] As expected for a comparably low dimensional system, linear and nonlinear dimension reduction qualitatively show the same picture. The ISOMAP embedding however shows narrow transition paths between the more populated metastable states that are not visible in the PCA embedding.

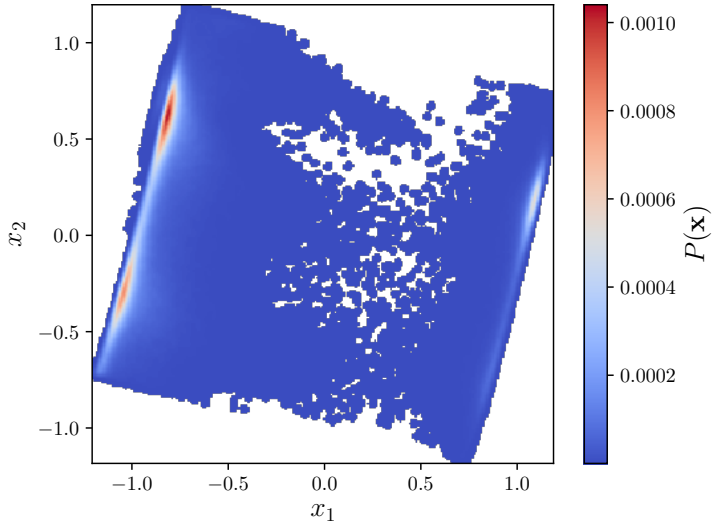

Figure 3.4: Kernel density estimation of the two dimensional PCA embedding of trialanine, using the Epanechnikov kernel with the bandwidth estimated by Scott's rule

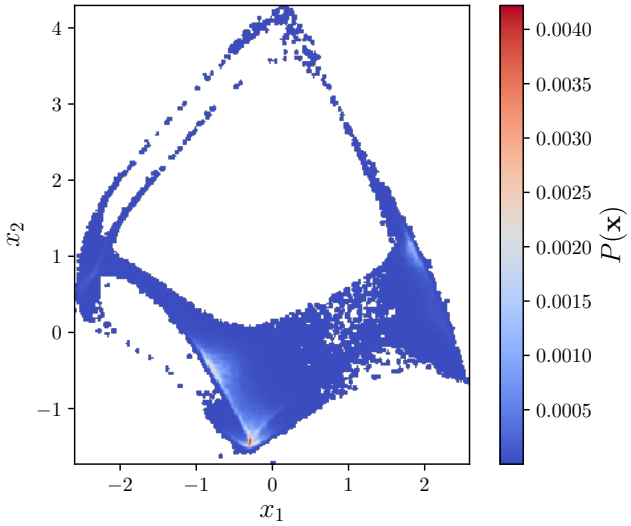

Figure 3.5: Kernel density estimation of the two dimensional ISOMAP embedding of trialanine, using the Epanechnikov kernel with the bandwidth estimated by Scott's rule. The embedding was computed using $k = 15$ nearest neighbours and $n_l = 15000$ landmarks chosen by the kde weighted selection.

3.2 Trp-cage

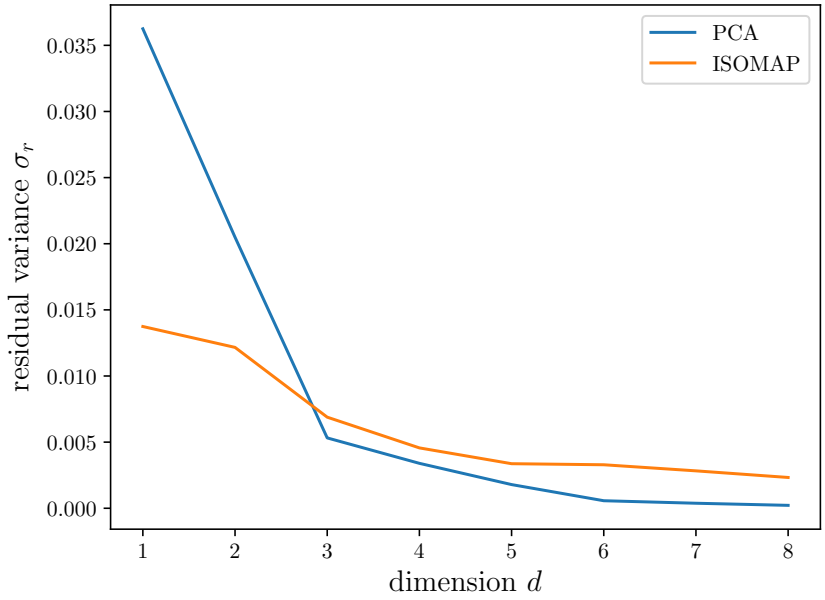

Figure 3.6: Residual variance σ_r against the number of dimensions d for both PCA and ISOMAP for a single time series of the Trp-cage molecule. The ISOMAP algorithm used $k = 20$ neighbouring points and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

A single time series of the unfolding of the Trp-cage molecule was performed at a temperature of 500 K. The length of the simulated time series is 200 ns, with an integration time step $\Delta t_i = 0.002$ ps and an output time step $\Delta t_o = 1$ ps.

For both the Trp-cage and the Aβ42 molecules, the embedding was only calculated on the atoms forming the peptide's backbone. Effectively this constitutes an initial dimension reduction. The positions of an atom in the amino acid is highly correlated to that of the backbone atom.

To find the embedding dimension of a single time series of the Trp-cage molecule, the residual variances for up to eight dimensional embeddings are calculated using both the PCA and the ISOMAP algorithm. A comparison of the residual variances of both algorithms can be seen in figure 3.6. The residual variance computed using the principal components embedding shows a prominent knee at three dimensions, entering a plateau afterwards.

This cannot be seen in the residual variances of the ISOMAP embedding, contrasting the results in sections 3.1 and 3.3. For two and three dimensions the ISOMAP algorithm produces a more accurate embedding of the system, this changes for higher dimensions, where PCA has a lower residual variance.
However it is possible, that the underlying system is one dimensional. In that case the residual variance has already entered the plateau in one dimension. This could especially be the case since, compared to the trialanine system in section 3.1, the residual variances are quite low.
Figure 3.7 suggests that the number of landmarks used has no great effect on the residual variance of the embedding. The embedding using 8000 landmarks forms an outlier in the residual variance. Since the landmarks were chosen using the Kernel Density Weighted Random Selection, it is possible, that in this case the landmarks are badly chosen and are misleading for the ISOMAP algorithm. All the other residual variances are close, especially the residual variances associated with the highest numbers of landmarks, 10000 and 15000 landmarks, are practically identical.
A very similar picture can be seen for a varying number of neighbours considered for the graph building. Figure 3.8 shows the residual variance for various numbers of neighbours k. All curves lie close to each other. For none of the values of k the residual variance shows a distinct plateau.
In figures 3.7 and 3.8 the difference between the choice of parameters almost vanishes for dimensions larger than five.
The two dimensional PCA embedding of the Trp-cage time series can be seen in figure 3.9. It shows a bent path snaking through the embedding space. There are several areas in the embedding, where the system seems to accumulate. It could however be, that these are artefacts of the projection to low dimensional space, and states are represented close to each other, that are in fact not closely related. Contrasting this the ISOMAP embedding shows, as suggested above, a nearly one dimensional manifold, apart from two protrusions in the interval $x_1 = 0, ..., 500$. As discussed in section 2.1.3 PCA is not able to unravel curved manifolds, whereas ISOMAP is able to recover these pathways even though they are bent in several dimensions.

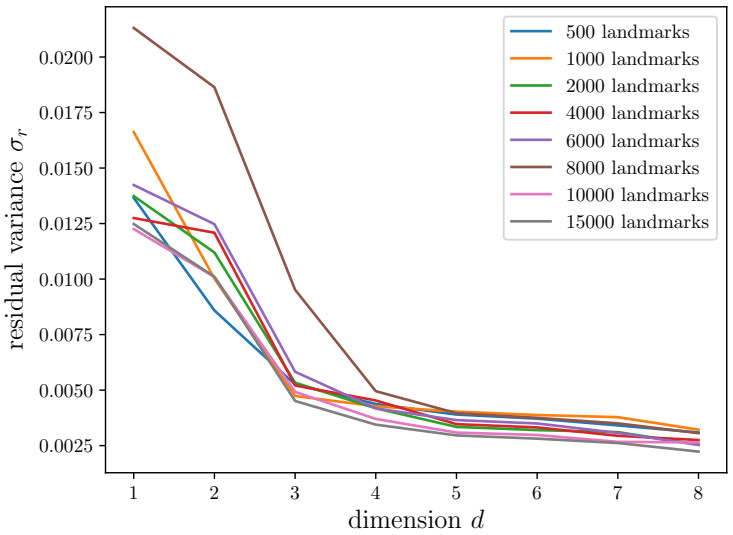

Figure 3.7: Residual variance σ_r against the number of dimensions d for a single time series of the Trp-cage molecule. The ISOMAP algorithm used $k = 20$ neighbouring points and various numbers of landmarks n_l chosen with the Kernel Density Weighted Random Selection.

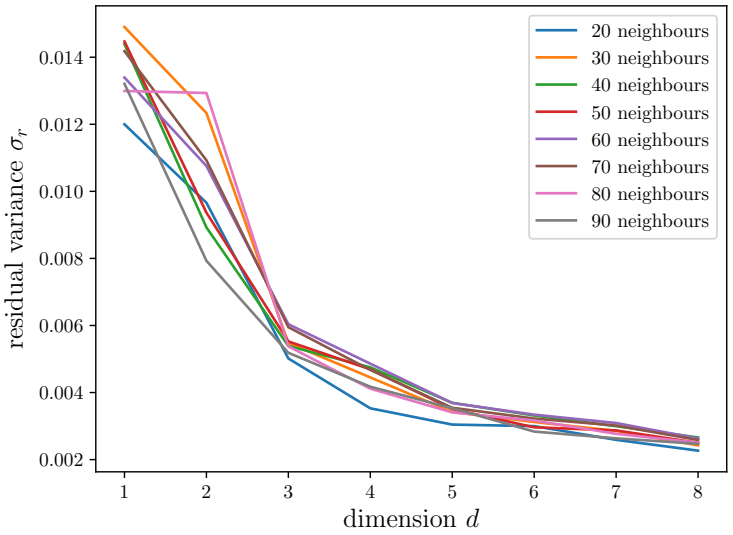

Figure 3.8: Residual variance σ_r against the number of dimensions d for a single time series of the Trp-cage molecule. The ISOMAP algorithm used a varying number of neighbouring points k and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

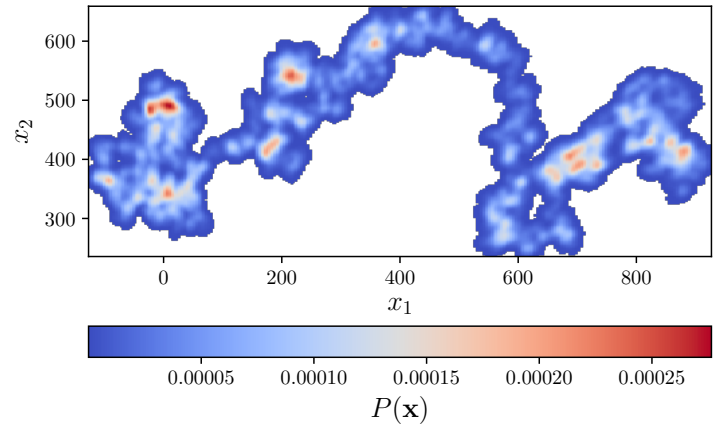

Figure 3.9: Kernel density estimation of the two dimensional PCA embedding of Trp-cage, using the Epanechnikov kernel with the bandwidth estimated by Scott's rule

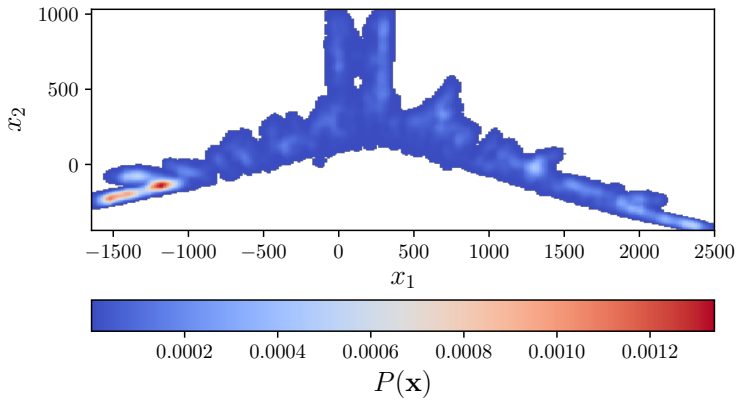

Figure 3.10: Kernel density estimation of the two dimensional ISOMAP embedding of Trp-cage, using the Epanechnikov kernel with the bandwidth estimated by Scott's rule. The embedding was computed using $k = 20$ nearest neighbours and $n_l = 15000$ landmarks chosen by the kde weighted selection.

3.3 Aβ42

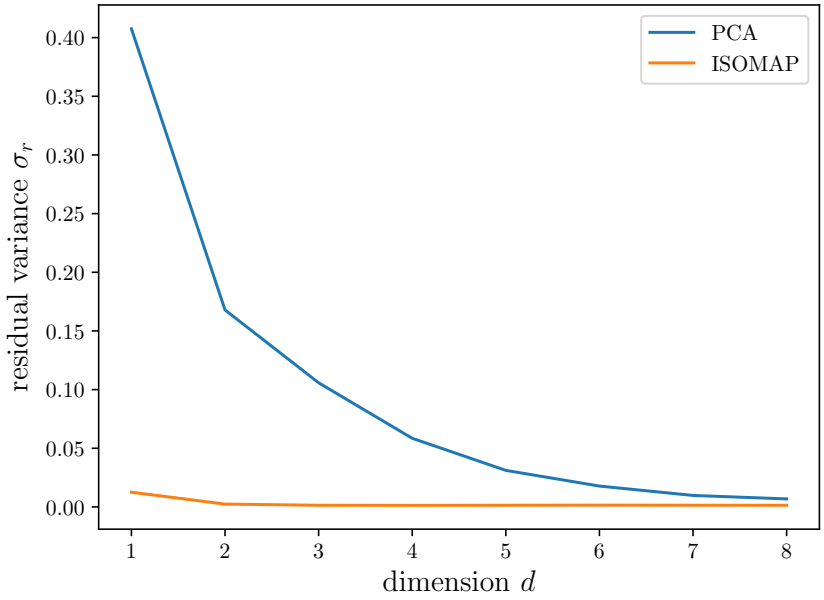

Figure 3.11: Residual variance σ_r against the number of dimensions d for both PCA and ISOMAP for a single time series of the Aβ42 molecule. The ISOMAP algorithm used $k = 20$ neighbouring points and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

The single time series of the unfolding of Aβ42 was obtained by an MD-simulation at 303 K. The time series is 100 ns long, with an integration time step $\Delta t_i = 0.002$ ps and an output time step $\Delta t_o = 1$ ps.

The residual variance of the dimension reduction of a single time series of the Aβ42 molecule calculated for the first eight embedding dimensions for both PCA and ISOMAP can be seen in figure 3.11. The nonlinear ISOMAP performs much better than the linear PCA. Figure 3.11 suggests that the problem is two dimensional since the residual variance σ_r changes only marginally for higher dimensions. In figure 3.12 the residual variance can be seen for different numbers of landmarks. As expected, the residual variance decreases with an increase in landmarks used for the embedding. In the case of 15000 landmarks used the residual variance plot even suggests a one dimensional embedding.

The residual variance is smallest for 20 or 30 neighbours used to build the

network (see figure 3.13). The clear gap between the residual variances for 20 and 30 neighbours, and the other values of k might indicate an edge of the network that shortcuts through the manifold, which is added as more neighbours are considered and thus increases the estimated dimension.
As figure 3.12 suggests, the time series of the Aβ42 molecule considered here can be described in only one dimension, if enough landmarks are used. This only becomes visible when using 15000 or more landmarks, as a denser set of landmarks prevents shortcutting errors.
Figures 3.14 and 3.15 show the two dimensional PCA and ISOMAP embeddings. The ISOMAP embedding shows a more or less straight line, as is to be expected for a one dimensional data set. The initially 501 dimensional time series, again only backbone atoms were considered, can be described in only one dimension, apart from two outliers. A different picture is drawn by the PCA embedding, which, according to figure 3.11 suggests a much higher dimensional embedding than ISOMAP. Consequently the two dimensional PCA embedding shows a more ambiguous picture showing a curved area along which ISOMAP presumably puts its first component.

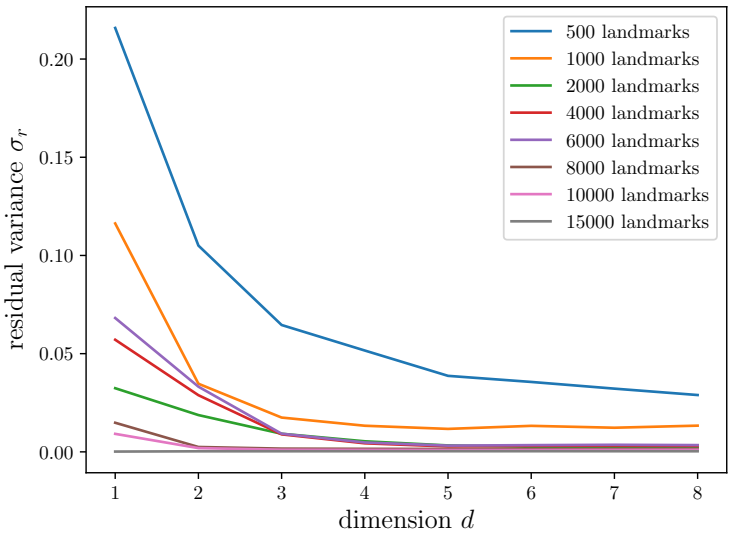

Figure 3.12: Residual variance σ_r against the number of dimensions d for a single time series of the Aβ42 molecule. The ISOMAP algorithm used $k = 20$ neighbouring points and various numbers of landmarks n_l chosen with the Kernel Density Weighted Random Selection.

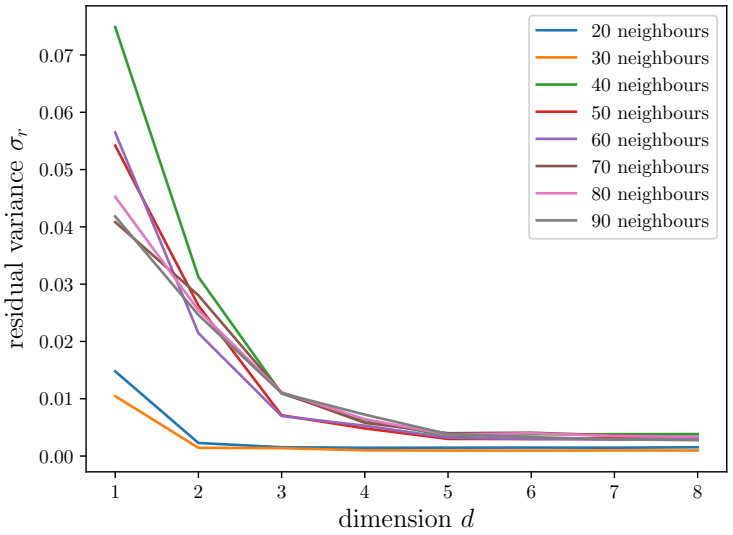

Figure 3.13: Residual variance σ_r against the number of dimensions d for a single time series of the Aβ42 molecule. The ISOMAP algorithm used a varying number of neighbouring points k and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

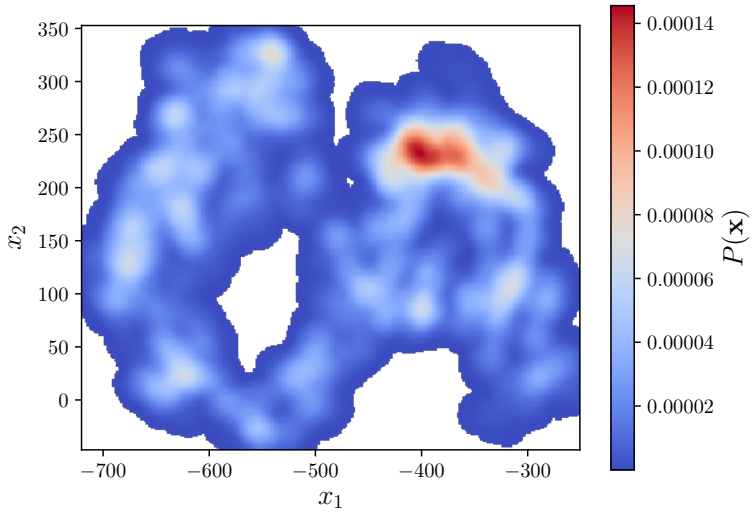

Figure 3.14: Kernel density estimation of the two dimensional PCA embedding of Aβ42 , using the Epanechnikov kernel with the bandwidth estimated by Scott's rule

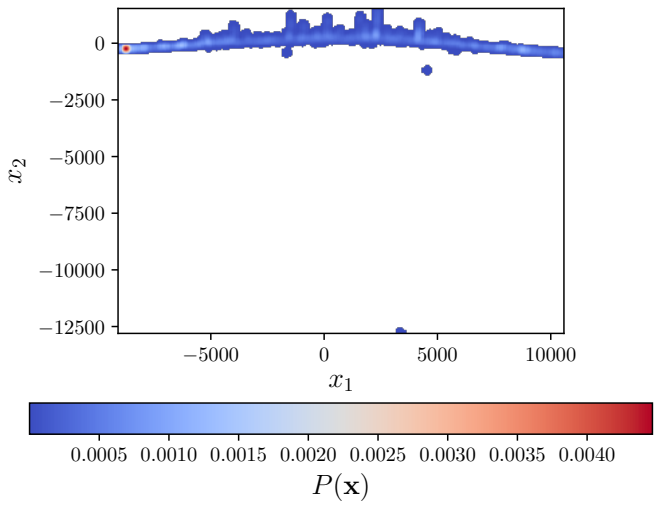

Figure 3.15: Kernel density estimation of the two dimensional ISOMAP embedding of Aβ42 , using the Epanechnikov kernel with the bandwidth estimated by Scott's rule. The embedding was computed using $k = 20$ nearest neighbours and $n_l = 15000$ landmarks chosen by the kde weighted selection.

Chapter 4

Dimension Reduction of multiple time series

4.1 Trialanine

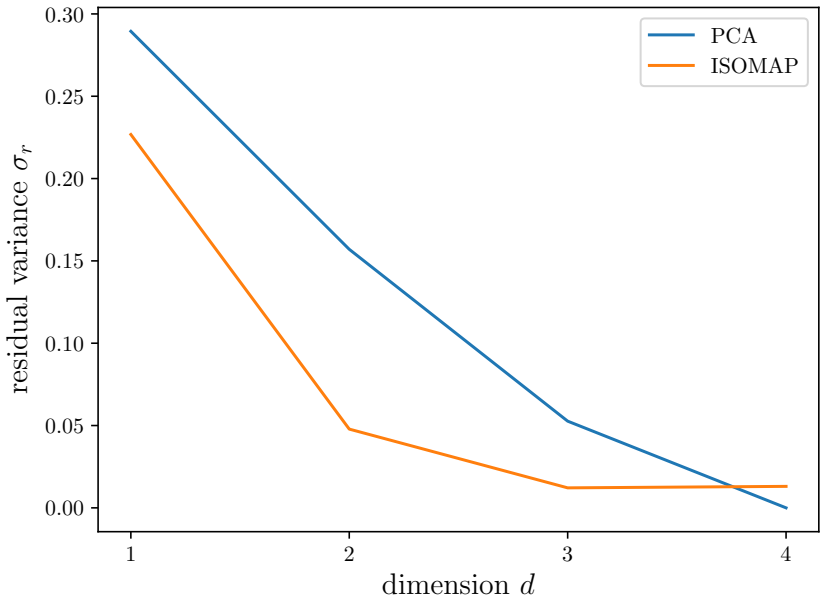

Figure 4.1: Residual variance σ_r against the number of dimensions d for both PCA and ISOMAP for multiple time series of the trialanine molecule. The ISOMAP algorithm used $k = 20$ neighbouring points and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

To study the behaviour of the dimension reduction algorithms being given multiple short time series, the time series of the trialanine molecule from section 3.1 was split into 100 short time series with 1000 steps each.
The system was then analysed in a similar manner as in the previous section. The residual variance of the PCA and ISOMAP embeddings is compared in figure 4.1. Again the nonlinear ISOMAP algorithm performs better in the one to three dimensional embeddings. Only in the four dimensional embedding PCA outperforms ISOMAP, due to the input data being four dimensional.
The picture changes slightly regarding the influence of the number of landmarks (figure 4.2). The residual variance decreases drastically as more landmarks are used. In the case of a single time series (see figure 3.2) the residual variance increased slightly for a higher number of landmarks, but the change is not as pronounced as in the multi time series case.
The residual variances for different numbers of neighbours can be seen in figure 4.3. The residual variance for $k = 10$ nearest neighbours forms an outlier compared to the other residual variances, which lie close together.
The two dimensional embeddings obtained by PCA and ISOMAP can be seen in figures 4.4 and 4.5. While the PCA embedding looks similar to that of the one obtained by embedding the whole time series, there are obvious differences in the ISOMAP embedding. While the short transition paths are still embedded sufficiently, the long ones look like they are cut open. In a sense that is exactly what happened. Since the transitions are already rare, the transition paths are only sparsely populated. By splitting the single trialanine time series into 100 shorter ones, it seems that crucial information about these transitions was lost. With a part of the transition path missing, the nearest neighbour network has no connecting path on this network and thus embeds the loose ends, as if they were not belonging together.

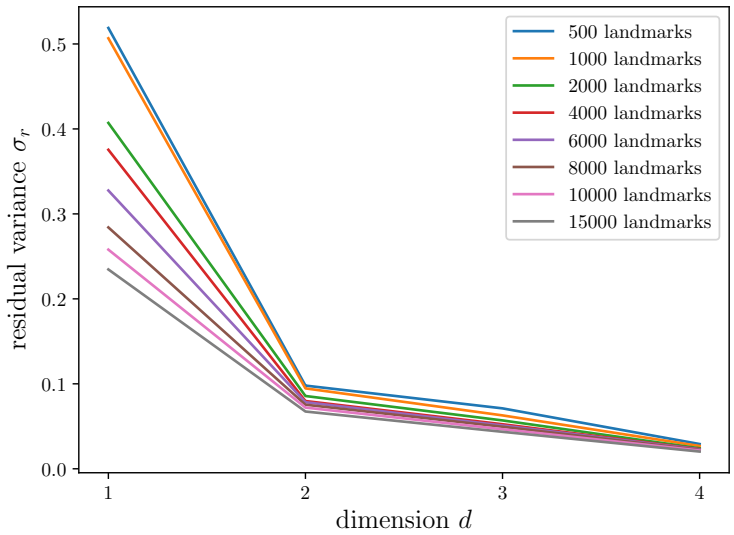

Figure 4.2: Residual variance σ_r against the number of dimensions d for multiple time series of the trialanine molecule. The ISOMAP algorithm used $k = 40$ neighbouring points and various numbers of landmarks n_l chosen with the Kernel Density Weighted Random Selection.

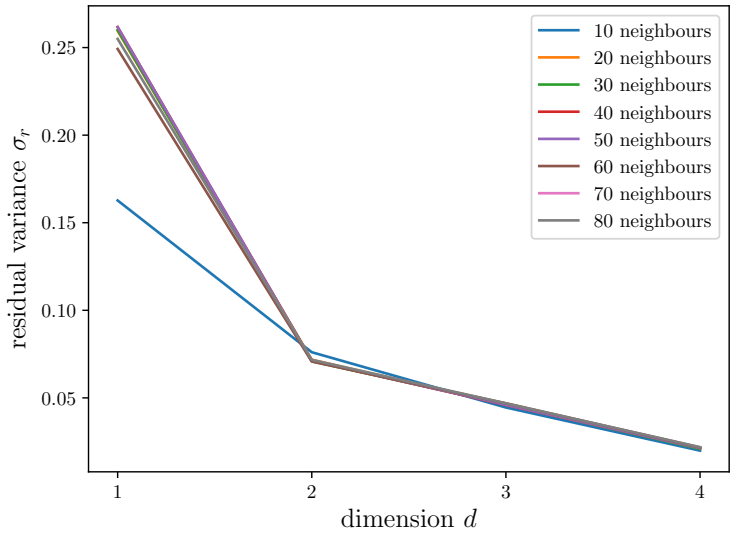

Figure 4.3: Residual variance σ_r against the number of dimensions d for multiple time series of the trialanine molecule. The ISOMAP algorithm used a varying number of neighbouring points k and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

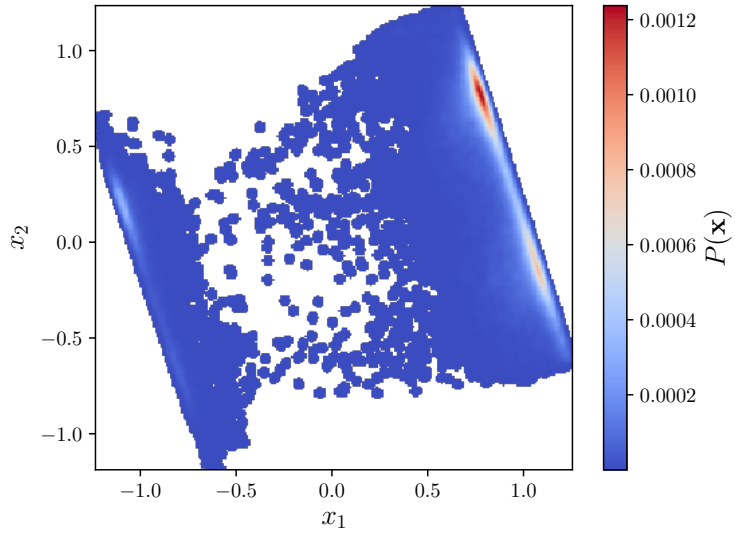

Figure 4.4: Kernel density estimation of the two dimensional PCA embedding of Aβ42 , using the Epanechnikov kernel with the bandwidth estimated by Scott's rule

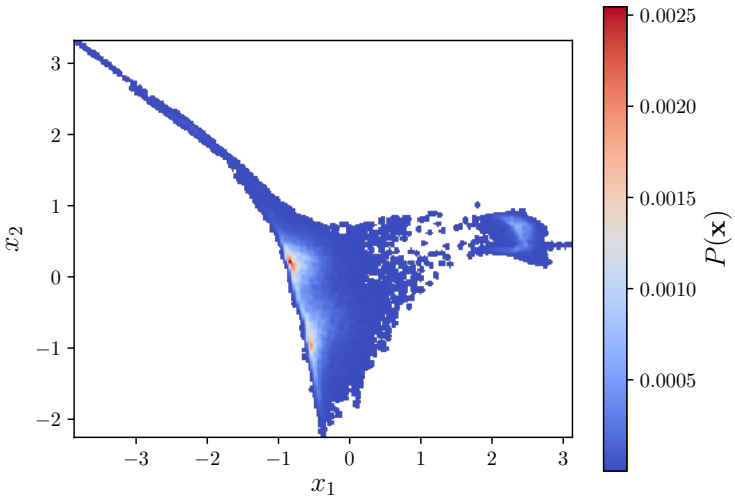

Figure 4.5: Kernel density estimation of the two dimensional ISOMAP embedding of Aβ42 , using the Epanechnikov kernel with the bandwidth estimated by Scott's rule. The embedding was computed using $k = 10$ nearest neighbours and $n_l = 15000$ landmarks chosen by the kde weighted selection.

4.2 Trp-cage

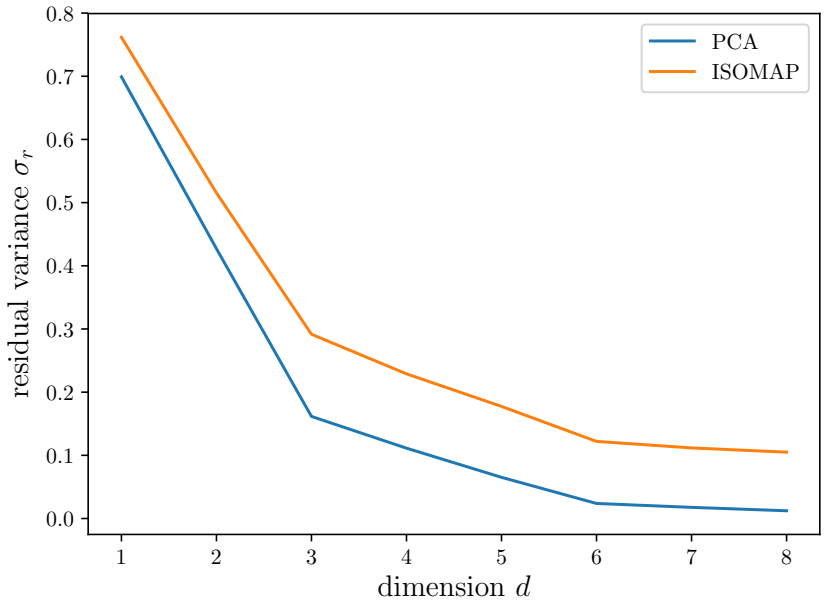

Figure 4.6: Residual variance σ_r against the number of dimensions d for both PCA and ISOMAP for multiple time series of the Trp-cage molecule. The ISOMAP algorithm used $k = 20$ neighbouring points and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

To obtain several short time series of the Trp-cage molecule, a MD-simulation at a temperature of 450 K was performed. From this time series 313 configurations were sampled. Each of these configurations provided the initial step for a separate MD-simulation. These simulations were performed at 300 K for 1 ns, with an integration time step $\Delta t_i = 0.002$ ps and an output time step $\Delta t_o = 0.002$ ns. In order to remove effects from the movement of a molecule as a whole, the coordinates of the atoms were transformed to coordinates relative to the molecule's centre of mass.

The dimensionality of the thus prepared library data was then reduced using the PCA and ISOMAP algorithms. The resulting residual variances are compared in figure 4.6. Rather unexpectedly the residual variance of the PCA embedding is, at least in the first 8 dimensions, lower than that of the ISOMAP embedding. Both residual variances show two knees, decreasing less after 3 dimensions, and again much less after 6 dimensions.

Again the influence of the number of landmarks n_l on the residual variance is explored. The residual variances of embeddings calculated using 200 to 10000 landmarks can be seen in figure 4.7. In contrast to the results of the trialanine molecule (figure 4.2) the residual variance increases for an increasing number of landmarks, albeit the increase is only slight. All chosen values of n_l produce embeddings with a similar residual variance.
The effect of the number of nearest neighbours on the embedding of Trp-cage (see figure 4.8) contrasts the trialanine results as well. Here for an increasing number of neighbours, the residual variance decreases. It seems that even for a comparably high number of neighbours, no shortcuts are made, that artificially increase the estimated embedding dimension.
The two dimensional embeddings calculated by the PCA (figure 4.9) and ISOMAP (figure 4.10) algorithms show a somewhat similar picture. The individual time series form a coherent area in both embedding methods. One of the main differences is the distribution of highly populated states. In the PCA embedding there are highly populated states scattered around the centre of the embedding space. The ISOMAP embedding shows much less states with high population. Still the time series spend most of the time in the center of the embedding space. Both embeddings show a few holes, where no information about the dynamics is provided by the library data. This could prove difficult for the drift diffusion estimation in section 5, making it necessary to exclude some of the time series to avoid undesired behaviour.

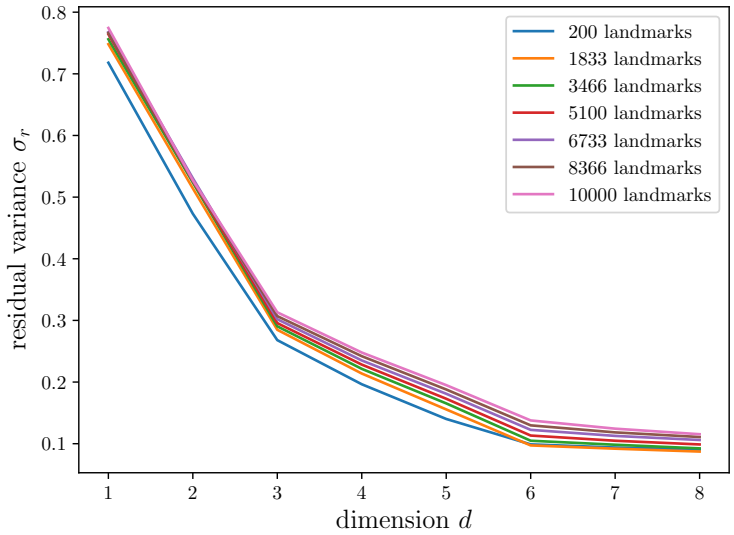

Figure 4.7: Residual variance σ_r against the number of dimensions d for multiple time series of the Trp-cage molecule. The ISOMAP algorithm used $k = 40$ neighbouring points and various numbers of landmarks n_l chosen with the Kernel Density Weighted Random Selection.

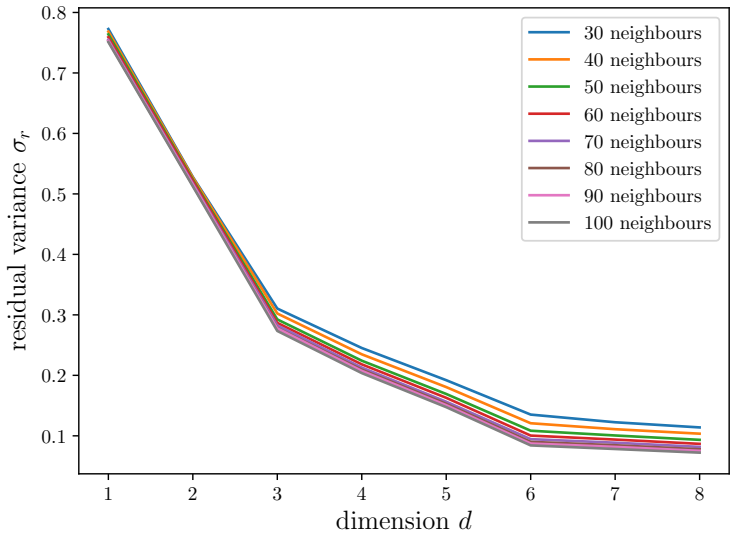

Figure 4.8: Residual variance σ_r against the number of dimensions d for multiple time series of the Trp-cage molecule. The ISOMAP algorithm used a varying number of neighbouring points k and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

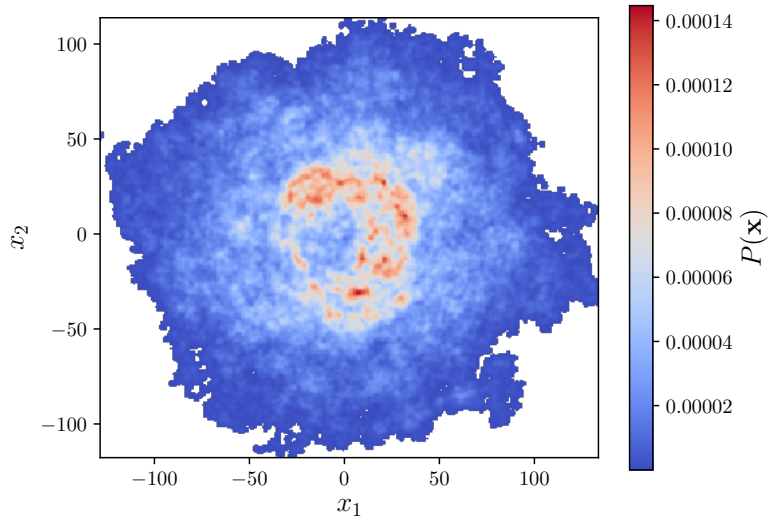

Figure 4.9: Kernel density estimation of the two dimensional PCA embedding of Trp-cage, using the Epanechnikov kernel with the bandwidth estimated by Scott's rule

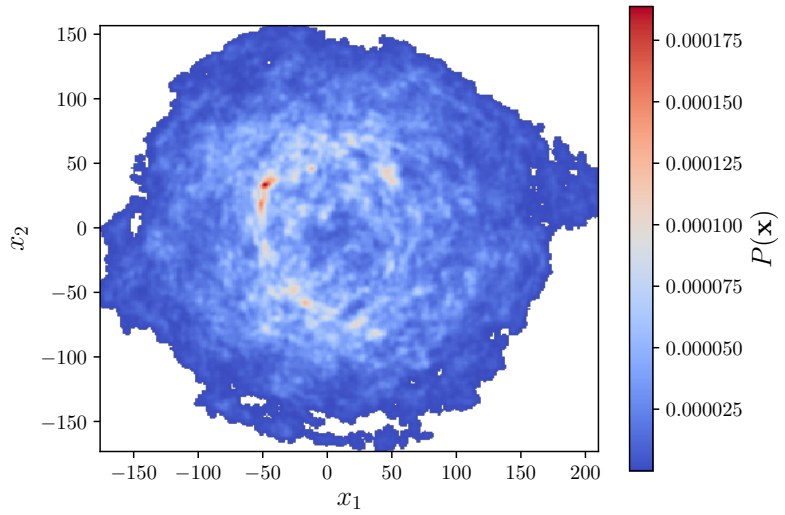

Figure 4.10: Kernel density estimation of the two dimensional ISOMAP embedding of Trp-cage, using the Epanechnikov kernel with the bandwidth estimated by Scott's rule. The embedding was computed using $k = 100$ nearest neighbours and $n_l = 10000$ landmarks chosen by the kde weighted selection.

4.3 Aβ42

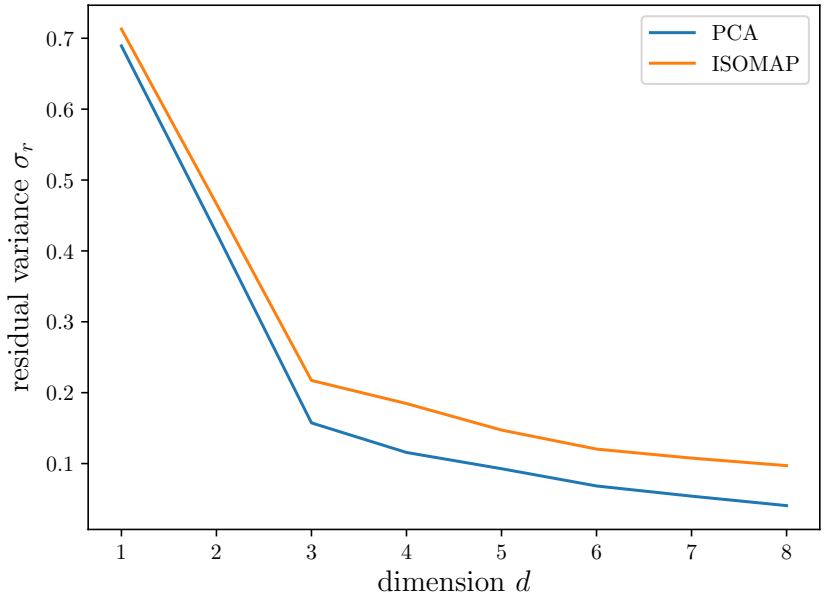

Figure 4.11: Residual variance σ_r against the number of dimensions d for both PCA and ISOMAP for multiple time series of the Aβ42 molecule. The ISOMAP algorithm used $k = 20$ neighbouring points and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

Similarly to section 4.2 the initial conditions for the multiple time series of the Aβ42 molecule were sampled from a high temperature simulation, again at 450 K. From this time series 208 initial configurations were extracted. These were again used as start points for separate MD-simulations. Each of these simulations was done at 300 K with an integration time step $\Delta t_i = 0.002$ ps and an output time step $\Delta t_0 = 0.002$ ns. The thus calculated time series are each 4 ns long.

These time series were again used as library data on which the dimensionality reduction algorithms PCA and ISOMAP were performed. Similar to the Trp-cage molecule in the previous section, the residual variance of the PCA embedding is, in the first 8 dimensions, lower than that of the ISOMAP embedding (figure 4.11). Both residual variances show a knee at 3 dimensions, they are however not very low, in both cases around 20% (PCA: 15.7%, ISOMAP: 21.7%) of the variance of the data is not explained by the low

dimensional embedding. However adding new dimensions only very slowly explains more variance.

A look at the residual variances using different numbers of landmarks (see figure 4.12) again shows the residual variance slightly increasing for a higher number of landmarks. The residual variances of the different embeddings are however still relatively close to each other.

A similar picture as in the previous chapter can be seen in figure 4.13 for a varying number of neighbours. The residual variance decreases for an increasing number of landmarks. There is no sharp increase in the estimated plateau dimension for a higher number of neighbours. That makes it likely, that no false connections are made in the network building step of the ISOMAP algorithm.

The two dimensional embeddings obtained by PCA (figure 3.14) and ISOMAP (figure 3.15) show very similar pictures, in the centre there are a few higher populated states, while the population decreases towards the outer regions of the embedding space. In these outer regions there are, in both embeddings, trajectories from single time series that traverse otherwise empty space. Again the gaps in the embedding can lead to unwanted artefacts in the drift diffusion simulation, these can either be dealt with by deleting the outlying time series or sampling additional time series in the undersampled regions.

Similar to the single time series cases in sections 3.2 and 3.3, PCA represents these outliers as trajectories that are quite bent in the embedding space, while ISOMAP is able to more or less straighten them out.

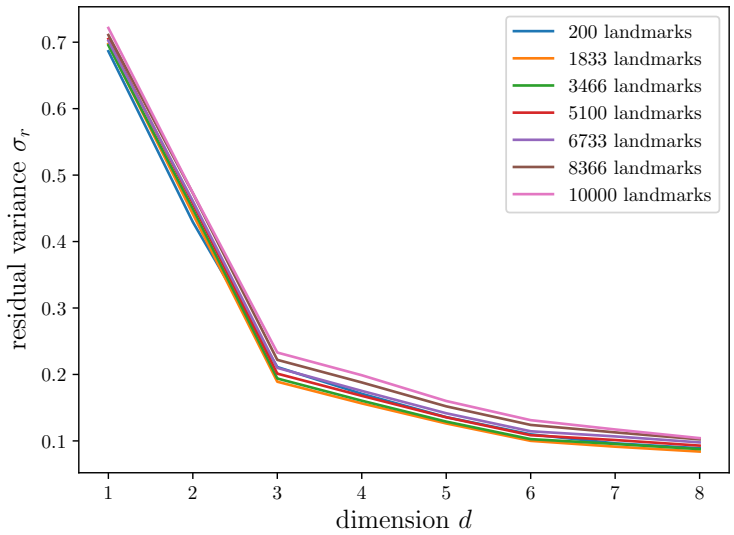

Figure 4.12: Residual variance σ_r against the number of dimensions d for multiple time series of the Aβ42 molecule. The ISOMAP algorithm used $k = 40$ neighbouring points and various numbers of landmarks n_l chosen with the Kernel Density Weighted Random Selection.

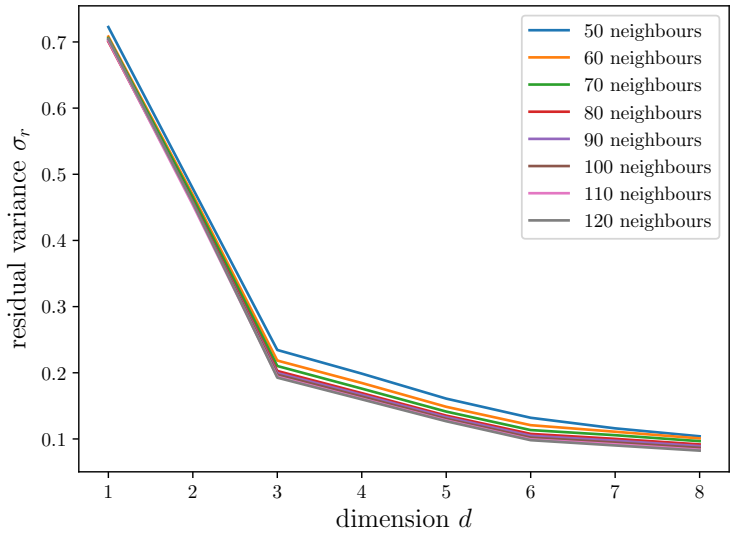

Figure 4.13: Residual variance σ_r against the number of dimensions d for multiple time series of the Aβ42 molecule. The ISOMAP algorithm used a varying number of neighbouring points k and $n_l = 10000$ landmarks chosen with the Kernel Density Weighted Random Selection.

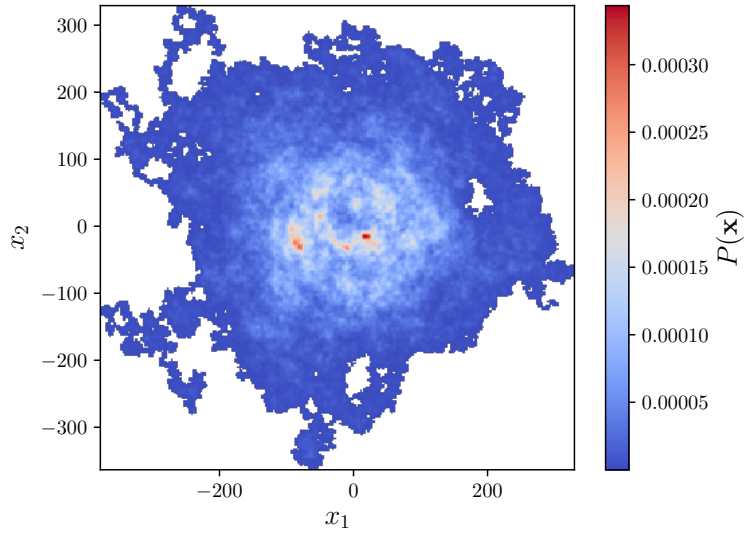

Figure 4.14: Kernel density estimation of the two dimensional PCA embedding of Aβ42 , using the Epanechnikov kernel with the bandwidth estimated by Scott's rule

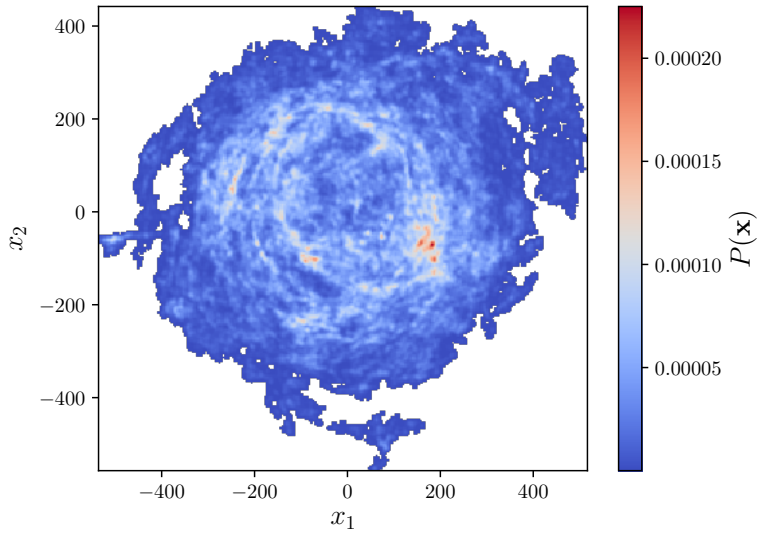

Figure 4.15: Kernel density estimation of the two dimensional ISOMAP embedding of Aβ42 , using the Epanechnikov kernel with the bandwidth estimated by Scott's rule. The embedding was computed using $k = 60$ nearest neighbours and $n_l = 10000$ landmarks chosen by the kde weighted selection.

Chapter 5

Reconstruction of Dynamics

5.1 Trialanine

5.1.1 Drift fields in the two dimensional representations

The drift fields $\boldsymbol{h}(\boldsymbol{x})$ of the trialanine time series on the two embeddings were calculated according to equation 2.15. Kernel density estimations of the PCA and ISOMAP embeddings overlaid with streamline plots of these fields can be seen in figures 5.1 and 5.2. At a first glance, both show a similar picture with the streamlines pointing towards highly populated states. Both show no drift away from the populated area which is very important for the step-wise drift-diffusion estimation (see section 2.2.3). If there would be drift pointing away from the populated area, the step-wise drift diffusion estimation could produce a time series that is moving away from the area where information about the dynamics of the system is present. As well both figures show no areas the system cannot escape using the stochastic driving. If there were an inescapable area, the step-wise drift diffusion estimation would yield a time series, that, after an initial transient, would spent all its time in this area, without exploring other parts of the phase space. Whether this happens cannot be ruled out completely at this point, since only the drift field is estimated.

A major difference between the two embeddings can be seen in the transition paths between the highly populated states. As already stated in section 3.1 the ISOMAP embedding shows distinct pathways between the metastable

states that cannot be distinguished in the PCA embedding. The estimated drift field reveals that these pathways are one-way streets connecting the states. Additionally, the streamlines of the drift field are less twisted in the ISOMAP embedding.

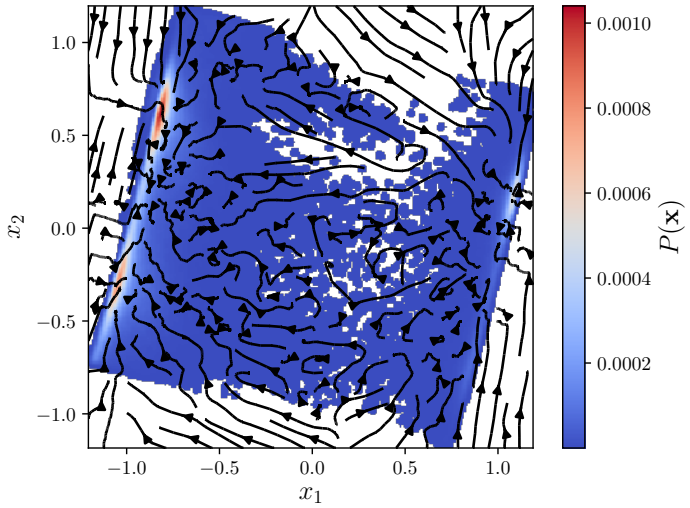

Figure 5.1: Kernel density estimation of the PCA embedded trialanine library time series overlaid with the drift field estimated according to eq. 2.15. $k = 100$ nearest neighbours were used for the local average.

5.1.2 Reproduction of known results

This section will describe the reproduction of known results from [11] in order to validate the code used throughout this thesis. As a first step, the dimensionality of the trialanine data set was reduced using the PCA algorithm (see sections 2.1.1 and 3.1) and high frequency noise was removed from this data set by a low-pass filter. Noise reduction became necessary because the high frequency noise caused the estimated time series to move towards areas not populated by the library data. On this dimension reduced and filtered data set, the step-wise Langevin algorithm, described in section 2.2.3 is used to estimate a new time series from the library data. A comparison between the library data and the rebuilt time series can be seen in figure 5.3. The distribution of the first two principal components is well resolved in the Langevin estimated time series.

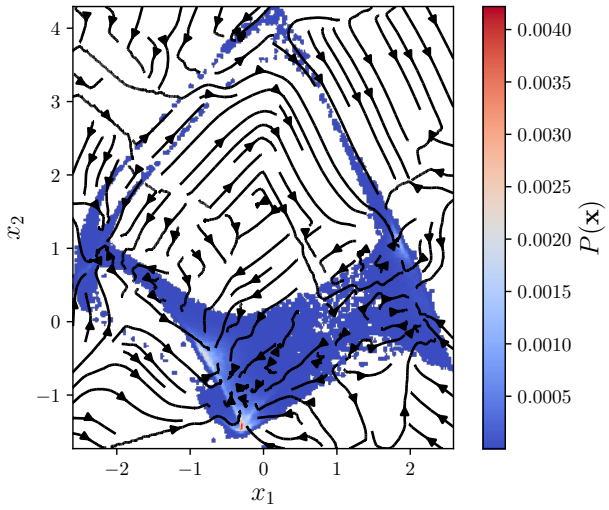

Figure 5.2: Kernel density estimation of the ISOMAP embedded trialanine library time series overlaid with the drift field estimated according to eq. 2.15. $k = 100$ nearest neighbours were used for the local average.

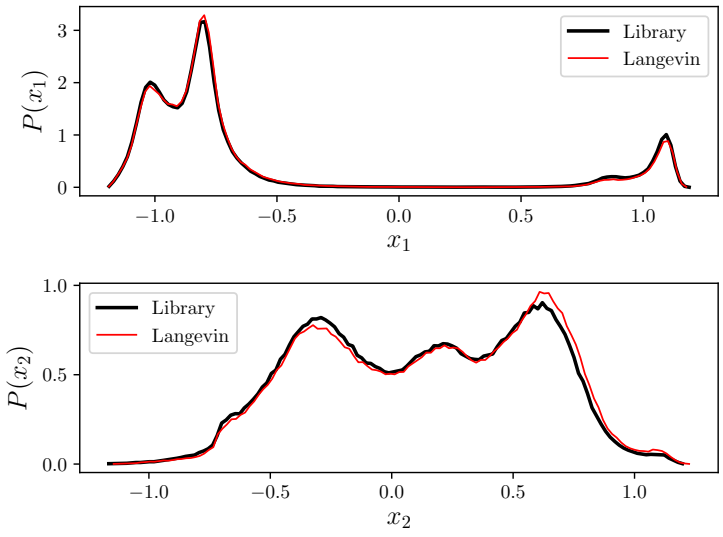

Figure 5.3: Distribution $P(\boldsymbol{x})$ of the first two principal components of a test trajectory of trialanine and the library data. 10 nearest neighbours were considered for the Langevin algorithm. A 500000 points long time series was created.

5.1.3 Dynamics on the curved embedding

To test, whether the ISOMAP embedding is suitable for the purposes of the step-wise drift diffusion estimation, a 500000 points time series was estimated, similarly to the estimation on the PCA embedding done previously. For the local average the $k = 10$ nearest neighbours were considered. The distribution of this estimated time series, along with the distribution of the library time series can be seen in figure 5.4. Similarly to the PCA embedding, the three metastable states of the trialanine molecule are well resolved in the estimated time series. The leftmost peak in the second collective coordinate is overestimated compared to the library data, that might be due to its high localization. However it is evident that, the collective variables calculated by the ISOMAP algorithm are suitable for the Langevin algorithm.

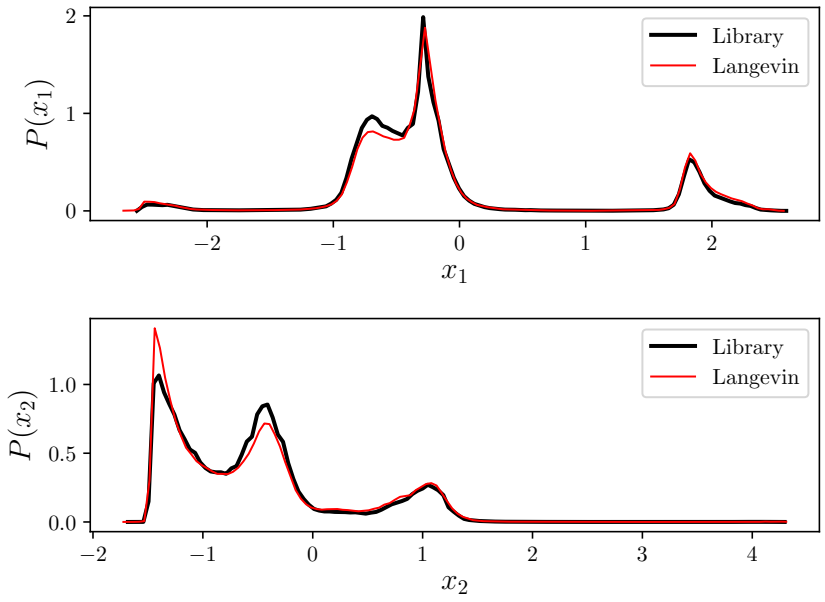

Figure 5.4: Distribution $P(\boldsymbol{x})$ of the first two ISOMAP coordinates of a test trajectory of trialanine and the library data. 10 nearest neighbours were considered for the Langevin algorithm. A 500000 points long time series was created.

5.1.4 Dynamics using a reduced library

Since the step-wise Langevin estimation of the dynamical properties of the system are only taken into account locally, it is not necessary, that the library data is correctly Boltzmann-weighted. The algorithm only takes into account local velocity information to estimate a time series from which, e.g. the probability distribution of the system can be calculated.
Therefore it is possible to discard redundant information from meta stable states, in order to significantly shorten the library data. Important for a correct estimation are the transitions from one metastable state to another. Those have to be sufficiently sampled in the library time series. Ideally a MD simulation would provide several time series that sample the phase space well and together have a rather flat distribution, avoiding having highly redundant information in the library time series.
In order to test the performance of the Langevin algorithm using a drastically reduced library, 100000 pairs of points were drawn from the trialanine time series. For both embeddings they were drawn, such that in the first embedding dimension, the distribution of the library time series is approximately flat. This ensures a reduction of redundant points in the library time series, while at the same time keeping all points in the transition pathways between the metastable states. Of each point nothing more than the next step in the library is known. Using this library, the step-wise drift-diffusion estimation is performed, for both the PCA (figure 5.5) and the ISOMAP (figure 5.6) embedding. Each time $k = 10$ nearest neighbours were used for the local average, and 500000 steps were simulated. In both embeddings, the step-wise drift-diffusion estimation is able to recover the correct distribution along the embedding variables from a flat library time series. In the ISOMAP embedding the largest peak is not as overestimated as in the drift-diffusion estimation using the complete time series. This might however be a statistical artefact that vanishes, when the estimations are done for a longer time.
Both figures quite impressively show the Langevin algorithms ability to retrieve a correctly Boltzmann-weighted distribution, from library data that is far from equilibrium.

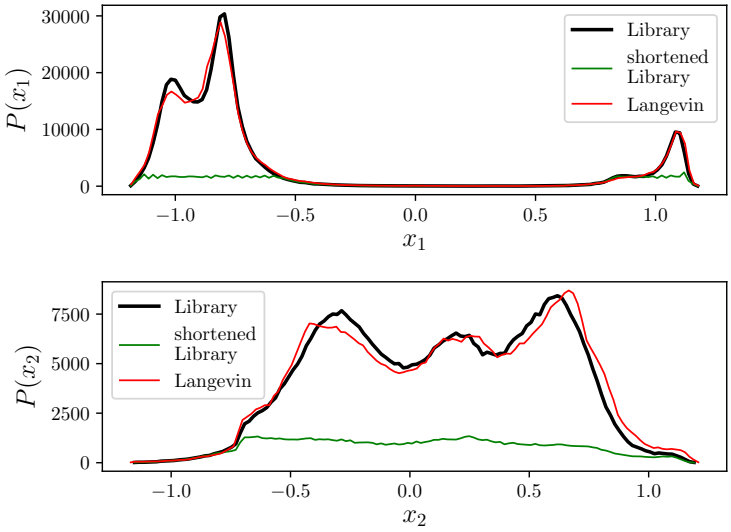

Figure 5.5: Probability distribution $P(\boldsymbol{x})$ of the first two PCA coordinates of trialanine. The original library time series is plotted in black. As input library a significantly shortened library time series was used (green). This shortened library is used as input for the Langevin algorithm. The distribution of the estimated time series is seen in red. $k = 10$ nearest neighbours were used in the local average.

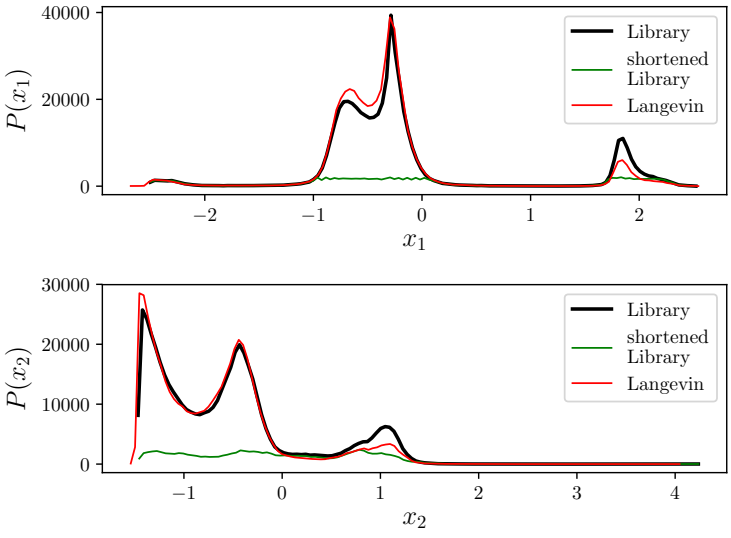

Figure 5.6: Probability distribution $P(\boldsymbol{x})$ of the first two ISOMAP coordinates of trialanine. The original library time series is plotted in black. As input library a significantly shortened library time series was used (green). This shortened library is used as input for the Langevin algorithm. The distribution of the estimated time series is seen in red. $k = 10$ nearest neighbours were used in the local average.

5.2 Trp-cage

5.2.1 Single time series drift fields in the two dimensional representations

The single time series kernel density estimations of the PCA and ISOMAP embeddings with overlaid streamline plots of the drift field $\boldsymbol{h}(\boldsymbol{x})$ can be seen in figures 5.7 and 5.8. The dynamics in the populated area are not as clearly visible as for the trialanine molecule (section 5.1.1). Again the drift field is pointing towards the populated area everywhere, ensuring the stability of the step-wise drift-diffusion estimation. However figure 5.8 might indicate an inescapable area in the lower right corner of the figure where the system might not be moved away by the stochastic driving.

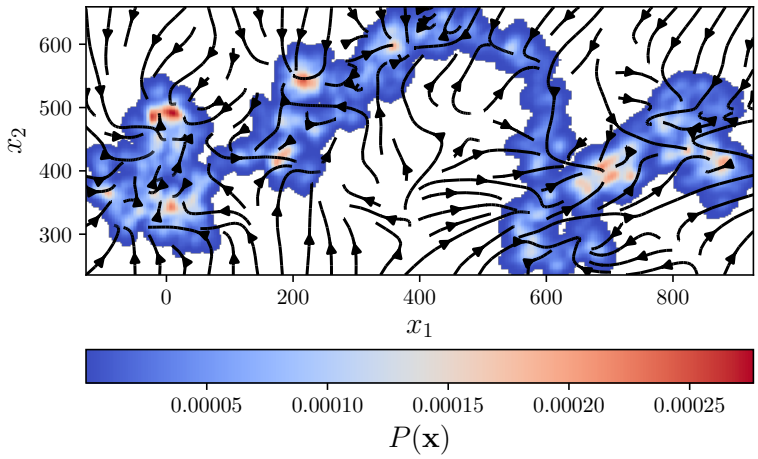

Figure 5.7: Kernel density estimation of the PCA embedded single Trp-cage library time series overlaid with the drift field estimated according to eq. 2.15. $k = 100$ nearest neighbours were used for the local average.

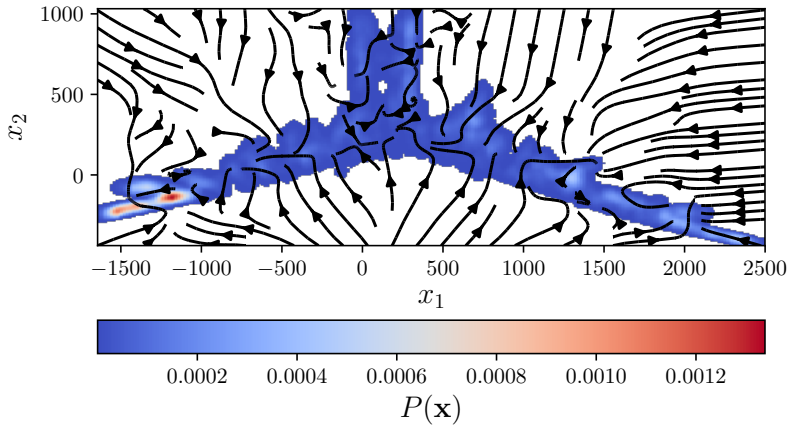

Figure 5.8: Kernel density estimation of the ISOMAP embedded single Trp-cage library time series overlaid with the drift field estimated according to eq. 2.15. $k = 100$ nearest neighbours were used for the local average.

5.2.2 Single time series step-wise drift-diffusion estimation

The single time series of the Trp-cage molecule was used as library data for the step-wise drift-diffusion estimation on both embeddings. The number of nearest neighbours considered in the local average was $k = 10$ and the start point for the estimated time series was chosen to be the first position of the library time series. The distribution $P(\boldsymbol{x})$ of the 2 μs long estimated time series can be seen in figure 5.9 for the PCA embedding and in figure 5.10 for the ISOMAP embedding. Both figures show a high peak, were the system accumulates after an initial transient, and a tail, which is formed by the points from said transient. If the time series were to be iterated further, the peak would only grow relative to the rest of the distribution. This is the behaviour that was expected from the drift fields in the previous section. The step-wise drift-diffusion estimation can only build trajectories that are already known in the library. Since there is no way back to other regions of the embedding space, the system is caught up at the end of the library time series. However the region at the end of the time series does not seem to be comprised of only one metastable state since in both figures there are at least to two discernible peaks. In the ISOMAP embedding the peaks are, in both dimensions, in a much smaller part of the embedding space. This might indicate, that the PCA embedding projects highly related states far from another. It is as well possible, that PCA projects unrelated conformations close to each other, leading to artefacts in the drift-diffusion estimation, that lead to a spacial spread of the estimated time series, that is not seen in the ISOMAP embedded drift-diffusion estimation.
However the purpose of the Langevin algorithm is the estimation of molecular dynamics on a large time scale and it is highly unlikely that the single time series has covered all of the relevant states of the system. This makes it necessary to sample a larger part of the embedding space, to obtain information about further transitions from one state to another.

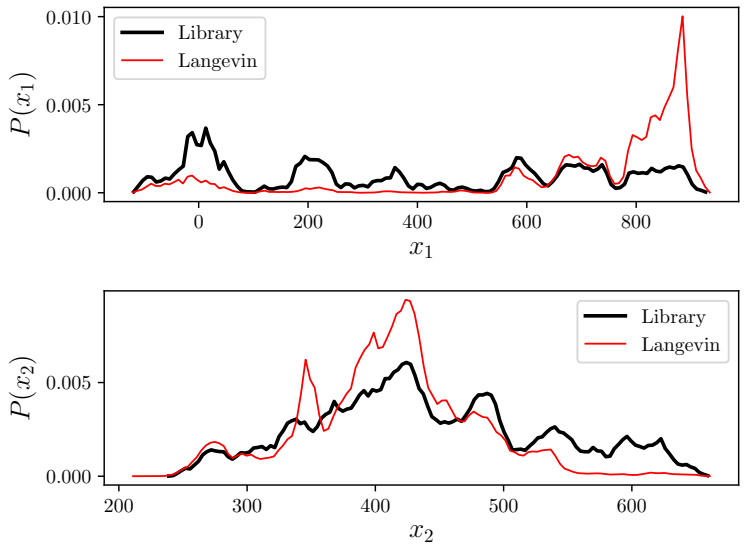

Figure 5.9: Distribution $P(\boldsymbol{x})$ of the first two principal components of a test trajectory and the library data of the single Trp-cage time series. 10 nearest neighbours were considered for the Langevin algorithm. A 2000000 points long time series was created.

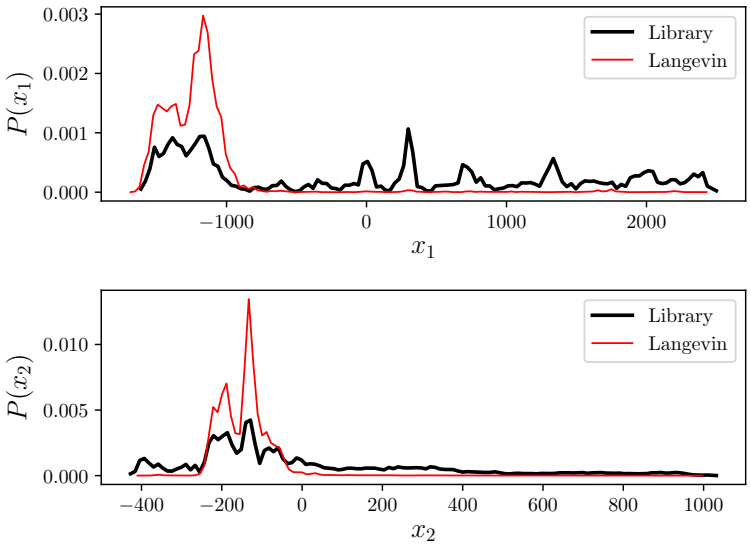

Figure 5.10: Distribution $P(\boldsymbol{x})$ of the first two collective coordinates of a test trajectory and the library data of the single Trp-cage time series. 10 nearest neighbours were considered for the Langevin algorithm. A 2000000 points long time series was created.

5.2.3 Multiple time series drift fields in two dimensions

To overcome the previously described problems of the single time series library data, 313 time series of length 1 ns were calculated (see section 4.2). These were then assembled to form the library data needed for the drift and diffusion estimations. The drift fields $\boldsymbol{h}(\boldsymbol{x})$ of the PCA and ISOMAP embeddings can be seen in figures 5.11 and 5.12. To estimate these fields the drift was calculated using a local average that considered the 10 nearest neighbours.

In both embeddings there are streamlines leading away from the populated area, which could cause problems in the step-wise drift-diffusion estimation. This estimation will however be done in a three dimensional embedding, as indicated in section 4.2. The streamlines leading away from the populated area could well be an artefact in a too low dimensional embedding that vanishes in an embedding with more appropriate dimensionality. Additionally the drift field shows no apparent inescapable areas, that would need further treatment. These two assertions will be validated by a step-wise drift-diffusion estimation on the three dimensional embedding.

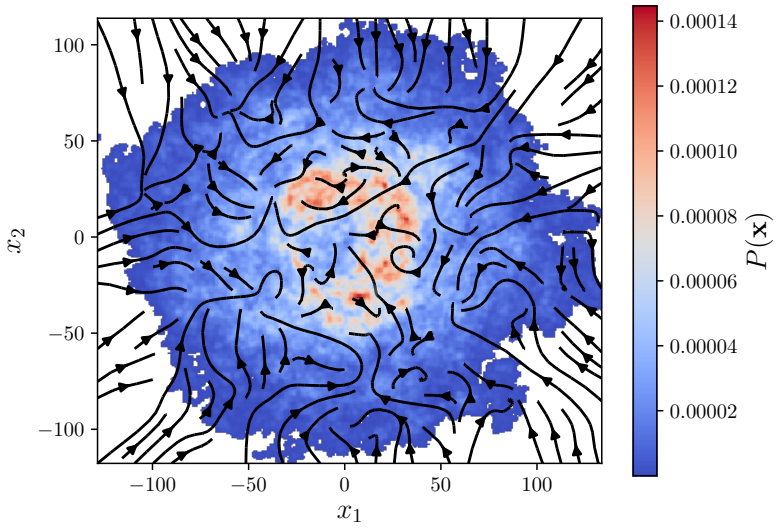

Figure 5.11: Kernel density estimation of the first two principal components of the PCA embedded multiple Trp-cage library time series overlaid with the drift field estimated according to eq. 2.15. $k = 100$ nearest neighbours were used for the local average.

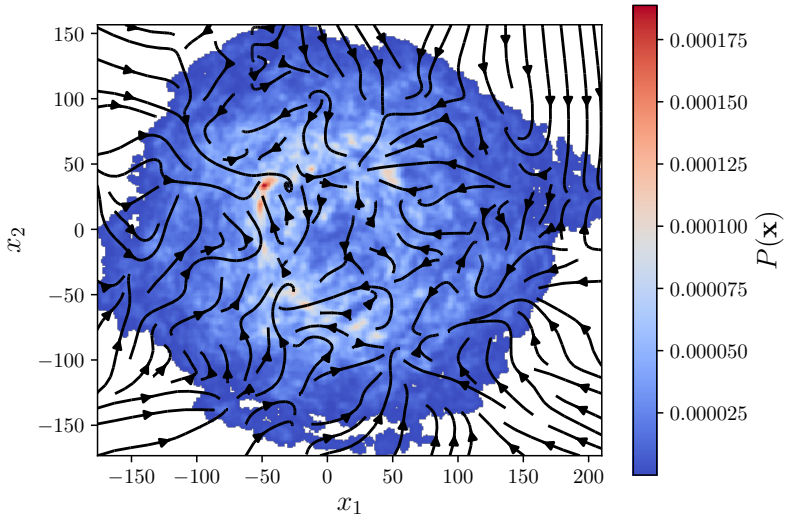

Figure 5.12: Kernel density estimation of the first two collective coordinates of the ISOMAP embedded multiple Trp-cage library time series overlaid with the drift field estimated according to eq. 2.15. $k = 100$ nearest neighbours were used for the local average.

5.2.4 Multiple time series drift-diffusion estimation

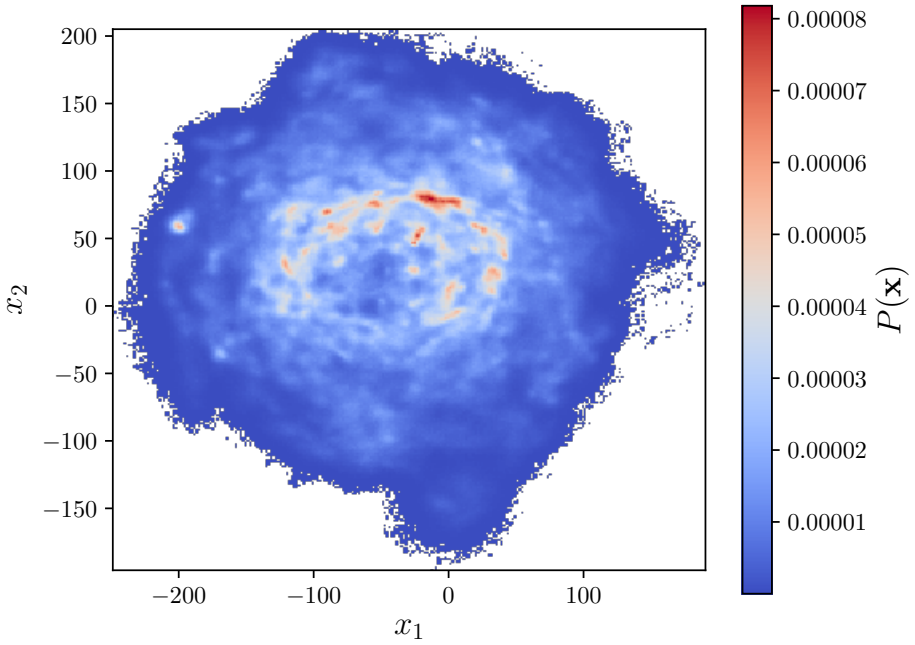

Figure 5.13: Distribution $P(\boldsymbol{x})$ of an estimated 40 μs trajectory of the Trp-cage molecule, projected onto the first two collective coordinates . $k = 10$ nearest neighbours were used for the local average.

The distribution $P(\boldsymbol{x})$ of a 40 μs long estimated time series using the ISOMAP embedded multiple time series of the Trp-cage molecule as input data can be seen in figure 5.13. $k = 10$ nearest neighbours were used to calculate the local averages. It shows several peaks around the centre of the embedding space. One additional peak is located at $(x_1, x_2) = (-200, 50)$. Since this is outside of the area described by the library time series, it is highly likely that this peak is due to an artefact. The local average of the drift at this point is calculated using far away library points, whose contribution to the drift cancels, leading to an accumulation. Apart from this peak, the distribution is largely in line with what was to be expected looking at the drift field in figure 5.12. The streamlines leading out of the populated area seen there seem to play no role in the drift diffusion estimation. This might be due to them being artefacts from the projection of the three

dimensional fields to two dimensions for visualization purposes. The actual step-wise drift-diffusion estimation was however done in three dimensions, possibly eliminating these artefacts.
There are however some differences visible between distribution of the library time series and that of the estimated time series. While the bulk of conformations is centred around $(x_1, x_2) = (-50, 50)$ in the library time series, the estimated time series spends most of its time around $(x_1, x_2) = (0, 75)$. That point is close to an area where many streamlines led to in figure 5.12, the exact position of this region might again be shifted by projection artefacts.

5.3 Aβ42

5.3.1 Drift fields in the two dimensional representations

The drift field $\boldsymbol{h}(\boldsymbol{x})$ is again calculated for both embedding methods for the single time series of the Aβ42 molecule (see figures 5.14 and 5.15). For the calculation of the local average 100 nearest neighbours were used. In both figures there are streamlines leading away from the populated area. Additionally the PCA embedding shows a few inescapable areas.
The drift field of the ISOMAP embedding in figure 5.15 nicely shows the essentially one dimensional nature of the time series. Points starting away from the populated area decay quickly towards this area and afterwards move with the flow on the x_1 axis. On the positive end of the x_1 axis the library time series provides no way back and the estimated dynamics will leave the boundaries of the system. Alternatively the drift towards infinity might be counteracted by the diffusive term in the Langevin equation (2.12), causing the estimated time series to accumulate at the end of the library time series.

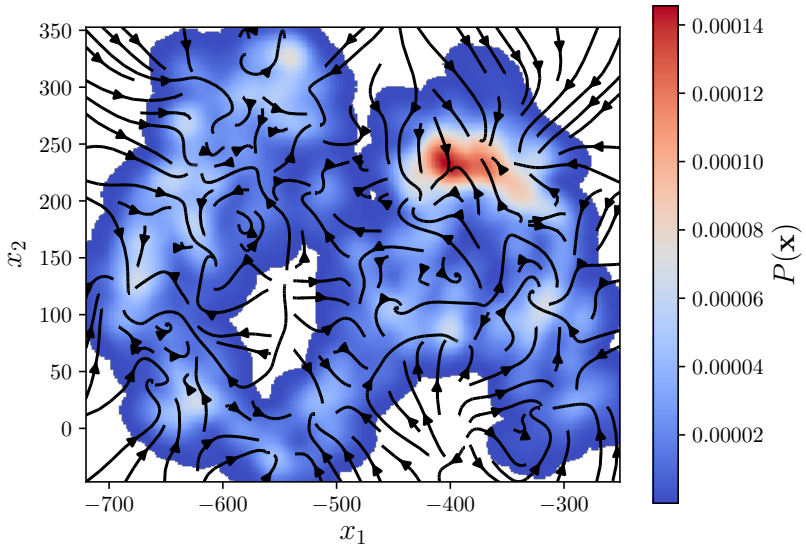

Figure 5.14: Kernel density estimation of the first two principal components of the PCA embedded single Aβ42 library time series overlaid with the drift field estimated according to eq. 2.15. $k = 100$ nearest neighbours were used for the local average.

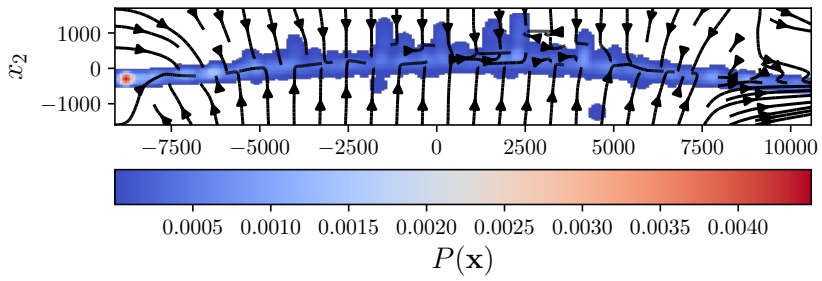

Figure 5.15: Kernel density estimation of the first two collective coordinates of the ISOMAP embedded single Aβ42 library time series overlaid with the drift field estimated according to eq. 2.15. $k = 100$ nearest neighbours were used for the local average.

5.3.2 Single time series step-wise drift-diffusion estimation

Similarly to section 5.2.2 a step-wise drift-diffusion estimation is performed, now with the single time series of the Aβ42 molecule. Again the number of nearest neighbours considered in the local average was chosen to be $k = 10$ and the start point of the estimated time series was the first point of the library time series. The length of the estimated time series is 1 μs. The distribution of the time series estimated using the PCA embedded library (see figure 5.16) shows two larger peaks in the first principal component and a wide distribution in the second. Contrasting this, the time series estimated using the ISOMAP embedded library (see figure 5.17) shows two peaks. One small peak is located at the start of both library and estimated time series around $x_1 = -9000$. Much more prominent however is the large peak around $x_1 = 10500$. It is very sharp indicating a low fluctuation in this state once it is reached.

The differences between the two embeddings that were already visible in the drift fields (section 5.3.1) become more evident in the drift diffusion estimation. While the time series estimated using the ISOMAP embedded library shows almost no fluctuation, once the area around the last point of the library time series is reached, there is no similar point in the PCA embedding. This is likely due to artefacts from the projection done by the PCA, that places conformations close to each other that show no related motion. This in turn increases the stochastic contribution in the step-wise drift-diffusion estimation, which might cause the broader peaks in the PCA embedding.

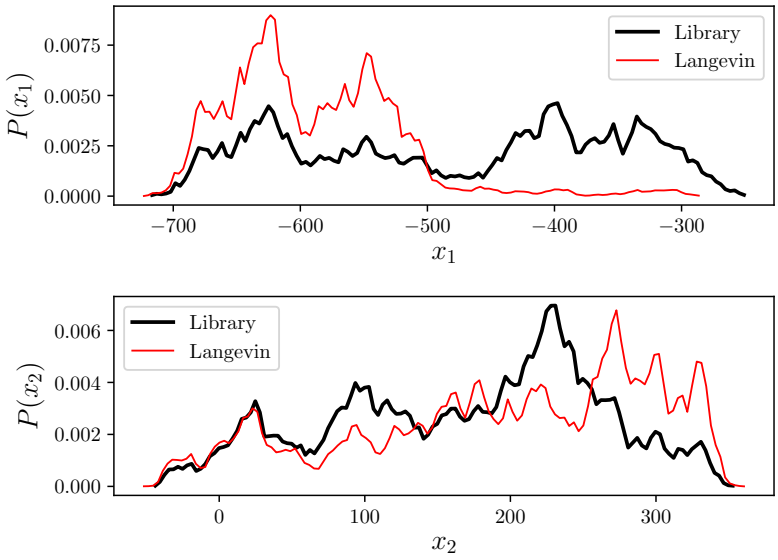

Figure 5.16: Distribution $P(\boldsymbol{x})$ of the first two principle components of a test trajectory and the library data of the single Aβ42 time series. 10 nearest neighbours were considered for the Langevin algorithm. A 2000000 points long time series was created.

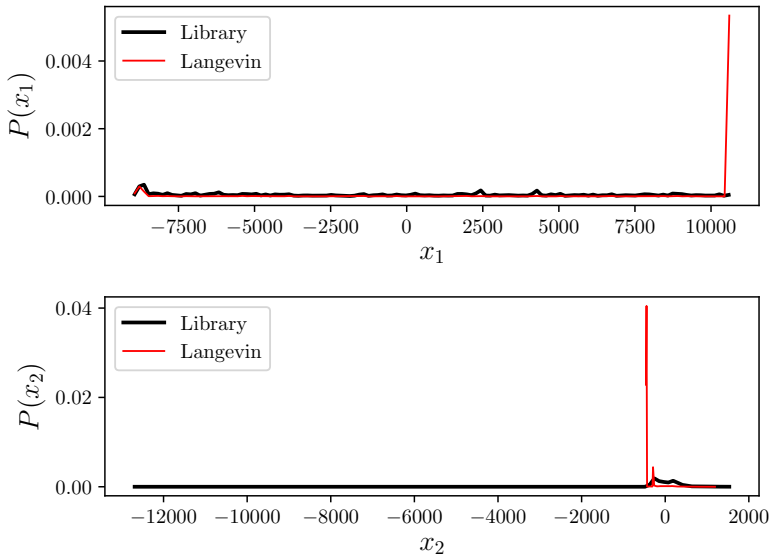

Figure 5.17: Distribution $P(\boldsymbol{x})$ of the first two collective coordinates of a test trajectory and the library data of the single Aβ42 time series. 10 nearest neighbours were considered for the Langevin algorithm. A 2000000 points long time series was created.

5.3.3 Multiple time series drift fields in two dimensions

In section 5.2 the single time series of the Trp-cage molecule proved inappropriate for the step-wise drift-diffusion estimation. This is again the case for the Aβ42 molecule. As described in section 4.3, 208 time series, each of length 4 ns were calculated. These were then assembled to form the library data for the estimation of drift and diffusion. The drift fields of the two embeddings can be seen in figures 5.18 (PCA) and 5.19 (ISOMAP). Both embeddings show some streamlines leading away from the populated area. Especially in the ISOMAP embedding there are several areas, were the step-wise drift-diffusion estimation could get stuck and be unable to escape driven by the stochastic part of equation 2.12. These areas are around $(x_1, x_2) = (-500, -80), (0, -80)$ and $(400, 150)$. They will be precisely identified in the following section.

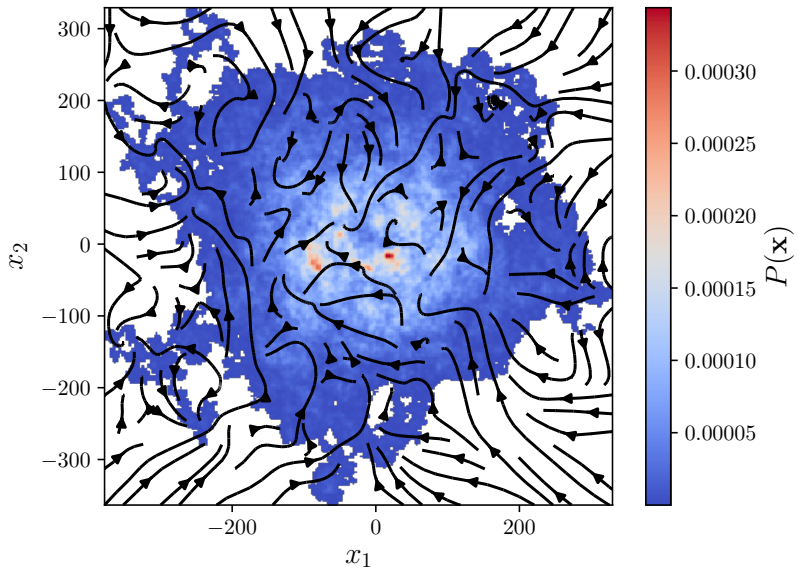

Figure 5.18: Kernel density estimation of the first two principal components of the PCA embedded multiple Aβ42 library time series overlaid with the drift field estimated according to eq. 2.15. $k = 100$ nearest neighbours were used for the local average.

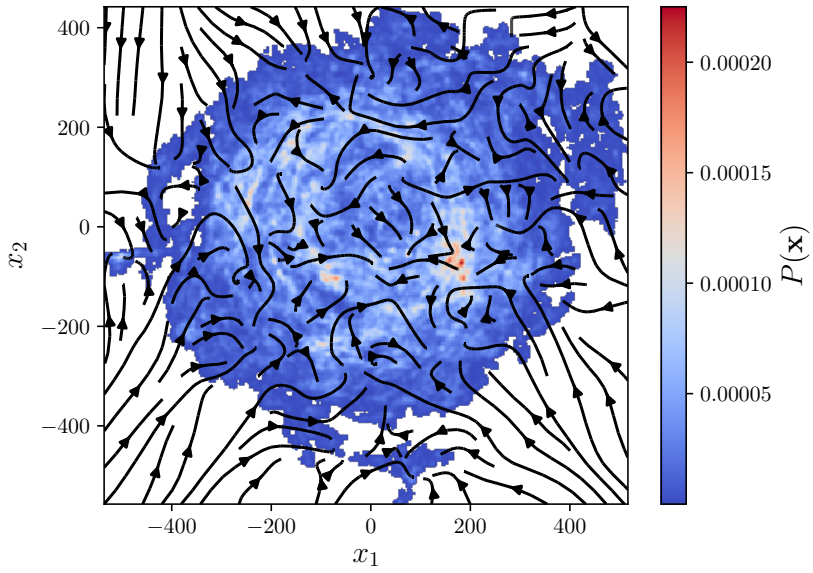

Figure 5.19: Kernel density estimation of the first two collective coordinates of the ISOMAP embedded multiple Aβ42 library time series overlaid with the drift field estimated according to eq. 2.15. $k = 100$ nearest neighbours were used for the local average.

5.3.4 Multiple time-series drift-diffusion estimation

Similar to the Trp-cage molecule in section 5.2.4, estimated time series were calculated in three dimensions using the ISOMAP embedded multiple time series of the Aβ42 molecule as library data. Again the number of neighbours in the local average was chosen to be $k = 10$. At first, thirty shorter estimated time series were calculated, starting from random initial conditions. After approximately 1500000 steps all of them were caught up in one of three inescapable regions. All of these accumulation areas where at the border of the area populated by the library time series, making a similar cause to that seen in the single time series case (section 5.3.2) likely. The individual library time series in the area around these three accumulation points were identified and all of the three points were lying at the end of a time series with no other time series close by. Thus these time series locally cause a situation similar to the endpoint of the single time series (see sections 5.2.2 and 5.3.2). To circumvent the problems arising from these time

series they were excluded from the library time series to ensure a meaningful estimation of the distribution $P(\boldsymbol{x})$.
This estimation yielded a 40 μs long time series whose distribution $P(\boldsymbol{x})$ can be seen in figure 5.20. It shows several highly populated states around the centre of the embedding.

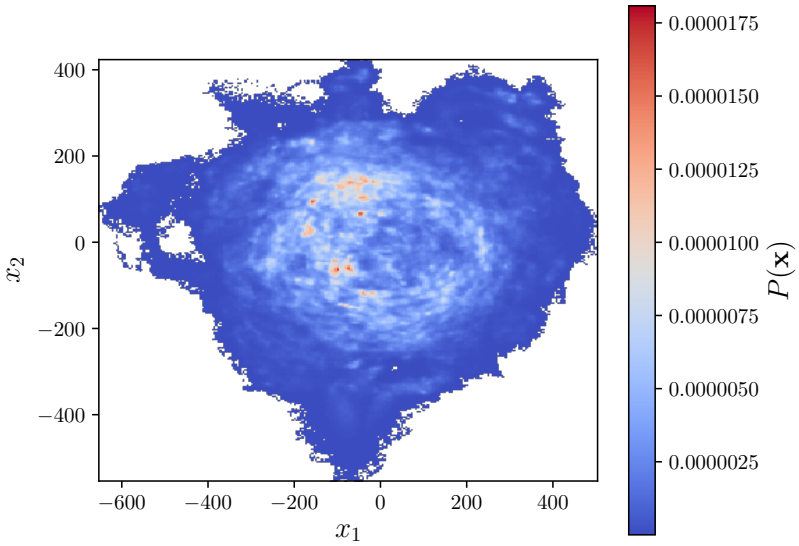

Figure 5.20: Distribution $P(\boldsymbol{x})$ of an estimated 40 μs trajectory of the Aβ42 molecule, projected onto the first two collective coordinates . $k = 10$ nearest neighbours were used for the local average.

Again there are differences between the distribution of the library time series and that of the estimated time series. While the main accumulations in figure 5.19 are around $(x_1, x_2) = (200, -100)$, they are spread wider in the estimated time series in figure 5.20. There they are mainly in an area between $x_1 = -200, ..., 0$ and $x_2 = -100, ..., 200$.
Similar to the Trp-cage molecule, the estimated time series briefly traverses areas, that are not covered by the library time series. There are however no significant peaks in these areas. Interestingly, one of these excursions is exactly in the area where one of the problematic time series was removed. The streamlines leading out of the populated area, seen in figure 5.19 did not affect the drift-diffusion estimation and might be due to an projection artefact.

Chapter 6

Conclusion

In the first part of this thesis the use of two different dimension reduction technique for molecular trajectories was demonstrated. The nonlinear ISOMAP algorithm initially showed a promising performance, in terms of the resulting embedding's residual variance, when simpler systems were considered in chapter 3. However for more complex systems, comprised of multiple shorter trajectories of a molecule, the linear PCA embedding had a lower residual variance. This is probably due to the approximative nature of the ISOMAP embedding. The pure ISOMAP algorithm already approximates distances on the low dimensional manifold by path distances. The landmark ISOMAP version takes the approximation another step further, embedding only a subset of the data points. Only those are used to form the network on which the embedding is performed. The remaining points are then reinserted into the embedding by a GPS-like triangulation. This approximation is however not only a weakness of this method but a strength as well, since, in principle, it enables the embedding of additional time series, after the initial network building and eigendecomposition of the Gramian matrix took place. This offers a computationally rather cheap way to enrich the library time series after an initial assessment of possible problematic areas.

The network building offers room for improvement as well. When multiple time series are used, they are not necessarily overlapping each other. It can take a large number of nearest neighbours, until all clusters of trajectories

are connected to each other and form one adjacency network. This is necessary for the distance calculation. Points from unconnected clusters have an infinite path length between them, which is impossible to embed in a meaningful way.

Isolated regions of the embedding space make a larger number of nearest neighbours necessary, which in other regions of the embedding space can lead to shortcutting artefacts, that might artificially increase the embedding dimension. A more sophisticated graph building method could circumvent these problems, by locally varying the number of neighbours k. An initial network of the landmarks with more neighbours than needed could be build. The landmarks are however not the only points that are embedded. The other points can now be used for an additional selection of the connections of the network. One can for example check, whether a path of the network traverses empty space, where no points, landmarks or not, are present. If that is the case this path needs to be removed from the network, since it is likely to form a shortcut.

Another approach would be to start the network with a relatively low number of neighbours. Then the unconnected clusters are identified and paths are added to the network, that connect these clusters to each other.

Both of these methods described above could be able to further reduce the residual variance of the ISOMAP embedding. It would however need to be determined which one performs better both computationally and in terms of the embedding's quality.

The efficient search for nearest neighbours in high dimensional space is still a major computational problem. The performance of the kd-tree nearest neighbour algorithm used here decreases drastically once the dimensionality reaches around 20 dimensions. This forms a major computational bottleneck of the ISOMAP algorithm. The speed of the algorithm could be increased drastically by making a further approximation. Instead of looking for the exact nearest neighbours, an approximate nearest neighbours algorithm can be employed. Whether this affects the embedding's accuracy to an intolerable extent needs to be determined.

Another way to reduce the computational time of the ISOMAP algorithm is the use of parallel computing whenever possible. The landmark ISOMAP

algorithm contains two main points, where parallelization yields a drastic improvement in speed. The first is the calculation of the eigendecomposition. For the calculation of the first P collective coordinates the P largest eigenvalues of the Gramian matrix S need to be computed. There are efficient parallel algorithms to compute the largest eigenvalues of a matrix already present in most software packages, like `scipy.linalg`.

A second part of the landmark ISOMAP algorithm that can be sped up by parallelization is the reinsertion of points into the landmark embedding. This is calculated for each point individually and therefore offers a huge potential for parallelization. So far the code used for this thesis employs parallelization on the different cores of the CPU. The reinsertion step, which still makes up a large part of the computational time, is almost a textbook example for parallelization on a GPU, since it contains N individual matrix multiplications for the embedding of a time series with N steps.

For more complex systems (see section 4) the PCA algorithm performed more accurate embeddings, in terms of residual variance, than the ISOMAP algorithm. This is in accordance to results from Duan et al. (see [8]). They reduced the dimensionality of data from peptide folding-unfolding simulations. Amongst other evaluation criteria, they compared the residual variance of several nonlinear embedding methods to that of PCA. Regarding the residual variance of the PCA and ISOMAP embedded data they had similar results as those presented here.

It is necessary to find additional measures that, apart from the preservation of the input distances, measure the embedding quality, for example using successful cluster separation. Another possibility would be the algorithms ability to separate transition paths, like in the simple trialanine case, where ISOMAP (figure 5.2) could distinguish pathways between the metastable states, while PCA (figure 5.1) showed rather a transition area than distinct pathways.

An altogether different problem is the interpretation of the collective variables retrieved by the ISOMAP algorithm. A possible ansatz for this interpretation could again make use of the library data. Since the original single-atom coordinates of the library data are known, one could visualize the molecules along a collective coordinate. This visualization could then

provide insights into the features of the molecule that vary along the collective coordinates.

Despite the room for improvement described above both dimension reduction methods were able to reduce the dimensionality of a given time series drastically from 237 (Trp-cage backbone) and 501 (Aβ42 backbone) to three dimensions.

In the second part of this thesis, drift and diffusion fields of the low-dimensional embeddings were estimated using the Langevin algorithm described by Hegger and Stock (see [11]). In a first step, their results were reproduced in section 5.1.2. The results here were in good agreement with those presented by Hegger and Stock. Furthermore, the step-wise drift-diffusion estimation worked as well on a drastically reduced library. This showed the algorithms ability to work on library data, that is comprised of several short time series. That ability became important when larger molecules where considered. The single time series of these molecules proofed inappropriate for the step-wise drift-diffusion algorithm (see sections 5.2.2 and 5.3.2). Both ended in a state from where the library knew no way back, causing the trajectory to become practically stationary there. In order to be able to run the Langevin algorithm on the embeddings of larger molecules, multiple shorter time series were calculated and assembled to form one library. This library was then in sections 5.2.4 and 5.3.4 used to perform simulations of a length that would not be practical using MD simulations. In both cases the estimated time series was 40 μs long, while all library time series together covered 606 ns (Trp-cage) and 820 ns (Aβ42).

The sampling of the library time series can be improved as well. The start positions for the time series could be chosen in a more sophisticated manner. Additionally, the simulation of a single time series could be stopped, once a metastable state is reached and the molecule does not significantly change for a prolonged period of time. This could save both computational time and memory.

In this thesis the time series in the Aβ42 system that caused the step-wise drift-diffusion estimation to get stuck at the end of that time series were removed. An alternative approach would be, to identify the problematic time series and run additional simulations close by, in order to add a way back to

the rest of the embedding. Whether the full landmark ISOMAP procedure has to be rerun, or if the points could be embedded by the triangulation step, would then need to be determined.

All in all the methods described in this thesis perform reasonably well. Parallelization of large parts of the landmark ISOMAP algorithm allows it to be run in a moderate amount of time, even for more extensive simulations of larger molecules. In addition to that, the step-wise drift-diffusion estimation allows one to calculate a 40 μs molecular trajectory in 100 minutes on a standard PC, rather than several nanoseconds of molecular trajectory being calculated in two days on a high performance computing system like PALMA II. Albeit the obtained trajectory is in a low dimensional representation needing additional interpretation. However some form of dimension reduction is always necessary for the interpretation of high dimensional systems like the proteins studied here.

The combination of dimension reduction and drift-diffusion estimation algorithms, employed to study molecular dynamics is a promising approach to these complex systems. It can be used similarly to this thesis, to obtain an impression of the long term evolution of complex systems. Another application could be studying the change of both the embedding and the drift field, when a parameter in the MD simulation, like the temperature, is changed. However the application of these algorithms is not limited to the study of molecular systems. Any high dimensional system that evolves with time can be examined using these methods. The possible application range from the molecular systems treated here, to larger systems, such as the human body, where EEG or ECG data could be analysed, or huge systems such as the global climate.

Bibliography

[1] A. Amadei, A. B. M. Linssen, and H. J. C. Berendsen. Essential dynamics of proteins. *Proteins: Structure, Function, and Genetics*, 17(4):412–425, dec 1993.

[2] J. Blackburn and E. Ribeiro. Human motion recognition using isomap and dynamic time warping. In A. Elgammal, B. Rosenhahn, and R. Klette, editors, *Human Motion – Understanding, Modeling, Capture and Animation*, pages 285–298, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[3] M. Ceriotti, G. A. Tribello, and M. Parrinello. Simplifying the representation of complex free-energy landscapes using sketch-map. *Proceedings of the National Academy of Sciences*, 108(32):13023–13028, 2011.

[4] R. R. Coifman and S. Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, jul 2006.

[5] P. Das, M. Moll, H. Stamati, L. E. Kavraki, and C. Clementi. Low-dimensional, free-energy landscapes of protein-folding reactions by nonlinear dimensionality reduction. *Proceedings of the National Academy of Sciences*, 103(26):9885–9890, 2006.

[6] V. De Silva and J. B. Tenenbaum. Sparse multidimensional scaling using landmark points. Technical report, Stanford University, 2004.

[7] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271, dec 1959.

[8] M. Duan, J. Fan, M. Li, L. Han, and S. Huo. Evaluation of Dimensionality-Reduction Methods from Peptide Folding–Unfolding

Simulations. *Journal of Chemical Theory and Computation*, 9(5):2490–2497, may 2013.

[9] R. Friedrich, J. Peinke, M. Sahimi, and M. Reza Rahimi Tabar. Approaching complexity by stochastic methods: From biological systems to turbulence. *Physics Reports*, 506(5):87–162, sep 2011.

[10] J. Gradišek, S. Siegert, R. Friedrich, and I. Grabec. Analysis of time series from stochastic processes. *Physical Review E*, 62(3):3146–3155, sep 2000.

[11] R. Hegger and G. Stock. Multidimensional Langevin modeling of biomolecular dynamics. *The Journal of Chemical Physics*, 130(3):034106, jan 2009.

[12] C. Lee and S. Ham. Characterizing amyloid-beta protein misfolding from molecular dynamics simulations with explicit water. *Journal of Computational Chemistry*, 32(2):349–355, jan 2011.

[13] J. A. Lee and M. Verleysen. *Nonlinear Dimensionality Reduction*. Springer, 2007.

[14] J. W. Neidigh, R. M. Fesinmeyer, and N. H. Andersen. Designing a 20-residue protein. *Nature Structural Biology*, 9(6):425–430, jun 2002.

[15] E. Parzen. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, sep 1962.

[16] M. Rosenblatt. Remarks on Some Nonparametric Estimates of a Density Function. *The Annals of Mathematical Statistics*, 27(3):832–837, sep 1956.

[17] S. T. Roweis. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290(5500):2323–2326, dec 2000.

[18] A. J. Rzepiela, N. Schaudinnus, S. Buchenberg, R. Hegger, and G. Stock. Communication: Microsecond peptide dynamics from nanosecond trajectories: A Langevin approach. *The Journal of Chemical Physics*, 141(24):241102, dec 2014.

[19] M. Samiuddin and G. M. El-Sayyad. On nonparametric kernel density estimates. *Biometrika*, 77(4):865–874, 1990.

[20] N. Schaudinnus, B. Bastian, R. Hegger, and G. Stock. Multidimensional Langevin Modeling of Nonoverdamped Dynamics. *Physical Review Letters*, 115(5):050602, jul 2015.

[21] N. Schaudinnus, A. J. Rzepiela, R. Hegger, and G. Stock. Data driven Langevin modeling of biomolecular dynamics. *The Journal of Chemical Physics*, 138(20):204106, may 2013.

[22] S. Siegert, R. Friedrich, and J. Peinke. Analysis of data sets of stochastic systems. *Physics Letters A*, 243(5-6):275–280, jul 1998.

[23] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for non linear dimensionality reduction. *Science*, 290(December):2319–2323, 2000.

[24] G. R. Terrell and D. W. Scott. Variable Kernel Density Estimation. *The Annals of Statistics*, 20(3):1236–1265, 1992.

[25] X.-W. Wang, D. Nie, and B.-L. Lu. Emotional state classification from EEG data using machine learning approach. *Neurocomputing*, 129:94–106, apr 2014.

Acknowledgements

Last but not least I want to express my gratitude to all those who were involved in the process of researching and writing this thesis:

... Dr. Oliver Kamps for the continuous support and supervision and many helpful hints and motivation

... Prof. Dr. Andreas Heuer for agreeing to supervise this project and examine my thesis

... Alexander Kötter for his patience teaching me how to run MD simulations

... Lukas Ophaus and Sarah Trinschek for welcoming me in their office and the warm and productive atmosphere there

... Franziska Alberts, Christopher Henkel and Jonas Plate for helpful discussions about physical and non-physical questions during extended coffee breaks

... Prof. Dr. Uwe Thiele for the opportunity to write my masters thesis in his research group

... My parents, Dr. Peter Schütz and Dr. Anke Holl, as well as my flatmate Wolfgang Pölking for proofreading my thesis and spotting many mistakes and typos

Plagiatserklärung des Studierenden

Hiermit versichere ich, dass die vorliegende Arbeit *Estimating low dimensional dynamical models for molecules* selbstständig verfasst worden ist, dass keine anderen Quellen und Hilfsmittel als die angegebenen benutzt worden sind und dass die Stellen der Arbeit, die anderen Werken - auch elektronischen Medien - dem Wortlaut oder Sinn nach entnommen wurden, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht worden sind.

_____

(Datum, Unterschrift)

Ich erkläre mich mit einem Abgleic der Arbeit mit anderen Texten zwecks Auffindung von Übereinstimmungen sowie mit einer zu diesem Zweck vorzunehmenden Speicherung der Arbeit in eine Datenbank einverstanden.

_____

(Datum, Unterschrift)

Declaration of Academic Integrity

I hereby confirm that this thesis on *Estimating low dimensional dynamical models for molecules* is solely my own work and that I have used no sources or aids other than the ones stated. All passages in my thesis for which other sources, including electronic media, have been used, be it direct quotes or content references, have been acknowledged as such and the sources cited.

_____

(Date, Signature)

I agree to have my thesis checked in order to rule out potential similarities with other works and to have my thesis stored in a database for this purpose.

_____

(Date, Signature)