Conformational Dynamics of a Peptide Ligand in Solvent and in Complex with a MHC-I Protein

November 17, 2010

Diplomarbeit

Institut für Theoretische Physik, Freie Universität Berlin

vorgelegt von

Martin Kavalar, Reichenberger Str. 62, 10999 Berlin

Betreuer: Dr. Frank Noé, Prof. Dr. Holger Dau

Abstract

The effect of the dynamics of proteins and peptides has long been suspected to be of great importance when explaining their function. The influence of their dynamical behavior remains mostly unknown as the vast majority of studies are restricted to the static structure determination of biomolecules. Recent advancement on computer simulation techniques as well as profound analysis carried out with the help of Markov state models allows to study the dynamics of a peptide ligand that is bound to a MHC-I protein. We are interested in finding out how the dynamics of the ligand is influenced when complexed with a protein compared to its uncomplexed dynamics. Starting from a crystal structure of the complex that hints to several stable conformations of the ligand, the ligand's dynamics will be investigated.

Several long-lived, "metastable" states and their corresponding probabilities as well as important transition pathways between those states could be extracted for the uncomplexed case. In comparison, when complexed with the MHC-I protein, the ligand's dynamics were much more restrained but conformational changes between different metastable states occurred so rarely, that the simulations that could be performed until now turned out to be insufficient in order to construct a reliable model of its dynamics. This is where future work using advanced adaptive sampling techniques or simulations an order of magnitude longer should be able to provide more insight.

Contents

List of Figures

1 Introduction

1.1 Motivation

Understanding the function of the human immune system on a molecular level is a very important subject of current research. Especially autoimmune diseases, which are related to a malfunction of the adaptive immune system, leading to an immune reaction against the healthy cells are not fully understood, yet.

This study is concerned with a few specific proteins that are important in this immune reaction, which belong to the class of MHC-I molecules. Each of these MHC proteins can bind smaller molecules, called peptides. We were provided with data gained from X-ray crystallography carried out on different MHC-I molecules complexed with various peptides. These experiments were not able to fully resolve the atomic structure. Specifically this was data from three crystallography experiments of MHC-I complexes (also called Human leukocyte antigen, abbr. HLA) which are all subtypes of HLA*B27 (B*2705:pB27, B*2705:pCP and B*2704:pB27).

The aim of this study is to assist the crystallographers by analyzing the dynamics of these complexes. By comparing the dynamics of the different complexes it will hopefully be possible to gain a deeper understanding of the autoimmune disease called ankylosing spondylitis, which is known to be related to these subtypes for more than 35 years [25, 5]. Even though many studies have been conducted on these subtypes complexed with various subtypes [12, 11, 18, 27, 29, 35, 44, 43], this issue is still puzzling.

Using the Molecular dynamics (MD) simulation software gromacs [17], simulations of all three system were executed.

Beyond standard MD analysis techniques performed on similar systems [33], we are attempting to perform a more substantial analysis of the peptides conformation dynamics with the help of a so called Markov state model. This approach should allow us to obtain important information regarding these systems that cannot be derived using the traditional approaches. This includes the identification of long-lived, also called metastable states and the

derivation of molecular observables of interest, such as stationary probabilities, free energies, transition rates and transition pathways, including their errors [34, 32].

While this technique has been applied to protein folding recently [31, 4, 3], it has not, to our knowledge, been used to study the dynamics of a peptide bound by a MHC protein. We also attempted to compare the dynamics of the pB27 peptide in water to the case where it resides in the MHC's binding pocket.

1.2 Amino acids, peptides and proteins

Amino acids are a class of organic molecules characteristically built of an amino group (NH_2), a carboxyl group (COOH) and a side chain. There are about 20 different standard amino acids encoded directly into the genetic code. When amino acids form peptide bonds, they are referred to as peptides, polypeptides or proteins. These words are a bit ambiguous, one generally refers to a protein when talking about a biomolecule consisting of many amino acids, while a peptide is often referred to as a short chain of amino acids (up to 30 amino acids). One individual amino acid of a protein is then referred to as a residue. A peptide bond is a covalent binding where the carboxyl group reacts with the amino group of another amino acid, releasing a water molecule (H_2O). This is illustrated in Fig. 1.

The end of a protein or peptide with a free amino group is referred to as the N-terminus or amino terminus, while the end with free carboxyl group is called the C-terminus. The amino acids sequence of a protein is generally presented in the N-to-C direction, written from left to right. Each amino acid can be abbreviated by either a three-letter or one-letter code. When listing the amino acid sequence of peptides or proteins we usually use the one-letter representation of a peptide. The so-called backbone of a protein is the linked series of carbon, oxygen and nitrogen atoms. Figure 2 shows three different representations of a peptide. pB27 peptide, its whose acid sequence would read RRKSSGGKGGSY.

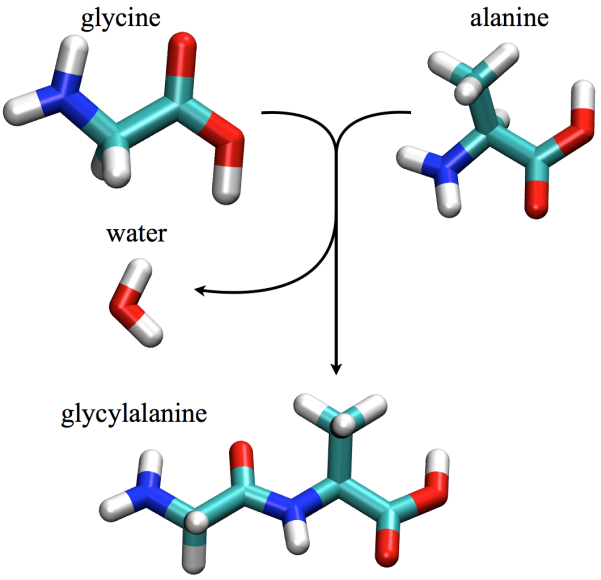
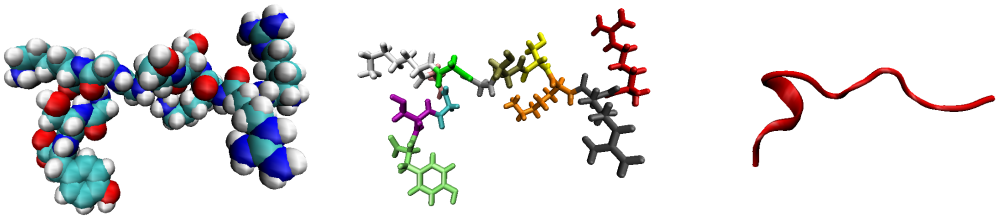


Figure 1: Peptide bond of two amino acids (glycine and alanine) to form the dipeptide glycylalanine. A water molecule is lost in the condensation reaction. The different atom types are shown in different colors (H white, O red, C green and N blue).



(a) Van-der-Waals (b) Bonds (c) Cartoon

Figure 2: Three representations of the pB27 peptide. The one-letter amino acid code of pB27 is RRKSSGGKGGSY.

Proteins are produced in the cell as a long chain of amino acids linked by peptide bonds. This amino acid sequence is known as the primary structure. As a result of weaker bonds including hydrogen bonds, ionic bonds and van der Waals attractions as well as hydrophobic side chains, proteins generally have structures that are energetically favorable. The rearrangement into this stable conformation is called folding while the three-dimensional alignment

is then called secondary structure. Two important patterns of secondary structures often occurring in proteins include the formation of α-helices and β-sheets which are shown in Fig 3.

1.3 The adaptive immune system

In the following section we shall briefly sketch the functionality of the human immune system and in order to explain the function of the MHC complexes. For a thorough explanation on this subject see standard textbooks on biochemistry, e.g. Alberts [1].

The human immune system can be classified in two parts. The innate immune system is a relatively simple defense strategy mostly made up of protective barriers such as skin and mucosa as well as pagocytic cells that can ingest and destroy invading microorganisms. This part of the immune system is unspecific.

The adaptive part on the other hand is a highly sophisticated system that serves the task of destroying invading pathogens and toxic molecules they produce. These adaptive immune responses are highly specific to the pathogen that induced them and can also provide long-lasting protection. Because this is a destructive response it is very important that it only responds to molecules foreign to the host and not molecules of the host itself. The ability to distinguish what is *foreign* and what is *self* is a fundamental feature of the adaptive immune system. One speaks of an autoimmune disease when this distinction is incorrect, i.e. the system reacts destructively against the host's own molecules. Mounting of an adaptive immune response against harmless foreign molecules is pointless and can be potentially dangerous. This can lead to allergic conditions such as asthma.

The adaptive immune response is carried out by white blood cells called lymphocytes. One can differentiate two classes of such an immune response. The antibody response is carried out by B cells. The cell-mediated immune response the other hand is mounted by T cells. We will take a closer look at this cell-mediated response, since the MHC-I proteins under investigation in this work play a very important role in this process.

In this cell-mediated response activated T cells react directly against a foreign antigen that is presented to them on a host cell. It can either kill a cell which has viral antigens presented on its surface directly, therefore eliminating the infected cell before the virus had a chance to replicate. Otherwise it can produce signal molecules that activate macrophages to destroy the invading microbes.

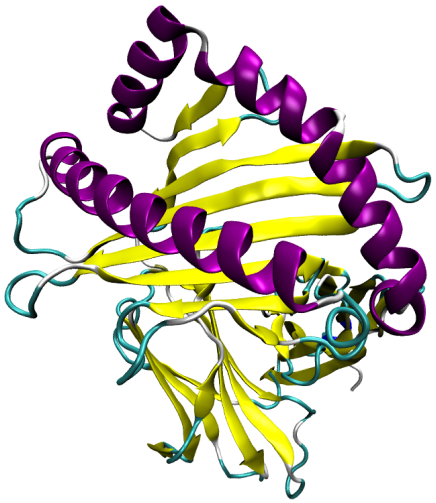


Figure 3: A MHC-I protein in its native (folded) state. The backbone of the protein aligns in such a way that it becomes very stable. We can see different secondary structures such as α-helix (purple) and β-sheets (yellow).

MHC complexes The recognition of pathogens is tied to the recognition of pathogen specific antigens. The Major Histocompatibility Complex (MHC) molecules studied in this thesis play an important role in triggering a cellular immune response and can be differentiated in two classes. Class I MHC-molecules are found on the surface of (almost) all cells (containing a nucleus). Class-II complexes are only found on antigen-presenting cells (APCs). When a foreign virus infects a cell it typically abuses the host cells protein synthesis for its own reproduction. Therefore an infected cell will produce non-self proteins which are spread in the cytosol. Fractions of those non-self proteins (peptides) can be bound and transported by the MHC-I complex to the cells surface. Only the MHC-I bound non-self peptide is detected by the T-cells,

triggering an immune response. Therefore it is also called Human Leukocyte Antigen (HLA) in humans. Figure 3 shows the secondary structure of a MHC-I complex.

2 Methods

This section will introduce the methods necessary to understand the analysis following. Therefore, a brief introduction on how X-Ray crystallography is used in order to provide an insight into the structure of many biomolecules is given in the first part. Furthermore, it will be shown how molecular dynamics simulation can be employed to extend that knowledge beyond the static crystallography case to study the time-development of such a biomolecular system at physiological conditions. Lastly, the reader is introduced to the mathematical tool set that can be used to investigate the conformation dynamics of a biomolecule, including how a so called Markov State Model (MSM) can be constructed from simulation data.

2.1 X-Ray crystallography

X-ray crystallography is the most-used method of examining the atomic structure biomolecules. In 1962, Max Perutz and Sir John Cowedery Kendrew received the Nobel Prize in Chemistry for their structure analysis of the myoglobin molecule [24]. The Protein Data Bank, where results of such structure analysis are published, currently lists 63,876 structures available, of which more than 87 % where created using this method. We will first go into describing why X-rays crystallography often is the method of choice to determine 3-D structures of crystals, explain the physics involved and lastly how we obtain a 3-D structure of a macromolecule using this experimental method.

2.1.1 Why X-rays?

X-rays are well-suited for crystallography because their wavelength is in the magnitude of Å (about 1-100 keV), whereas visible light has a wavelength of $\sim$ 5000 Å. As we will see in the next subsection, the wavelength used is directly related to the resolution of our image obtained. Furthermore, X-rays are easily produced using X-ray tubes. X-rays are emitted when charged particles such as electrons are accelerated and deflected by another particle

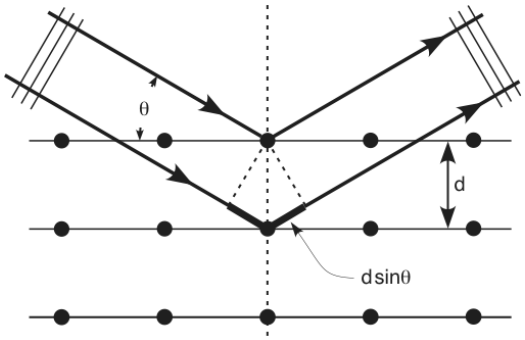


Figure 4: Bragg diffraction

such as a neutron. This effect is known as *Bremsstrahlung*.

2.1.2 Physical background

In the typical X-Ray crystallography experiment the crystal is mounted on a goniometer and rotated as a monochromatic X-ray beam is projected onto a crystal. The X-ray radiation is scattered elastically at the atoms electron shell, meaning its wavelength does not change. A detector records a diffraction pattern at different angles, giving us a function of intensity over a 2-D-angle of the reflected X-rays beams. Depending on the angle, peaks can be measured. Using Bragg's diffraction theory, the source of the peaks turn out to occur when the beams, reflected at different lattice planes, interfere constructively. For a Bravais crystal, whose lattice plane distance is d, peaks can be measured when

$$2dsin\theta = n\lambda \; . \tag{1}$$

This is illustrated in Fig. 4.

The peaks intensity I_{hkl} is proportionally related to the structure factors F_{hkl} by

$$I_{hkl} \propto |F_{hkl}|^2 \; . \tag{2}$$

The structure factor its self is the Fourier transform of the electron density ρ

$$F_{hkl} = \int_0^a \int_0^b \int_0^c \rho(x,y,z) \, \exp\left[2\pi i \left(\frac{hx}{a} + \frac{ky}{b} + \frac{lz}{c}\right)\right] \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z \; , \tag{3}$$

whereas h, k, l denotes the *Miller indices*, denoting crystal planes, a, b, c the basis vectors and x, y, z the coordinates. Each peak has an associated amplitude, wavelength and phase. Because we are measuring intensity as the square absolute value of the complex structure factors, we lose all phase information. This keeps us from doing a simple Fourier re-transformation to obtain the original crystal structure and is known as the *phase problem*. Multi-wavelength anomalous dispersion (MAD), Multiple isomorphous replacement (MIR) and Molecular replacement (MR) are all methods using the *Patterson map* to work around this problem by using either heavy atoms or similar previously solved structures (MR).

2.1.3 Protein crystallography

For a simple Bravais crystal the process of determining its structure is straightforward. When looking a biomolecules such as a protein on the other hand one faces several challenges before X-ray crystallography can be applied. First of all, the biomolecule has to be built into a crystal which has to be large enough (about 0.1 mm in all dimensions). This process, known as crystallization, is critical to obtaining good results. Proteins, which are mostly grown in solution are crystallized by gradually lowering the solubility. Finding good parameters that have an influence on crystallization such as temperature, pH, solvent type and added ions or ligands can be a tedious task. Impurity or non-uniform orientation can lead to a decrease in resolution or fuzziness.

The general process of getting to the 3-D structure of a protein is illustrated in Fig. 5. Once the protein has been crystallized, a diffraction pattern is generated using the setup described above. Chemical information or knowledge gained from previous experiments is used to produce and iteratively refine a 3-D model.

Another problem which is of special interest in this thesis is the change of conformations of a protein.When the crystal is produced, we tend to randomly freeze the each protein at its current conformation. As we are measuring an ensemble average, areas of the protein where the alignment of atoms

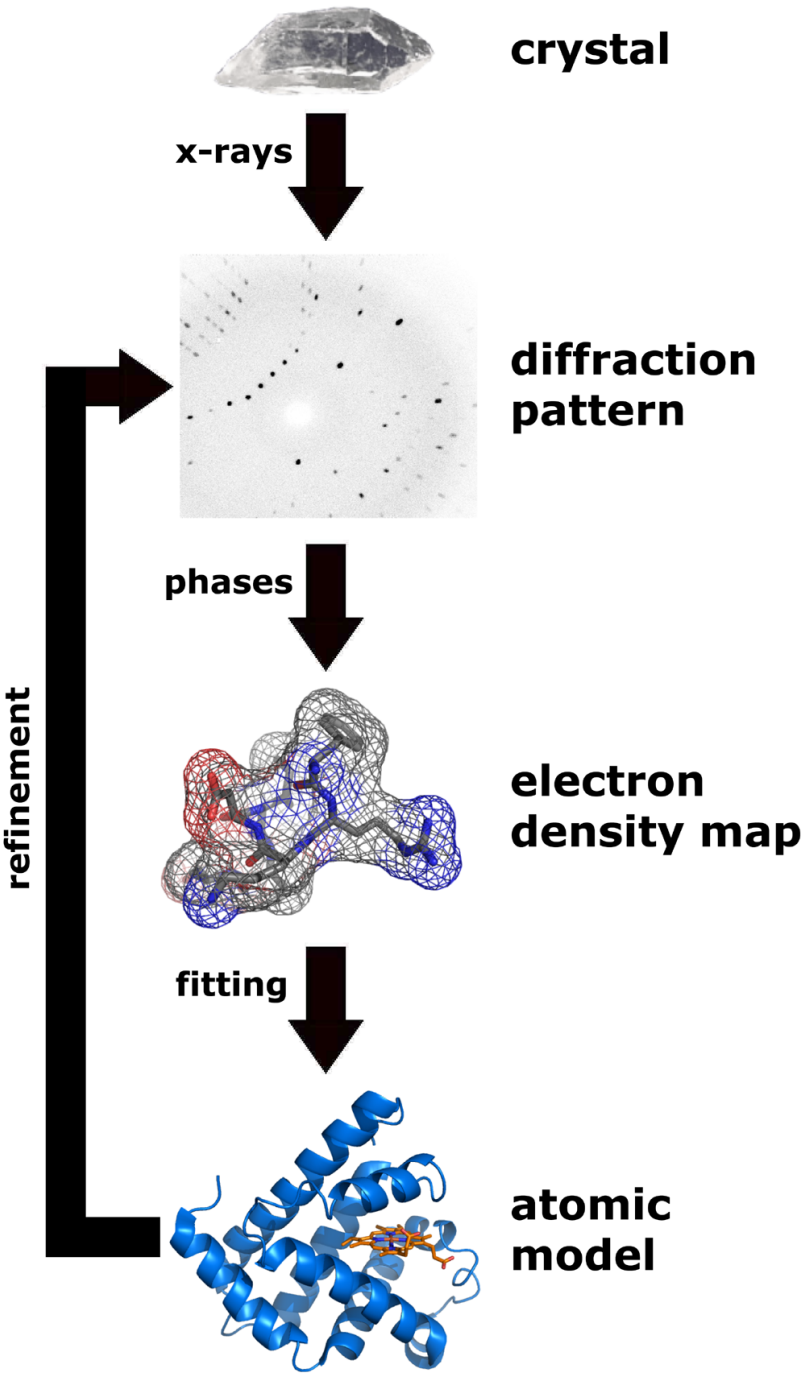


Figure 5: Process of X-ray crystallography

differs, will generally lead to blur in our X-ray image. Also we can imagine that the crystallization temperature as well as other solutes employed to improve the crystallization might have an effect on the behavior and alignment of the molecule. In summary, it can be said that while X-ray crystallography is the method of choice for producing a static three-dimensional atomic model of a protein, we can hardly gain any insight on its dynamics.

2.2 Molecular Dynamics

Molecular Dynamics (MD) is a computer simulation method to solve the time development of a molecular system. Over the last few decades it, and its enclosing scientific field *molecular modeling*, led to respectable progress on important topics such as protein folding [23, 9] as well as practical applications such as drug design and giving insights into complex processes such as human immune reactions. A brief overview on how MD simulation works and what current shortcoming exist will be given, mostly based on the textbooks by Frenkel and Smit [15] as well as Schlick [36].

2.2.1 Basics

MD simulations are similar to real experiments. We start out the the preparation of our sample and connect it to a measuring instrument. Then, we run the experiment (simulation) as long as it takes to get reliable averages for the quantities we want to measure. As in real experiments, the measurements in simulations are disturbed by statistical noise. In an actual simulation one prepares a sample and starts the equilibration process. Once the system is equilibrated one can start to measure the physical quantities of interest.

From a physical point of view a first idea to simulate a molecular system would be to solve the time-dependent Schrödinger equation numerically. Such a method does exist and is called *ab initio* or quantum MD, but it is computationally extremely demanding. Even when dealing with very small model systems, which are hardly of any practical use, one force calculation can take days or even weeks using modern hardware. Therefore it is desirable to simplify the model as much as possible while still producing realistic results.

Classical MD simulations, based on Newtons equations of motion, have often proven to be sufficiently accurate in order to tackle many problems in biophysics, often showing a good agreement with experiments [38].

Structure data gained X-ray crystallography does usually not contain hydrogen atoms, which are added to the atoms first. Equilibration usually involves minimizing the potential energy of the system, allowing the molecules

to relax. After choosing the number of particles N, the density or the size of the simulation box, the pressure and/or the temperature we can place the particles into the simulation box. This can also involve placement of solvent molecules in the system.

The next step is to distribute the initial velocities. The velocities are related to the desired temperature through the equipartition theorem via

$$\langle \frac{1}{2} m v_\alpha^2 \rangle = \frac{1}{2} k_B T, \tag{4}$$

where m is the particle mass, v_α is the velocity, k_B is the Boltzmann constant and T is the temperature. We can connect this relation to the current state of the system by

$$k_B T(t) = \sum_{i=1}^{N} \frac{m v_\alpha^2(t)}{N_f} \tag{5}$$

with the simulation time t and the number of degrees of freedom N_f. Using this relation we can assign the velocities in order to obtain a desired temperature or we can measure the temperature by looking at the velocities.

2.2.2 Force calculation

From the initial state on we want to develop the system in time. To get the time evolution of each single particle we need to compute the forces acting on it. These forces can be computed by the relation

$$f_\alpha(r) = -\frac{\partial U(r)}{\partial r_\alpha} = -\left(\frac{r_\alpha}{r}\right)\left(\frac{\partial U(r)}{\partial r}\right). \tag{6}$$

The value r is the distance between two particles. It is important to note that we usually simulate a finite number of particles in a limited simulation box. In order to omit large scale finite size effects one uses periodic boundary conditions. These are applied by computing the shortest distance between two particles from all their images. In Fig. 6 we show a simple realization of periodic boundary conditions. The shaded box is the actual simulation

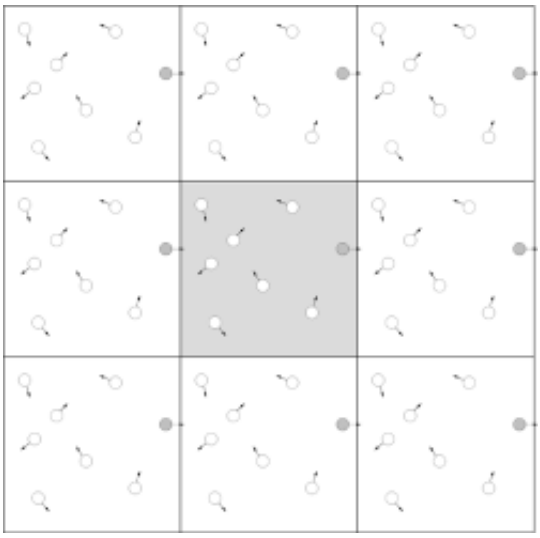


Figure 6: Illustration of periodic boundary conditions

box, whereas all white boxes represent images of the simulation box. If one computes the distance from the gray shaded particle to another particle the closest particle in the same periodic box or the surrounding ones is taken.

Force field In order to compute the forces acting on each atom in reasonable time, a force field is used. Here the Optimized Potentials for Liquid Simulations all-atom (OPLS-aa) force field will be applied. It approximates the potential energy of each atom, using different terms for bonded and non-bonded terms as follows:

$$U = E_{\text{bonded}} + E_{\text{non-bonded}}. \tag{7}$$

The bonded term can be split up into three terms:

$$E_{\text{bonded}} = E_{\text{bond-stretch}} + E_{\text{angle-bend}} + E_{\text{rotate-along-bond}}, \tag{8}$$

which are illustrated in Fig. 7.

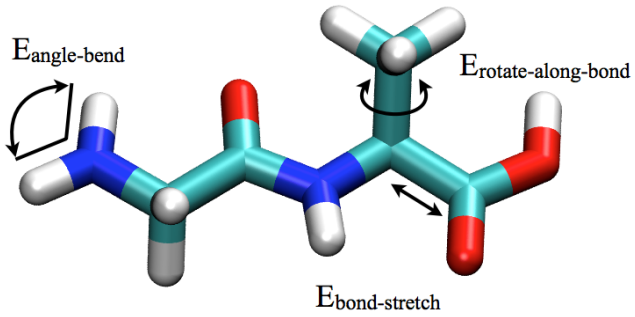


Figure 7: Illustration of the different terms contributing to the change of potential energy

All three terms can be configured using a parameter k. The first term is a harmonic potential representing the length displacement from their ideal value b_0 of atomic pairs that are covalently bound:

$$E_{\text{bond-stretch}} = \sum_{\text{covalent bonds}} k_b(b - b_0)^2. \tag{9}$$

The second term account for the alteration of bond angles θ from ideal values θ_0 and is also represented by a harmonic potential:

$$E_{\text{angle-bend}} = \sum_{\text{angles}} k_\theta(\theta - \theta_0)^2 \tag{10}$$

Lastly, the third term is assumed to be periodic and therefore often expressed by a cosine function and accounts for twisting a bond:

$$E_{\text{rotate-along=bond}} = \sum_{\text{torsions}} k_\phi(1 - cos(n\phi)). \tag{11}$$

The non-bonded term is the sum of Van der Waals and electrostatic interaction:

$$E_{\text{non-bonded}} = E_{\text{van-der-Waals}} + E_{\text{electrostatic}}. \tag{12}$$

The van der Waals interaction is often modeled using the Lenard-Jones

potential using atom-type dependent constants A and C:

$$E_{\text{van-der-Waals}} = \sum_{\text{nonbonded pairs}} \left(\frac{A_{ik}}{r_{ij}^{12}} - \frac{C_{ik}}{r_{ij}^{6}} \right). \tag{13}$$

For electrostatic interaction the Coulomb potential is used, with D the effective dielectric function for the medium and the atom distance r between the two atoms charged with q_i and q_k:

$$E_{electrostatic} = \sum_{\text{nonbonded pairs}} \frac{q_i q_k}{D r_{ik}}. \tag{14}$$

A detailed description of optimal parameters used can be found in the work by Jorgensen et al. [21, 20].

Because the force field is a simplification of the quantum mechanical description, we must also look at what limitations it imposes and in what cases we assume the simplifications to bias the results.

Since we are using *Newton's mechanics* to describe our system it should be logical that no quantum-mechanical effects, such as proton tunneling, will occur in the simulation. These classical approximations are suitable for many atoms at normal temperatures, but problems are expected when dealing with e.g. liquid helium at low temperatures.

More troublesome however is that classical mechanics delivers significantly different results from a quantum mechanical calculation as at lower temperatures and/or higher frequencies the discrete energy levels become more and more relevant. As long as the vibration frequency ν is smaller, so that

$$\frac{h\nu}{k_B T} \ll 1 \tag{15}$$

is true, we can assume our approximation to hold. Here k_B denotes Boltzmann's constant, h Planck's constant and T the temperature. For bonds with a ratio $\frac{h\nu}{k_B T}$ of more than one (a O-H stretch has about 17 at 300K) we cannot simply apply classical treatment. Workarounds for that problem can involve applying quantum mechanical energy correction or putting constraints on the bonds in the equations of motion.

When looking at the modeling of covalent bonds by harmonic oscillators (Eq. 9) it should become clear that bonds will never break up. This means that no chemical reactions will occur in our simulations.

The atoms are described only by their *nuclear* motion. The *Born-Oppenheimer* approximation which separates electron movement from nucleus movement (the lighter electrons follow the nucleus instantly) is applied. As a result, no effects that need a more subtle electron model, e.g. polarization. Also binding effects involving repositioning of electrons such as a delocalized π electron cloud will not be modeled correctly.

A current review of the state and drawbacks of classical MD simulations can be found in the work by Freddolino et al. [14].

2.2.3 Long range interactions

Long range interactions pose a special challenge in MD simulations. Interactions are called long range when they decay slower than $1/r^3$. The Coulomb interaction U_{Coul} of a point charges q_i decays with $1/r$, making it long-range. We have a total of N particles in our simulation box and we assume that the total system is does not carry a net charge. The Coulomb interaction in the system is then

$$U_{coul} = \frac{1}{2}\sum_{i=1}^{N} q_i \Phi(r_i) \tag{16}$$

with

$$\Phi(r_i) = \sum_{j,\mathbf{n}}{}' \frac{q_j}{|r_{ij} + \mathbf{n}L|}. \tag{17}$$

This sum goes over all periodic images of the simulation box, indicated by sum over $\mathbf{n}L$ since the interaction is not vanishing after the distance of half the simulation box. The prime in the summation makes sure that we do not sum over the particle $i = j$ in the central simulation box $\mathbf{n} = 0$, but of itself with all its periodic images. This sum is poorly convergent and we have to take into account a large number of periodic images. This is a major problem, even for modern computers. Advanced summation techniques were introduced to overcome this computational drawback when dealing with long

range interactions.

One starts by introducing a charged cloud around all point charges. The charged cloud is the opposite sign then the point charges and guarantees a faster convergence of the potential of the point charges by screening it. Obviously the new charges introduced into the system have to be corrected in order to maintain the physical correctness. For this purpose additional charged clouds are introduced to compensate the first clouds. Thus, we extend the amount of charges by two components what leads finally to three parts, the point charges q_i, the screening clouds $-q_i$ and the counter screening clouds q_i. Usually one chooses a Gaussian distribution for the charged clouds. The advantage of this new formulation is that the potential in the central simulation box decays fast enough to cut it off and to separate the simulation boxes from each other. The correction is periodic and can be expressed in terms of a Fourier transform. The Fourier transform is rapidly convergent compared to the original sum. The exact formulation of the so called Ewald summation goes far beyond the scope of the work and can be found in the work of Ewald [10].

For larger systems ($N > 10^4$) the Ewald summation becomes inefficient. To improve the performance one can basically interpolate the particles on a regular mesh by introducing a charge assignment function. By this means a Fast Fourier Transformation can be applied instead of a normal Fourier Transformation in the case of the Ewald summation technique. This method is called the Particle Mesh Ewald summation and is applied in our simulations to handle all long range interactions.

2.2.4 Integrating the equations of motion

Once the force acting on each particle is computed, Newton's equations of motion can used to get the new positions. We start by expanding the coordinates in a Taylor series

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \Delta t \left[\dot{\mathbf{r}}(t) + \frac{\Delta t}{2} \ddot{\mathbf{r}}(t) \right] + \mathcal{O}(\Delta t^3) \tag{18}$$

and by using the simple equality $\dot{\mathbf{r}}(t) + \frac{\Delta t}{2}\ddot{\mathbf{r}}(t) = \dot{\mathbf{r}}(t + \frac{\Delta t}{2})$ as well as $\ddot{\mathbf{r}}(t) = \frac{\boldsymbol{F}(t)}{m}$ we obtain

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \dot{\mathbf{r}}(t + \frac{\Delta t}{2})\Delta t \tag{19}$$

and

$$\dot{\mathbf{r}}(t + \frac{\Delta t}{2}) = \dot{\mathbf{r}}(t - \frac{\Delta t}{2}) + \frac{\boldsymbol{F}(t)}{m}\Delta t. \tag{20}$$

The algorithm is called leap-frog, since it uses coordinates $\boldsymbol{r}$ at time t and velocities $\dot{\mathbf{r}}$ at time $t - \frac{\Delta t}{2}$. It is a popular MD integration algorithm also used by gromacs.

2.2.5 Thermostats

When simulating biomolecules we often want to simulate at constant temperate to match the physiological conditions. Our simulations will be performed at constant particle number, constant volume and constant temperature, NVT which is the case of the canonical ensemble. This energy exchange with the surrounding heat bath is ensured by using a thermostat. While many different thermostats for simulations exist only the Langevin thermostat, which is used in the simulations following, will briefly be described here. A comprehensive review of current thermostat algorithms and their applications can be found in the work of Hünenberger [19].

Langevin thermostat The Langevin equation describes the motion of a large particle trough a continuum of small particles. The motion of the large particle is changed accordingly to $-\gamma p$, what describes actually the friction produced by hits with the smaller particles. In MD simulations the Langevin equation can be used if we assume that the particles we simulate are in a sea of much smaller fictional particles, that we do not want to simulate explicitly, but it can also be used as a heat bath. The Langevin equation changes the momentum accordingly to

$$\Delta p = (f - \gamma p + \delta p)\Delta t. \tag{21}$$

The motion of the particles is damped by the factor $-\gamma p$ and δp is a Gaussian distributed random number with probability

$$p(\delta p) = \frac{1}{\sqrt{2\pi}\sigma_p} \exp(-\delta p^2/2\sigma_p^2), \tag{22}$$

where $\sigma_p = 2\gamma m k_B T$. This thermostat corrects the velocities in every simulation step according to the correct canonical distribution. The force of the thermostat can be adjusted by the parameter γ. When simulating a good value for this parameter is 0.5 ps^{-1} since this results in a friction lower than the internal friction of water.

2.3 Mathematical description

The aim of this section is to give a brief overview of the mathematical background necessary in order to analyze the conformation dynamics of a protein. Traditional analysis of MD simulations, that often involves visual inspection of compound molecular trajectories and computing specific properties such as the distance to a reference structure, often tend to oversimplify or hide important dynamical properties of a molecule. By describing a molecule in terms of a stochastic process, or more specific, a Markov process, many important statistical quantities, including their uncertainties, of interest can be computed. The resulting Markov state model (MSM) can also be validated and checked for errors afterwards.

Based on the textbooks by Gardiner [16] and Van Kampen [42], the general stochastic process and the special case of the Markov process will first be introduced. Following, fundamental stochastic equations, namely the Chapman-Kolmogorov and Master equation are derived. Finally, based on the reference on this subject by Prinz et al. [34], it is shown how this can be applied to biomolecules in order to study the conformation dynamics.

2.3.1 Stochastic process

Definition

Given an experiment of outcome $\omega \in \Omega$, where Ω is the set of all possible outcomes. A function $\xi(\omega, t)$ can be assigned to every outcome ω. In physical systems t is referred to as time with $t \in \mathbb{R}_{0+}$. The family of all functions $\xi(\omega, t)$ is called a stochastic process. The process is a function of two variables, ω, t. Such a process can be regarded in two different ways, either by fixing ω or the time t. In the first case we obtain $\xi(\omega, t) = \xi^{\omega}(t)$, which is a function of time and called a realization or sample function. By fixing t we get $\xi(\omega, t) = \xi_t(\omega)$. This is a function of a random variable depending upon the time.

Now we can introduce some necessary formal definitions that allow us to

deal with stochastic processes. The distribution function is defined by

$$P(\mathbf{x};t) = W\{\boldsymbol{\xi}(t) < \mathbf{x}\}. \tag{23}$$

We note a state of the system by $\mathbf{x} \in \mathbb{R}^n$, because a physical process can usually be described by a vector that describes the state of our system completely. In the case of a protein this could be a vector of positions and momenta of each atom. The distribution functions describes the probability of the event $\{\boldsymbol{\xi}(t) < \mathbf{x}\}$ that consists of all outcomes such that $\boldsymbol{\xi}_t(\omega) < \mathbf{x}$. Using the protein as an example again we could think of a set of structures that are similar to a reference structure with respect to some metric and a threshold. We also define the probability density corresponding to this distribution

$$p(\mathbf{x};t) = \frac{\partial W(\mathbf{x};t)}{\partial \mathbf{x}}. \tag{24}$$

If we consider several outcomes of an experiment we need a joint probability distribution

$$P(\mathbf{x},\mathbf{y};t,t+\tau) = W\{\boldsymbol{\xi}(t) < \mathbf{x};\boldsymbol{\xi}(t+\tau) < \mathbf{y}\} \tag{25}$$

with the analogous probability density

$$p(\mathbf{x},\mathbf{y};t,t+\tau) = \frac{\partial^2 W(\mathbf{x},\mathbf{y};t,t+\tau)}{\partial^2 \mathbf{x}}. \tag{26}$$

In the same way higher joint probability distributions and densities can be defined. An important quantity regarding stochastic processes is the conditional probability density, that is defined by

$$p(\mathbf{y},t+\tau|\mathbf{x},t) = \frac{p(\mathbf{x},\mathbf{y};t,t+\tau)}{p(\mathbf{x},t)}. \tag{27}$$

The conditional probability expresses the probability of being in state $\boldsymbol{y}$ at a later time $t+\tau$, given we were in state $\boldsymbol{x}$ at an earlier time t. Using a straightforward rearrangement we can also express the joint probability

density in terms of a conditional probability by

$$p(\mathbf{x}, \mathbf{y}; t, t+\tau) = p(\mathbf{y}, t+\tau|\mathbf{x}, t) \cdot p(\mathbf{x}, t). \tag{28}$$

While the above concepts are shown for two realizations the extension to n realizations straightforward.

In real physical systems we deal with many random variables or numerous realizations. That is, we usually measure moments of a stochastic process. The mean of a stochastic process is defined as follows

$$\mu(t) = \langle \boldsymbol{\xi}(t) \rangle = \int_{-\infty}^{\infty} \mathbf{x} p(\mathbf{x}, t) d\mathbf{x}. \tag{29}$$

The correlation of two realizations is defined as

$$B(t, t+\tau) = \langle \boldsymbol{\xi}(t), \boldsymbol{\xi}(t+\Delta t) \rangle = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mathbf{x}\mathbf{y} p(\mathbf{x}, \mathbf{y}; t, t+\tau) d\mathbf{x} d\mathbf{y}, \tag{30}$$

and the covariance as

$$C(t, t+\tau) = \langle \{\boldsymbol{\xi}(t) - \boldsymbol{\mu}(t)\} \cdot \{\boldsymbol{\xi}(t+\tau) - \boldsymbol{\mu}(t+\tau)\} \rangle = \langle \langle \boldsymbol{\xi}(t), \boldsymbol{\xi}(t+\tau) \rangle \rangle. \tag{31}$$

Stationary and Ergodic theorem

A stochastic process is called stationary if all possible distribution functions defining $\boldsymbol{\xi}(t)$ remain unchanged for shifts in time

$$P(\mathbf{x}_1, \ldots, \mathbf{x}_n; t_1 + \tau, \ldots, t_n + \tau) = P(\mathbf{x}_1, \ldots, \mathbf{x}_n; t_1, \ldots, t_n). \tag{32}$$

This statement leads for instance in a one dimensional distribution to $P(\mathbf{x}, t) = P(\mathbf{x})$, a complete time independent distribution and in the two dimensional case to a distribution that depends on the time difference only. One important property of stationary stochastic processes is the Ergodic Theorem. The theorem connects the mathematical averages defined in the previous section with the real physical measurements. The theorem states that for a stationary stochastic process the time average equals the average

over all realizations

$$\langle \boldsymbol{\xi}(t) \rangle = \lim_{T\to\infty} \frac{1}{T} \int_0^T \boldsymbol{\xi}(t) dt, \tag{33}$$

and

$$\langle \boldsymbol{\xi}(t)\boldsymbol{\xi}(t+\tau \rangle = \lim_{T\to\infty} \frac{1}{T} \int_0^T \boldsymbol{\xi}(t)\boldsymbol{\xi}(t+\tau) dt. \tag{34}$$

The great importance of this theorem can is backed up by the fact that physical measurements are usually taken as time averages.

2.3.2 Markov process

Assume an increasing time ordering for the stochastic process by $t_1 < t_2 < \cdots < t_n$. We can identify several characteristic processes. The purely random process is characterized by the fact that subsequent values of $\boldsymbol{\xi}(t)$ are statistically independent

$$p(\mathbf{x}_1, \ldots, \mathbf{x}_n; t_1, \ldots, t_n) = p(\mathbf{x}_1; t_1) \cdot \cdots \cdot p(\mathbf{x}_n; t_n). \tag{35}$$

This process is completely separable and all information is carried by the first order density.

If the conditional probability density has the property

$$p(\mathbf{x}_n, t_n | \mathbf{x}_1, \ldots, \mathbf{x}_{n-1}; t_1, \ldots, t_{n-1}) = p(\mathbf{x}_n, t_n | \mathbf{x}_{n-1}, t_{n-1}) \tag{36}$$

the stochastic process is called Markov process. The conditional probability density at time t_n given the value at $\mathbf{x}_{n-1}$ at time t_{n-1} is not affected by values at earlier times. This is why the Markov process is often described as memoryless. A Markov process is thus determined by the two functions $p(\mathbf{x}_1, t_1)$ and $p(\mathbf{x}_2, t_2 | \mathbf{x}_{1,} t_1)$ because the whole hierarchy can be constructed from them

$$\begin{aligned} p(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3; t_1, t_2, t_3) &= p(\mathbf{x}_3, t_3 | \mathbf{x}_1, \mathbf{x}_2; t_1, t_2) \cdot p(\mathbf{x}_2, t_2 | \mathbf{x}_1, t_1) \cdot p(\mathbf{x}_1; t_1) & (37) \\ &= p(\mathbf{x}_3, t_3 | \mathbf{x}_2, t_2) \cdot p(\mathbf{x}_2, t_2 | \mathbf{x}_1, t_1) \cdot p(\mathbf{x}_1, t_1). & (38) \end{aligned}$$

This makes the Markov process manageable and very important for the anal-

ysis of stochastic processes in physical systems.

2.3.3 Chapman-Kolmogorov equation

In order to derive the Chapman-Kolmogorov equation, we start by noting that we can obtain a joint probability density of less variables by integrating over some variables, e.g.

$$p(\mathbf{x}_1, \mathbf{x}_3) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4) d\mathbf{x}_2 d\mathbf{x}_4. \tag{39}$$

Now we turn to a Markov process, which is fully determined by $p(\mathbf{x}_1; t_1)$ and $p(\mathbf{x}_2; t_2|\mathbf{x}_1; t_1)$. These two functions describe completely a Markov process, however have to obey two conditions. The first one is

$$p(\mathbf{x}_1, \mathbf{x}_2; t_1, t_2) = p(\mathbf{x}_2; t_2|\mathbf{x}_1; t_1) \cdot p(\mathbf{x}_1; t_1), \tag{40}$$

where $p(\mathbf{x}_2; t_2|\mathbf{x}_1; t_1)$ is called the transition probability, since it describes the probability to be at state $\mathbf{x}_2$ at t_2 if it was at state $\mathbf{x}_1$ at time t_1. By integrating over $\mathbf{x}_1$ we obtain

$$p(\mathbf{x}_2; t_2) = \int_{-\infty}^{\infty} p(\mathbf{x}_2; t_2|\mathbf{x}_1, t_1) \cdot p(\mathbf{x}_1; t_1) d\mathbf{x}_1. \tag{41}$$

The second condition we obtain by using Eqs. (38,40) which leads to

$$p(\mathbf{x}_3; t_3|\mathbf{x}_1; t_1) = \int_{-\infty}^{\infty} p(\mathbf{x}_3; t_3|\mathbf{x}_2, t_2) \cdot p(\mathbf{x}_2; t_2|\mathbf{x}_1; t_1) d\mathbf{x}_2. \tag{42}$$

This equation is called the Chapman-Kolmogorov equation. A process that obeys these two conditions in Eqs. (41, 42) is markovian.

2.3.4 Master equation

The Chapman Kolmogorov equation allows us to build up the transition probabilities iteratively at all times, if we know the transition probabilities

at small times. For small times the transition probability can be written as

$$p(\mathbf{x}|\mathbf{z},\tau') = (1 - a_0\tau')\delta(\mathbf{x}-\mathbf{z}) + \tau' w(\mathbf{x}|\mathbf{z}) + \sigma(\tau') \tag{43}$$

where τ' is a small time step, $w(\mathbf{x}|\mathbf{z})$ is the transition probability from $\mathbf{z} \to \mathbf{x}$ and $a_0(\mathbf{z}) = \int_{-\infty}^{\infty} w(\mathbf{x}|\mathbf{z})d\mathbf{x}$. By substituting this approximation into the Chapman Kolmogorov equation we obtain

$$p(\mathbf{z}|\mathbf{x},\tau+\tau') = (1 - a_0(\mathbf{z})p(\mathbf{z}|\mathbf{x},\tau) + \tau' \int_{-\infty}^{\infty} w(\mathbf{z}|\mathbf{y})p(\mathbf{y}|\mathbf{x},\tau)d\mathbf{y}. \tag{44}$$

Some simple transformations lead to

$$\frac{\partial}{\partial\tau}p(\mathbf{z}|\mathbf{x},\tau) = \int_{-\infty}^{\infty}[w(\mathbf{z}|\mathbf{y})p(\mathbf{y}|\mathbf{x},\tau) - w(\mathbf{y}|\mathbf{z})p(\mathbf{z}|\mathbf{x},\tau)]d\mathbf{y}. \tag{45}$$

This is called the Master equation. For discrete systems the integral becomes a sum:

$$\frac{d}{dt}p_n(t) = \sum_m [w_{nm}p_m(t) - w_{mn}p_n(t)]. \tag{46}$$

This equation describes the gain loss in probability of all states noted by n. The change of all probabilities of states n is determined by all transitions into state n from all other states m (gain) and by all transitions out of state n into all other states m (loss). This equation allows us to fully describe the time evolution of a Markov process, even by knowing only transition probabilities at small time scales.

In most cases a process can be described in discrete states. The Master equation can be written in matrix notation for continuous time. We start with

$$\frac{\partial}{\partial t}p_n = \sum_{n'} w_{nn'}p_{n'} - w_{n'n}p_n, \tag{47}$$

where $p_n \in \mathbb{R}^n$ is the state vector that describes the probability to be in state n and $w_{nn'}$ is the transition probability. By introducing $\mathbb{W} = w_{nn'} -$

$\delta\left(\sum_{n''} w_{n''n}\right)$ we end up with

$$\frac{\partial}{\partial t} p_n = \mathbb{W} \cdot p_n. \tag{48}$$

Thus the time evolution can be written as

$$p(t) = \mathbb{W}^t p(0) \tag{49}$$

and the steady state distribution p^s can be obtained from the corresponding eigenvector to the eigenvalue $\lambda = 1$. The existence of a stationary distribution is guaranteed for a irreducible matrix by the Perron Frobenius theorem.

Another very important condition for closed, isolated and finite physical systems under certain restrictions is the condition of detailed balance. This condition can be written as

$$w_{nn'} p^s_{n'} = w_{n'n} p^s_n \tag{50}$$

where $w_{nn'}, w_{n'n}$ are the transition probabilities and $p^s_{n'}, p^s_n$ are the stationary probabilities, respectively. This condition guarantees the time reversibility of a Markov process and says that no probability can be produced or destroyed.

The Master equation for discrete state in Eq.(48) can be solved by expansion in Eigenfunctions, by

$$p(t) = \sum_{\lambda} c_\lambda \phi_\lambda e^{-\lambda t}, \tag{51}$$

where λ are all possible eigenvalues and ϕ_λ the corresponding eigenvectors. The coefficients can be determined by $p(0) = \sum_\lambda c_\lambda \phi_\lambda$. This solution shows us that the time evolution at long time scales is determined by the largest eigenvalues and they thus play a key role in analyzing Markov processes.

2.3.5 Continuous dynamics

In the previous sections 2.3.3 - 2.3.4 general stochastical methods were derived. Now we shall turn to the more specific application of conformation dynamics to a biomolecule.

First, we consider a state space Ω that contains all dynamical variables to describe the current state of the system which is still continuous in time. Again Ω could contain all position and momenta of the atoms of molecules under investigation, as well as surrounding bath particles. We define the dynamical process that is considered by $\mathbf{x}(t) \in \Omega$. The state vector $\mathbf{x}(t)$ has some important properties:

- $\mathbf{x}(t)$ is a Markov process as defined in Section 2.3.2. We write the transition probability as

$$p(\mathbf{x}, \mathbf{y}; \tau)d\mathbf{y} = P[\mathbf{x}(t+\tau) \in d\mathbf{y} | \mathbf{x}(t) = \mathbf{x}], \tag{52}$$

  where $\mathbf{x}, \mathbf{y} \in \Omega, \tau \in \mathbb{R}^0_+$. This function describes the probability that the system is at state $\mathbf{x}$ at time t and is in an infinitesimal region dy around $\mathbf{x}$ at point $\mathbf{y}$ at time $t + \tau$.

- For a subset $A \subset \Omega$ the transition probability is defined as

$$p(\mathbf{x}, A; \tau) = P[\mathbf{x}(t+\tau) \in A | \mathbf{x}(t) = x] = \int_{\mathbf{y} \in A} d\mathbf{y} p(\mathbf{x}, \mathbf{y}; \tau). \tag{53}$$

- $\mathbf{x}(t)$ is ergodic, meaning that our state space Ω does not have dynamically disconnected subsets. With $t \to \infty$ we are able to visit all states infinitely often. The stationary distribution in equilibrium is given by $\mu(\mathbf{x}) \in \Omega$. For molecular dynamics at constant temperature T the stationary density is a function of T:

$$\mu(x) = Z(\beta)^{-1} exp(-\beta) H(x)) \tag{54}$$

  with the Hamiltonian $H(x)$ and $\beta = 1/k_B T$, the Boltzmann constant k_B , the thermal energy $k_B T$ as well as the partition function $Z(\beta) =$

$\int exp(-\beta H(x))dx$.

- $\mathbf{x}(t)$ is time reversible. As defined in Sec. 2.3.4 that is $\mathbf{x}(t)$ fulfills the detailed balance condition

$$p(\mathbf{x}, \mathbf{y}; \tau)\mu(\mathbf{x}) = p(\mathbf{y}, \mathbf{x}; \tau)\mu(\mathbf{y}). \tag{55}$$

These conditions are necessary to use the mathematical methods described above and define the system in a physical reasonable sense.

2.3.6 Transfer operator

We consider an ensemble of molecular systems at time t described by the probability density $p_t(\mathbf{x})$. After the time interval τ the probability density has changed accordingly to the transition probability density $p(\mathbf{x}, \mathbf{y}; \tau)$. The system has thus evolved or relaxed towards the equilibrium distribution, which can be described by the action of a continuous operator

$$p_{t+\tau}(\mathbf{y}) = \mathcal{Q}(\tau) \circ p_t(\mathbf{y}) = \int_{\mathbf{x}\in\mathbf{\Omega}} d\mathbf{x} p(\mathbf{x}, \mathbf{y}; \tau) p_t(\mathbf{x}). \tag{56}$$

An equivalent form of this equation can be obtained by using the transfer operator, which is defined as

$$u_{t+\tau}(\mathbf{y}) = \mathcal{T}(\tau) \circ u_t(\mathbf{x}) = \frac{1}{\mu(\mathbf{y})} \int_{\mathbf{x}\in\Omega} d\mathbf{x} p(\mathbf{x}, \mathbf{y}; \tau)\mu(\mathbf{x}) u_t(\mathbf{x}). \tag{57}$$

The functions $u_t(\mathbf{x})$ are connected to the probability densities by

$$p_t(\mathbf{x}) = \mu(\mathbf{x}) u_t(\mathbf{x}). \tag{58}$$

Both operators $\mathcal{Q}(\tau)$ and $\mathcal{T}(\tau)$ fulfill the Chapman-Kolmogorov equation (Eq. 42) expressed by

$$p_{t+k\tau}(\mathbf{x}) = [\mathcal{Q}(\tau)]^k \circ p_t(\mathbf{x}), \tag{59}$$

$$u_{t+k\tau}(\mathbf{x}) = [\mathcal{T}(\tau)]^k \circ u_t(\mathbf{x}), \tag{60}$$

where $[\mathcal{T}(\tau)]^k$ is the k-fold application of the operator. We note ϕ_i and λ_i as the eigenvectors and the corresponding eigenvalues from the operator $\mathcal{Q}(\tau)$ and as ψ_i and λ_i the eigenvectors and eigenvalues of $\mathcal{T}(\tau)$. The operators can then be written as $\mathcal{Q}(\tau) \circ \phi_i(\mathbf{x}) = \lambda_i \phi_i$ and $\mathcal{T}(\tau) \circ \psi_i(\mathbf{x}) = \lambda_i \psi_i$. By the condition of detailed balance the dynamics are reversible and thus all λ_i are real and $-1 < \lambda_i < 1$ and the eigenvectors of both operator are connected by $\phi_i(\mathbf{x}) = \mu(\mathbf{x})\psi_i(\mathbf{x})$. By the condition of detailed balance the existence of one eigenvalue $\lambda = 1$ is guaranteed. The corresponding eigenvector represents the stationary distribution of the operator.

By the decomposition in eigenvalues and eigenvalues we obtain a subset of slow processes and one of the remaining fast processes. The decomposition can be written as

$$
\begin{aligned}
u_{t+k\tau}(\mathbf{x}) &= \mathcal{T}_{\text{slow}}(k\tau) \circ u_t(\mathbf{x}) + \mathcal{T}_{\text{fast}}(k\tau) \circ u_t(\mathbf{x}) && (61)\\
&= \sum_{i=1}^{m} \lambda_i^k \langle u_t, \phi_i \rangle \psi_i(\mathbf{x}) + \mathcal{T}_{\text{fast}}(k\tau) \circ u_t(\mathbf{x}) && (62)\\
&= \sum_{i=1}^{m} \lambda_i^k \langle u_t, \psi_i \rangle_\mu \psi_i(\mathbf{x}) + \mathcal{T}_{\text{fast}}(k\tau) \circ u_t(\mathbf{x}). && (63)
\end{aligned}
$$

In the above equation λ_i are the eigenvalues of $\mathcal{T}_{\text{slow}}(\tau)$ and ψ_i, ϕ_i the corresponding left and right eigenvectors, respectively. By $\langle \cdot \rangle$ we denote the normal scalar product and $\langle \cdot \rangle_\mu$ denotes the weighted scalar product defined by $\langle f, g \rangle_\mu = \int \mu(x) f(x) g(x) dx$. The physical interpretation of the above equations is that the slow dynamics that are dominant in the system are describe by the largest eigenvalues $\lambda_i, i = 1, ..., m$ and the corresponding eigenvectors ψ_i, ϕ_i. Each λ_i corresponds to a physical process with a timescale, that indicates how fast this process equilibrating. This can be described by the implied timescale that is defined as

$$
t_i = -\frac{\tau}{\ln \lambda_i}. \qquad (64)
$$

By using this definition we can write Eq. 63 as

$$u_{t+k\tau}(\mathbf{x}) = 1 + \sum_{i=2}^{m} \exp\left(-\frac{k\tau}{t}\right) \langle u_t, \psi_i \rangle_\mu \psi_i(\mathbf{x}) + \mathcal{T}_{\text{fast}}(k\tau) \circ u_t(\mathbf{x}). \tag{65}$$

This formulation can now be used to analyze data that is obtained from simulation.

2.3.7 Discretization

When looking at the amounts of data generated by molecular dynamics simulation often consisting of hundreds of thousands atoms each with their positions and momenta at each time step, the need for a simplified model in order to make it human understandable becomes obvious. A first step in this simplification is a conversion of space-continuous (limited only by numerical precision) trajectories to a discrete set of states, called microstates. The assignment of each frame of the trajectory to such a microstate if often solely based on the atom positions of the molecules at interest, thus disregarding velocities as well as bath molecules. While the full description of the system markovian by construction, disregarding information such as velocity will certainly affect the markovianty of our model. First we want to introduce the discretization of the model.

We start by discretizing the state space Ω into n sets. These sets are described by $S = \{S_1, \ldots, S_n\}$ partitioning the state space by $\bigcup_{i=1}^{n} S_i = \Omega$ without overlap $S_i \cap S_j = \emptyset, \forall i \neq j$. We define functions $\{v_1(\mathbf{x}), \ldots, v_n(\mathbf{x})\}$ with $v_i(\mathbf{x}) \geq 0, \forall \mathbf{x} \in \Omega$, that measure the similarity of points $\mathbf{x}$ to each of the n sets. The probability of $\mathbf{x}$ belonging to set i is

$$\chi_i(\mathbf{x}) = \frac{v_i(\mathbf{x})}{\sum_{j=1}^{n} v_j(\mathbf{x})}. \tag{66}$$

Here we will use a crisp discretization by using simple step functions

$$v_i(\mathbf{x}) = \chi_i(\mathbf{x}) = \begin{cases} 1 & \mathbf{x} \in S_i \\ 0 & \mathbf{x} \notin S_i \end{cases}. \tag{67}$$

The stationary distribution π_i to be in set i is then given as

$$\pi_i = \int_{\mathbf{x}\in\mathbf{S_i}} d\mathbf{x}\mu(\mathbf{x}) \tag{68}$$

and the local stationary distribution $\mu_i(\mathbf{x})$ in set i is

$$\mu_i(\mathbf{x}) = \begin{cases} \frac{\mu(\mathbf{x})}{\pi_i} & \mathbf{x} \in S_i \\ 0 & \mathbf{x} \notin S_i \end{cases}. \tag{69}$$

2.3.8 Transition matrix

The discrete analogon to our transfer operator $\mathcal{T}(\tau)$ as described in Sec. 2.3.6 can now be approximated via:

$$T_{ij}(\tau) = (\chi_i \cdot \mathcal{T}(\tau) \circ \chi_j). \tag{70}$$

A element $T_{ij}(\tau)$ can physically be interpreted as the conditional or transfer probability that the system is in state j at time $t + \tau$ given that it was in state i at time t. Using the definition of the conditional probability from Eq. 27, we can express the transition matrix as:

$$\begin{aligned} T_{ij}(\tau) &= P[\mathbf{x}(t+\tau) \in S_i | \mathbf{x}(t) \in S_i] && (71) \\ &= \frac{P[\mathbf{x}(t+\tau) \in S_i \cap \mathbf{x}(t) \in S_i]}{P[\mathbf{x}(t) \in S_i]} && (72) \\ &= \frac{\int_{\mathbf{x}\in S_i} d\mathbf{x}\mu_i(\mathbf{x}) p(\mathbf{x}, S_i, \tau)}{\int_{\mathbf{x}\in S_i} d\mathbf{x}\mu_i(\mathbf{x})}. && (73) \end{aligned}$$

Note that we only need the local equilibrium distribution $\mu_i(\mathbf{x})$ and not the global distribution $\mu(\mathbf{x})$ what is very important from a practical point of view. The transition matrix can be obtained by independent simulation runs but we can construct the transition matrix from each subset using the equations above.

This mathematical framework established here will be used in Sec. 3.2 in order to build a Markov state model of the conformation dynamics of a peptide.

Now we can show the relation between the transition matrix and the rate matrix as used in Eq. 48. By writing Eq. 49 as

$$\frac{dp_i(t)}{d(t)} = p^T(t)\mathbb{W} \tag{74}$$

we can write the transition matrix as

$$T(\tau) = \exp(\tau\mathbb{W}). \tag{75}$$

By eigendecomposition of this equation we obtain finally

$$[\psi_1, \ldots, \psi_n]\text{diag}\{\lambda_1, \ldots, \lambda_n\}[\psi_1, \ldots, \psi_n]^{-1} = \tag{76}$$

$$[\psi_1, \ldots, \psi_n]\text{diag}\{\exp(\tau\gamma_1), \ldots, \exp(\tau\gamma_n)\}[\psi_1, \ldots, \psi_n]^{-1} \tag{77}$$

The γ_i are the eigenvalues of $\mathbb{W}$ and have the physical meaning of rates and are equivalent to the inverse time scales $t_i^{-1} = -\gamma_i$ and we get the relation $\exp(\tau t_i^{-1}) = \lambda_i(\tau)$ what leads to the form of the implied time scales, see Eq. 64. By the eigenvalues we can thus identify the different timescales underlying the system. The first m eigenvalues that are close to 1 represent the slowest processes that dominate the system. Of special interest is the second largest eigenvalue that generates the slowest relaxation time t_2. This value is the worst case of equilibration. If one wants to take an ensemble average of an observable $\langle A \rangle$ out of a simulation one has to simulate the trajectory many times t_2 in order to get good estimates of $\langle A \rangle$.

2.3.9 Perron cluster cluster analysis

Biomolecules often show to have long-lived, also called metastable states. Identification and extraction of those metastable states based on the Markov model has recently been addressed by Deuflhard et al. [8] with a method called Robust Perron Cluster Cluster Analysis.

We assume that our transition matrix is a stochastic matrix. The Perron Frobenius theorem states that for such a matrix an eigenvalue $\lambda_1 = 1$ that is simple and dominant, i.e. $|\lambda_i| < 1$ for all other eigenvalues, exists. This

dominant eigenvalue is called the Perron root. The corresponding left eigenvector π_1 represents the stationary distribution and the corresponding right eigenvector is $e_1 = (1, ..., 1)^T$.

Uncoupled Markov chains We now assume that the underlying Markov chain consists of k uncoupled aggregates or clusters. This means that there are no transitions between these clusters. One can prove that such a matrix can be transformed by decomposition into block diagonal form

$$T = \begin{pmatrix} D_{11} & 0 & \dots & 0 \\ \vdots & D_{22} & \dots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \dots & D_{kk} \end{pmatrix}, \tag{78}$$

where each block D_{ii} is a square stochastic matrix. Due to the Perron Frobenius theorem, such a matrix has an k-fold eigenvalue $\lambda = 1$. Each block has a corresponding right eigenvector $e_i = (1, \dots, 1)^T$ with length $\dim(D_{ii})$. The eigenvectors of matrix T corresponding to $\lambda = 1$ is spanned by the vectors

$$\chi_{A_i} = (0, \dots, 0, e_i^T, 0, \dots, 0)^T, i = 1, \dots, k. \tag{79}$$

The eigenvectors can be interpreted as characteristic functions of the uncoupled aggregates or clusters. Any basis $\{X_i\}_{i=1,\dots,k}$ of the eigenspace can be written as linear combination of the eigenvectors

$$X_i = \sum_{j=1}^{k} \alpha_{ij} \chi_{A_i}, i = 1, \dots, k \tag{80}$$

where $\alpha_{ij} \in \mathbb{R}$ are the coefficients. With this expression we can analyze the aggregates by a collection of states with a common sign structure.

Nearly uncoupled Markov chains The transition matrix generated from simulations should never be fully uncoupled. When this is the case, we can clearly assume that the trajectory is not ergodic, as there are states that

cannot be reached from other states. In order to perform meaningful analysis, we then either perform more simulations or adjust the discretization.

Using a coupled transition matrix, our aim is to reorder the elements so that we have aggregates D_{ii} with high transition probabilities within these sets, whereas $|E_{ij}| \leq \mathcal{O}(\epsilon)$ are small transitions probabilities between these blocks.

$$T = D + E = \begin{pmatrix} D_{11} & E_{12} & \dots & E_{1k} \\ \vdots & D_{22} & \dots & E_{2k} \\ E_{i1} & E_{i2} & \ddots & \vdots \\ E_{k1} & E_{k2} & \dots & D_{kk} \end{pmatrix}. \tag{81}$$

When ϵ is sufficiently small, the eigenvalues are continuous in ϵ and the spectrum of T can be divided in three parts. First the Perron root $\lambda = 1$, second a cluster of $k-1$ eigenvalues $\lambda_2(\epsilon), ..., \lambda_k(\epsilon)$ that approach 1 for $\epsilon \to 0$ and the remaining spectrum which is smaller than 1 for $\epsilon \to 0$. The Perron Cluster Cluster Analysis (PCCA) algorithm then works as follows:

- Compute all eigenvalues of the transition matrix
- Find the cluster of eigenvalues close to the Perron eigenvalue $\lambda = 1$
- Analyze the sign structure of the corresponding right eigenvectors and assign each microstate according to the signs of these eigenvectors

This method has shown to be numerically unstable when applied to real data. That is why an improvement to this method known as Robust Perron Cluster Cluster Analysis (PCCA+) was developed. PCCA+ performs the assignment of microstates to cluster not based on the sign structure, but by performing an optimization in such a way that the metastability of the system is minimized. This means that it finds a decomposition into subsets that minimizes transitions between these sets.

2.3.10 Transition-Path Theory (TPT)

Understanding how a molecules transitions between different metastable states is one of the major challenges in protein folding today. This can be addressed

with a method called Transition-Path Theory (TPT) [31]. Given two sets of microstates, a base set A, and a target set B we want to know the probability distribution of the trajectories leaving set A and entering set B.

First we define a forward-committor probability q_i^+ which tells us, being in a microstate i, what is the probability that the system will reach set B rather than A. By definition we set

$$q_i^+ = \begin{cases} 0 & i \in A \\ 1 & i \in B \end{cases}. \tag{82}$$

To compute q_i^+ for all intermediate states not part of A or B, the following system of equations has to be solved:

$$-q_i^+ + \sum_{k \in I} T_{ik} q_k^+ = -\sum_{k \in B} T_{ik}. \tag{83}$$

The backward-committor q_i^- is the probability of when being at state i, gives us the probability of having been in set A before, as opposed to B. For a system in equilibrium this simply becomes $q_i^- = 1 - q_i^+$. Using both committors, we can now define the effective flux f_{ij} as probability flux along a edge i,j contributing to the transition A$\rightarrow$ B as:

$$f_{ij} = \pi_i q_i^- T_{ij} q_j^+. \tag{84}$$

Because the effective flux may contain unnecessary detours f_{ij} as well as f_{ji} both will be positive for a pair i,j in the intermediate set of states, it is reasonable to consider only the net flux f_{ij}^+ given by

$$f_{ij}^+ = max\{0, f_{ij} - f_{ji}\}. \tag{85}$$

Using these equations, transition pathways between different sets can be computed.

3 Results

The following section will describe the simulations and analysis conducted on the systems consisting of a MHC-I protein of class B27 as well as a peptide. To assist the work of matching the electron density maps of these MHC-I-bound peptides, simulations of 100 ns length were performed. The first part will discuss these simulations and analyze them using standard MD techniques. Furthermore, using the data of a long 2 µs simulation of the pB27 peptide in water, a MSM could be constructed and many important statistical properties could be derived. Lastly, I will discuss the attempt to build a MSM of the MHC-I-bound pB27 peptide. Unfortunately it turned out that the simulations performed during the time of this study were insufficient to accomplish that task. Nevertheless, an outlook and estimation on how that might be accomplished in the future will be given.

3.1 100 ns simulations of MHC complexes

3.1.1 Systems

The simulated systems are two human MHC class I subtypes (HLA-B*2704 and HLA-B*2705) complexed with two different peptides (pB27 and pCP). pB27 and pCP are peptides built of 12 and 11 amino acids respectively. In short amino acid sequence notation they can be written as RRKSSG-GKGGSY (pB27) and RRFKEGGRGGK (pCP). While HLA-B*2704 and HLA-B*2705 differ only in very few amino acids (see Table 1), X-ray crystallography of both HLAs complexed with the same pB27 was performed with varying resolution: While two peptide conformations could be fully resolved in HLA-B*2705:pB27, only few amino acids of pB27 could be resolved in HLA-B*2704. Crystallography as well as simulation has previously been performed on similar systems [28, 27, 43, 33], but the three combinations of MHC molecules and peptides has not been simulated, yet.

It should be noted that the provided PDB files of HLA-B*2705 complexed with pB27 did not contain the terminal Methionine residue, which is in contrast present in B*2705:pCP (Met277). Unfortunately, this difference was

chain	$\alpha 2$	$\alpha 2$	$\alpha 3$
residue id	77	152	211
B*2704	Ser	Glu	Gly
B*2705	Asp	Val	Ala

Table 1: Difference between HLA-B*2704 and HLA-B*2705

noticed only recently. However, as the residue is in far distance from the binding groove, we expect it not to affect the peptide dynamics.

3.1.2 Molecular Simulations

As start structures we received different PDB files for HLA-B*2705:pB27 (in two peptide conformations A and B, corresponding to chain C and D in the given PDB file pB27_oct_6_refmac13.pdb) and HLA-B*2705:pCP. As for HLA-B*2704:pB27, where the peptide structure could not be reconstructed, we used the peptide from HLA-B*2705:pB27 in conformation A and aligned it in the binding groove.

All simulation steps were performed using gromacs 4.0.5 [17]. A similar setup as described by Pöhlmann et al. [33] who previously simulated HLA*B27 subtypes using gromacs. We set gromacs to use the OPLS all-atom force field [22]. First, the system was prepared by running a short 250 step vacuum energy minimization using the steepest descent integrator. Simulations where then performed in a periodic box sized 8.9 x 8.3 x 10.0 nm and solvated using TIP4P [21] water molecules, corresponding to a water shell thickness of at least 1,4 nm around the protein. The whole system consisted of around 97,000 atoms. To compensate the net negative charge of the MHC molecules Na^+- and Cl^--Ions were added. The solvated system was also relaxed using a steepest descent integrator with 250 steps, followed by a 2500-step position restrained simulation that fixed all bonds and allows the water molecules align without clashes. To allow for a integration step of 2 fs we applied the SHAKE algorithm to all hydrogen bonds. Electrostatic interactions were calculated explicitly within a cut-off radius of 1 nm and using the Particle Mesh Ewald Method [6] for long-range interactions. Tem-

perature coupling at 310 K was provided by the Langevin thermostat [7] by using the stochastic dynamics (SD) integrator with a friction constant of 0 $\frac{u}{ps}$ and setting tau_t to 2 ps. The simulations were carried out on four nodes of the biocomputing cluster in parallel, each node consisting of eight Intel Xeon CPU cores clocked at 2.33 GHz. 100 ns simulations where completed for all systems, except HLA-B*2705:pCP, for which 71 ns are currently completed. For a detailed explanation of all steps executed and the reference of gromacs parameter files, see Appendix A.

3.1.3 Visual analysis

A first visual analysis carried out using VMD showed very different outcomes for the simulated systems. While the N-terminus (Argenine residue id 376) of pB27 remains in place simulated both in HLA-B*2704 and HLA-B*2705, C-terminus (Tyrosine) remains only in place in HLA-B*2704 and is much more flexible in HLA-B*2705 (see Fig. (8)). While it still remains somewhat in place in the simulation started from conformation A it is almost completely flexible in the simulation started from conformation B. It appears that pB27 in B*2704 is stabilized by a H-bond at residue id 152.

3.1.4 Flexibility analysis

Minimal Root Mean Square Deviation

Going beyond the visual analysis it is necessary to back this information up using a metric. When comparing two conformations of the same molecule the minimal Root Mean Square Deviation (minimal RMSD) is the most widely accepted distance measure. Given two structures x and y of the same molecule consisting of N atoms, assuming that x_i and y_i represent the position vector of the ith atom, the RMSD computes as follows

$$d_{rmsd} = \sqrt{\frac{1}{N} \sum |x_i - y_i|^2}. \tag{86}$$

To account for the translation and rotation invariance of the complete molecule, the minimal RMSD is computed by aligning both structures in a way that

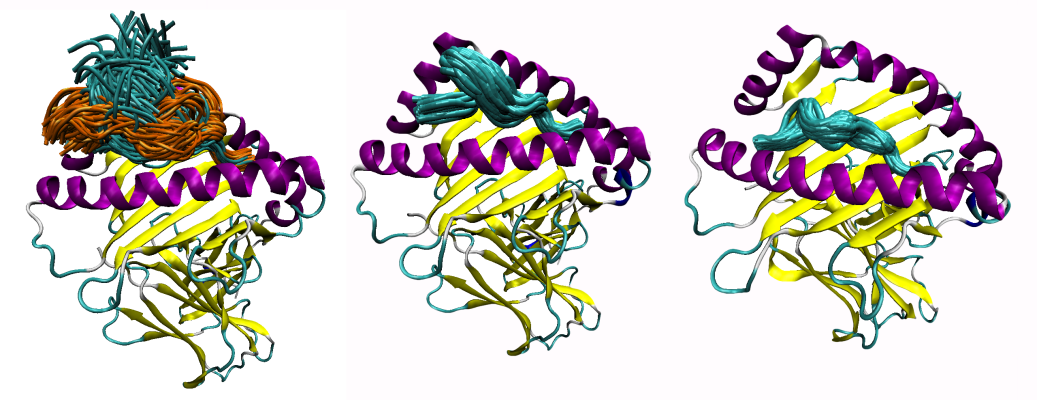


Figure 8: Backbone conformations visualized in MHC complex of HLA-B*2705:pB27 (orange: conformation A, green: conformation B) HLA-B*2704:pB27 and HLA-B*2705:pCP (from left to right). These images were generated by showing the backbone of every 500th frame in VMD and show the much greater flexibility of B*2705:pB27 compared to the two other systems.

this value becomes minimal.

Regular Spatial Clustering

We used EMMA [13] to convert the continuous simulation data into a trajectory of discrete states. We employed the regular spatial algorithm using minimal RMSD metric on the backbone atoms with a minimum distance $d_{min} = 1$ Å. The algorithm works as follows:

- Use the coordinates in the first frame and assign it as the first element of a list of cluster centers
- Compute minimum RMSD distance from coordinates of following frames to all cluster centers
- if distance is larger d_{min} create a new cluster center add it to the list
- After all cluster centers have been found, assign each frame to its closest cluster center

Table 2 outlines the number of clusters found for each simulation, as expected the most clusters occur in B*2705:pB27, indicating a high backbone

system	# clusters
B*2705:pB27 conformation A	240
B*2705:pB27 conformation B	298
B*2704:pB27	26
B*2705:pCP	13

Table 2: Regular Spacial Clustering

flexibility. In B*2704:pB27 and B*2705:pCP less clusters are found, suggesting stronger restrained peptides. The number of unique clusters discovered with progressing simulation time are shown in Fig. 9.

Peptide backbone flexibility

To analyze the peptides backbone flexibility, we computed average peptide backbone *root mean square deviation* (RMSD) distances to a reference structure for simulation snapshots taken in nanosecond intervals. The results are depicted in Fig. 10. As reference structure backbone coordinates of the peptide after a 1 ns equilibration phase were used.

The RMSD plot shows that HLA-B*2705:pB27 is more flexible when starting the simulations from conformation B than it is when starting from conformation A and B*2705:pCP, while B*2704:pB27 as well as B*2705:pCP are both less flexible. These results are in contrast to X-ray crystallography experiments where B*2704:pB27 could not be reconstructed. Fig. 11 shows the average root mean square fluctuation per amino acid during the 100 ns simulations. In B*2705:pB27 starting from conformation B especially the Tyorsine at the C-terminus appears to be very flexible, with an average RMSD of over 4 Å. On the contrary, B*2705:pCP and B*2704:pB27 do not exhibit RMS fluctuations greater than 1 Å.

3.1.5 Highly populated structures

In order to examine which peptide conformations occur in the simulation trajectories, they were again clustered spatially. For this purpose the full trajectories were thinned out by considering only structures at every 100th

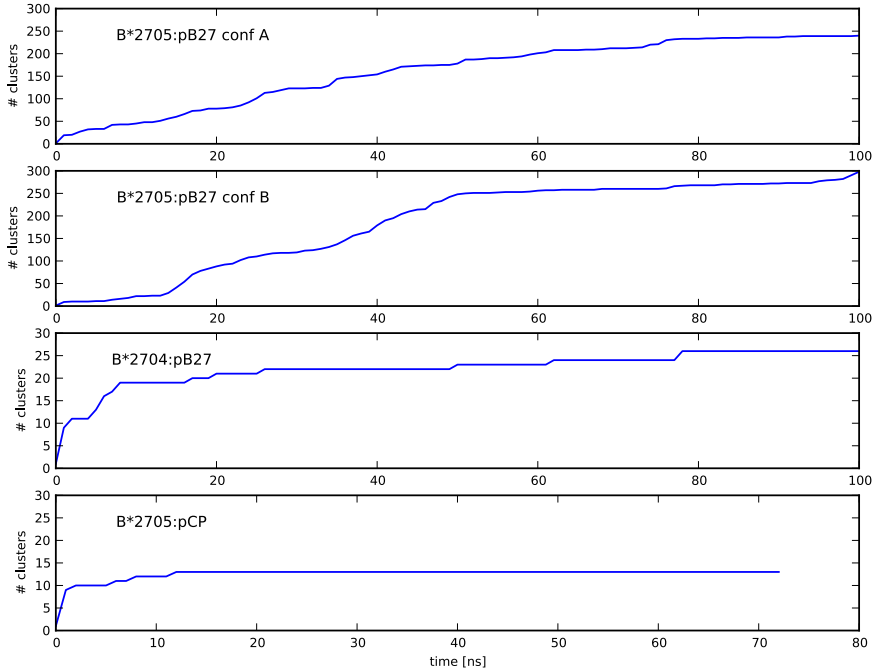
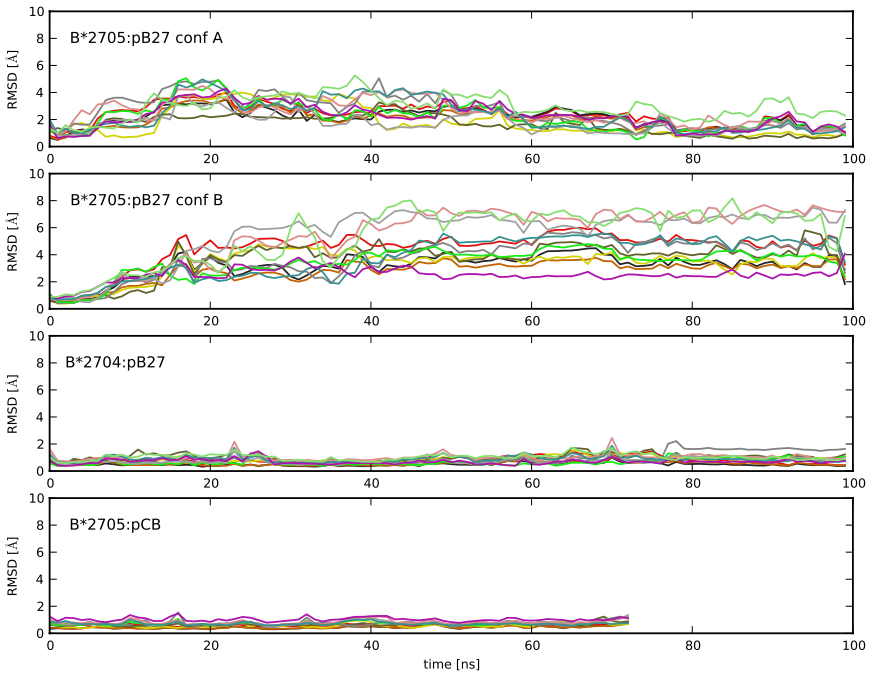
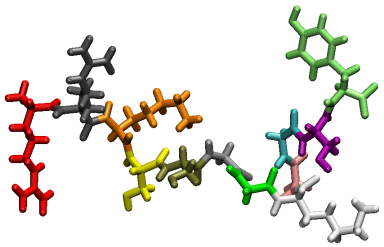


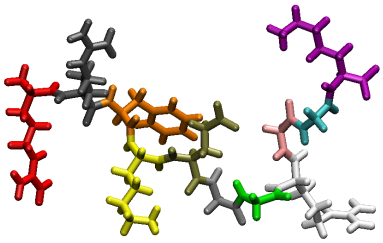
Figure 9: Number of unique clusters over time as given by regular spatial clustering. While B*2704:pB27 or B*2705:pCP look saturated after about 80 and 13 ns respectively, for B*2705:pB27 there still new unique clusters at the very end of the simulation.



(a) Backbone RMSD



(b) pB27 (RRKSSGGKGGSY)



(c) pCP (RRFKEGGRGGK)

Figure 10: Peptide backbone RMSD and the peptides. The peptides amino acid colors correspond to the line of the same color in the RMSD plot. It is obvious that B*2705:pB27 exhibits a much higher flexibility than both B*2704:pB27 and B*2705:pCP.

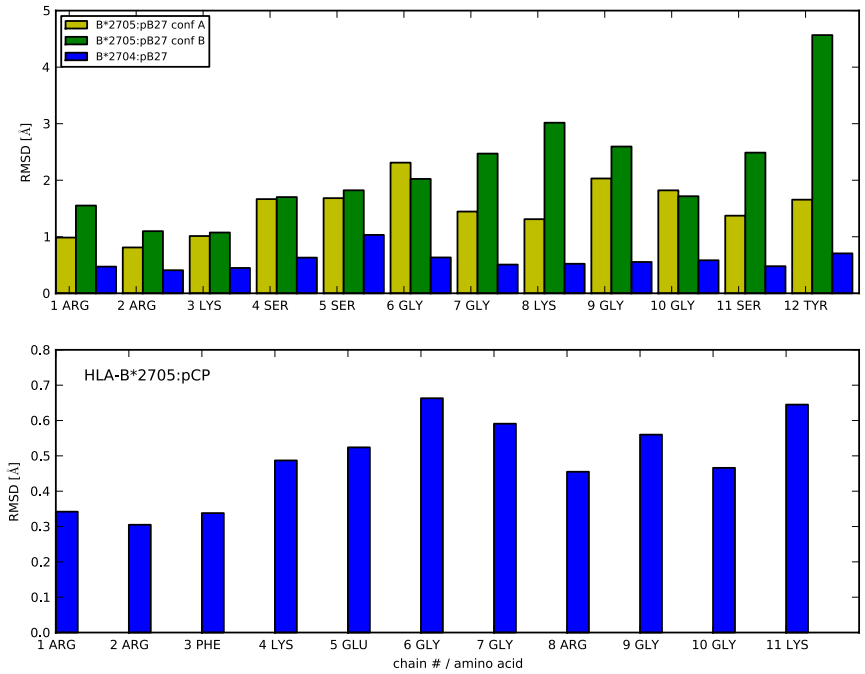
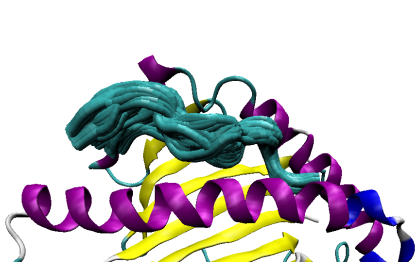
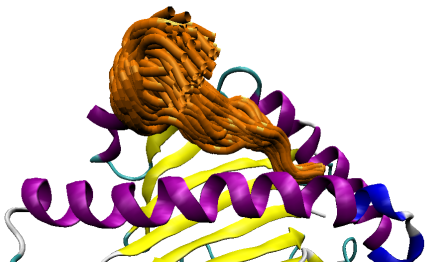


Figure 11: Average RMSD per amino acid for all three simulated systems. Note how the C-terminus of B*2705:pB27 in conformation B (12 Tyrosine) has the largest RMSD, supporting the visual analysis that it can move freely in this simulation.

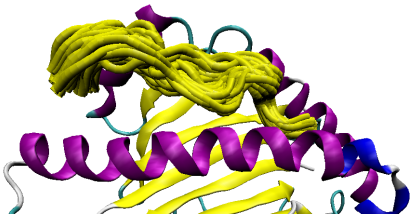
ps. Conformation A and B trajectories of B*2705:pB27 were concatenated beforehand. The minimal distance in the regular spatial clustering algorithm was set to $d_{min} = 1.8$ Å for all systems under consideration. For B*2705:pCP and B*2704:pB27 this procedure resulted in a single cluster. For B*2705:pB27 we selected the five highest populated clusters of the 32 found. The total number of structures in these five clusters corresponds to 51% of all structures in the thinned trajectory. These backbones of all three systems are shown in Fig. 12.



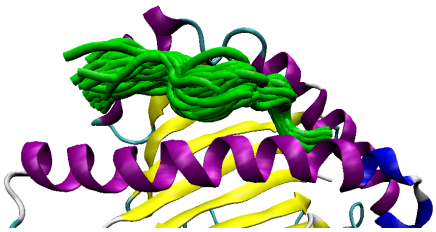
(a) B*2705:pB27 cluster 1



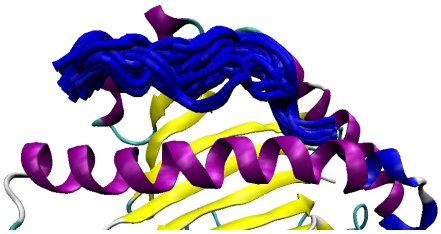
(b) B*2705:pB27 cluster 2



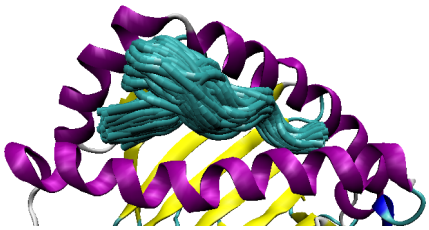
(c) B*2705:pB27 cluster 3



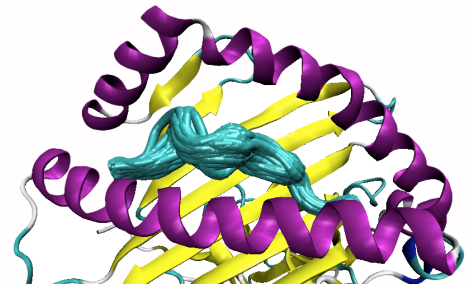
(d) B*2705:pB27 cluster 4



(e) B*2705:pB27 cluster 5



(f) B*2704:pB27



(g) B*2705:pCP

Figure 12: Clustered backbones of simulated systems

3.2 Conformation dynamics of pB27 in water

3.2.1 Simulation parameters and data

In order to construct a Markov State Model (MSM) from simulation data, a simulation of 2 µs length was started of the pB27 peptide in water. The solvated system consisted of around 130,000 atoms. This simulation took about 120 days to complete, using eight cores of one of the nodes at our CMB cluster. The starting structure was the peptide extracted from the B*2705:pB27 PDB file. Except for the smaller periodic box size, the same parameters as in Sec. 3.1 were used. The following analysis was mostly carried out using the EMMA software package [13].

3.2.2 Discretization

When converting the trajectory data (taken at every ps) into discrete states the aim is to find a discretization that is just sufficiently fine in order to avoid the assignment of distinct states into the same cluster, which would introduce memory to our system. If discretization becomes too fine, leaving only very few visits of each state, the error grows very large. The computational cost of further analysis also grows with the number of clusters. The first 10 ns were cut off from the beginning of the trajectory to allow the system to equilibrate. Analysis was based on heavy atoms only, meaning the H-atoms were removed from the trajectory.

It was found that regular spatial clustering combined with a distance parameter of 3 Å computed using the minimal RMSD metric produced best results. This resulted in a trajectory made up of 1348 discrete microstates. This discrete trajectory was therefore used in all further analysis.

3.2.3 Count and transition matrix estimation

Using the discretized trajectory at time resolution Δt (in our case 1 ps), the next step is to estimate a transition matrix $\hat{T}(\tau)$. The method used here, called window count, estimates a count matrix by sampling the discrete trajectory using a window of width τ and shifting it along the trajectory:

$$c_{ij}^0(\tau) = c_{ij}^0(l\Delta t) = \sum_{k=0}^{N-l} \chi_i(x_k)\chi_j(x_{k+l}). \tag{87}$$

Here χ_i is a step function with x_k being microstate the found in the discrete trajectory at time $k\Delta t$:

$$\chi_i(x_k) = \begin{cases} 1 & x_k = i \\ 0 & \text{otherwise} \end{cases}. \tag{88}$$

Here we are shifting a count window of width τ at intervals Δt along the trajectory and observe if a transition $i \to j$ occurs. This means that c_{ij}^0 is the number of times the trajectory was observed in state i at time t and in state j at time $t+\tau$, summed over all times t. A estimation of the transition matrix $\hat{T}$ is given by the estimator

$$\hat{T}_{ij} = \frac{c_{ij}^0}{c_i^0}. \tag{89}$$

Here, c_i^0 denotes the row sums of our count matrix

$$c_i^0 = c_i^0(\tau) := \sum_{k=1}^{n} c_{ik}^0, \tag{90}$$

which is the number or times the trajectory was in state i. It can be shown that this estimator leads to the maximum probability matrix $\hat{T} = \arg\max p(T|C^0)$ [34]. This estimator does however not necessarily fulfill the detailed balance condition $\pi_i \hat{T}_{ij} = \pi_j \hat{T}_{ji}$ even if the underlying dynamics is in equilibrium [2]. When we can assume that we have achieved equilibrium sampling, a simple way to ensure the detailed balance condition is met is to add reverse counts as they would occur when reversing the trajectory, by adding the transposition of the count matrix

$$C_{\text{reversible}} = C_{\text{forward}} + C_{\text{forward}}^T. \tag{91}$$

For simulations not in equilibrium this will lead to strong bias of our

model. However, it is possible to derive an maximum probability optimal reversible transition matrix from a non-symmetrical count matrix. Because no closed-form solution exists this is done by using a constrained iterative algorithm that is described in detail in the work of Prinz et al. [34].

3.2.4 Implied timescales

Since our discretization into microstates is solely based on the atom positions, we want to know at what minimum lagtime τ our model can be assumed to be markovian. Based on the eigenvalues of transition matrices computed for different lagtimes and using on Eq. 64 we can plot the lagtime dependence of the implied timescales. Markovianity can be assumed when these timescales become constant. These implied timescale plots for regular spatial clustering with the distance parameter of 3 Å. are shown in Fig. 13.

All three counting methods show a good convergence of the implied timescales at a lag time of 5 to 8 ns. The method of forward counting seems to estimate the real timescales of our system badly, while reversible counting, as well as forward counting with the computation of the optimal reversible transition matrix are consistent. All further analysis was thus carried out on the optimal reversible calculated transition matrix based on forward counting using a lagtime of 7 ns.

The slowest process in this model has a corresponding timescale of about 1 µs. While we did not expect the slowest process to be that slow, visual analysis of the trajectory confirmed that the peptide does indeed first contract to a more compact structure and extract again once during the simulation, at about 1 µs. Another test that can be carried in order to confirm the timescales are not artificial (e.g. due to bad discretization) is the projection on the first right eigenvectors (see Fig. 14). The first eigenvector is constant and the n + 1 right eigenvector corresponds to the n*th* slowest process. It can be seen that the trajectory switches between different values of the eigenvector (corresponding to different states) and relaxes there for some time.

3.2.5 Robust Perron Cluster Cluster Analysis (PCCA+)

In order to obtain a simple model that can be easily understood a classification into long-lived, "metastable" states is desirable. Therefore the PCCA+ algorithm developed by Deuflhard et al. as explained in Sec was used. 2.3.9. We chose to coarse-grain our model into 10 metastable sets. As this algorithm is implemented in EMMA, it could be used directly.

The probability of being in a certain set A can be computed by using the stationary probability π of the Markov model $\hat{T}(\tau)$ by

$$w_i^A = \begin{cases} \frac{\pi_i}{\sum_{j \in S} \pi_j} & i \in A \\ 0 & i \notin A \end{cases}. \tag{92}$$

Table 3 shows the PCCA+ sets with their probability and RMSD values. The probability is computed by summing up the stationary distribution values of all microstates found in each set (Eq. 92). The average RMSD in reference to the backbone complex structure is computed by averaging over all RMSD values over each frame belonging to a set. For each microstate we also computed an average RMSD per microstate. The table shows also the minimum and maximum values of this computation per set. It can be seen, that set #2 has the highest probability of 0,345. The set with the smallest average RMSD, set #7, also has the smallest probability. This makes sense since it the most extended structure which is very unstable due to the absence of a stabilizing secondary structure. For each set, we than extracted 100 random structures and visualized them using VMD. One representative structure is shown for each set and the together with the conformational flexibility of each set as indicated by the gray cloud in Fig. 15. Note how the most probable sets 2, 8 and 6, which make up about 78% of the total probability are stabilized by extensive secondary structures.

3.2.6 Chapman-Kolmogorov-Test

To validate our Markov model, it is essential to check if it is consistent with the data in the trajectory. Given a transition matrix $\hat{T}(\tau)$ estimated from

set#	probability	avg rmsd [Å]	min rmsd [Å]	max rmsd [Å]
0	0.020	5.37	3.39	5.87
1	0.019	5.32	3.22	5.92
2	0.345	5.37	2.62	7.40
3	0.036	5.65	2.38	6.29
4	0.075	6.34	2.77	7.22
5	0.025	5.18	4.44	6.13
6	0.192	6.39	4.00	7.39
7	0.016	4.21	2.67	5.09
8	0.245	5.81	2.74	7.10
9	0.029	6.75	4.50	7.04

Table 3: Probability and backbone RMSD values for the 10 PCCA+ sets.

data at lag time τ and $\hat{T}(k\tau)$ being the transition matrix estimated from the same data a longer lagtime $k\tau$ we can check how well the approximation

$$[\hat{T}(\tau)]^k \approx \hat{T}(k\tau) \tag{93}$$

holds within the statistical error. Since the estimated transition matrix can be error-prone, especially at states rarely visited, the test is best carried out on each of metastable sets identified by PCCA+. Using w^A as the initial probability vector for each set (Eq. 92), the probability of being at that set at a later times $k\tau$ is given by:

$$p_{MD}(A, A, k\tau) = \sum_{i \in A} w_i^A p_{MD}(i, A, k\tau) \tag{94}$$

where $p_{MD}(i, A, k\tau)$ is the estimation from trajectory data of the stochastic transition function Eq. (53):

$$p_{MD}(i, A, k\tau) = \frac{\sum_{j \in A} c_{ij}^0(k\tau)}{\sum_{j=1}^n c_{ij}^0(k\tau)} \tag{95}$$

where $c_{ij}^0(k\tau)$ is the number of transitions from state i to state j using a lag

time of $k\tau$. The same probability can be defined for our Markov model by:

$$p_{MSM}(A, A, k\tau) = \sum_{i \in A} [w^A T^k(\tau)]_{ii}. \tag{96}$$

Using the Chapman-Kolmogorov property, the Markov model can then be validated by checking how well the equality of

$$p_{MD}(A, A, k\tau) = p_{MSM}(A, A, k\tau) \tag{97}$$

holds. Due to the statistical uncertainties of the transitions probabilities from MD trajectories depending on the number of observed transitions between states this equality (97) cannot be expected to hold exactly. The error of the transition probabilities can therefore be computed as:

$$\epsilon(A, A; k\tau) = \sqrt{k \frac{p(A, A, k\tau) - [p(A, A, k\tau)]^2}{\sum_{i \in A} \sum_{j=1}^{n} c_{ij}(k\tau)}}. \tag{98}$$

This test was carried out on all three counting methods for which the implied timescales were computed (count forward, count reversible, count forward optimal reversible). Unfortunately the Markov test worked surprisingly bad for both reversible counting as well as forward counting with optimal reversible transition matrix estimation. The way this test should be carried out with respect to different counting methods is still subject of current research. However, for the forward counting the Markov test mostly agreed within the statistical errors on the trajectory data as depicted in Fig. 16.

3.2.7 Transition-Path Theory (TPT)

The method of Transition-Path Theory as explained in Sec. 2.3.10 was also applied. This allows us to understand, using what pathways does our peptide transition from its most probable microstate, to the one most similar to the structure found in the complex.

The most probable microstate is the one with the highest corresponding stationary distribution value was found in set 2. This defined set A, and this

state was removed from set 2. Because we wanted to extract 100 random representative structures for each set and the microstate with the smallest backbone RMSD to the complex reference structure was only found in 55 frames, set B was made up of the two closest microstates. These were both found in set 3. Again, these two microstates were removed from set 3. The other PCCA sets remained unchanged. Using these sets we computed the net flux on the transition matrix from the source set A to target set B using Eq. 85. This net flux per microstate was then course grained on the sets. This allowed us to visualize the transition paths and their fluxes as shown in Fig 17. The thickness of the arrows corresponds to the net flux along the pathways.

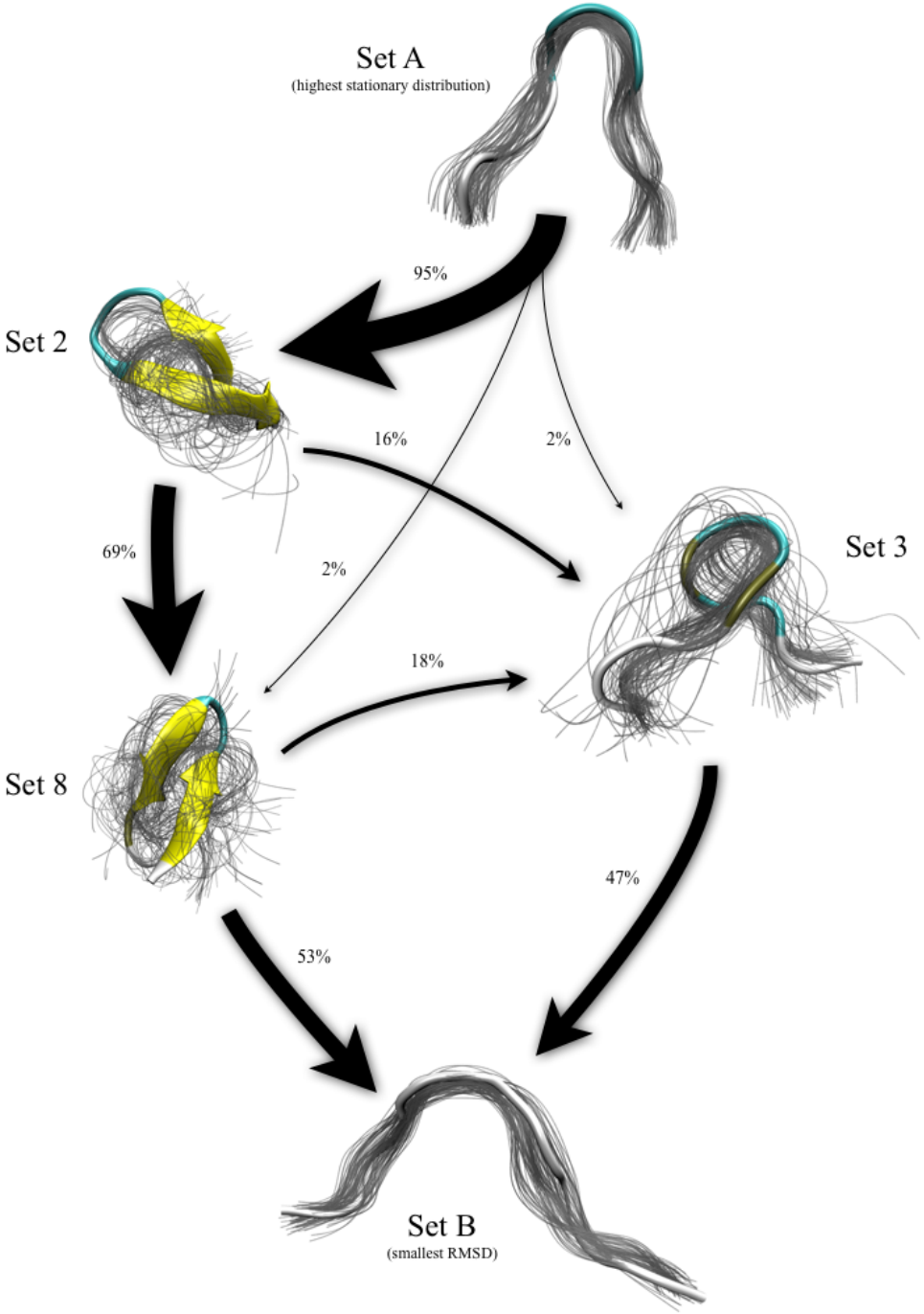


Figure 17: Most important transition pathways for the pB27 peptide going from the most probable microstate (set A) to the one most similar to the complex structure (set B). The thickness of the arrows correspond to the flux along each path.

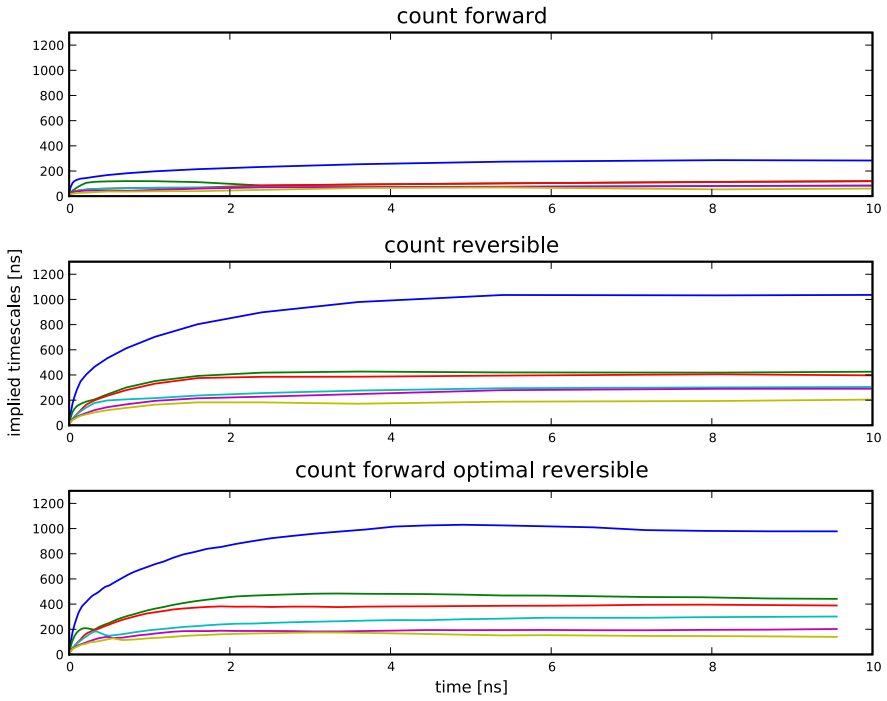


Figure 13: Implied Timescale plots of pB27 in water for three different counting methods. The blue line corresponds to the slowest process found in our system represented by the second eigenvalue. Accordingly the green, red, cyan, purple and beige lines are timescales given by the eigenvalues number 3, 4, 5, 6 and 7. There seems to be a good agreement of the count reversible method with count forward optimal reversible, estimating very similar timescales. The method of counting forward only seems to underestimate the real timescales of our system.

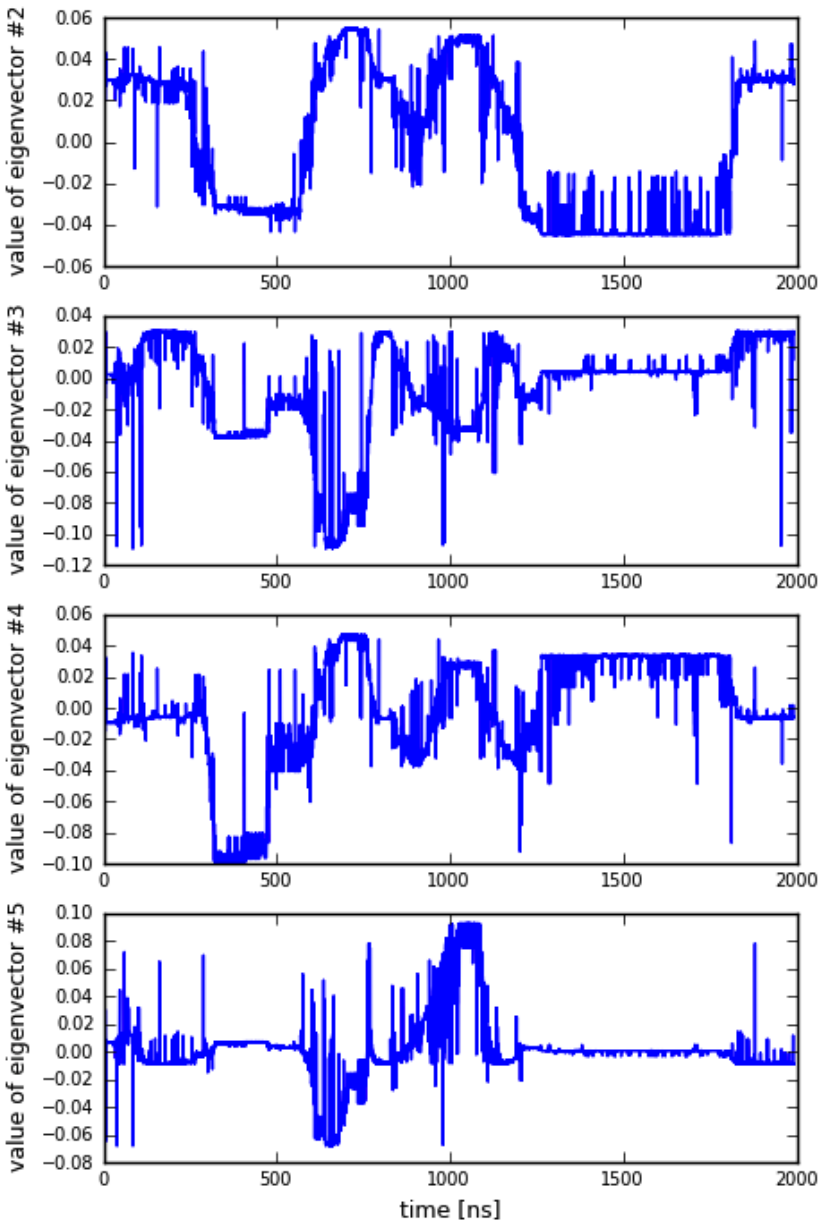
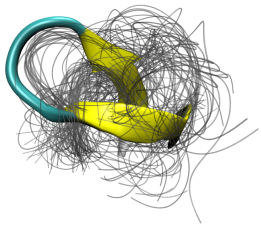


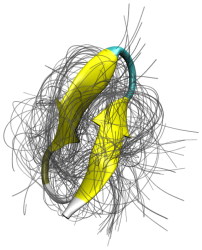
Figure 14: Trajectory of pB27 in water projected on right eigenvectors #2-#5. Switches between metastable states can be observed. The trajectory then relaxes in such a metastable state, which can be seen because the projection takes an almost constant value for several nanoseconds. The thin large peaks are an indication of recrossing, resulting from imperfect separation of states.

Figure 15: Visualization of the 10 PCCA+ sets sorted by their probability. For each set one representative structure was manually selected and is visualized by the secondary structure of its backbone. The grey lines show the expansion by 100 randomly selected backbone structures from each set. Note how the probability is related to the structure of the peptide. The existence of a secondary structure clearly leads to a higher probability for a set. The less compact or folded a the structures in a set are, the less probable they become.

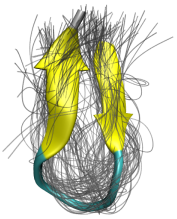
(a) set 2, p = 34,5%



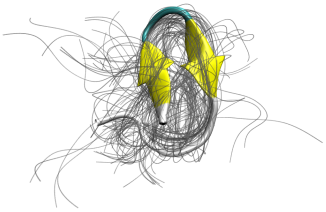
(b) set 8, p = 24,5 %



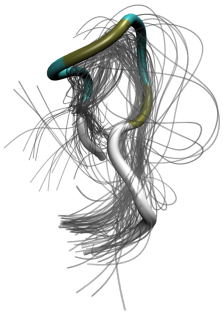
(c) set 6, p = 19,2 %



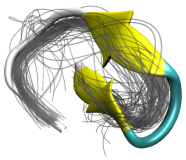
(d) set 4, p = 7,5 %



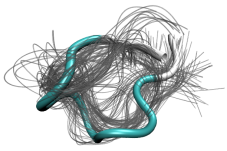
(e) set 3, p = 3,6%



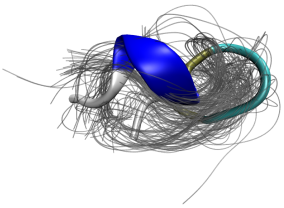
(f) set 9, p = 2.9 %



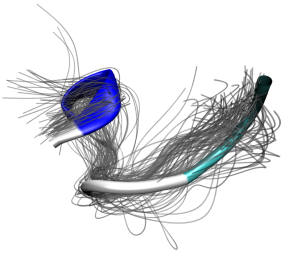
(g) set 5, p = 2,5 %



(h) set 0, p = 2.0 %



(i) set 1, p = 1,9 %



(j) set 7, p = 1,6%

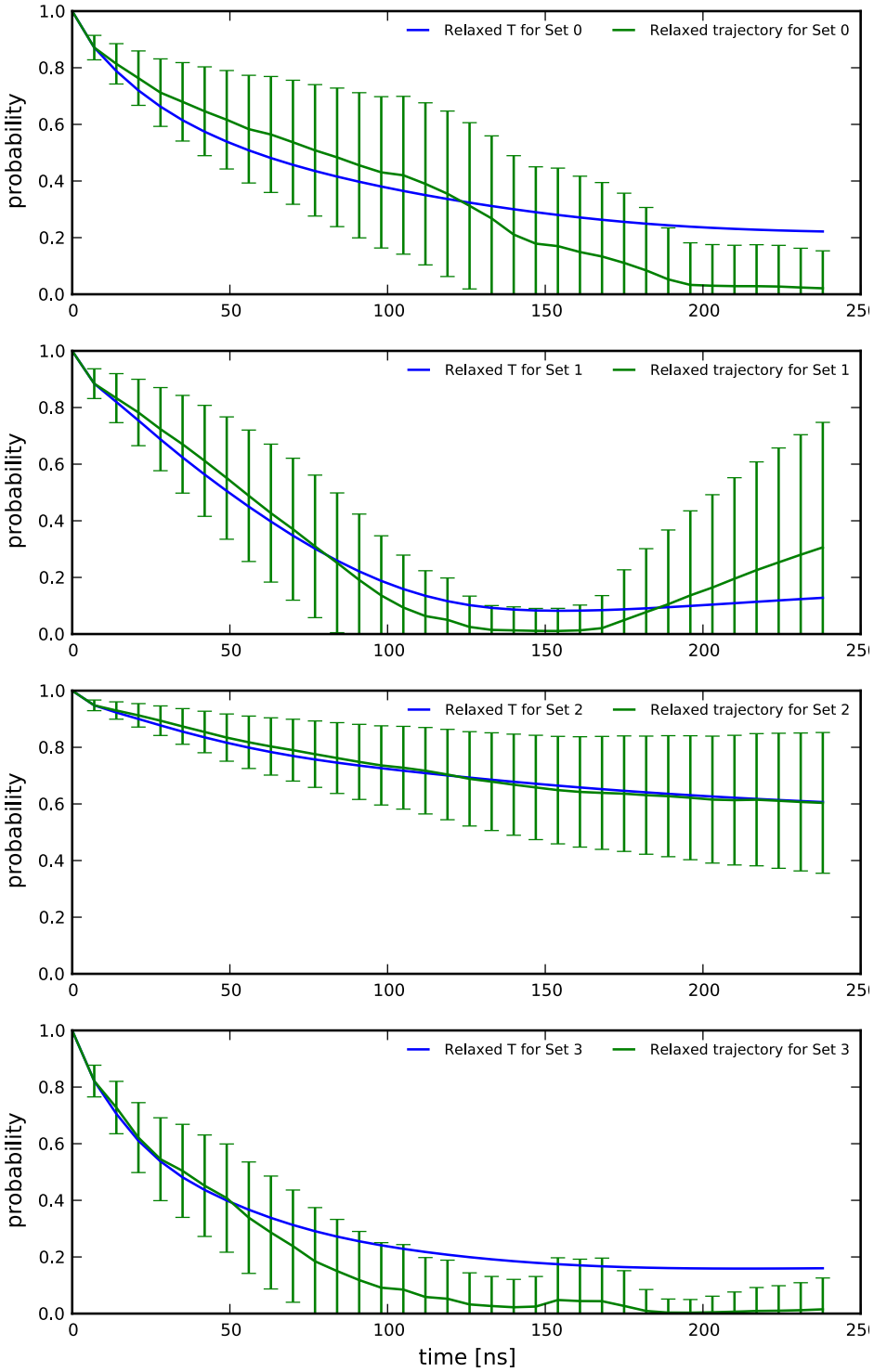


Figure 16: Markov Test for four PCCA+ Sets

3.3 Conformation dynamics of HLA-B*2705:pB27

The following section will focus on the HLA-B*2705:pB27 for which the 100 ns simulations were discussed previously in section 3.1. The aim of this analysis is to be able to generate a MSM of the peptide bound in the complex and compare its conformation dynamics to the case, were the peptide is solvated in water. Because of the much larger system size of around 96,000 atoms in the solvated complex compared to around 6,000 of the peptide in water, the simulation is much more time-consuming. Simulating one nanosecond took about 1,4 and 18 hours for the solvated peptide and the complex respectively. Using the same hardware, the simulation one continuous 2 µs trajectory would therefore take more than four years. It should also be noted that simulation of equilibrium trajectories contains large amounts of data where no new transitions between states occur, therefore consuming large amounts of computing resources where no additional information is gained. It is therefore evident, that a long equilibrium trajectory simulation is unfeasible and another approach has to be taken. It has previously been shown [37, 3, 31] that the long-time dynamics can be reconstructed from running short simulations, that sample different parts of conformational space.

3.3.1 Generation of random starting conformations using WindowMove

While it can sometimes be sufficient to start simulations from the same conformational structure using different initial velocities, simulation time can be further reduced by starting out from distinct conformations. For generation of these conformations, we used a program called WindowMove developed by Frank Noé during his PhD thesis [30].

The program reads in a structure PDB file of heavy atoms and treats them as solid spheres each with their van der Waals radius combined with the bonding information. One then defines a window in which the program will try to reposition the molecule. This is done by first by adjusting the backbone of molecule, using random adjustments of the dihedral angles. The sidechains are placed afterwards. The resulting conformation is only accepted when all

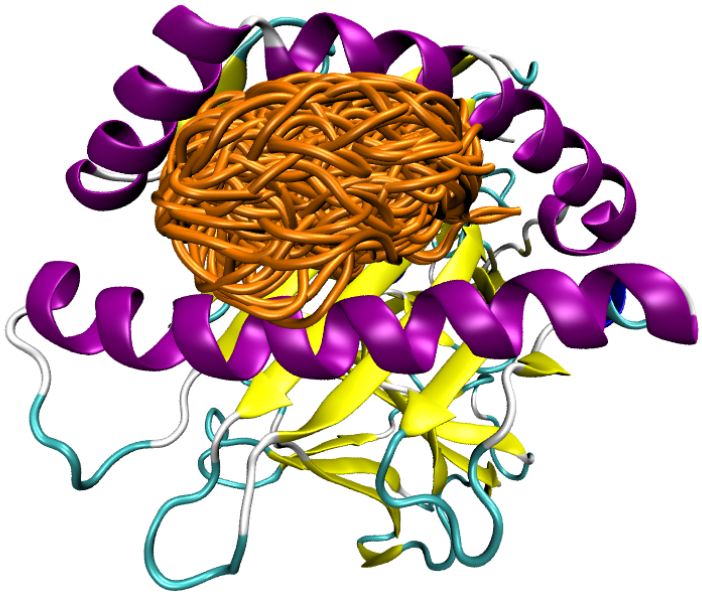


Figure 18: Different pB27 backbone conformations as generated by WindowMove

non-bonded atoms are further apart than the sum of their van der Waals radii. To ensure that no two conformations are too similar, I extended the program to reject all conformations that are not at least 60° apart considering RMSD of dihedral angles from every previously accepted structure. Using this method, 1000 conformations were generated of which 200 representatives are shown in Fig. 18.

3.3.2 Simulation and discretization

Each of these 1000 random conformations was prepared using a steepest decent energy minimization. This was not possible for all conformations, 125 of them turned out to be unstable and crashed during this energy minimization step. This can be due to overlapping atoms in conformations generated by WindowMove that produce clashes. From the remaining 875 starting conformations we started simulations of 1 ns length, after preparing them by adding periodic boundary conditions and solvent as explained in Sec. 3.1. Because these trajectories would only allow to carry out the analysis using

Table 4: simulated trajectories of HLA*B 2705:pB27

trajectory length	number of trajectories
100 ns	2
3 ns	86
1 ns	875

very short lagtimes (50% of the data would be disregarded using a lagtime of 0,5 ns), 86 trajectories of length 3 ns were also simulated. Using about ten nodes of the biocomputing cluster, these short simulations took about three months to complete. The total simulation time for the complex was therefore 1,333 ns of simulations of length as depicted in Table 4.

In order to be able to compare the peptides dynamics when complexed with the HLA to the unbound case described in Sec. 3.2, these trajectories were discretized using regular spatial clustering on heavy-atoms with an minimum RMSD distance of 3 Å as well. As the peptides movement is much more restricted, this resulted in 97 clusters.

3.3.3 Connectivity and the implications of random starting conformations

Because the simulation was started from conformations generated by WindowMove we cannot safely assume that the simulations start out from equilibrium starting points. Therefore only forward counting was used to construct a count matrix. We started to construct this count matrix using a lag time of 1 ps. From there we tested the connectivity of the count matrix. This is done by checking if there are any isolated sets in the count matrix from which no transitions to other states are observed. This would lead to a degenerated transition matrix with isolated substates, leading to infinitely long transition times. It turned out that tree trajectories contained one microstate each that where isolated from the rest of the simulations. Disregarding these trajectories resulted in a fully connected count matrix for a lag time of up to 300 ps.

A transition matrix was then computed from this count matrix. Then we

projected the trajectory on the first four right eigenvectors. When looking at the projection in Fig. 19 and comparing it to the same projection performed for the free pB27 peptide in Fig. 14 one can see that hardy any real state changes can be observed. To the contrary, we can see many small peaks, called recrossings, which result from an imperfect separation of states. This is an indication that the state switches that we observe are artificial and that not enough simulation data is available to perform meaningful analysis.

This assumption was confirmed when looking a the implied timescales in Fig. 20. Note that the implied timescale for eigenvector #2 is not shown in this plot, because its timescale was so large, reaching about 4 µs. Even with the slowest process omitted, which probably comes from an almost degenerated transition matrix, with two sets between which only very few transitions are observed, the implied timescale plots are clearly not converging within the 0,5 ns. This is another strong indication that more and longer trajectories are necessary since the short 1 ns trajectories cannot be analyzed at larger lagtimes.

In conclusion we can say that meaningful analysis cannot be carried out on this system using the trajectories currently available.

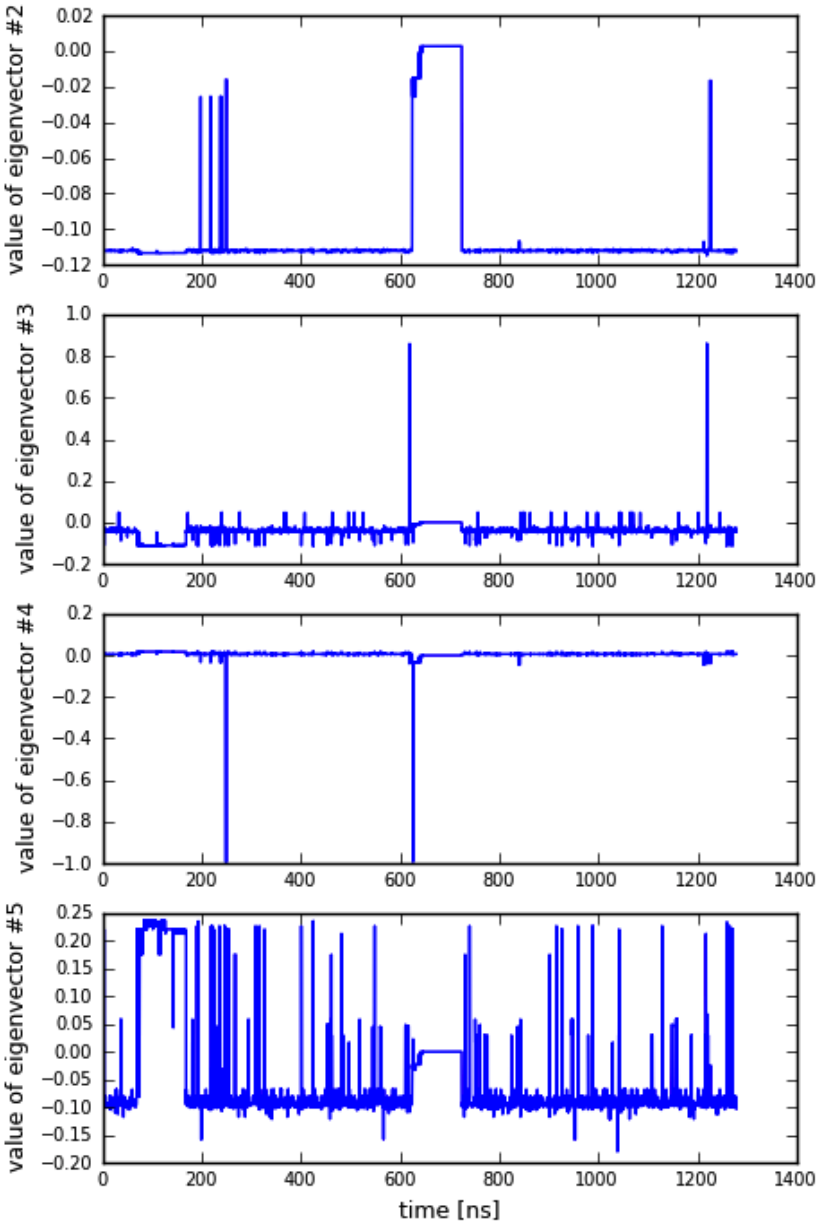


Figure 19: Projection of the concatenated trajectories on the first four non-constant right eigenvectors. The small peaks are a indication of recrossings, which are no real state changes but resulting from imperfect separation of states. The range of 620 to 720 ns corresponds to the 100 ns simulation started from conformation B. Here the projection does change its value for a longer time, but since this is an isolated trajectory, it is not a real state change.

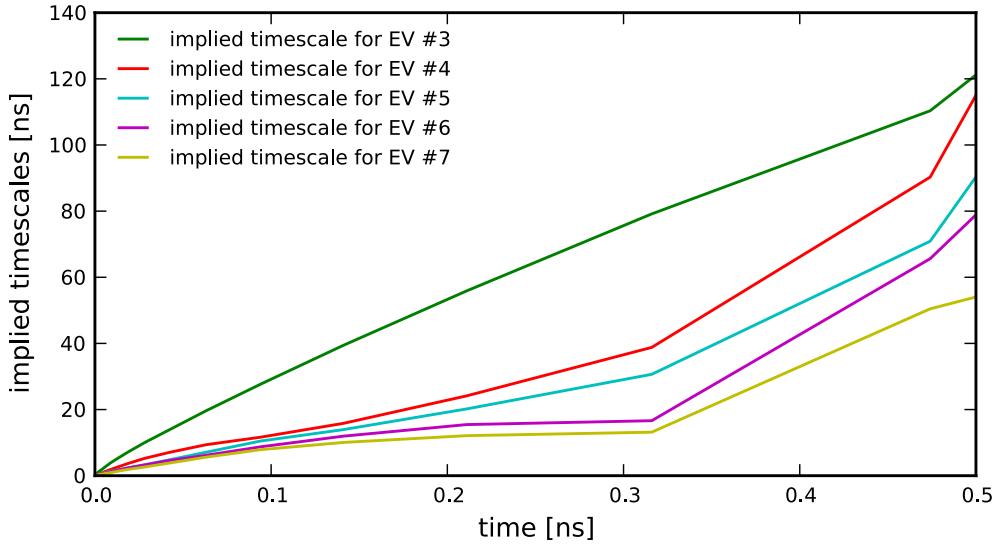


Figure 20: Implied timescales of the right eigenvectors #3 to #7 corresponding to the 2nd to 6th slowest process. The timescale of eigenvector #2 (the slowest process) is omitted here because at reaching about 4 µs it was so large that it would hide all other timescales.

4 Discussion and Outlook

Based on the structural data provided from X-ray crystallography experiments MD simulation was carried out on three HLA-B27 complexes as well as one free peptide. For all three complexes (B*2705:pB27, B*2705:pCP and B*2704:pB27) I was able to provide the crystallographers with representations of important conformations. These structures will hopefully be of assistance in resolving a larger amount of the yet unexplained electron density.

For the pB27 peptide in water a really profound analysis based on Markov state models (MSM) could be conducted. This included the identification of ten metastable states, which could also be visualized. We were also able to derive the stationary distribution and use our metastable sets to compute transition pathways from the most probable microstates to the ones most similar to the structure found in the complex. One thing that remains to be fully understood are the implications of different counting methods and estimation of transition matrices. Right now we cannot be sure as to why the Chapman-Kolmogorov test carried out in order to validate the markovianitiy of our model is only in agreement for the model generated from forward counting. The simulation is currently being extended to 4 µs. This additional data should allow for further refinement of the model, as well as a reduction of the sampling error. In summary we have shown, using this peptide as a test case, that Markov models can greatly assist us in understanding the conformation dynamics of peptides and proteins.

Secondly, we shall look at the analysis of the B*2705:pB27 complex, which was the main system under investigation. The results of these simulations delivered unexpected results. The C-terminus of the pB27 peptide was much more flexible than expected, leading to a very high configurational space. We also identified several important conformations which we gave the crystallographers. The usage of these conformations should support the structure detection of their crystallographic models, thus matching a higher percentage of the electron density. The pathological implications with respect to the autoimmune diseases of this flexibility will also be studied by them, and

hopefully deliver new insights. Despite having simulated over 1,300 ns, the attempt to construct a MSM of the peptides dynamics when bound to the complex was not successful, yet, mainly due to this high flexibility. Several hints lead to the conclusion that we need more and longer simulations in order to generate a valid Markov model of the system. First there is the fact that when taking data from all trajectories into account the resulting Markov model was uncoupled, resulting in sets of isolated microstates from which no transitions to other sets could be observed. This is a clear violation of the ergodic theorem. But even when excluding these isolated trajectories from the analysis, the eigenvector projection as well as the implied timescales further backed up the the need for longer simulations. The eigenvector projection showed no clear division between metastable states and the timescales where not converging using the maximum lagtime of 500 ps for which we could construct them. As we have seen that the regular spatial discretization works good for the peptide, it might impose a problem when looking at the peptide bound by the MHC complex: movement of the peptide within the MHC's binding pocket would not lead to distinct states. This might make it necessary to cluster in such a way that we first align the MHC complexes of the different simulations and use an euclidean distance metric (without further alignment) on the peptide afterwards. Running of simulations orders of magnitudes longer could be achieved in the future using resources of the Folding@home project in the near future [26]. This is a project developed at the Stanford university where simulations are distributed to users over the Internet using their workstations or even PlayStation systems to run thousands of simulations in parallel. Also, recent development to perform MD calculations on graphic cards shows considerable speedup and looks very promising [41]. Furthermore, adaptive sampling techniques can help to further reduce the total simulation time needed. This could include the usage of metadynamics to sample the complete energy landscape of the bound peptide much quicker [39, 40] as well as the start of new simulations guided by the current model. For metadynamics this usually involves a modification of the potential in order to drive the peptide towards conformational changes while new simulations could be started from points where little or no states changes can yet

be observed.

For the other two systems, B*2704:pB27 and B*2705:pCP, one concise conformation was identified for each peptide. This should be of great help for the crystallographic studies, especially for B*2704:pB27, where the peptide could hardly be resolved at all. The sampling technique of WindowMove could also be applied to both these systems and could lead to either the discovery of new metastable states that were not covered by the current simulations or verify that both these peptides really are much more restrained in the MHC's binding groove than in the case of B*2705:pB27. The investigation of these two systems played only a side-role in my thesis however, as the clear focus was to study B*2705:pB27.

When considering the bigger picture of explain autoimmune diseases at a molecular level we must acknowledge that we are currently investigating a fraction of the molecules involved when only simulating the MHC:peptide complex. Computational resources are currently insufficient to simulate the whole immune response including the T cell receptor. Conformation dynamics might play a key role in understanding how these immune reactions really work.

A Gromacs MD simulation

This appendix shall give a detailed overview of how the molecular dynamics simulations were performed using gromacs [17]. Using one system as an example, all important commands executed as well as the used parameter files are listed.

System preparation Using VMD, Glycerol residues originating from crystallography had to be removed. For B*2705:pB27, which contained more than one conformation in one PDB file, only one conformation had to be selected and saved. Because the pB27 peptide in B*2704:pB27 could not be resolved, the MHC complex B*2704:pB27 was aligned with the one from B*2705:pB27 and later saved using the B*2704 MHC molecule together with the pB27 peptide from B*2705.

Structure conversion The PDB files received from crystallography did not contain any hydrogen atoms. Adding hydrogen atoms is done using the gromacs routine pdb2gmx.

```
pdb2gmx -f 2705pB27.pdb -ff oplsaa -water tip4p \
    -o 2705pB27.gro -p 2705pB27.top -i 2705pB27 -ignh
```

Here the usage of the opls all-atom force field is specified along with the the tip4p water model.

Vacuum energy minimization A first energy minimization is the next step. When the unconstrained production simulation run is started, there should be no more large forces on any of the atoms. The programs `grompp` and `mdrun` perform a first vacuum energy minimization.

Listing 1: Gromacs parameter file for vacuum energy minimization

```
; steepest descent minimization of solvated system in 250 steps
integrator      = steep
emtol           = 1.0
nsteps          = 250
nstenergy       = 0
energygrps      = System
```

```
; Parameters describing how to find the neighbors
; of each atom and how to calculate the interactions
ns_type         = simple
coulombtype     = PME
rcoulomb        = 1.0
rvdw            = 1.0
constraints     = none
fourierspacing  = 0.12
pme_order       = 4
ewald_rtol      = 1e-5

; no periodic boundary conditions
pbc             = no
```

The vacuum energy minimization is performed using the steepest decent integrator with a parameter file as given in Listing 1. Here forces that might have been implied through the now removed residues or solutions will be relaxed.

```
grompp -v -f parameters/vacuum.mdp -c 2705pB27.gro \
    -p 2705pB27.top -o 2705pB27-EM-vacuum.tpr
mdrun -v -deffnm 2705pB27-EM-vacuum \
    -c 2705pB27-EM-vacuum.pdb
```

Periodic boundary conditions Periodic boundary conditions are applied with the command `editconf`. Here, a cubic box with a minimum distance of 1.4 nm from the protein is specified.

```
editconf -f 2705pB27-EM-vacuum.pdb \
    -o 2705pB27-PBC.gro -d 1.4
```

Solvent addition The system is then solvated using the command `genbox`.

```
genbox -cp 2705pB27-PBC.gro -cs tip4p.gro \
    -p 2705pB27.top -o 2705pB27-water.pdb
grompp -v -f parameters/solvated.mdp -c 2705pB27-water.pdb \
    -p 2705pB27.top -o 2705pB27-water.tpr
```

Ion addition Because the MHC complex carries a net negative charge and electrostatic forces are long ranged the system has to be neutralized. The

`genion` command can be used to add Na^+ and Cl^- ions in order to ensure the whole system is uncharged.

```
genion -s 2705pB27-water.tpr -o 2705pB27-solvated.pdb \
    -conc 0.15 -neutral -pname NA+ -nname CL-
```

Solvent system energy minimization Once the ions are added, another energy minimization step is performed. This is quite similar to the vacuum energy minimization step except that we are now using the periodic boundary conditions. This means that the parameter file used is almost identical to the one in Listing 1 except the last line specifies pbc = xyz instead of no. The following commands execute this minimization. The switch -np 8 tells gromacs to use 8 CPU cores.

```
grompp -v -f parameters/solvated.mdp -c 2705pB27-solvated.pdb \
    -p 2705pB27-ions.top  -o 2705pB27-EM-solvated.tpr \
    -po 2705pB27-solvated
mpirun -np 8 mdrun -v -deffnm 2705pB27-EM-solvated \
    -c 2705pB27-EM-solvated.pdb -pd
```

Position restrained MD simulation After the solvent addition another energy minimization step has to be performed. This time, a different set of parameters is used.

Listing 2: Gromacs parameter file for position restrained energy minimization

```
; MD integrator with Reaction-Field electrostatics
integrator               = md
dt                       = 0.0001
nsteps                   = 2500
nstenergy                = 0
energygrps               = Protein Non-Protein
coulombtype              = PME
rcoulomb                 = 1.4
epsilon_rf               = 78
vdw-type                 = Cut-off
rvdw                     = 1.4
; Temperature coupling using simple Berendsen thermostat
tcoupl                   = Berendsen
tc-grps                  = Protein  Non-Protein
tau_t                    = 0.1      0.1
```

```
ref_t                    = 310      310
; Treat all bonds as fixed
constraints              = all-bonds
; Generate random velocities for 310K
gen_vel                  = yes
gen_temp                 = 310.0
```

Using the parameter file of Listing 2 we apply constraints to all bonds, fixing the lengths and angles. This allows the solvent molecules to align around the protein. We are also using a thermostat at physiological temperature of 310K.

```
grompp -v -f parameters/position_restrained.mdp \
    -c 2705pB27-EM-solvated.pdb \
    -p 2705pB27-ions.top -o 2705pB27-PR.tpr
mpirun -np 8 mdrun -v -deffnm 2705pB27-PR -pd
```

Production simulation run As our system is now in stable state where no more large forces put strain on our protein, we are now ready to start the production simulation run. Again we are simulating at physiological temperature of 310 K. We are using the stochastic dynamics integrator to act as a Langevin thermostat. We are writing out a complete trr trajectory which allows us to resume the simulation every 50 ps. A compressed xtc trajectory containing only the protein coordinates is written every ps. The SHAKE algorithm is used to allow for a integration step of 2 fs.

Listing 3: Gromacs parameter file for production simulation in NVT ensemble

```
; SD integrator with langevin thermostat at 310 K
integrator               = sd
dt                       = 0.002
nsteps                   = 50000000
bd_fric                  = 0
tcoupl                   = no
tc-grps                  = Protein  Non-Protein
tau_t                    = 2.0      2.0
ref_t                    = 310      310
nstlist                  = 5
ns-type                  = Grid
pbc                      = xyz
rlist                    = 1.0
```

```
; No pressure coupling (NVT ensemble)
pcoupl                   = no
; PME electrostatics
coulombtype              = PME
rcoulomb                 = 1.0
epsilon_rf               = 78
vdw-type                 = Cut-off
rvdw                     = 1.0
; write data to resume simulation every 50 ps
; write compressed trajectory of protein every ps
nstxout                  = 50000
nstvout                  = 50000
nstfout                  = 50000
nstlog                   = 50000
nstenergy                = 50000
xtc_precision            = 1000
nstxtcout                = 500
energygrps               = Protein Non-Protein
xtc_grps                 = Protein
; use SHAKE algorithm for H-bonds
constraints              = hbonds
constraint_algorithm     = SHAKE
; Generate random velocities for 310K
gen_vel                  = yes
gen_temp                 = 310.0
```

Using the parameter file from Listing 3 the production simulation run is the executed using the following command.

```
grompp -v -f parameters/nvt.mdp -c 2705pB27-PR.gro \
    -p 2705pB27-ions.top -o 2705pB27-NVT.tpr
mpirun -np 8 mdrun -v -deffnm 2705pB27-NVT
```

References

[1] Bruce Alberts. *Molecular biology of the cell*. Garland Science, 2002.

[2] GR Bowman, KA Beauchamp, G Boxer, and VS Pande. Progress and challenges in the automated construction of markov state models for full protein systems. *The Journal of Chemical Physics*, 131:124101, 2009.

[3] J Chodera, W Swope, J Pitera, and A Dill. Long-time protein folding dynamics from short-time molecular dynamics simulations. multiscale model. *en.scientificcommons.org*, Jan 2008.

[4] JD Chodera, N Singhal, VS Pande, KA Dill, and WC Swope. Automatic discovery of metastable states for the construction of markov models of macromolecular conformational dynamics. *The Journal of Chemical Physics*, 126:155101, 2007.

[5] Ana Cipriani, Sergio Rivera, Manzur Hassanhi, Georgina Márquez, Rosaura Hernández, Cecilia Villalobos, and Milagros Montiel. Hla-b27 subtypes determination in patients with ankylosing spondylitis from zulia, venezuela. *Hum Immunol*, 64(7):745–9, Jul 2003.

[6] T Darden, D York, and L Pedersen. Particle mesh ewald: An n log (n) method for ewald sums in large systems. *The Journal of Chemical Physics*, 98:10089, 1993.

[7] Ruslan L Davidchack, Richard Handel, and M. V Tretyakov. Langevin thermostat for rigid body dynamics. *J Chem Phys*, 130(23):234101, Jan 2009.

[8] P Deuflhard and M Weber. Robust perron cluster analysis in conformation dynamics. *Linear Algebra and its Applications*, Jan 2005.

[9] Guy G Dodson, David P Lane, and Chandra S Verma. Molecular simulations of protein dynamics: new windows on mechanisms in biology. *EMBO Rep*, 9(2):144–50, Feb 2008.

[10] PP Ewald. Die berechnung optischer und elektrostatischer gitterpotentiale. *Annalen der Physik*, 369(3):253–287, 1921.

[11] Heinz Fabian, Hans Huser, Bernhard Loll, Andreas Ziegler, Dieter Naumann, and Barbara Uchanska-Ziegler. Hla-b27 heavy chains distinguished by a micropolymorphism exhibit differential flexibility. *Arthritis Rheum*, 62(4):978–87, Apr 2010.

[12] Heinz Fabian, Hans Huser, Daniele Narzi, Rolf Misselwitz, Bernhard Loll, Andreas Ziegler, Rainer A Böckmann, Barbara Uchanska-Ziegler, and Dieter Naumann. Hla-b27 subtypes differentially associated with disease exhibit conformational differences in solution. *J Mol Biol*, 376(3):798–810, Feb 2008.

[13] Martin Fischbach and Frank Noé. Emma 1.0 markov model algorithms tutorial and documentation. pages 1–19, May 2010.

[14] Peter L Freddolino, Christopher B Harrison, Yanxin Liu, and Klaus Schulten. Challenges in protein-folding simulations. *Nature Physics*, 6:751, Oct 2010. (c) 2010: Nature.

[15] Daan Frenkel and B. Smit. *Understanding Molecular Simulation, Second Edition: From Algorithms to Applications (Computational Science Series, Vol 1)*. Academic Press, 2 edition, November 2001.

[16] Crispin Gardiner. *Handbook of Stochastic Methods: for Physics, Chemistry and the Natural Sciences (Springer Series in Synergetics)*. Springer, 3rd edition, April 2004.

[17] Berk Hess, Carsten Kutzner, David van der Spoel, and Erik Lindahl. Gromacs 4: Algorithms for highly efficient, load-balanced, and scalable molecular simulation. *J Chem Theory Comput*, 4(3):435–447, Jan 2008.

[18] Martin Hülsmeyer, Karin Welfle, Thomas Pöhlmann, Rolf Misselwitz, Ulrike Alexiev, Heinz Welfle, Wolfram Saenger, Barbara Uchanska-Ziegler, and Andreas Ziegler. Thermodynamic and structural equiva-

lence of two hla-b27 subtypes complexed with a self-peptide. *J Mol Biol*, 346(5):1367–79, Mar 2005.

[19] PH Hünenberger. Thermostat algorithms for molecular dynamics simulations. *Advanced Computer Simulation*, pages 105–149, 2005.

[20] W Jorgensen, D Maxwell, and J Tirado-Rives. Development and testing of the opls all-atom force field on conformational energetics and properties of organic liquids. *J Am Chem Soc*, Jan 1996.

[21] WL Jorgensen, J Chandrasekhar, JD Madura, RW Impey, and ML Klein. Comparison of simple potential functions for simulating liquid water. *The Journal of Chemical Physics*, 79:926, 1983.

[22] GA Kaminski, RA Friesner, J Tirado-Rives, and WL Jorgensen. Evaluation and reparametrization of the opls-aa force field for proteins via comparison with accurate quantum chemical calculations on peptides, Jan 2001.

[23] Martin Karplus and J Andrew McCammon. Molecular dynamics simulations of biomolecules. *Nat Struct Biol*, 9(9):646–52, Sep 2002.

[24] JC Kendrew, G BODO, HM Dintzis, RG Parrish, H Whyckoff, and DC Phillips. A three-dimensional model of the myoglobin molecule obtained by x-ray analysis. *Nature*, 181(4610):662–6, Mar 1958.

[25] Muhammad Asim Khan, Alessandro Mathieu, Rosa Sorrentino, and Nurullah Akkoc. The pathogenetic role of hla-b27 and its subtypes. *Autoimmun Rev*, 6(3):183–9, Jan 2007.

[26] Stefan M Larson, Christopher D Snow, Michael Shirts, and Vijay S Pande. Folding@home and genome@home: Using distributed computing to tackle previously intractable problems in computational biology. *eprint arXiv*, 0901:866, Jan 2009.

[27] B Loll, R Misselwitz, and B Uchanska-Ziegler. Implications of structural and thermodynamic studies of hla-b27 subtypes exhibiting differential

association with ankylosing spondylitis. *Molecular Mechanisms of ...*, Jan 2009.

[28] Bernhard Loll, Anna Zawacka, Jacek Biesiadka, Cordula Petter, Christine Rückert, Wolfram Saenger, Barbara Uchanska-Ziegler, and Andreas Ziegler. Preliminary x-ray diffraction analysis of crystals from the recombinantly expressed human major histocompatibility antigen hla-b*2704 in complex with a viral peptide and with a self-peptide. *Acta Crystallogr Sect F Struct Biol Cryst Commun*, 61(Pt 10):939–41, Oct 2005.

[29] Daniele Narzi, Kathrin Winkler, Jürgen Saidowsky, Rolf Misselwitz, Andreas Ziegler, Rainer A Böckmann, and Ulrike Alexiev. Molecular determinants of major histocompatibility complex class i complex stability: shaping antigenic features through short and long range electrostatic interactions. *J Biol Chem*, 283(34):23093–103, Aug 2008.

[30] F Noé. Transition networks: Computational methods for the comprehensive analysis of complex rearrangements in proteins. *ub.uni-heidelberg.de*, Jan 2006.

[31] Frank Noé, Christof Schütte, Eric Vanden-Eijnden, Lothar Reich, and Thomas R Weikl. Constructing the equilibrium ensemble of folding pathways from short off-equilibrium simulations. *Proc Natl Acad Sci USA*, 106(45):19011–6, Nov 2009.

[32] V Pande, K Beauchamp, and G Bowman. Everything you wanted to know about markov state models but were afraid to ask. *Methods*, Jan 2010.

[33] Thomas Pöhlmann, Rainer A Böckmann, Helmut Grubmüller, Barbara Uchanska-Ziegler, Andreas Ziegler, and Ulrike Alexiev. Differential peptide dynamics is linked to major histocompatibility complex polymorphism. *J Biol Chem*, 279(27):28197–201, Jul 2004.

[34] Jan-Hendrik Prinz, Frank Noe, Martin Fischbach, Martin Held, and Christof Schütte. Markov models of molecular kinetics: Generation and validation. pages 1–39, Jun 2010.

[35] Christine Rückert, Maria Teresa Fiorillo, Bernhard Loll, Roberto Moretti, Jacek Biesiadka, Wolfram Saenger, Andreas Ziegler, Rosa Sorrentino, and Barbara Uchanska-Ziegler. Conformational dimorphism of self-peptides and molecular mimicry in a disease-associated hla-b27 subtype. *J Biol Chem*, 281(4):2306–16, Jan 2006.

[36] Tamar Schlick. *Molecular Modeling and Simulation*. Springer, 1 edition, August 2002.

[37] C Schutte, A Fischer, W Huisinga, and P Deuflhard. A direct approach to conformational dynamics based on hybrid monte carlo. *J Comput Phys*, 151(1):146–168, Jan 1999.

[38] David E Shaw, Paul Maragakis, Kresten Lindorff-Larsen, Stefano Piana, Ron O Dror, Michael P Eastwood, Joseph A Bank, John M Jumper, John K Salmon, Yibing Shan, and Willy Wriggers. Atomic-level characterization of the structural dynamics of proteins. *Science*, 330(6002):341–6, Oct 2010.

[39] V Spiwok, B Králová, and I Tvaroška. Continuous metadynamics in essential coordinates as a tool for free energy .... *Journal of Molecular Modeling*, Jan 2008.

[40] V Spiwok and I Tvaroška. Metadynamics modelling of the solvent effect on primary hydroxyl rotamer .... *Carbohydrate Research*, Jan 2009.

[41] John E Stone, David J Hardy, Ivan S Ufimtsev, and Klaus Schulten. Gpu-accelerated molecular modeling coming of age. *Journal of Molecular Graphics and Modelling*, 29(2):116–125, Sep 2010.

[42] N. G. Van Kampen. *Stochastic Processes in Physics and Chemistry, Third Edition (North-Holland Personal Library)*. North Holland, 3 edition, May 2007.

[43] A Ziegler, CA Müller, RA Böckmann, and B Uchanska-Ziegler. Low-affinity peptides and t-cell selection. *Trends in immunology*, 30(2):53–60, 2009.

[44] Andreas Ziegler, Bernhard Loll, Jacek Biesiadka, Wolfram Saenger, Thomas Kellermann, Rolf Misselwitz, and Barbara Uchanska-Ziegler. A cartilage-derived self peptide presented by hla-b27 molecules? comment on the article by atagunduz et al. *Arthritis Rheum*, 52(8):2581–2; author reply 2582–3, Aug 2005.

Acknowledgments

I want to thank Dr. Frank Noé and Dr. Holger Dau for supervising my thesis and allowing me to work on this very interesting subject. I am also obliged to Martin Held for answering numerous questions and offering many helpful suggestions.

Erklärung gemäß § 17, Abs. 7 DPO

Ich versichere hiermit, diese Arbeit selbständig verfasst und keine anderen als die angegeben Hilfsmittel benutzt zu haben.

Berlin, den 17.11.2010