Collective Langevin Dynamics of Conformational Motions in Proteins

Dissertation

Zur Erlangung des Doktorgrades

der Mathematisch-Naturwissenschaftlichen Fakultäten

der Georg-August-Universität zu Göttingen

Vorgelegt von

Oliver Lange

aus Kassel

Göttingen, 2005

D7
Referent: Prof. Dr. A. Zippelius
Korreferent: PD Dr. H. Grubmüller
Tag der mündlichen Prüfung: 1.12.2005

Contents

The important thing in science is not so much to obtain new facts
as to discover new ways of thinking about them
— Sir William Bragg

Chapter 1

Introduction

Proteins are biological macromolecules, which are mainly composed from polymeric chains of amino acids[1]. They are involved in a diversity of processes in living organisms. Although some play a mere structural role (e.g., collagen in tissues, or α-keratin in hair), the function of most others depends crucially on their dynamics. While for the many examples of motor proteins (e.g., kinesin and F1-ATPase) the connection to dynamics is obvious, the dynamics also plays an important role if primary function is not mobility itself. For example, the ability to change conformation is essential for the function of proteins involved in signal transduction or transport, for molecular recognition, e.g., in the immune system, and for the function of numerous enzymes[1]. In many enzymes, for instance, conformational changes serve to enclose the substrate, thereby preventing its release from the protein and optimally positioning it for the protein to perform its function, as in lysozyme.

To understand the mechanisms of protein function is an intriguing and formidable task. Although remarkable progress has been made in past decades, and despite the number and quality of available methods has been tremendously increased, most mechanisms are not understood on a physical basis, which would require models based on first principles allowing for a quantitative comparison with experimental results.

Experimental techniques made remarkable progress to unravel protein structures (e.g., Xray crystallography[2, 3] and nuclear magnetic resonance spectroscopy (NMR)[4, 5]) and, furthermore, even allow to probe dynamics (NMR relaxation[6], electron paramagnetic resonance (EPR)[7], neutron scattering[8, 9], as well as fluorescence spectroscopy[10]). In some instances different functional states of proteins were structurally characterized by trapping them in certain substates[11]. Furthermore, time-resolved Xray diffraction[12, 13] allows to follow the conformational protein motion with picoseconds time resolution. Wide-spread use of the latter two techniques is impeded, though, by the massive experimental effort involved.

In comparison to this tremendous experimental progress and the enormous variety of available techniques the theoretical treatment of protein dynamics strikes as underdeveloped. Only computer simulation techniques, and especially molecular dynamics (MD) simulations at atomic resolution, have been applied with noteworthy success to elucidate functional processes in recent years[14]. Therefore, advancements of theoretical methods and concepts are urgently required.

As will be described in Chapter 2, classical MD is an atomistic simulation method, which treats each atom as point mass and describes the interaction between atoms with simple force terms. Trajectories are generated by integrating Newton's equations of motions. It operates in the full $3N$ dimensional configurational space of the protein and the surrounding solvent molecules (where N is the number of atoms). The large number of pair-wise interactions to be evaluated and the short time steps enforced by the fastest motions entail very long computation times, which limits MD at present to systems of $10^5 - 10^6$ atoms and to timescales of several 100ns. Unfortunately, apart from a few exceptions, relevant biological processes, such as the gating of ion channels, allosteric interactions, ligand binding, molecular recognition, chemo-mechanical energy conversion and many more, occur on microsecond to seconds time scales, and thus are currently far out of reach for conventional MD.

This holds true despite considerable efforts to speed up the computations, particularly of the long-range Coulomb forces. Recent developments include efficient methods such as multiple step algorithms[15, 16, 17, 18, 19, 20], fast multipole methods[21, 22, 23, 24], and Ewald summation techniques[25]. Also, the use of constraints[26, 27, 28] helps to increase efficiency. Still, however, processes on time scales of microseconds and beyond can only be studied by resorting to certain 'tricks' to enhance sampling by speeding up conformational motions, as reviewed in Ref. [29]. Unfortunately, this kind of accelerated sampling necessarily implies loss of dynamical information and often loss of thermodynamical accuracy as well[29].

Statistical mechanics is the appropriate theoretical framework to understand the dynamics of many-particle systems such as proteins. One considers a macroscopic state as statistical ensemble of a large number of replica of a microscopic system, which evolve independently from each other. Macroscopic observations of relevant degrees of freedom are obtained by averaging the remaining ones over the statistical ensemble. The applied averaging yields energetics, which are influenced by entropic contributions, and hence free energies need to be considered. Seen from this perspective, protein function and the corresponding highly controlled conformational motions are driven by free energy differences between different substates of the solvated protein.

MD simulation, however, is not intrinsically a statistical mechanics approach, since it describes protein dynamics from a microscopic point of view. Rather, it is used as a 'brute-force' method to generate statistical ensembles. Although a statistical mechanics treatment can be attached to the MD results, this modus operandi impedes profiting from the elegance of this framework. It remains thus challenging to 'go the whole way' and consistently treat relevant degrees of freedom of protein dynamics with statistical mechanics.

In this thesis we advance the methodology beyond conventional 'brute force' MD by applying statistical mechanics to gain a drastic *reduction of the large number of degrees of freedom.* This implies two steps. First, to identify few appropriate slow and relevant degrees of freedom[30], which serve to define a reduced active space within which the dynamics is evolved without explicit treatment of the remaining orthogonal fast degrees of freedom. Second, to derive suitable equations of motion for these slow degrees of freedom.

As an illustration of our approach consider the well-known dimension reduced treatment of the motion of a Brownian particle, which is also based on a separation of slow and fast degrees of freedom. A Brownian particle is a large solute particle immersed in a fluid of much smaller particles, e.g., water. Its macroscopic erratic movement is the combined result of a large number of collisions with fluid particles. Because the motion of the macroscopic particle is much slower than that of the fluid particles, one can consider the slow and fast motions as uncorrelated. This justifies to treat the solvent coordinates as irrelevant and thus to replace their influence on the slow degree of freedom by a random force, which is memory free due to the separation of timescales. In contrast to our work described below, the trajectory of the Brownian particle is a random walk and can thus be described by a Markov process, i.e., its future evolution does not depend on its past, because (a) the random force is memory free and (b) the motion is overdamped.

To apply this concept — replacing fast degrees of freedom by a random force — to our case we have to be aware of the differences though. First, it is not at all clear how to select the slow degrees of freedom for the internal motion of a protein. All involved particles, regardless if constituting solvent or protein, are atoms of similar mass, which move with comparable speeds. Second, there will be no clear separation between fast and slow degrees of freedom due to the continuous spectrum of time scales covered by protein dynamics. An important consequence is that the random force contains memory effects. A third difference is that the motion is not overdamped, such that inertia effects matter. Thus, our treatment will have to account for this non-Markovian character of the slow dynamics.

The absence of canonical slow degrees of freedom has led to a diversity of phenomenologically motivated selections of the active space. These include implicit solvent[31], combined atom or bead models[32, 33, 34, 35], and the treatment of polypeptides as chains of stiff 'platelets', for which only ψ-φ backbone angles are retained as explicit degrees of freedom[36, 37]. A somewhat related approach is the gaussian network model[38].

However, by restricting the model to certain atoms or groups of atoms and omitting others, only a very small subset of all possible collective degrees of freedom is considered. One may, therefore, expect to derive improved dimension-reduced descriptions of protein dynamics by dropping this empirical restriction and considering as degrees of freedom m fully general functions $c_i = f_i(\mathbf{x}_1, \ldots, \mathbf{x}_N), \quad i = 1 \ldots m$, of the atomic positions $\mathbf{x}_j$. Linear f_i are widely considered, e.g., within the framework of principal components analysis (PCA)[39], which is often used to systematically derive slow and relevant (essential) collective degrees of freedom from MD simulations or structural ensembles[40]. Here we consider both, linear and non-linear collective degrees of freedom.

The general framework that allows to reduce the full dynamics of all atomic degrees of freedom to dynamics of the selected (collective) degrees of freedom is provided by the projection-operator formalism of Zwanzig and Mori[41, 42]. The resulting generalized Langevin equation (GLE)[43, 44, 45, 46, 47, 48] governs non-Markovian dynamics due to its generalized dissipative term, which is a convolution of the *memory kernel* with past velocities. We will show how the GLE is derived from Newton's equation of all degrees of freedom by separating the overall motion with the projection-operator into (a) an ensemble-averaged motion on the free energy surface, governed by the *potential of mean force*, and (b) a deviation from these average dynamics.

Combining these two concepts, generalized collective degrees of freedom and dimension reduced dynamics, we here develop the framework of Collective Langevin Dynamics (CLD), which describes protein dynamics in collective coordinates. The projection-operator formalism is used to derive the necessary parameters for the GLE, i.e., an appropriate potential of mean force and memory kernels from short MD simulations. Thereby, all parameters are systematically obtained from first principles, which allows to automate parameter extraction. By construction, there are no general parameters which hold for all proteins, but parameters need to be specifically extracted for the chosen molecular system and the selected set of collective coordinates. The low number of degrees of freedom will allow a computationally efficient generation of trajectories, thereby rendering microseconds timescales accessible.

The main tasks which need to be addressed in this thesis are (1) identification of suitable conformational coordinates, (2) extraction of memory kernels and (3) construction of a suitable free energy landscape from MD simulations, and (4) evaluation of CLD accuracy and performance.

Note that it is a huge task to develop CLD to full maturity, such that we here only attempt the first steps, which we outline below.

(1) Extraction of relevant degrees of freedom

Selection of suitable collective degrees of freedom crucially affects the strength and persistence of memory effects as well as the resolution of conformational states. Thereby, this choice determines the significance of the resulting CLD model for functionally relevant dynamics. Thus, we aim for collective modes which are as slow as possible. Moreover, the active subspace is ideally uncorrelated to those remaining fast degrees of freedom, which are not treated explicitly.

A well established method to identify functional relevant modes in MD trajectories is principal component analysis (PCA)[49, 39, 50, 51, 40]. Therefore, it is a natural choice to consider PCA as a candidate here. It selects those collective degrees of freedom which contribute most to the atomic motion seen in the trajectory by diagonalizing the covariance matrix of atomic displacements.

Whether and to what extent a separation of timescales can be achieved by application of PCA has not yet been systematically assessed. Furthermore, it is not clear, and subject to ongoing discussions[52], whether principal components extracted from short MD simulations can serve to describe protein dynamics at long time scales sufficiently well. In Chapter 2, we will shortly review the theory of PCA and address these questions.

Unfortunately, PCA does not yield fully uncorrelated collective modes[50], because the covariance matrix detects only *linear* correlations. Although the remaining *non-linear* and *multi-coordinate* correlations do not impede using principal components within CLD we might be able to advance the method by extracting coordinates that are correlated to a lesser extent.

Therefore, we will introduce in Chapter 3 the (Shannon) mutual information[53], which detects *any* correlation. Based on this measure we will develop in Chapter 4 Full Correlation Analysis (FCA), which extracts maximally uncoupled coordinates by minimizing the mutual information of the configurational ensemble.

That the extraction of collective coordinates relies heavily on correlation triggered the wish to find an experimental access to this observable. This would allow a direct verification of collective modes, and possibly an experimental access to collective modes without the requirement of an MD simulation.

Experiments probe correlations in the motion of *atoms* in three dimensional space[54, 55, 56], in contrast to the previously considered correlations between *coordinates*. In Chapter 3 we will show that the established method[57, 58] to quantify such correlations suffers from considerable inconsistencies, and thus misses over 50% of correlations. Since this impedes any meaningful comparison of this observable with experiments, we propose to apply mutual information also to this problem and define a generalized correlation coefficient. In this way we avoid not only the inconsistencies of the previous measure but also detect *non-linear* correlations.

Having then established a solid grasp of correlations on the simulation side, we will compare these with experimental data. A recently reported NMR relaxation experiment promised to probe correlated motions in proteins[59]. Whether the results were really related to correlated motion, however, could not be tested by experiment alone. Therefore, we address this issue by means of MD simulations in Chapter 5.

(2) Extraction of Memory Functions

Extraction of memory kernels from MD simulations is still a challenging problem. Despite considerable efforts, a generally accepted approach has not yet emerged[60, 61, 62, 63]. Thus, we will study different memory extraction schemes, and evaluate their performance within the framework of CLD in Chapter 7.

To our knowledge, all existing algorithms are based on either the Memory equation[62, 64, 65, 66, 67, 68, 61, 69], or on a direct relation of the memory kernel with force autocorrelation functions[63, 70]. We assess both approaches, which have different merits and flaws in the context of CLD. Because exploiting the Memory equation requires solution of an inverse problem, we need to study regularization techniques for its stable and robust numerical solution.

(3) Free energy surface

Free energy surfaces of the conformational coordinates can be estimated by molecular dynamics sampling[71, 72]. More efficient, however, are enhanced sampling techniques[29], for instance, multicanonical methods (e.g., replica exchange MD (REMD)[73]), smart Monte Carlo (SMC)[74], or umbrella sampling[75]. These techniques are complementary to CLD, because they yield canonical ensembles, but do not yield dynamical information. CLD, on the other hand, yields proper dynamical information, but relies on already known canonical ensembles.

Due to the abundance of available techniques it was not necessary to treat this topic in detail.

(4) Evaluation of CLD models

Assessment of the quality of the obtained dimension-reduced description is non-trivial in itself. Clearly, direct comparison of the observed CLD trajectory with explicit (deterministic) MD simulations is not meaningful, because the underlying GLE governs a stochastic process and because the dynamics is chaotic. Rather, suitable observables such as averages over many realizations of the stochastic process, or time averages such as time correlation functions, transport coefficients, or transition rates should be used[76].

However, one has to take care not to check observables which were used to parameterize the CLD model, rendering the selection of test-observables a delicate choice. Velocity autocorrelation functions, for example, are used to extract memory kernels from MD simulation and, thus, do not represent a rigorous test of CLD. In Chapter 8, we use conformational transition rates as observables, which are fully unrelated to the input - yet statistically meaningful. They are compared to reference rates obtained from a long explicit MD simulation. Additionally, we compare positional autocorrelation functions as a probe of long time correlations, because these are not resolved by the velocity autocorrelation functions used as input.

I do not fear computers. I fear the lack of them.
— Isaac Asimov

Chapter 2

Principal Component Analysis

Principal component analysis (PCA) is a well-established technique for reducing dimensionality. Its applications include data compression, image processing, data visualization, exploratory data analysis, pattern recognition and time series prediction[77]. In this chapter we elucidate whether PCA can be applied to extract from short MD simulations slow slow collective degrees of freedom to treat protein dynamics within the proposed framework of collective Langevin dynamics (CLD).

For analysis of protein dynamics principal component analysis (PCA)[49, 39, 50] is an established method based on the notion that the biggest positional fluctuations occur along collective degrees of freedom. This was first realized by normal mode analyses of small proteins[78, 79, 80]. In normal mode analysis, the potential energy is approximated harmonically and the collective modes are obtained by diagonalizing the Hessian matrix in a local energy minimum. PCA, and the related quasi-harmonic analysis[81, 82, 83, 84] and singular value decomposition[85, 86], have shown that even beyond the harmonic approximation protein dynamics are dominated by few collective modes. In particular, these methods showed that it was generally possible to describe about 90% of the total atomic displacement of a protein with only 5-10% of the collective degrees of freedom[50, 87]. This has led to the concept of the *essential* subspace, which is spanned by a small number of the PCA modes with the highest fluctuational amplitudes. It could be shown that in this way PCA separates protein dynamics in two kinds of modes. The fluctuational distribution of the non-essential (small amplitude) modes is well approximated by a Gaussian. Thus, these modes are called *quasi-harmonic*, and are considered to constitute near constraint degrees of freedoms[88, 89]. For the large amplitude (essential) modes, on the contrary, this approximation is inaccurate, such that they are called *anharmonic*[88, 89]. It was argued that only the latter describe functional relevant motion, since the anharmonicity results from rare transitions between multiple minima, while the motion within the minima is rather quasi-harmonic[88, 89, 90].

As a consequence, the dynamics in the essential subspace, denoted as essential dynamics, are often in the primary focus of computational studies[91, 92, 93, 94, 95], enhanced sampling techniques[96, 97, 98] or simple models of protein dynamics[90, 99, 100, 101]. To investigate whether the essential PCA modes are suitable to serve as conformational coordinates within the CLD framework, we need to establish that (a) the timescales between essential and non-essential PCA modes must be partially separated, and (b) the essential modes must describe also the long time dynamics sufficiently well.

Essential PCA modes indeed describe slow motion, because via the equipartition theorem the large amplitude modes are connected with a slow effective frequency $\omega_i^{\text{eff}} = \sqrt{\frac{kT}{\langle c_i^2 \rangle}}$[87]. However, the timescale separation needs to be investigated more systematically, because a slow effective frequency does not rule out minor but significant fast contributions to the dynamics of an essential mode. Therefore, we are going to analyze in Sec. 2.4 the power spectra of all principal modes, in order to see whether and under which conditions PCA is able to filter out purely slow motions.

Furthermore, we need to establish that the essential subspace obtained from a short MD simulation describes a considerable and sufficient amount of the overall protein motion observed on long time scales. This question about the convergence of principal modes has led to considerable dispute. Amadei et al. advocated a fast convergence[102], whereas Balsera et al. strongly questioned the suitability of principal modes to describe protein dynamics on long time-scales[52]. This controversy stems from a different perception of convergence of principal components. Amadei et al. found in 2 ns simulations evidence on a remarkably stability of the directions of single eigenvectors[102]. Balsera's rejection of principal modes, however, was mainly based on the slow convergence of the fluctuational amplitudes[52]. Because these amplitudes are not important for the use of the PCA modes within the CLD framework, the findings of Amadei et al. are more relevant to us.

Nevertheless, both antagonistic studies are based on short simulations — due to the limited computer power at their time — rendering the judgment of the suitability of principal modes for description of protein motion on long time scales a precarious extrapolation. Therefore, we resolve this question by analyzing in Sec. 2.5 how well principal components computed from short MD simulations can describe the dynamics observed in a much longer (i.e., 450ns) MD simulation of crambin. Besides, we depart from Amadei's work not only by means of much longer simulation time, but also by adopting a new measure of stability (cf. Sec. 2.3.3) that is particular suited to answer our question.

In the subsequent section we introduce PCA as maximization of fluctuational amplitude and report its basic properties. Since the following investigations are based on extended MD simulation, we sketch its principles in Sec. 2.2 and use the opportunity to introduce in Sec. 2.3.1 all simulation systems used within this work.

2.1 Theory of principal component analysis

We shortly review the most common derivation of PCA to illustrates its basic properties. PCA is applied to ensembles of protein structures $\left\{\mathbf{r}^{(k)}\right\}_{k=1\ldots M}$, where $\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)^{\mathrm{T}}$ denotes the positions of its N atoms in three dimensional space and angular brackets denote the ensemble average $\langle f(\mathbf{r}) \rangle = M^{-1} \sum_{k=1}^{M} f\left(\mathbf{r}^{(k)}\right)$. PCA aims at finding linear orthogonal projections $c_i = \mathbf{a}_i^{\mathrm{T}} (\mathbf{r} - \langle \mathbf{r} \rangle)$, where the $\mathbf{a}_i$ are unit-vectors, such that the cumulative variances of the projections to the first m modes, $\sigma_m^2 = \left\langle \sum_{i=1}^{m} c_i^2 \right\rangle$, are maximized for all $m = 1 \ldots 3N$. The c_i are then called *principal* *components.*

Now we show that the mode vector $\mathbf{a}_i^{\mathrm{T}}$ corresponds to the normalized eigenvector associated with the i-th largest eigenvalue of the covariance matrix of atomic displacements,

$$\mathbf{C} = \langle (\mathbf{r}- \langle \mathbf{r} \rangle)(\mathbf{r}- \langle \mathbf{r} \rangle)^{\mathrm{T}} \rangle .$$

Without loss of generality it is assumed that $\langle \mathbf{r} \rangle = 0$, i.e., $\mathbf{C} = \left\langle \mathbf{r}\mathbf{r}^{\mathrm{T}} \right\rangle$. First, the variance of the first principal component c_1 is maximized, i.e., $m = 1$, and, subsequently, the other principal components are obtained by a simple repetition of the steps with $m > 1$.

A maximizer $\mathbf{a}_1$ of the variance $\sigma_1^2 = \left\langle \mathbf{a}_1^{\mathrm{T}} \mathbf{r} \left(\mathbf{a}_1^{\mathrm{T}} \mathbf{r} \right)^{\mathrm{T}} \right\rangle = \left\langle \mathbf{a}_1^{\mathrm{T}} \mathbf{r}\mathbf{r}^{\mathrm{T}} \mathbf{a}_1 \right\rangle = \mathbf{a}_1^{\mathrm{T}} \left\langle \mathbf{r}\mathbf{r}^{\mathrm{T}} \right\rangle \mathbf{a}_1$ under the constraint $\|\mathbf{a}_1\|^2 = 1$ must solve the normal equation

$$0 = \left\langle \mathbf{r}\mathbf{r}^{\mathrm{T}} \right\rangle \mathbf{a}_1 - \lambda \mathbf{a}_1, \tag{2.1}$$

which is obtained by differentiation with respect to $\mathbf{a}_1$ of the Lagrangian function

$$L_\lambda (\mathbf{a}_1) = \mathbf{a}_1^{\mathrm{T}} \left\langle \mathbf{r}\mathbf{r}^{\mathrm{T}} \right\rangle \mathbf{a}_1 - \lambda \left(\|\mathbf{a}_1\|^2 - 1 \right),$$

where λ denotes a Lagrange multiplier. Eq. (2.1) yields the necessary condition that the maximizer $\mathbf{a}_1$ has to be an eigenvector of the covariance matrix $\left\langle \mathbf{r}\mathbf{r}^{\mathrm{T}} \right\rangle$ corresponding to an eigenvalue λ. Moreover, $\lambda = \mathbf{a}_1^{\mathrm{T}} \left\langle \mathbf{r}\mathbf{r}^{\mathrm{T}} \right\rangle \mathbf{a}_1 = \sigma_1^2$, i.e., the maximum variance is given by the largest eigenvalue and its corresponding eigenvector.

After the first $d - 1$ projection vectors $\mathbf{a}_i$ have been identified, the subsequent vector $\mathbf{a}_m^{\mathrm{T}}$ is obtained by maximizing the cumulative variance $\sigma_m^2 = \left\langle \sum_{i=1}^d c_i^2 \right\rangle = \sigma_{m-1}^2 + \left\langle \mathbf{a}_m^{\mathrm{T}} \mathbf{r} \left(\mathbf{a}_m^{\mathrm{T}} \mathbf{r} \right)^{\mathrm{T}} \right\rangle$ keeping the first $m - 1$ vectors, and thus the variance σ_{m-1}^2, fixed. This yields, repeating the steps above, that $\mathbf{a}_m^{\mathrm{T}}$ is the eigenvector of $\mathbf{C}$ corresponding to the m-largest eigenvalue.

An illustrative alternative is to define principal components as the projections c_i for which the reproduction error $\left\langle \|\mathbf{r} - \hat{\mathbf{r}}_m\|^2 \right\rangle$ is minimized. The m-dimensional reproduction $\hat{\mathbf{r}}_m = \mathbf{A}^{\mathrm{T}} (c_1, c_2, \ldots, c_m, 0, \ldots, 0)^{\mathrm{T}}$, where the rows of $\mathbf{A}$ are formed by the vectors $\mathbf{a}_i^{\mathrm{T}}$, is the motion in the original $3N$-dimensional space, which can be described using only m degrees of freedom c_i. The minimization of the reproduction error is equivalent to maximization of σ_m^2

$$\left\langle \|\mathbf{r} - \hat{\mathbf{r}}_m\|^2 \right\rangle = \left\langle \mathbf{r}^{\mathrm{T}}\mathbf{r} - \mathbf{r}^{\mathrm{T}}\hat{\mathbf{r}}_m - \hat{\mathbf{r}}_m^{\mathrm{T}}\mathbf{r} + \hat{\mathbf{r}}_m^{\mathrm{T}}\hat{\mathbf{r}}_m \right\rangle ,$$

which yields with $\mathbf{r} = \mathbf{A}^{\mathrm{T}} (c_1, c_2, \ldots, c_{3N})^{\mathrm{T}}$ due to $\left\langle \mathbf{r}^{\mathrm{T}}\hat{\mathbf{r}}_m \right\rangle = \left\langle \hat{\mathbf{r}}_m^{\mathrm{T}}\mathbf{r} \right\rangle = \left\langle \hat{\mathbf{r}}_m^{\mathrm{T}}\hat{\mathbf{r}}_m \right\rangle = \left\langle \sum_{i=1}^m c_i^2 \right\rangle = \sigma_m^2$ that

$$\left\langle \|\mathbf{r} - \hat{\mathbf{r}}_m\|^2 \right\rangle = \left\langle \mathbf{r}^{\mathrm{T}}\mathbf{r} \right\rangle - \sigma_m^2 .$$

Thus, using PCA the dimension reduced description of the protein dynamics has the smallest reproduction error that is possible to achieve with a given number m of collective degrees of freedom.

This property of PCA renders it particularly useful in the context of CLD. The covariance matrix $\mathbf{C}$, which yields the principal modes by diagonalization, is computed from an MD ensemble. Therefore, we describe in the following section the method of MD simulations, which is used to generate MD ensembles as reported in Sec. 2.3.1, and explain the computation of $\mathbf{C}$ from the generated MD ensembles in Sec. 2.3.2.

2.2 Principles of Molecular Dynamics Simulation

Classical molecular dynamics (MD) is an atomistic simulation method, where:

- each atom is treated as a point mass,
- simple force rules describe the interactions between atoms
- trajectories are generated integrating Newton's equations, and
- thermodynamic statistics and kinetics are extracted from the motion of the atoms.

Since the details of MD simulations are not of central importance for our work, we just shortly summarize its principles and refer to the plentiful literature for detailed accounts[103, 104, 105, 106].

The goal of MD simulations of proteins is an accurate description of the dynamics of molecular systems containing about 10^3 to 10^6 interacting atoms. The large number of interacting particles requires basically three drastic approximations upon the exact description with the time-dependent Schrödinger equation. First, the Born-Oppenheimer approximation separates off the electronic degrees of freedom and describes their effect on the nucleic degrees of freedom $\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)$ in form of a potential energy surface $V(\mathbf{r})$. Second, the motion of the nuclei in this potential energy surface is described classically by Newton's equations of motion

$$m_i \frac{d^2\mathbf{r}_i(t)}{dt^2} = -\nabla_i V(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N),$$

where m_i and $\mathbf{r}_i$ is the mass and the position of the i-th nucleus. These two approximations are the basis of most so called *Quantum Mechanics* molecular dynamics methods[105]. However, to obtain the potential energy and its gradient by solving the time independent Schrödinger equation for the electronic degrees of freedom is computationally expensive and limits those methods to small systems and short simulation times. Therefore, a further approximation is applied, that is the introduction of a semi-empirical force field, which approximates $V(\mathbf{r})$ by a large number of functionally simple energy terms, e.g., we use here,

$$V = \sum_{\text{bonds i}} V_B^i + \sum_{\text{bond angles j}} V_\alpha^j + \sum_{\text{extra planar angles k}} V_{\text{imp}}^k + \sum_{\text{dihedrals l}} V_D^l + \sum_{\text{pairs } \alpha,\beta} \left(V_q^{\alpha,\beta} + V_{\text{vdW}}^{\alpha,\beta} \right).$$

The simple energy terms, are harmonic (e.g., V_B, V_α and V_{imp}) or motivated by physical laws (e.g., Coloumb V_q and Lennard-Jones V_{vdW}). They are defined by their functional form and a small number of parameters, e.g., an atomic radius for V_{vdW}. The number of energy terms, their functional forms, and their individual parameters can differ substantially between different force fields. Thereby, the single parameter, e.g., an atomic radius, carries only limited information on its own, but is an essential part to yield the correct dynamics in the context of the whole force field. The parameters are usually determined together in an iterative process, using experimental data, quantum-chemical calculations, or comparisons of thermodynamic data with suitable averages of small molecule MD ensembles. A large number of such force fields have emerged, e.g., OPLS[107], CHARMM[108], GROMOS[109], AMBER[110], MM3[111]. Here OPLS and GROMOS were used.

Any observable that can be connected to macroscopic experiments has to be defined as an ensemble average $\langle A \rangle_{\text{ensemble}}$, as prescribed by statistical mechanics. The ensemble average, however, cannot be obtained *directly* from the single replica of the protein system described by MD. Nevertheless, the ergodic hypothesis, which is generally assumed to apply for protein dynamics, allows the *indirect* computation of ensemble averages as time averages from such single replica MD simulations

$$\langle A \rangle_{\text{time}} = \lim_{T \to \infty} \frac{1}{T} \int_{t=0}^{T} A\left(\mathbf{r}(t), \mathbf{p}(t)\right) dt.$$

Alternatively, non-equilibrium observables, e.g., escape times, can be obtained by averaging over a large number of relaxation simulations, whose starting conditions are drawn randomly from an appropriate ensemble[112, 113].

The above approximations lay the foundation for a practical realization of molecular dynamics simulations of proteins, as it is done, e.g., in the GROMACS software package[114], which was used in this work and whose algorithms and methods will be sketched in the following:

Newton's equations of motion are iteratively solved in steps on the femto-second scale by means of the leap-frog algorithm[115], which has the advantage that the expensive force calculation is done only once per integration step. To avoid artifacts from the boundaries such as evaporation, high pressure due to surface tension, and preferred orientations of solvent molecules on the surface, periodic boundary conditions are applied. In this way the simulation system does not have any surface. This, however, may lead to new artifacts if the molecules also interact with their periodic images due to the long-range electrostatic interactions. These periodicity artifacts are minimized by increasing the cell size. Different choices of unit cells, e.g., cuboid, dodecahedron, or truncated octahedron allow an improved fit to the shape of the protein, and, therefore, allow reduction of the number of solvent molecules, while simultaneously keeping the crucial protein-protein distance high.

A solution of Newton's equations conserves the total energy of the system (NVE ensemble). However, in real systems a molecular subsystem of the size studied in the simulation constantly exchanges energy with its surrounding. To be closer to reality, this energy exchange should therefore be introduced to the simulation. Thus, the temperature T of the system is kept constant by applying one of many proposed *thermostats*[116, 117, 118]. The popular *Berendsen thermostat*, which simply rescales the velocities each step, was applied here[117].

In addition to the heat bath coupling, real biological systems are subjected to a constant pressure of usually 1 atm. Therefore, in the simulations, isobaric ensembles were generated by applying the *Berendsen barostat*, which rescales the coordinates each step[117]. Thus, NPT ensembles are created.

Additionally a couple of measures are taken purely to increase computationally efficiency. These are parallelization, constraining bonds to increase the time-step, reduction of particle number by replacing aliphatic carbon centers with compound atoms (for GROMOS96 force field), and the special treatment of non-bonded forces with Ewald-Summation techniques.

2.3 Methods

2.3.1 Generation of MD ensembles

In the following we report the particular methodical details for all molecular dynamics (MD) simulation carried out within this work using the GROMACS simulation suite[114].

Lincs and Settle[28, 27] were applied to constrain covalent bond lengths, allowing an integration step of 2 fs. Electrostatic interactions were calculated using the Particle-Mesh-Ewald method[119, 120]. The temperature was kept constant by separately coupling ($\tau = 0.1$ ps) the peptide and solvent to an external temperature bath[117]. The pressure was kept constant by weak isotropic coupling ($\tau = 0.1$ ps) to a pressure bath[117].

Crambin

Two molecular dynamics simulations of crambin, CR1 and CR2, were started from the crystal structure (Protein Data Bank entry 1CBN[121]). The simulations were carried out with the GROMOS96 force field F49A1[122]. The protein was solvated in 2718 SPC water molecules[123]. The total system size comprised 8563 atoms. The simulations were carried out using periodic boundary conditions in a dodecahedronal box. Simulation CR1 was run for 450 ns, and coordinates were recorded every 0.1 ps. To obtain high-resolution Fourier spectra, an additional simulation, CR2, was performed for 100 ps, with coordinates and velocities recorded every 2 fs timestep.

Neurotensin

Several molecular dynamics simulations of neurotensin were carried out, using the OPLS all atom force field [107]. Neurotensin, a peptide with the sequence Ac-RRPYIL[124], was solvated with 2246 TIP4P water molecules[125] and 2 Cl^- counter ions in a cubic box. A first simulation was started from an extended configuration and equilibrated for 10 ns. The 90 ns simulation NT1 was started from the last snapshot of the equilibration, and coordinates were recorded every 1ps. A second simulation NT2 of length 63 ns, was started from the last snapshot of NT1, and positions and velocities were recorded every $10\,\mathrm{fs}$, which allowed for the computation of velocity autocorrelations without aliasing artefacts.

Additionally, eight $500\,\mathrm{ps}$ simulations, NTS$_i$, $\quad i = 1 \ldots 8$, were started from snapshots of NT1 selected for their mutually large root mean square differences, and positions and velocities were recorded every $10\,\mathrm{fs}$.

T4 Lysozyme

A molecular dynamics simulation of coliphage T4 lysozyme (T4L), $117\,\mathrm{ns}$ long using the OPLS all atom force field [107], was started from the crystal structure of a M6I mutant (PDB entry 150L chain D[126]). The protein was solvated in 8898 TIP4P water molecules[125] and 8 Cl$^-$ counter ions using a rectangular box.

B1 domain of Protein G

Two Molecular dynamics simulations of the B1 domain of streptococcal protein G (GB1 and GB1/2), using the OPLS all atom force field [107], were started from the crystal structure (Protein Data Bank entry 1PGB[127]). The protein was solvated in 4651 TIP4P water molecules[125] using a cubic box. Four sodium ions were added to the simulation system in order to compensate for the net negative charge of the protein. Simulations were run for $100\,\mathrm{ns}$ (GB1/2) and $200\,\mathrm{ns}$ (GB1), respectively.

2.3.2 Recording of trajectory data

Sampling frequency

In those instances, where we are interested in autocorrelation functions or frequency distributions of the motion, special care was taken to avoid aliasing artifacts. In signals sampled with a finite step size Δt any frequency component f above the *Nyquist frequency* $f_c = (2\Delta t)^{-1}$ will be indistinguishable from an oscillation, which differs from f by a multiple of Δt^{-1}, e.g., a slow oscillation in the range $0 \ldots f_c$[128]. Thus, in sampled data high frequencies above f_c contribute spuriously to the low frequency spectrum, which is called *aliasing.* Note that velocity autocorrelation functions can suffer from aliasing effects, too.

To avoid any aliasing we sampled all velocities with a timestep of $\Delta t = 10\,\mathrm{fs}$, which corresponds to a Nyquist frequency of $50\,\mathrm{ps}^{-1}$. Test computations with sampling-timestep $\Delta t = 2\,\mathrm{fs}$ asserted that all observed frequency contributions were well below $50\,\mathrm{ps}^{-1}$.

Collective Coordinates with principal component analysis

We used PCA to extract collective coordinates from MD simulations. Since we were interested in the internal protein motion only, overall translational and rotational motion was removed in a first step from the recorded position $\tilde{\mathbf{r}}$. This was achieved by moving the center of mass $\tilde{\mathbf{r}}_{\mathrm{cm}}$ into the origin and subsequent least squares fitting to a reference structure $\mathbf{r}_{\text{ref}}$, which yields a rotation $\mathbf{R}$, such that $\mathbf{r} = \mathbf{R}\,(\tilde{\mathbf{r}} - \tilde{\mathbf{r}}_{\text{cm}})$. As reference structure we chose the crystal structure or, if not available, the starting structure of the simulation.

Principal component analysis (PCA) was carried out by diagonalizing the covariance matrix $\mathbf{C} = \langle (\mathbf{r} - \langle \mathbf{r} \rangle)\,(\mathbf{r} - \langle \mathbf{r} \rangle)^{\text{T}} \rangle$, where angular brackets denote averaging over an MD trajectory. The eigenvectors of $\mathbf{C}$ yielded the PCA modes $\{\mathbf{a}_j\}_{j=1\ldots 3N}$, and positions projected onto mode j were obtained as $c_j = \mathbf{a}_j^{\text{T}}\,(\mathbf{r} - \langle \mathbf{r} \rangle)$.

For consistency with the positions, the rotational and translational motion had to be removed from the recorded velocities $\tilde{v}(t_i)$. Therefore, the translational and rotational contributions to the velocities were computed from the time dependence of both, the displacement vector $\tilde{\mathbf{r}}_{\text{cm}}(t_i)$ and the rotation matrix $\mathbf{R}(t_i)$. Corrected velocities $\mathbf{v}(t_i)$ were given by removing these contributions

$$\mathbf{v}(t_i) = \tilde{\mathbf{v}}(t_i) - \Delta t\,[\tilde{\mathbf{r}}_{\text{cm}}(t_{i-1}) - \tilde{\mathbf{r}}_{\text{cm}}(t_i) + \mathbf{R}(t_{i-1})\mathbf{x}(t_i) - \mathbf{R}(t_i)\mathbf{x}(t_i)]\,, \tag{2.2}$$

where Δt denotes the sampling interval. Thus, for the projected velocities $\dot{c}_j(t) = \mathbf{a}_j^{\text{T}}\mathbf{v}_c(t)$ consistency with the projected positions was achieved, i.e., $c_j(t) = \int_0^t \dot{c}_j(\tau)d\tau + c_j(0)$.

2.3.3 Convergence of conformational subspaces

Here we describe the stability measure that was used to quantify how well principal components derived from short MD simulations can also describe the fluctuations observed in long MD simulations. In particular, this measure should quantify the fraction of the protein dynamics that can be described with a given subset of principle components, which is not achieved by the usually employed measures[50, 102, 129, 130]. Instead, we computed the average reduction in observed fluctuation amplitude upon projection P on to a subspace of given dimension m, as the similarity

$$\Upsilon = \sqrt{\left\langle \frac{\|P(\mathbf{x})\|^2}{\|\mathbf{x}\|^2} \right\rangle}. \tag{2.3}$$

In order to use the similarity Υ as stability measure, we applied the projection P, given by a set of principle components derived from a short simulation, to an ensemble derived from a (different) long-time simulation. Therefore, the average was evaluated by projecting each fitted (see above) position $\mathbf{x}$ of the ensemble CR1 to PCA modes $\{\mathbf{a}_j\}_{j=1\ldots m}$, such that $\|P(\mathbf{x})\|^2 = \sum_{j=1}^m (\mathbf{a}_j \cdot \mathbf{x})^2$. Note, that our similarity measure differs from the one previously proposed[50],

$$\sqrt{\frac{\left\langle \|P(\mathbf{x})\|^2 \right\rangle}{\left\langle \|\mathbf{x}\|^2 \right\rangle}} = \sqrt{\frac{\sum_{i=1}^m \sigma_i^2}{\sum_{j=1}^{3N} \sigma_j^2}},$$

where σ_i^2 denotes the eigenvalue of PCA mode $\mathbf{a}_i$. The advantage of the measure Eq. (2.3) is that it goes beyond a Gaussian approximation of the ensemble density. It also improves upon the measure proposed in Ref. [102], by taking into account amplitudes.

The PCA modes were obtained from short fragments of CR1 with differing length T ranging from 100ps to 400ns. The mean similarity $\bar{\Upsilon}$ was computed from M fragments of the same length, with error bars computed as $\Delta\Upsilon = \left(\sum_i^M \frac{1}{M(M-1)} \left(\Upsilon - \bar{\Upsilon}\right)^2\right)^{-1/2}$. For fragment sizes below 50ns, 5 fragments were chosen randomly, for larger fragment sizes, 2-5 (overlapping) fragments were chosen with a separation of 50ns. Snapshots from the fragments were taken every 0.1ps for $T < 500$ps and every 1ps for $T > 500$ps, respectively.

2.3.4 Frequency Spectra

Spectral densities g_j of a PCA mode $\mathbf{a}_j$ were computed from the discrete Fourier transform of the projected velocity $\mathbf{v}_k$ as

$$g_j(\omega) = \frac{|X_j(\omega)|^2}{2\pi},$$

where $X_j(\omega) = \sum_{k=0}^{N-1} \mathbf{v}_k \cdot \mathbf{a}_j \exp\left(-i\omega k \Delta t / N\right)$, where $\mathbf{v}_k$ denote M velocities sampled with intervals of Δt.

2.4 Separation of timescales

In this section we investigate whether and how principal component analysis (PCA) can be applied to identify slow collective modes suitable for CLD by considering molecular dynamics of the protein Crambin. To this end the vibrational density of states along different PCA modes was analyzed. Furthermore, because usually PCA is carried out on subsets such as C_α-atoms only[40, 131], we also analyzed the influence of such a preselection of atoms on the timescale separation properties.

Figure 2.1 (a-d) shows examples of frequency distributions of the MD trajectory CR2 projected on single PCA modes. Panel (a) and (c) show the first mode of PCA carried out on all C_α-atoms and heavy atoms, respectively. A corresponding high index mode (C_α: 84th / 138 modes and heavy atoms: 601th/ 981 modes) was plotted in panel (b) and (d). The first mode of the PCA carried out on C_α-atoms, i.e., mode 1/C_α, (panel a) showed the expected slow contributions $\nu < 5$ps^{-1}. With similar weight, however, intermediate and also fast dynamics $\nu \approx 20$ps^{-1} contributed to this mode. The latter are likely to result from angle vibrations, which occur at such timescales. Higher frequencies, corresponding to bond vibrations, were hardly seen, because these are suppressed by the used constraints. The density of states of mode 84/C_α in panel (b) lacks contribution of the slowest motions, but shows hardly any change to mode 1/C_α in the distribution of the remaining frequencies.

In contrast, the two corresponding modes obtained by PCA carried out on all heavy atoms showed a significantly improved separation of spectra. Both showed narrower frequency distributions than the C_α-based modes. The spectrum of mode 1/heavy (panel c) contained only frequencies below $\nu < 5$ps^{-1}, whereas mode 601/heavy showed only frequencies above $\nu > 10$ps^{-1}.

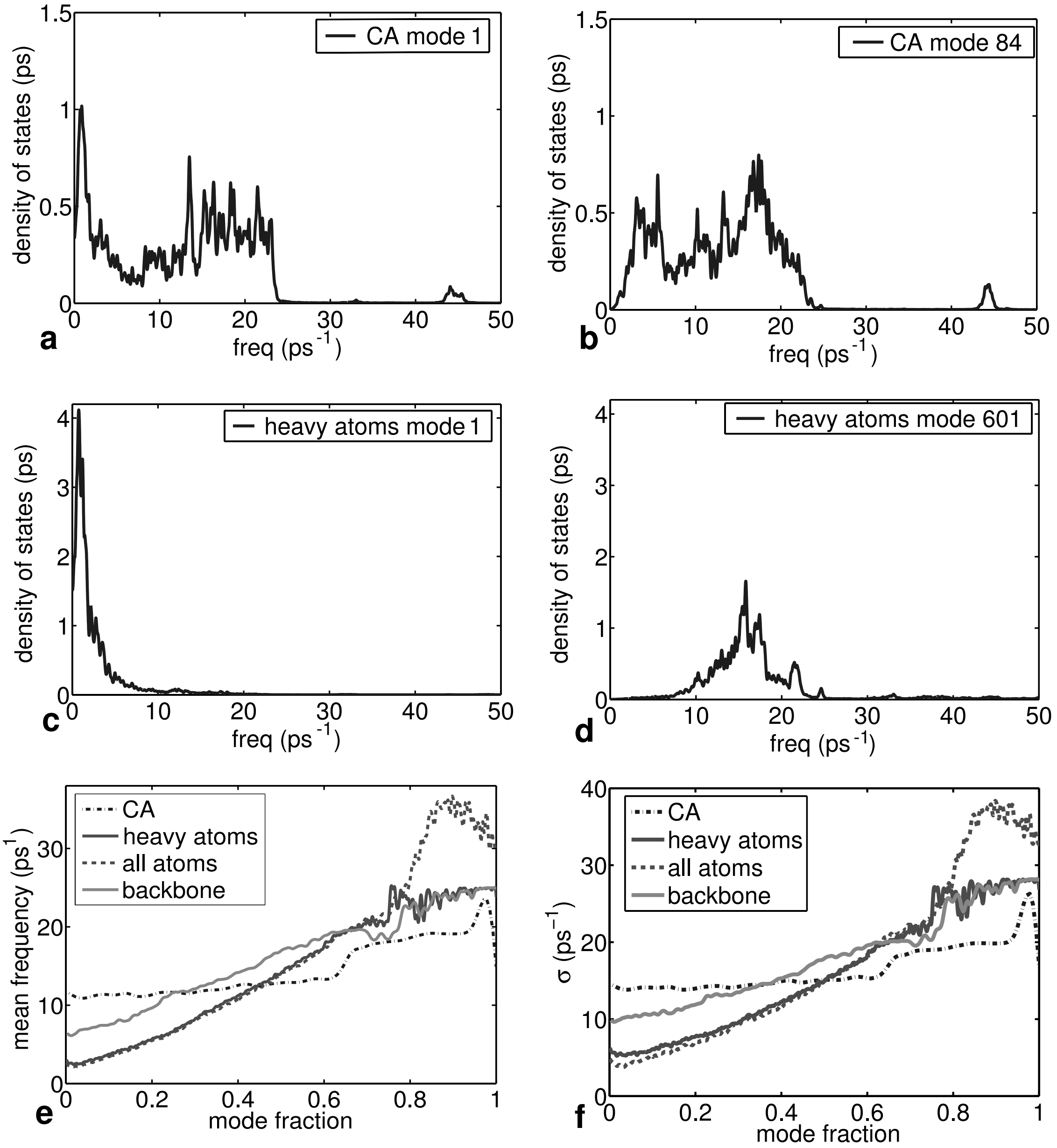


Figure 2.1: Comparison of spectral densities for different PCA modes. PCA was carried out on the four different atom sets, C_α-atoms, backbone, heavy atoms, and all atoms. Sample densities of states corresponding to these PCA modes are shown exemplary for individual modes **(a-d)** and are characterized by their averages **(e)** and widths **(f).** To facilitate comparison despite different number of modes, the mode axis was normed to 1.

To gain a more systematic overview we plotted the mean (Fig. 2.1e) and width (Fig. 2.1f) of the frequency distribution for every mode and for the four analyzed atom sets, C_α-atoms, backbone, heavy atoms, and all atoms. For the C_α-atoms, the nearly constant mean and the constantly large width underscored the lack of proper timescale separation. In contrast, for the heavy atoms, the strong dependence of the average frequency on the mode index, together with the initially small widths, shows that, indeed, a much improved separation is achieved, as was suggested already by the examples (cf. Fig. 2.1c,d). An intermediate result is obtained for backbone atoms; the mean of the slightly broader frequency distribution increases, but with a smaller slope.

Obviously, the separation of time scales improved with the number of atoms used for the PCA. To rule out that this is merely due to the different number of degrees of freedom, we carried out a similar analysis for neurotensin (6 residues) and HLA (385 residues) (the MD simulation of HLA-B27 is described in [132]). Both systems showed the same dependency of the time scale separation on the selected atom set (results not shown). In particular the first of the 1155 C_α modes showed strong high frequency contributions. This confirmed that the selection of the correct atom set is crucial to extract slow modes with PCA, independent of the system size. In all cases the best - and sufficient - timescale separation is achieved only if the PCA involves all heavy atoms.

Does inclusion of hydrogen atoms further improve the time scale separation? Figure 2.1e shows that the improvement is small. The high frequency motion of these light particles is largely uncoupled from the slow modes. This, is reflected in an increased mean frequency only in the last quarter of the modes. Thus, an exclusion of the hydrogen atoms does not change the dynamics of the slower modes.

These findings show that PCA is able to identify systematically slow modes describing conformational motion, as expected. Moreover, the analysis revealed a strong improvement of the quality of this separation if all heavy atoms of the protein participated in the collective modes.

An explanation needs to be found for the counterintuitive intrusion of high frequency motion in modes like $1/C_\alpha$. From the effective frequency ω_{eff} and accordingly the equipartition theorem one would expect a separation of timescales, which is, however, only seen for PCA carried out on all heavy atoms. Possibly, the reason for that is that in the first case atoms are excluded from the PCA which are strongly coupled to the analyzed ones. For example consider motion in a three dimensional highly elliptical harmonic well, whose cartesian degrees of freedom are highly coupled, such that the principal axes are very different from the coordinate axes. If PCA is applied to all degrees of freedom, it finds the three principal axes of this ellipse. One very short, with fast frequency, and the others longer, with slower frequencies. However, if PCA is carried out on only two of the three original degrees of freedom then obviously the true principal axes which are not in the plane of the considered two degrees of freedom cannot be found. Therefore, high frequency motion is mixed into the slow frequency motion and cannot be separated off anymore. This simple example of three highly correlated 1-d atoms illustrates what might cause the break-down of PCA if strongly coupled atoms are excluded, as it is the case with PCA on C_α-atoms. On the other hand, exclusion of hydrogen atoms only, does not impede the expected separation of timescales, because their motion does not

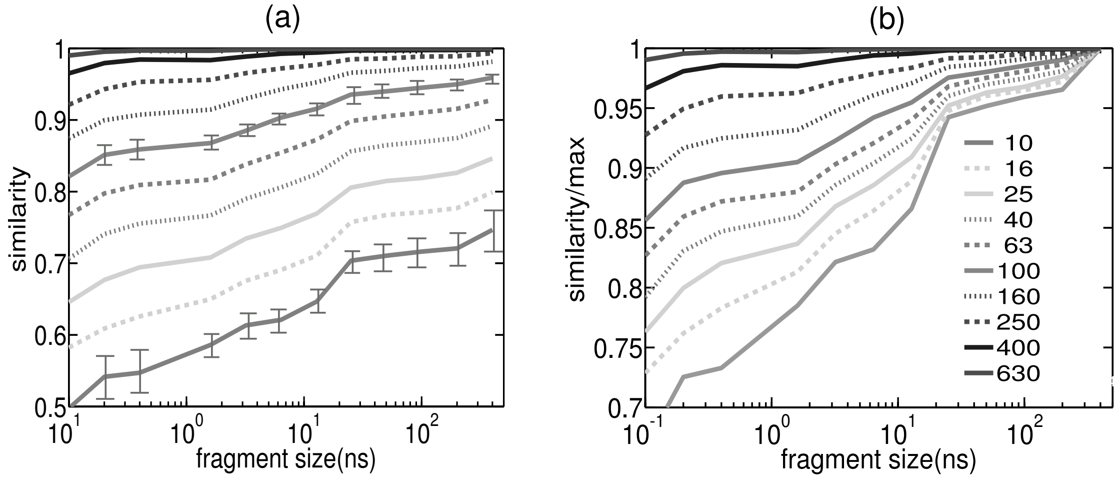


Figure 2.2: Convergence of conformational subspaces for Crambin. **(a)** Similarity (Υ, Eq. (2.3)) between PCA subspaces of different dimensionality (cf. legend in (b)) obtained from varying short fragments (cf. abscissa) of the 450ns-trajectory CR1 and the whole ensemble **(b)** same as figure (a), but the similarities are normalized by the maximally achievable similarity for the respective subspace dimensionality.

couple strongly enough to the heavy atom motion.

2.5 Convergence of conformational subspaces

For the formulation of the CLD we proposed to use as conformational subspace the low-frequency principal components or, embedded within this space, curved coordinates. Therefore, we need to test whether and to what extent a small number of principal components obtained from short MD simulations can describe the major part of protein dynamics also on relatively long timescales. To this end we carried out PCA on varying short fragments of the 450ns MD simulation, CR1, of the protein Crambin.

The similarity Υ between the whole ensemble and subspaces obtained from such fragments were computed for a wide range of subspace dimensionalities $m = 10 \ldots 680$, i.e., number of principal components used to describe the protein motion. Figure 2.2a shows that these similarities were monotonically increasing with larger fragment size (horizontal axis), and that also for a given fragment size the similarity increased if the PCA subspace is enlarged (different curves). The results for the largest fragment size, thereby, reflected the well-known result that ~5% of the eigenvectors describe ~90% of the motion[50] (e.g., the curve corresponding to m=40 reaches 0.9 in Figure 2.2a).

Focussing on the dependence of the similarity on the fragment size, Figure 2.2b shows the curves normalized by their respective maximum similarity. In particular, a PCA subspace of $m = 100$, i.e., 10% of all eigenvectors, computed from a short MD simulation of length 3ns could describe 86 % of the whole ensemble generated in a 450ns simulation, which was 90% of the maximally achievable limit of 96% for a subspace of that size. Apparently, already subspaces from short simulations describe a large fraction of the long time protein dynamics.

Note that for all subspace dimensions two regimes of the similarity curves are seen. Above 25ns, their slope decreases significantly, and, therefore, not much is gained using larger trajectory fragments. A similar leveling off is found between 300ps $- 2$ns, but only for the larger subspaces.

These results show, at least for the protein Crambin, a remarkably fast convergence of the conformational subspace. Hence, a few ns of simulation time suffice to derive a suitable conformational subspace for long time CLD simulation with PCA.

This remarkable results needs to be discussed in the light of the arguments put forward by Balsera et al.[52]. They claimed, in contrary to us, that slow convergence of the fluctuation amplitudes along the largest PCA modes, would prevent such determination of long time-scale modes from short MD simulations. Moreover, they compared the directions of eigenvectors of two independent 235ps simulations of a 375 residue protein and asserted from missing overlap that no convergence of the directions was reached.

However, we do not agree with their conclusion that PCA subspaces of short MD simulations cannot describe long time protein motion.

Firstly, the eigenvalues of the principal modes are not important for CLD. Secondly, the single direction of an eigenvector is not relevant, but rather the whole space spanned by the most important principal modes. For instance, the new direction of the first principal mode due to a freshly sampled conformational substate, is nevertheless, as showed by our results, likely to be contained in the conformational subspace spanned by the m largest modes already, if m is sufficiently large. Thirdly, judging from our results obtained for a 46 residue protein, the small simulation time of 235ps for a much larger protein was slightly to short to see an onset of convergence.

To summarize, despite the well-known slow convergence of the complete information within PCA, in particular its eigenvalues[102, 130, 129], about 5ns of MD simulation of crambin suffice to define a suitable subspace for CLD. This does not imply that a good sampling of the configurational space was reached, but simply that several nanoseconds suffice for a good sampling of the near constraints subspace, as already pointed out earlier[102].

2.6 Conclusions

We have shown that principal component analysis (PCA) is a suitable method to extract from molecular dynamics (MD) simulations collective coordinates for the proposed framework of collective Langevin dynamics (CLD). In particular, a few extracted coordinates are able to describe a large fraction of the overall atomic displacement. As shown, this even holds true for remarkably long time scales. For the protein crambin ten percent of the principal components obtained from MD simulations shorter than 5 ns were able to describe over 85% of the total atomic displacement observed in a 450 ns MD simulation. Furthermore, we were able to show that PCA, if based on the covariance matrix of the fluctuations of all heavy atoms, is able to partially separate timescales. Thus, low indexed modes constitute slow degrees of freedom, which are free of contributions from fast vibrational dynamics, and are, therefore, suitable for CLD.

...everything that living things do can be understood in terms of the jigglings and wigglings of atoms.
— Richard P. Feynman

Chapter 3

Generalized Correlation of Biomolecular Dynamics

Correlated motions in biomolecules, in particular proteins, are ubiquitous and often essential for biomolecular function[133]. Examples are allosteric signal transduction, as in G protein coupled receptors (GPCRs)[134], or mechanical/thermodynamic energy transport, as in F_0/F_1-ATPase[135]. Furthermore, the energetics of protein function is often dominated by entropic contributions, which are directly linked to correlated atomic motion[136, 137, 138]. Correct assessment of correlated motions, both experimentally and from theory and simulations, is therefore crucial for a quantitative understanding of biomolecular function.

Collective Langevin dynamics (CLD) intrinsically describes correlated motions, since it is based on collective coordinates as degrees of freedom. How accurate the correlated motions are represented, however, is determined to a large extent by the choice of the collective coordinates. Thus, it is essential for CLD to extract from molecular dynamics (MD) simulations such coordinates which describe the correlations well.

This in turn requires that the MD trajectory used for extraction describes the correlations accurately, which is optimally checked by a direct comparison of this observable with experiments that probe correlations in the motions of $atoms$ in three dimensional space[54, 55, 56]. The established method to quantify these correlations from MD simulations, in analogy to the *Pearson correlation coefficient*, rests on calculation of the normalized covariance matrix of atomic displacements, $C_{ij} = \langle \mathbf{x}_i \cdot \mathbf{x}_j \rangle / \sqrt{\langle \mathbf{x}_i^2 \rangle \langle \mathbf{x}_j^2 \rangle}$, where $\mathbf{x}_i$ and $\mathbf{x}_j$ are the positional displacement vectors of atoms i and j, respectively, in the molecular fixed frame[58, 57]. As will be shown in the Theory Section, this established approach, however, misses a considerable fraction of the correlated motions and, therefore, usually underestimates atomic correlations. This limitation is mainly due to two assumptions.

First, estimates of correlations from the Pearson coefficient are only strictly valid if $\mathbf{x}_i$ and $\mathbf{x}_j$ are co-linear vectors, as already pointed out by Ichiye and Karplus[57]. Improved results are obtained with the method of *canonical* correlations[139] by choosing so called *canonical* variables which furnish average co-linearity, i.e., for every pair of atoms a different coordinate transformation is applied.

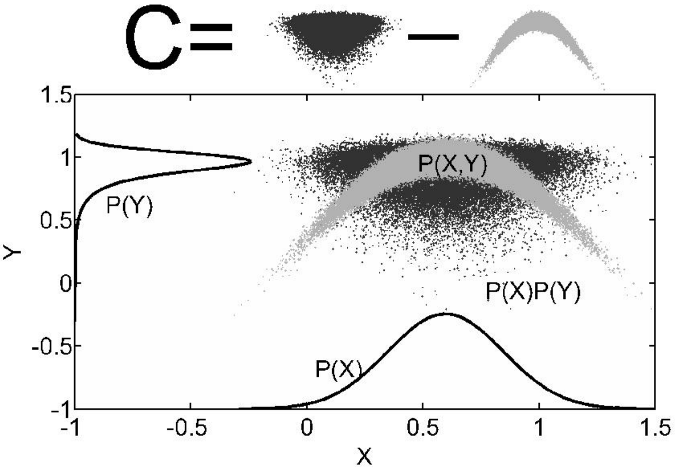


Figure 3.1: Correlations of random variables are defined as deviation of their probability distribution from the hypothetical probability distribution of the independent random variables. In the sketch, the correlation between variables and (gray in the scatter plot) is to be quantified. From the marginal distributions and (black curves) one computes the hypothetical joint distribution for independent variables (black points). The difference between the given joint distribution and the hypothetical uncorrelated joint distribution yields the correlation measure, as illustrated at the top of the graphic.

In contrast to the Pearson correlation coefficient, canonical correlations do not differentiate between correlated and anticorrelated, i.e., *positively correlated,* and *negatively correlated* motion. Such a distinction becomes problematic in the multidimensional case, and thus has to be dropped for *any* meaningful correlation measure. Consider, e.g., two atoms which oscillate perfectly correlated in parallel directions. If the oscillation direction of one atom is rotated until both atoms oscillate antiparallel, the Pearson correlation coefficient changes from 1 to -1 and therefore has to cross zero, usually after rotation by $90°$, i.e., when the directions are perpendicular. In this case, the vanishing correlation coefficient is highly misleading, because the motion of the two atoms is still perfectly correlated.

Second, use of the covariance matrix implies a Gaussian approximation of the underlying configurational space density. Therefore, this approach treats correlations in a quasi-harmonic, i.e., linear, approximation. Thus, the Pearson correlation coefficient, as well as the canonical correlation method, miss non-linear correlations. Higher moment corrections are conceivable, but notoriously suffer from dramatic combinatorial increase of computational effort, and slow convergence, which renders the treatment of large systems such as proteins impossible.

As an efficient alternative, we propose here a general approach to quantify any correlated motion.

The proposed generalized correlation measure rests on the fundamental definition of independence of random variables. Accordingly, two random variables are independent, if and only if their joint distribution is a product of their marginal distributions, $P(X,Y) = P(X)P(Y)$. The basic idea is to quantify the correlation between variables X, Y as the deviation between both sides of the above equation, i.e., by the deviation from the case of two independent random variables (Fig. 3.1). As will be shown in Sec. 3.1.2, this definition is equivalent to defining a correlation C as the well-known (Shannon) mutual information (MI)[53], $C\,[X,Y] = H\,[X] + H\,[Y] - H\,[X,Y]$, where H denotes the entropy of the random variables. This definition rests on the well-known inequality $H\,[X,Y] \leq H\,[X] + H\,[Y]$, which becomes an equality if and only if both variables are independent.

This formulation is equivalent to an infinite moment expansion. Truncation at second moments yields a linearized mutual information which will be defined in Sec. 3.1.3.

In Sec. 3.1.1 we will review the definition of the Pearson correlation coefficient and its canonical interpretation. Following this interpretation we will define the *generalized correlation coefficient* which maps the mutual information with values in the range $0, \ldots, \infty$ onto the more convenient interval $[0, 1)$ to allow a direct comparison with the Pearson correlation coefficient.

The impact of the known[57] problems of the Pearson correlation coefficient seems largely underrated, and the canonical correlation approach[139], is generally not applied. Here we quantify the inconsistencies and shortcomings of the Pearson correlation coefficient when applied to protein dynamics. To this end, two examples are studied, the B1 domain of Protein G and T4 Lysozyme. Using these examples, we will also show that our generalized correlation measure does not suffer from these shortcomings and, therefore, provides an accurate and complete quantification of correlations in protein dynamics. Note that the B1 domain of Protein G was chosen, because its experimental data was available for the aspired comparison presented in Chapter 5.

3.1 Theory of correlation measures

At first we introduce some notation. In this chapter we focus on correlations of atomic displacements, i.e., of vectors in 3-dimensional space. In Chapter 4 it will be necessary to discuss also correlations between one-dimensional variables. Therefore, we use the following notation. All positions of atoms (or other variables) are denoted by a vector $\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)^{\mathrm{T}}$ with N components $\mathbf{r}_i \in \mathbb{R}^d$, with $d = 3$ for atoms ($d = 1$ for coordinates). We refer to positional displacements, i.e., the deviation from the mean, $\mathbf{x} = \mathbf{r} - \langle \mathbf{r} \rangle$, with

$$\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)^{\mathrm{T}} = \left(x_1^{(1)}, x_1^{(2)}, \ldots, x_1^{(d)}, \ldots, x_N^{(1)}, \ldots, x_N^{(d)} \right)^{\mathrm{T}}$$

and $\langle . \rangle$ denoting the ensemble average. With $p(\mathbf{x})$ we denote the corresponding probability density, which in the context of biomolecular dynamics is the canonical ensemble density $p(\mathbf{x}) = Z^{-1} \exp(-\beta V(\mathbf{x} + \langle \mathbf{r} \rangle))$, where Z is the partition function, β the inverse temperature, and V the potential energy. Further we denote the marginal probability density by $p_i(\mathbf{x}_i) = \int p(\mathbf{x}) d\mathbf{x}_{j \neq i}$.

3.1.1 Pearson correlation coefficient

The established and intuitive method[57, 58] to quantify the correlation between pairs of components (i, j) of the displacement vector $\mathbf{x}$ is

$$r\left[\mathbf{x}_i, \mathbf{x}_j\right] = \langle \mathbf{x}_i \cdot \mathbf{x}_j \rangle / \left(\langle \mathbf{x}_i^2 \rangle \langle \mathbf{x}_j^2 \rangle \right)^{1/2}, \tag{3.1}$$

where the square brackets indicate the dependence on the whole ensemble of $\mathbf{x}_i, \mathbf{x}_j$.

In the one-dimensional case, r is called the *Pearson coefficient*, and it has a very straightforward and fairly general interpretation: Under the assumption that at least one variable is normally distributed, it yields the *coefficient of non-determination,*

$$1 - r^2 = \frac{\left\langle (x_j - f(x_i))^2 \right\rangle}{\left\langle x_j^2 \right\rangle}, \tag{3.2}$$

of the best *linear* fit $f\left(x_i\right)$ to x_j. For the multidimensional case, an analogous interpretation of the Pearson coefficient is possible provided that the atoms (i, j) have unit variance $\left\langle x_i^{(k)} x_i^{(l)} \right\rangle = \delta_{kl}$ and the displacements are co-linear $\left\langle x_i^{(k)} x_j^{(l)} \right\rangle = r_k \delta_{kl}$, i.e., the part of their covariance matrix containing cross-correlations is diagonal. Then

$$\frac{\left\langle (\mathbf{x}_j - f(\mathbf{x}_i))^2 \right\rangle}{\left\langle \mathbf{x}_j^2 \right\rangle} = \frac{1}{d} \sum_{k=1}^{d} \left(1 - r_k^2\right),$$

which simplifies in the case of identical correlation coefficients $r_k = r$ ($k = 1, \ldots, d$) to the *coefficient of non-determination* for single variate variables, Eq. (3.2).

In the following discussion, and in accordance with common practice, we will use Eq. (3.1) also in the multi-variate cases to define a *Pearson coefficient,* due to its similarity with the usual single-variate definition. However, in these cases several problems arise, which seemingly have not yet impeded widespread use[57, 58, 140]. Firstly, the conditions co-linearity and unit variance are generally not satisfied, thus invalidating the interpretation as a coefficient of non-determination. This raises serious doubts regarding any conclusions drawn from this measure, particularly because any value for it can be obtained for a given ensemble by scaling single coordinates. Secondly, the Pearson coefficient is limited to detect *linear* correlations, i.e., it yields the coefficient of non-determination regarding the best *linear* fit. *Non-linear* fits, which can yield much lower coefficients of non-determinations, are therefore not considered. This latter problem applies also to the one dimensional case. Consider, e.g., two atoms oscillating in parallel direction, but with a $90°$ phase shift. They will give rise to a vanishing correlation matrix element $\langle \sin(\omega t) \sin(\omega t + \pi/2) \rangle = 0$, and, thus, this fully correlated motion would also not be detected. In configurational space, this motion generates an ensemble distributed along the perimeter of a circle, which cannot be captured by the Gaussian approximation implied in any formulation of correlated motion based on second moments.

3.1.2 Mutual information

Among the measures of correlation between random variables, mutual information (MI) is singled out by its information theoretical background[53]. Accordingly, the joint probability distribution $p(\mathbf{x})$ is the product of the marginal distributions $p_i(\mathbf{x}_i)$,

$$p(\mathbf{x}) = \prod_{i=1}^{N} p_i\left(\mathbf{x}_i\right) \tag{3.3}$$

if and only if the components $\mathbf{x}_i$ are independent, i.e., uncorrelated. Because Eq. 3.3 can be rewritten as

$$\ln \frac{p(\mathbf{x})}{\prod_{i=1}^{N} p_i\left(\mathbf{x}_i\right)} = 0,$$

the ensemble-averaged deviation from the uncorrelated distribution is given by the mutual information (MI)[141, 53],

$$I\left[\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\right] = \int p(\mathbf{x}) \ln \frac{p(\mathbf{x})}{\prod_{i=1}^{N} p_i\left(\mathbf{x}_i\right)} d\mathbf{x}. \tag{3.4}$$

Only for fully uncorrelated motions, MI vanishes.

Evaluation of the right hand side of Eq. (3.4) relates MI to the more widely known measure of information content (entropy) $H[\mathbf{x}] = -\int p(\mathbf{x}) \ln p(\mathbf{x}) d\mathbf{x}$,

$$I\left[\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\right] = \sum_{i=1}^{N} H[\mathbf{x}_i] - H[\mathbf{x}]. \tag{3.5}$$

In contrast to the Pearson coefficient, this measure is scale-invariant. Even individual linear coordinate transformations in the d-dimensional subspaces, i.e., $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N) \mapsto (\mathbf{T}^{(1)}\mathbf{x}_1, \mathbf{T}^{(2)}\mathbf{x}_2, \ldots \mathbf{T}^{(N)}\mathbf{x}_N)$, as given by $d \times d$-matrices $\mathbf{T}^{(i)}$, leave the mutual information invariant, as little algebra shows. Here, we focus on the correlation between pairs of atoms,

$$I\left[\mathbf{x}_i, \mathbf{x}_j\right] = H[\mathbf{x}_i] + H[\mathbf{x}_j] - H[\mathbf{x}_i, \mathbf{x}_j]. \tag{3.6}$$

For higher order correlations we refer to Ref. [142].

Having established that the mutual information provides us with a well defined and complete measure of correlation, we note that it yields values in the range $[0 \ldots \infty)$, which is unfamiliar and has no obvious interpretation. Therefore we develop below an interpretation in terms of a coefficient of non-determination, r_{MI}, which quantifies how well the best *non-linear* model can describe the data. To this aim, we generalize the above one-dimensional linear case, for which the Pearson coefficient r directly allows this interpretation (Eq. (3.2)). In particular, we suggest to relate $I\left[\mathbf{x}_i, \mathbf{x}_j\right]$ to a more intuitive Pearson-like coefficient $r_{\text{MI}}\left[\mathbf{x}_i, \mathbf{x}_j\right]$ such that also in multidimensional and for non-linear fit-functions f, the connection to the *coefficient of non-determination* holds, i.e.,

$$1 - r_{\text{MI}}\left[\mathbf{x}_i, \mathbf{x}_j\right]^2 = \frac{\left\langle (\mathbf{x}_j - f(\mathbf{x}_i))^2 \right\rangle}{\left\langle \mathbf{x}_j^2 \right\rangle}. \tag{3.7}$$

For fully correlated motions, this *generalized correlation coefficient* r_{MI} equals 1 and vanishes for fully uncorrelated motion.

To this end we exploit that in the special case of Gaussian distributions ($d = 1$) or co-linear Gaussian distributions of unit variance ($d = 3$) the Pearson correlation coefficient ($r = \langle \mathbf{x}_i \cdot \mathbf{x}_j \rangle = \langle |\mathbf{x}_i| \, |\mathbf{x}_j| \rangle$) captures all correlations. For this special case, one derives a one-to-one relationship between MI and the value of the Pearson correlation,

$$I_{\mathrm{Gauss}} \left[\mathbf{x}_i, \mathbf{x}_j \right] = -\frac{d}{2} \ln(1 - r^2). \tag{3.8}$$

Starting from this relationship we define the *generalized correlation coefficient*, r_{MI}, as the Pearson coefficient of such a multi-dimensional Gaussian distribution, whose mutual information equals the one we wish to interpret. From Eq. (3.8),

$$r_{\mathrm{MI}} \left[\mathbf{x}_i, \mathbf{x}_j \right] = \left(1 - e^{-2I[\mathbf{x}_i, \mathbf{x}_j]/d} \right)^{-\frac{1}{2}}, \tag{3.9}$$

which, as it is derived from the mutual information, contains all correlations. Therefore, for vectors of unit variance, $r_{\mathrm{MI}} \left[\mathbf{x}_i, \mathbf{x}_j \right]$ is always larger than $r \left[\mathbf{x}_i, \mathbf{x}_j \right]$. For multi-variant cases, this rule may be violated due to the inconsistent scaling properties of r, which is repaired by r_{MI}. Note that the Gaussian distribution used to define r_{MI} will generally have a larger covariance than the original distribution, because Gaussians have the highest covariance compared to all possible distributions with the same MI[143].

We now turn to numerically estimating the mutual information from a given ensemble or molecular dynamics trajectory. For high-dimensional variables, crude approximations, such as cumulant expansions, are available[143]. For the correlation analysis of macro-molecular dynamics, however, and in particular for the assessment of the correlated motion of atom pairs, density estimates for six-dimensional subspaces suffice. Approaches resting on k-nearest neighbor distances[144] or kernel density estimators[145] have proven to provide sufficiently accurate results for this purpose. The required accuracy is indeed very high, particularly for small correlations, for which the entropies involved nearly cancel out, hence small errors of the relatively large entropy terms lead to large errors in the estimated mutual information. This problem is aggravated due to the large slope of the transformation Eq. (3.9), in the low-correlation regime, which further amplifies errors considerably. These strict accuracy requirements hold also for many other applications of the concept of mutual information, which recently instigated many developments[146, 147, 148, 149, 150, 144, 145].

3.1.3 Linear mutual information

The quite general and rigorous framework of Mutual Information also serves to single out non-linear contributions to correlations. To this aim recall that the Pearson coefficient suffers from two flaws, its inability to detect non-linear correlations and its unwanted dependency on the relative orientation of the displacements. Thus, to separate the former from the latter, a reference quantity is required that suffers only from one of the two flaws. The linear mutual information defined below serves this purpose. It has the additional advantage that its calculation does not require highly accurate and computationally demanding density estimates. Rather it rests on the same Gaussian approximation implied by the computationally much more efficient calculation of the covariance matrix, namely

$$g(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{(2\pi)^d \det\left(\mathbf{C}_{(ij)}\right)} \exp\left(-\frac{1}{2}\left(\mathbf{x}_i, \mathbf{x}_j\right) \mathbf{C}_{(ij)}^{-1} \left(\mathbf{x}_i, \mathbf{x}_j\right)^{\mathrm{T}}\right),$$

with the pair-covariance matrix $\mathbf{C}_{(ij)} = \left\langle \left(\mathbf{x}_i, \mathbf{x}_j\right)^{\mathrm{T}} \left(\mathbf{x}_i, \mathbf{x}_j\right) \right\rangle$. This gaussian is the harmonic approximation to the canonical density of atomic motion. Thus, the mutual information, which can be computed analytical from this approximation, contains only linear correlations. The marginal probabilities are computed accordingly, using marginal-covariances $\mathbf{C}_{(i)} = \langle \mathbf{x}_i^{\mathrm{T}} \mathbf{x}_i \rangle$. In contrast to the (general) mutual information, here, the required entropies are obtained analytically from the Gaussian density approximations, i.e., from the covariance matrices,

$$H\left(\mathbf{x}_i, \mathbf{x}_j\right) = \frac{1}{2}\left[2d\left(1 + \ln\, 2\pi\right) + \ln\, \det \mathbf{C}_{(ij)}\right].$$

From Eq. (4.1), the linear mutual information,

$$I_{\mathrm{lin}}(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{2}\left[\ln\, \det \mathbf{C}_{(i)} + \ln\, \det \mathbf{C}_{(j)} - \ln\, \det \mathbf{C}_{(ij)}\right], \tag{3.10}$$

is obtained.

Similarly to the interpretation of (general) MI, the coefficient of non-determination for the best multivariate *linear* fit is defined by Eq. (3.7) and (3.9). Linear MI (LMI) is a strict lower bound to MI, because the Gaussian distribution maximizes entropy under the constraints of a given mean and variance[53]. This is consistent with the definition of the generalized correlation coefficient and its interpretation, because the inclusion of non-linear models will generally yield a higher coefficient of determination than restriction to linear models.

3.2 Methods

Computation of Correlation Coefficients from Molecular Dynamics Simulations.

After a 5ns equilibration phase, coordinates every 10ps were used from MD simulations GB1 and T4L. Thus 19500 and 11200 snapshots were used from GB1 and T4L, respectively. Translational and rotational motions were removed by least squares fitting to the C_α-atoms of the respective crystal structures. The average structure $\langle \mathbf{r} \rangle$ was subtracted from the coordinates $\mathbf{r}$ to obtain centered atomic displacements $\mathbf{x}$. Correlations between displacements of the C_α-atoms were quantified by Pearson coefficients, Eq. (3.1), by linearized mutual information, Eq. (3.10), and by mutual information. For the latter, the density estimator by Kraskov et al.[144] was used with nearest neighbor parameter

$k = 6$.

3.3 Correlated motion in Protein G

We first compare both correlation measures, the Pearson coefficient, Eq. (3.1), and the generalized correlation coefficient, r_{MI}, Eq. (3.9), for the B1 domain of Protein G (Fig. 3.2a). As expected, all correlations detected by the Pearson coefficient are also seen with the generalized correlation coefficient. Many additional correlations are revealed by r_{MI}, however, which are not revealed by the Pearson coefficient. Furthermore, as will be analyzed in detail below, the purely geometrical (orientational) perturbation of the Pearson coefficient creates patterns in the Pearson matrix which actually are unrelated to any correlation and in this sense artificial.

Correlations detected by both methods are found along the diagonal and in two bands perpendicular to the main diagonal. The latter are due to the hydrogen bonded contacts between different strands of the four-stranded β-sheet. The correlations between strands β_1-β_2 and β_3-β_4 are pronounced, whereas the correlations between hydrogen bonding partners of the central neighbors β_1-β_4, showing up as band parallel to the diagonal, are weaker.

The broad region of high correlation along the main diagonal between residues 22 and 38 is caused by the close packing of residues in the α-helix. The correlation between hydrogen bonded residues in the helix is slightly weaker than correlation between opposing C_α-atoms in β-sheets. The reason for this is that in β-sheets, both neighbors of the C_α-atom are tightly hydrogen-bond coupled to one residue of the parallel strand, whereas in the helix the two neighbors couple to two different residues in opposite direction.

New, so far undetected correlations, are seen in the generalized correlation matrix. These include — less pronounced, but significant — correlations between the α-helix and the first double strand of the β-sheet (β_1, β_2), which are absent for the second double strand (β_3, β_4). This finding can also be explained in terms of geometrical proximity. The helix of GB1 traverses diagonally one half of the β-sheet; starting above residues 50 and 1 of strand β_4 and β_1, respectively, it extends outwards ending near residue 13 of β_2 (cf. Fig. 3.2c). Therefore, the larger part of the helix is located far from strands (β_3, β_4) and closely to strands (β_1, β_2), yielding correlations with the latter only, whereas the residues in the preceding loop and the adjacent part of the helix are close enough to (β_3, β_4) to also cause correlations with these strands.

In summary, the largest correlated motions observed in GB1 are rather due to geometrical proximity than due to collective conformational motion, and, are in this sense, trivial. These large correlations are, not surprisingly, captured by both measures, Pearson coefficient and MI. However, while the Pearson coefficient focuses on the correlation inside secondary structure elements, mutual information reveals many new and non-trivial medium strong correlations between different secondary structure elements.

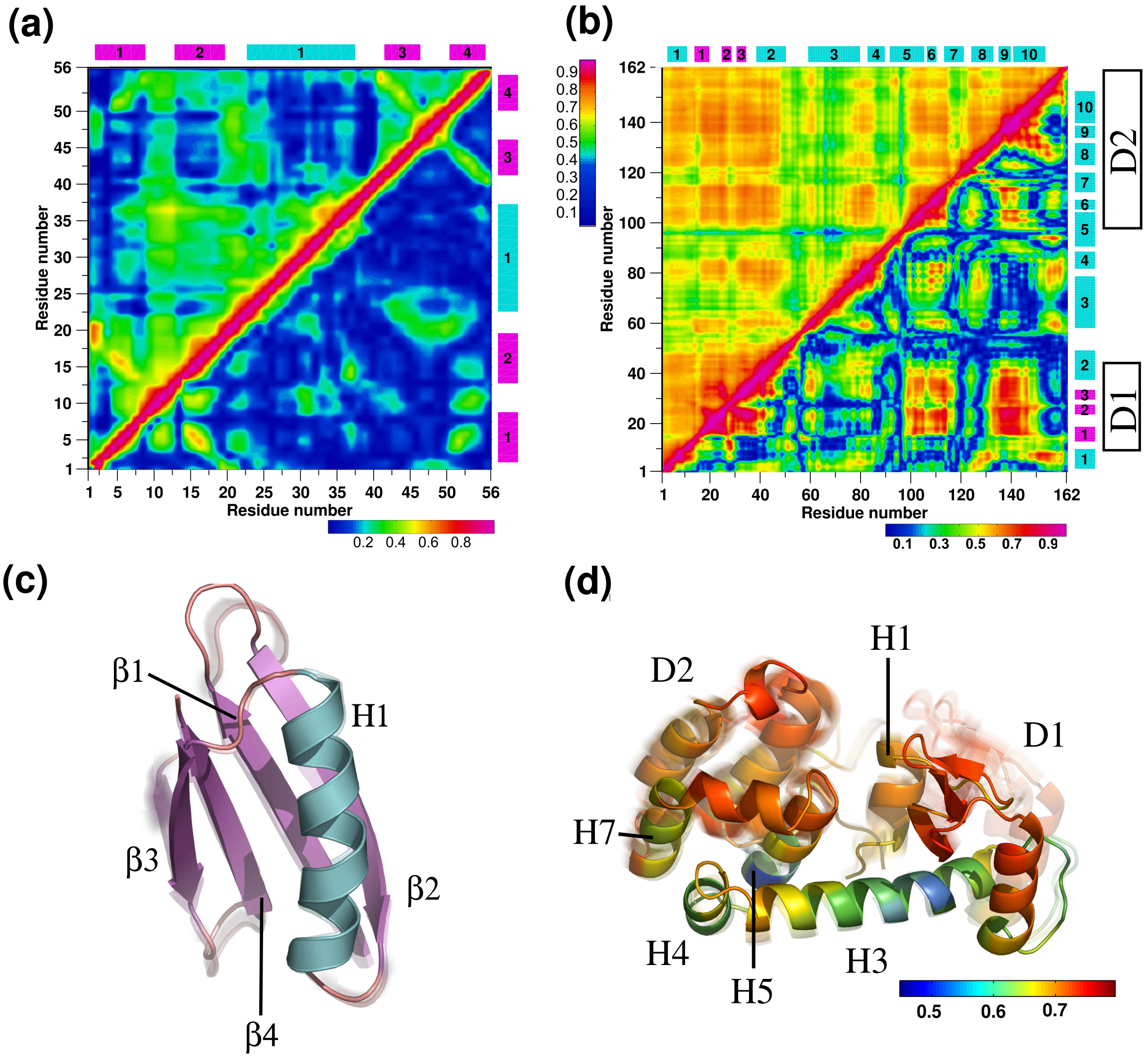


Figure 3.2: **(a,b)** Generalized correlation coefficient r_{MI} (upper left triangle) and Pearson coefficient $|r|$ (lower right triangle) correlation matrices for **(a)** the B1 domain of Protein G (GB1) and for **(b)** T4 lysozyme (T4L)**.** The strength of the computed correlation between two respective residues is color-coded, see color bars; note that different color mappings are used to enhance contrast. Secondary structure elements are indicated by bars in magenta (β-sheets) and cyan (α-helices). **(c,d)** Structure and superimposed three frames from the Protein G (GB1) and lysozyme (T4L) trajectory, respectively, indicating the amplitude of the observed motion. **(d)** For every residue the mean correlation with residues of the two domains D1 (15-46) and D2 (100-160) was computed and color-coded.

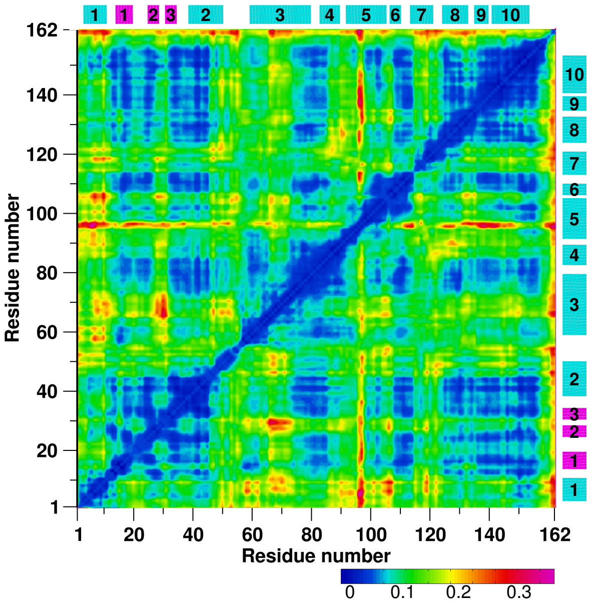


Figure 3.3: Matrix of purely non-linear contributions $r_{\mathrm{MI}} - r_{\mathrm{LMI}}$ to the correlations between atom pairs n T4 lysozyme, see text.

3.4 Correlated motion in Lysozyme

The single protein domain GB1 characterized above is intrinsically rigid. Now we turn to T4 lysozyme (T4L), which exhibits two well-separated domains and significant conformational inter-domain motions[151, 152]. Experimental and theoretical studies have shown that these domain motions are essential for the function of this enzyme, allowing the substrate to enter and the products to leave the active site[153, 154, 155, 156]. Atomic correlations have been analyzed extensively for lysozyme using the Pearson coefficient matrix (Fig. 3.4b, lower right)[58]. Here we have calculated the mutual information based generalized correlation coefficient matrix and focus on the new features this analysis has revealed.

Figure 3.2b (upper triangle) shows that the mutual information successfully quantifies the highly correlated motion within and between the two domains, D1, residue 13-50, and D2, residue 100-162, of T4L (cf. Fig. 3.2). The second domain (D2) moves as two rigid blocks, formed by H4-H6 and H8-H9 respectively, which are weakly linked by residues 118-121. Interestingly, the less correlated linker residues are part of H7 and not, as one might expect, part of the loop region between helices.

Furthermore, the generalized correlation matrix shows that the inter-domain motion is not just a simple hinge motion[154, 155, 157] with H3 and H4 (residue 62-90) forming the hinge region, as one might expect. In fact, a typical hinge motions would imply smaller correlations for the hinge, as indeed found for residues 85-98 (part of H4 and H5) and residues 62-74 (part of H3). Instead, part of the hinge regions, namely adjacent parts of H3 and H4 (residues 75-84) correlate strongly with the overall domain motion, which would not be the case for a simple hinge motion.

The N-terminal helix H1, which contains active site residues, moves correlated with both domains

D1 and D2. However, in contrast to these domains, it shows only weak correlation to the aforementioned linker region around H4. These results are consistent with a previously conducted principal component analysis[131], where the conformational motion of T4L could be described by a rigid body closure and twist motion of domains D1 and D2. That study showed rigid co-motion of H1 and D2 for the closure motion, and, for the twist motion, H1 moves with D1. This splitting up of the H1 correlation can now be understood by considering the non-linear contributions to the overall correlation obtained as difference between mutual information and linear mutual information (Fig. 3.3). As can be seen, the correlations between domains D1 and D2 are mostly linear in nature, while the correlation of H1 with both domains has significant non-linear contributions. This explains why the rather non-linear correlation of H1 with the domains was found to be distributed over two *linear* principal modes[131].

In contrast to the complete quantification of correlated motions by the generalized correlation coefficient, the Pearson coefficient picks up only parts of these correlations, and many remain undetected. This can give rise to a rather inconsistent picture, i.e., patterns in the results not reflecting patterns of correlation, which is particular pronounced for T4 lysozyme (cf. Fig. 3.2b, lower triangle). Although the two domains move as relatively rigid units, the Pearson coefficient quantifies the correlations within the domains rather incompletely. While the Pearson coefficient does show correlations within the first part of D1, the correlations between the first part of D1 and its last 10 residues (40-50) seen by the generalized correlation coefficient are missing. Moreover, most correlations within domain D2 are undetected. A particularly striking and obvious inconsistency would be the violation of transitivity, i.e., two regions between which no correlated motions are detected, but which are both correlated to a third one. Such a situation is indeed purported by the Pearson correlation coefficient, which indicates a high correlation of D1 with the two regions, residues 100-118 and residues 130-150 of D2, but misleadingly low correlation between these two regions. Finally, the Pearson coefficient does not detect H1 to be correlated with D1, and detects only a small fraction of the correlations between H1 and residues of D2. In some instances the Pearson correlation measure yields higher values than the generalized correlation, which should, in principle, not happen. The two possible reasons, the scaling dependency of the Pearson measure or numerical inaccuracies in the estimation of mutual information, are discussed further below.

Thus, the proposed generalized correlation coefficient based on mutual information yields a much more complete picture of the correlated motions which is consistent with — and extends — previously applied principal component analysis[131]. In particular, whereas the Pearson correlation coefficient captures most of the correlated motions within the B1 domain of Protein G, it misses many pronounced correlated motions of lysozyme, which involve all active site residues and are likely to be functionally important. The nature of this failure and the question under which conditions it is to be expected, deserves closer inspection.

3.5 Analysis of the failures of the Pearson coefficient

Figure 3.4 compares as scatter plots all elements of the generalized correlation coefficient matrices with the respective elements of the Pearson correlation matrices, both shown in Fig. 3.2a,b. Results are shown for both GB1 and T4L (Fig. 3.4a,b). For large correlations ($r_{\text{MI}} \geq 0.8$) the Pearson coefficient r and the generalized correlation coefficient r_{MI} give comparable results. For less correlated motions ($r_{\text{MI}} \leq 0.8$), the Pearson coefficient rarely captures the full correlation, and often underestimates r_{MI} considerably, yielding any value between zero and r_{MI}. In fact, as quantified by the average underestimation $\frac{1}{N}\sum_{ij}|r_{ij}|/r_{\text{MI}}^{ij} = 0.48$, only less than half of the correlations are revealed by the Pearson coefficient. Below we analyze the causes for the erratic occurrences of their drastic underestimation.

As discussed in Methods, possible causes are a) the dependence on the relative orientation of the displacements, b) the presence of non-linear correlations and c) lack of scaling invariance. We will demonstrate below that the dependence on direction is in fact the main cause of the underestimation enhanced by the presence of non-linear correlations.

We start the analysis by separating the effect of non-linear correlations from the purely linear contributions. To this end, Figure **??** compares the generalized correlation coefficient discussed above with the corresponding coefficient based on *linear* mutual information (see Methods). As can be seen, both agree well for GB1 except for numerical inaccuracies within the low-correlation regime. In contrast, clear deviations for lysozyme point towards significant non-linear correlations. Indeed as qualified by the histogram of deviation (inset) or quantified by $\frac{1}{N}\sum_{ij}(r_{\text{MI}} - r_{\text{LMI}})/r_{\text{MI}}^{ij} = 0.09$, the non-linear part of the correlation contributes up to 10% to the overall correlation and, therefore, accounts for a significant part of the correlation not described by the Pearson coefficient. (cf. crosses in Fig. 3.4). Since both, r_{LMI} and r, rely on the linear quasi-harmonic approximation, the remaining approximately 40% of the undetected correlations — in fact the largest part — cannot be explained by non-linear effects.

To quantify the (geometrical) effect of relative orientation of the atomic displacements on the Pearson coefficient, the latter was separated into correlations of distances,

$$r_{\text{abs}}\left[\mathbf{x}_i, \mathbf{x}_j\right] = \langle|\mathbf{x}_i| \cdot |\mathbf{x}_j|\rangle \,/\, \left(\langle\mathbf{x}_i\rangle \langle\mathbf{x}_j\rangle^2\right)^{1/2}, \tag{3.11}$$

and average co-linearity

$$r_{\text{dir}}\left[\mathbf{x}_i, \mathbf{x}_j\right] = \left\langle\left|\frac{\mathbf{x}_i}{|\mathbf{x}_i|} \cdot \frac{\mathbf{x}_j}{|\mathbf{x}_j|}\right|\right\rangle. \tag{3.12}$$

Figure 3.5 compares the correlations of distances, r_{abs}, with both, the linear generalized correlation coefficient r_{LMI} (black) as well as the Pearson coefficient r (red). As can be seen r_{abs} is more closely linked to r_{LMI} than to the Pearson coefficient, as is also quantified by correlation coefficients of 0.88 vs. 0.64, respectively. In contrast, the average co-linearity is more linked to the Pearson coefficient (correlation coefficients 0.47 vs. 0.87, respectively, data not shown), thus confirming that the relative orientation of the atomic displacements perturbs the Pearson coefficient considerably. Indeed,

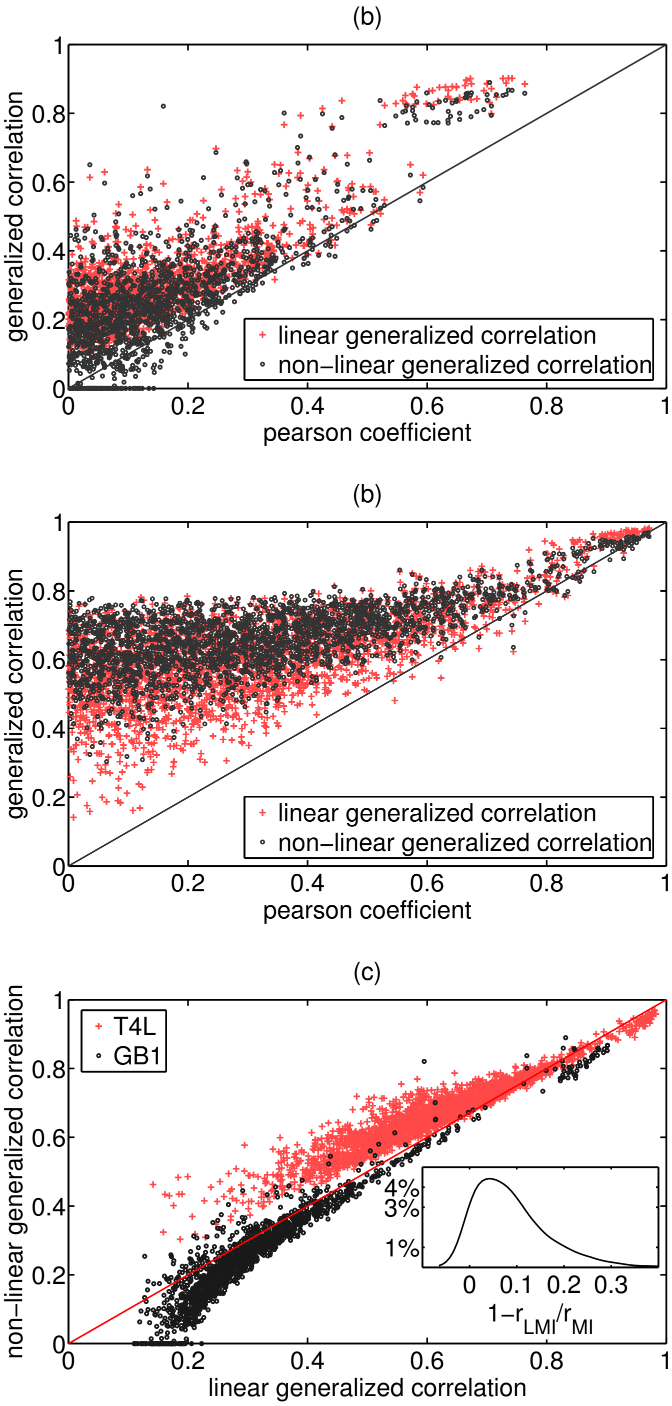


Figure 3.4: Comparison of mutual information based correlation measures with Pearson correlation coefficients. For pairs of C_α−atoms of **(a)** GB1 and **(b)** T4L, both generalized correlation coefficients r_{MI} (dark gray circles) and r_{LMI} (red crosses) are plotted against the Pearson correlation coefficient. **(c)** Comparison between linear and non-linear mutual information. For GB1 (black) and T4L (gray) the generalized correlation coefficients computed from linear mutual information are plotted against non-linear generalized correlation. For T4L, the inset shows a histogram of the differences between both coefficients, with maximum of the distribution at 0.04 and a mean of 0.09.

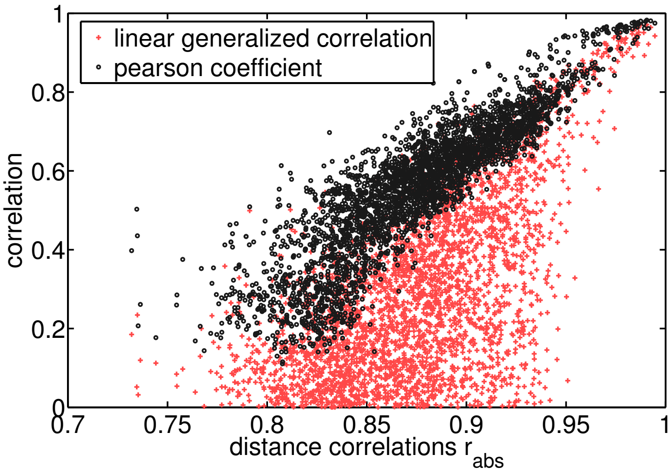


Figure 3.5: Distance correlations r_{abs} compared to the two linear correlation measures r_{LMI}(black) and Pearson coefficient (red).

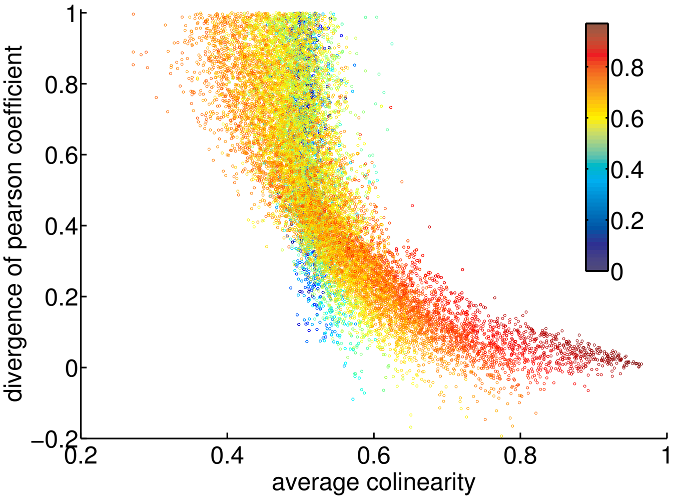


Figure 3.6: Relative (linear) divergence of the Pearson coefficient $\Delta |r| = 1 - |r| / r_{\mathrm{LMI}}$ as a function of the average co-linearity r_{dir}. The colors quantify full (non-linear) correlation r_{MI}.

as shown in Fig. 3.6, the average orientation is closely linked to the divergence of the Pearson coefficient from the generalized correlation, quantified by a correlation coefficient of -0.78. Knowledge of the relative orientations of the displacements alone, therefore, allows to predict when the Pearson coefficient will fail to detect correlations. For high co-linearity, the Pearson coefficient quantifies the correlation relatively well, whereas it systematically underestimates the correlation in cases where the displacements are nearly perpendicular to each other.

Interestingly, additional consideration of the generalized correlation coefficient in Figure 3.6 (color-coded) shows that the very high correlation in lysozyme coincides exclusively with co-linear motions — in which case the Pearson coefficient performs quite well. This is explained by the fact that, for the case of protein dynamics, these high correlations can only arise from atoms confined within secondary structure elements. That only medium strong correlations are missed by the Pearson coefficient is, therefore, rather due to the specific properties of protein dynamics and not a merit of the Pearson coefficient. Similarly, the high performance of the Pearson coefficient at high correlations may hold only for dynamics and is not a general property of the Pearson coefficient.

The only defect in the definition of the Pearson coefficient not discussed so far is its lack of scaling invariance. Closer inspection of Figure 3.4 reveals atom pairs for which the Pearson coefficient is slightly higher than the r_{LMI}. Since both measures are based on the same linear, i.e., harmonic approximation, this cannot be explained with numerical estimation inaccuracies. We suggest as the likely cause the improper scaling behavior, which yields an overestimation of the Pearson coefficient. However, since the effect is small, we were not able to separate it from the other effects discussed above, so that this hypothesis could not be proven.

We finally discuss the numerical inaccuracies mentioned above and described in Methods. For certain atom pairs, the non-linear correlation is actually lower than the linear correlation (Fig. **??**), which should not occur since linear mutual information is a strict lower bound to non-linear mutual information (see Methods). However, here the mutual information is estimated from a finite number of frames, which implies statistical inaccuracies. Because mutual information is a difference of relatively large entropies the relative error increases for small correlations, which explains the deviations seen in Fig. **??** for low mutual information $r_{\mathrm{LMI}} < 0.3$. At higher correlations ($r_{\mathrm{MI}} \gtrsim 0.7$) a small systematic underestimation of MI is observed, as discussed in [144]. Taken together, accuracy can be enhanced by using the larger value of both, the (analytical) linear and the (numerical) non-linear mutual information.

3.6 Conclusions

We have derived a generalized correlation measure based on mutual information, which allows for complete characterization and quantification of atomic correlations in proteins and other macromolecular motion. It provides a consistent framework for analyzing correlations between coordinates, atoms, and groups of atoms, and thereby overcomes the problems of the usually employed Pearson correlation coefficient.

Firstly, both linear and non-linear contributions to correlation are accounted for. Moreover, a linearized generalized correlation coefficient was derived within the framework of mutual information which allowed separation of linear and non-linear contributions to correlation. For T4 lysozyme the latter account for roughly 10% of all correlations. Secondly, our generalized correlation coefficient does not suffer from the artifacts of the established method which originate from the relative orientation of the atomic displacements. This purely geometrical artifact of the Pearson coefficient typically leads to underestimation of the correlations by more than 40%. Taken together, more than 50% of the correlations remain undetected by the established method, but are fully accounted for by the generalized correlations coefficient.

Application to two proteins, the B1 domain of Protein G and coliphage T4 lysozyme, revealed new information on their functionally relevant collective dynamics. In particular for lysozyme, the established characterization of the domain motion in terms of a hinge motion has been extended towards a more complex pattern of collective motions. This pattern is not revealed by the conventional

Pearson coefficient matrix, which, in addition, conveyed misleading information.

The enhanced characterization of the collective motion provided by the generalized correlation matrix also complements the analysis of collective motions with principal component analysis (PCA). For example the assessment of non-linear correlations presented here, can explain the previous finding by PCA that for T4 lysozyme the helix H1 moves either rigidly together with domain D1, as shown by the first principal component, or, for the second principle component, H1 moves together with domain D2[131].

Overall, particularly many inter-domain motions were revealed by the generalized correlation coefficient. In contrast the Pearson correlation coefficient turned out to focus at the local correlations, which often are due to spatial proximity within secondary structure elements and in this sense virtually trivial. Particularly the inter-domain motions, however, tend to exhibit non-linear correlations, which can now be captured by the generalized correlation.

We note that the presented definition of mutual information can be generalized to higher dimensions. Accordingly, correlations between groups of atoms can also be quantified, e.g., between a ligand and selected residues of its binding pocket. To this aim, the application of linearized mutual information is straightforward. For the non-linear mutual information, the numerical estimation used here may become inaccurate for larger numbers of atoms per group. In this case, parametric entropy estimators will be superior[53].

The generalized correlation coefficient developed in this chapter is widely applicable to the exponentially growing amount of configurational ensembles provided by molecular dynamics simulations and from other sources such as NMR or Concoord[158]. This method will thus allow for the detection and characterization of a large number of new functionally important protein motions. Moreover, it facilitates direct comparison with experimental data, e.g., from X-ray diffusive scattering, NMR, or, via entropy, from calorimetry. In Chapter 5 we use it to compare the discussed correlated motions of the B1 domain of Protein G to measurements obtained with a new NMR method, which attempts to probe these correlations.

Discontent is the first necessity of progress.
—Thomas A. Edison

Chapter 4
Full Correlation Analysis

For Collective Langevin Dynamics (CLD) we require suitable slow collective coordinates, which enable us to describe the functionally relevant motions with a drastically reduced number of degrees of freedom. We aim to extract these coordinates from MD trajectories, which is not a trivial problem[51]. The two most widely used methods to determine these motions are normal mode analysis (NMA)[78, 79, 80] and principal component analysis (PCA) (cf. Chapter 2). We have shown that the latter method yields modes which are suitable candidates for a CLD approach.

However, it is not entirely clear which criterion one should apply to identify functionally relevant motions. Since many functional processes involve large and slow conformational changes (as opposed to small-amplitude fast thermal vibrations), one reasonable approach is to select those collective degrees of freedom which contribute most to the total atomic displacements seen in the trajectory. This is achieved by PCA. A different approach, motivated by the desire to simplify the treatment by separation into weakly coupled modes, is NMA. For small molecules, this approach reliably predicts infrared vibrational spectra, and it has also been successfully applied to calculate high frequency vibrational spectra of proteins[159]. However, it is unclear to what extent a harmonic approximation to a single minimum of the potential energy surface can characterize the functional motion on the complex frustrated multi-minima energy landscape of proteins[160].

This problem is partially circumvented by PCA, which rests on the covariance matrix of atomic displacements rather than on the Hessian matrix. Accordingly, PCA yields a multivariate Gaussian approximation to the canonical ensemble of the system, such that one can reinterpret principal components as (uncoupled) normal modes in an effective harmonic free energy surface[81, 161]. Due to this statistical mechanics approach to extract collective coordinates from MD simulations, PCA captures also motions that result from visits of multiple minima, which is a major advantage of PCA over NMA for applications to proteins.

Unexpectedly at first sight, seeking those collective modes which accumulate the largest atomic displacements, PCA is equivalent to diagonalizing the covariance matrix (cf. Sec. 2.1). Thus, PCA identifies exactly those collective modes whose covariances vanish. However, because the covariance matrix measures only *linear* correlations, *non-linear* correlations between the PCA modes can — and often do — persist, as already pointed out by Amadei et al.[50].

Here we present a new method, Full Correlation Analysis (FCA), to minimize *all* correlations between the collective degrees of freedom. Avoiding harmonic and linear approximations altogether, we combine the advantage of PCA to use a statistical mechanics approach with the original objective of NMA to yield uncoupled collective coordinates. The coupling between coordinates is quantified by mutual information (MI) and subsequently minimized. This measure of correlation, which is singled out by its information theoretical background (cf. Sec. 3.1.2), quantifies *any* correlation — in particular, non-linear correlations and multi-coordinate correlations. In Chapter 3, MI was successfully applied to quantify the correlation in the motion of *pairs* of protein atoms from MD simulations. Here, we carry this idea further to full generality and minimize the MI of the whole system to yield maximally uncoupled collective coordinates. This is achieved by selecting from all possible rotations of configurational space the transformation with lowest MI.

For the implementation of FCA we adopt an efficient algorithm to minimize MI from the signal processing field. There, independent sources from mixed signals are extracted, e.g., by blind source separation (BSS)[162] or independent component analysis (ICA)[141, 163]. The algorithms developed for these methods differ in three main aspects. First, the estimation of MI can be either cumulant based, parametric (e.g., FastICA[164]), or un-parametric (e.g., MILCA[165]). Second, for the minimization of MI, diverse methods like stochastic descent, gradient descent or a direct solution of the normal equations (e.g., FastICA) have been applied. Third, the resulting coordinates, can be linear or nonlinear, e.g., MISEP[166]. Combining and selecting suitable features out of these existing algorithms, we develop an algorithm tailored for the special requirements of FCA on biomolecular dynamics. As first step we consider linear collective coordinates.

The chapter is organized as follows. In Section 4.1 we develop the minimization algorithm. In the Section 4.3 its capability to extract uncoupled coordinates of a test-system with a known solution is shown. Subsequently, FCA is applied to the 117ns MD trajectory of a T4 bacteriophage lysozyme, which was already used in the previous chapter to analyze correlations of the atomic motions. Additionally, FCA is applied to a 100ns trajectory of the hexapeptide neurotensin (cf. Sec. 4.4). In the latter section, we compare, at first, the resolution of conformational subspaces of T4 lysozyme gained by FCA and PCA. Then, we obtain free energy surfaces for two-dimensional subspaces spanned by either FCA or PCA modes of neurotensin, and compare their ability to accurately describe the conformational transitions of the peptide. Subsequently, we quantitatively analyze the differences of amplitude, collectivity and anharmonicity of FCA and PCA modes, and assess the remaining coupling between pairs of modes. Finally, the convergence of FCA modes is elucidated by applying FCA to a multi-dimensional random walk.

4.1 An Algorithm for Full Correlation Analysis (FCA)

4.1.1 Minimization of mutual information

As shown in Sec. 3.1.2, *any* correlation between atomic displacements $\mathbf{x} = (x_1, x_2, \ldots, x_{3N})$, i.e., also *non-linear* and *multi-coordinate* contributions, can be quantified via the well-known (Shannon) mutual information (MI) (cf. Eq. (3.5))

$$I\left[x_1, x_2, \ldots, x_{3N}\right] = \sum_{i=1}^{3N} H[x_i] - H[\mathbf{x}], \tag{4.1}$$

where $H[\mathbf{x}] = -\int p(\mathbf{x}) \log p(\mathbf{x}) d\mathbf{x}$ denotes the entropy.

In the following we exploit this property by minimizing Eq. (4.1). In order to find those collective modes for which Eq. (4.1) is minimized, we look for an orthogonal coordinate transformation $\mathbf{R}$ of the cartesian displacement vector $\mathbf{x}$, such that the resulting coordinates

$$\mathbf{s}(t) = \mathbf{R}\mathbf{x}(t), \tag{4.2}$$

$\mathbf{s}(t) = \left(s_1(t), s_2(t), \ldots, s_{3N}(t)\right)$, minimize $I\left[s_1, s_2, \ldots, s_{3N}\right]$.

This overall rotation $\mathbf{R}$ is gained iteratively by carrying out a sequence of rotations which respectively act on two coordinates x_i and x_j, i.e., $\mathbf{R} = \prod_{k=1}^{N(N-1)/2} \mathbf{R}_{i_k j_k}(\varphi_k)$, where

$$\mathbf{R}_{ij}(\varphi) \cdot \left(x_1, \ldots, x_i, \ldots, x_j, \ldots, x_N\right)^{\mathrm{T}} = \left(x_1, \ldots, \tilde{x}_i, \ldots, \tilde{x}_j, \ldots, x_N\right)^{\mathrm{T}}, \tag{4.3}$$

with

$$\tilde{x}_i = \cos\varphi x_i + \sin\varphi x_j, \qquad \tilde{x}_j = -\sin\varphi x_i + \cos\varphi x_j.$$

For such a rotation the change in MI is given by

$$\Delta_I(\varphi) = I\left[\mathbf{R}_{ij}(\varphi)\mathbf{x}\right] - I\left[\mathbf{x}\right] = H\left[\tilde{x}_i\right] + H\left[\tilde{x}_j\right] - H\left[x_i\right] - H\left[x_j\right]. \tag{4.4}$$

To find in a specific rotational plane the global minimum of $\Delta_I(\varphi)$ the angle φ is optimized in two steps. At first the whole interval $\left[0, \frac{\pi}{2}\right]$ is sampled coarsely at 10 rotation angles $\{\varphi_l\}_{l=1\ldots10}$ and subsequently minimization is refined by the MATLAB$^{\mathrm{TM}}$-function `fminbnd` on interval $[\varphi_k - \Delta\varphi, \varphi_k + \Delta\varphi]$, where $\varphi_k = \min\{\varphi_l\}_{l=1\ldots10}$ and $\Delta\varphi$ the step-size. `fminbnd` uses a combination of golden section search and parabolic interpolation to achieve this goal[167]. Note that during the iteration process to find the optimal overall rotation $\mathbf{R}$ the same rotation plane is generally minimized several times, because its optimum changes as soon as one of its coordinates is rotated during a minimization of a different plane.

The algorithm can be summarized by the following steps:

(i) preprocessing (PCA is used to find an initial guess)

(ii) select rotation plane heuristically

(iii) find angle φ to minimize $\Delta_I(\varphi)$

(iv) repeat from (ii) until no further minimization is possible

4.1.2 An efficient selection of rotation planes

In applications to high dimensional protein ensembles the search space of the minimization algorithm becomes rather large. To cut down computational costs, the number of visited planes is kept small by applying heuristics, which select primarily the promising planes for minimization of $\Delta_I(\varphi)$.

In particular, at first those rotational planes (i, j) are selected which feature a high pairwise correlation $I_{ij} = I[\mathbf{x}_i, \mathbf{x}_j]$, since for these a relatively high loss of mutual information upon minimization is expected. Furthermore, unnecessary re-evaluation of already visited planes are avoided, by using a marker m_{ij}, which is initialized with one, and set to zero after minimization in the ij-plane. Because rotation in the ij-plane increases the likelihood that an already marked plane (ik) or (jk), $k \neq i, j$, allows further optimization, all respective markers are increased by $|\varphi|$, scheduling these planes for re-evaluation.

Taken together, planes are evaluated in the decreasing sequence $m_{i_1j_1}I_{i_1j_1} > m_{i_2j_2}I_{i_2j_2} > \ldots$, until 4 rotations with $|\rho| > 0.01$ have been found. Then the correlations r_{ij} are re-computed and a new succession $m_{i_1j_1}I_{i_1j_1} > m_{i_2j_2}I_{i_2j_2} > \ldots$ is devised.

Because only those pairwise correlations I_{ij} need to be updated, where coordinates changed substantially, a second book-keeping matrix o_{ij} was used to track these changes. o_{ij} is set to zero after computation of I_{ij} and increased by $|\varphi|$ if the rotation-angle corresponds to a (ik) or (jk) plane. As soon as $o_{ij} > 0.3$ the respective correlation I_{ij} is computed again.

The algorithm is terminated if all $m_{ij} \leq 0.01$.

4.1.3 Estimation of mutual information

The FCA algorithm described in Sec. 4.1.1 depends upon numerical estimates of entropies $H[x_i]$. Furthermore, the heuristical selection of rotational planes (cf. Sec. 4.1.2) requires an explicit computation of pair-wise mutual information $I[x_i, x_j]$, and therefore estimates of $H[x_i, x_j]$. Thus, densities $p_i(x_i)$ and $p_{ij}(x_i, x_j)$ of one or two dimensional distributions, respectively, have to be estimated.

On the one hand, FCA requires a computational efficient estimator, since a high number of evaluations are performed during the iterative minimization procedure. On the other hand, also the required accuracy is high, because small absolute errors of estimated entropies lead to large relative errors of their difference, i.e., the estimated MI. Nonetheless, in application of FCA to MD ensembles statistics are usually very good, i.e, usually ensembles with > 10000 structures are generated. Thus, sophisticated estimators, e.g., spacing estimates[150], k-nearest neighbor methods, or kernel density estimators[147] are not required, but would slow-down the computation considerably.

Instead, we chose a fast kernel-smoothed histogram estimation and will check its accuracy in Sec. 4.3.1 against an estimator based on a k-nearest neighbor approach.

The details of the estimation were as follows. The entropy $H\left[x_i\right]$ of a one dimensional ensemble $\{x_i(t_k)\}_{k=1\dots M}$ was estimated by counting occupations, n_b, of $b = 1 \dots L_{\mathrm{1D}}$ bins, with $L_{\mathrm{1D}} = 200$. The histogram was smoothed by convolution $p_b = \sum_{k=-m}^{m} n_{b+k} g_k / M$ with a discrete Gaussian function $g_l = \left(2\pi\sigma^2\right)^{-1/2} \exp\left(-(l\Delta x)^2/\sigma^2\right)$, with $\sigma = \Delta x$ the binning width, evaluated at points $l = -m \dots m$ with $m = 3$. From this the entropy of coordinate x_i was computed as $H\left[x_i\right] = -\Delta x \sum_{b=1}^{L} p_b \log p_b$.

Entropies of two dimensional ensembles $H\left[\mathbf{x}_i, \mathbf{x}_j\right]$ were estimated by choosing $L_{\mathrm{2D}} = 100$ bins for every dimension and bandwidths $\sigma_i = 1.8\Delta x_i$ and $\sigma_j = 1.8\Delta x_j$, respectively.

For efficiency reasons we did not implement a sophisticated optimal bandwidth selection as, e.g., in Ref. [168]. A computationally less expensive bandwidth selection scheme[147] was tested, but led to unacceptable inaccuracies for distributions which deviated too much from a Gaussian. Instead, bandwidth is selected by adapting the bin widths Δx_i and Δx_j such that a fixed number of bins (L_{1D} and L_{2D}) is placed between the extremes of a distribution. In this way, a good trade-off between efficiency, accuracy and robustness, has been achieved.

4.2 Methods

4.2.1 Preprocessing of FCA

Before minimization of MI commenced, PCA was applied to the C_α-atoms for the T4L example and to all non-hydrogen atoms of neurotensin, respectively. For efficiency only rotations within the subspace of the first 100 eigenvectors were considered in both cases. This is justified, because the small amplitude PCA modes are sufficiently uncoupled already.

4.2.2 Selection of essential FCA modes

FCA modes were ranked, such that the mode which is most likely to describe functional protein motion has the lowest index. Instead of following PCA to rank the modes by fluctuation amplitude $\langle x \rangle^2$[50], we rank by anharmonicity. The anharmonicity of a collective mode was quantified by its negentropy[163],

$$J\left[x_i\right] = \frac{1}{2}\left[1 + \log\left(2\pi\right) + \log\left(\left\langle x_i^2 \right\rangle\right)\right] - H\left[x_i\right], \tag{4.5}$$

i.e., the difference in the entropies of the observed density and its Gaussian approximation.

4.2.3 Selection of pairs of FCA modes

The subspace of relevant FCA modes is generally more than three dimensional, and thus, difficult to analyze visually. For exploratory data analysis it is, therefore, necessary to project the motion on pairs or triples of FCA modes, as it is usually done with PCA modes[50]. However, the amount of projections to analyze increases quickly, and many projection pairs are redundant. The MI used for FCA offers the advantage to select pairs of modes more targeted. In our experience a selection of modes with highest pairwise correlation is likely to convey the most information. Accordingly, we showed each of the first ten modes together with the highest correlating mode of smaller index.

4.2.4 A test-system for FCA

To test the FCA algorithm we constructed a set of 10 independent modes $\mathbf{s}(t) = (s_1(t), s_2(t), \ldots, s_{10}(t))$. These were mixed by applying a random orthogonal matrix $\mathbf{x} = \mathbf{A}\mathbf{s}$ and were subsequently recovered from the mixture $\mathbf{x}$ by FCA. For comparison we also tried to recover the independent coordinates with PCA.

Modes with non-gaussian distribution, $s_i(t)$ $(i = 1 \ldots 5)$, were obtained from five 300ns trajectories recorded every 10 ps generated with a one-dimensional CLD model of the conformational motion of the peptide neurotensin, which will be devised in Chapter 8.

Additionally, five quasi-harmonic distributions, $s_i(t)$ $(i = 6 \ldots 10)$, were drawn randomly from Gaussian densities of differing widths, i.e., yielding fluctuational amplitudes $\left(\langle x^2 \rangle\right)^{-1/2}$ =1, 0.8, 0.6, 0.4, and 0.2, respectively.

The random orthogonal 10×10 matrix $\mathbf{A}$ was generated by eigenvalue decomposition, i.e., $\mathbf{T}\mathbf{T}^{\mathrm{T}} = \mathbf{A}\Lambda\mathbf{A}^{\mathrm{T}}$, where $\mathbf{T}$ denotes a 10×20 matrix, whose elements are normal distributed random numbers with unit variance, and Λ a diagonal matrix.

From the mixed components $\mathbf{x} = \mathbf{A}\mathbf{s}$ we computed with FCA and PCA, $\mathbf{R}_{\mathrm{FCA}}$ and $\mathbf{R}_{\mathrm{PCA}}$, respectively. The results were validated by computing inner product matrices, $\mathbf{A}^{\mathrm{T}}\mathbf{R}_{\mathrm{FCA}}^{\mathrm{T}}$ and $\mathbf{A}^{\mathrm{T}}\mathbf{R}_{\mathrm{PCA}}^{\mathrm{T}}$, and recovered components, $\tilde{\mathbf{s}}_{\mathrm{FCA}} = \mathbf{R}_{\mathrm{FCA}}\mathbf{x}$ and $\tilde{\mathbf{s}}_{\mathrm{PCA}} = \mathbf{R}_{\mathrm{PCA}}\mathbf{x}$.

4.2.5 Collectivity of modes

We computed the collectivity Ω of a mode from its normed direction vector $\mathbf{d} = (d_1, d_2, \ldots, d_{3N})$. To this end, the squared motional contribution a_i^2 of atom i to mode s, was computed as the sum of the squared entries, which belong to atom i, i.e., $a_i^2 = \sum_{j=1}^{3} d_{3(i-1)+j}^2$. The collectivity was given by the entropy of the distribution of motional contributions. Thus,

$$\Omega(\mathbf{d}) = -\left(\log N\right)^{-1} \sum_{i=1}^{N} a_i^2 \log a_i^2,$$

where the normalization constant $\log N$ was chosen, such that a mode, whose atoms contribute equally to the motion, has a collectivity of one.

4.2.6 Free energy surface for projected motion of neurotensin

Calculation of free energy surfaces $G = \beta^{-1} \log \rho(s_1, s_2)$ of the projection of neurotensin dynamics onto pairs of modes (s_1, s_2) required the density $\rho(s_1, s_2)$ of the projected MD ensemble. It was estimated by smoothing a two dimensional histogram (150 $\times$ 150 bins) with a Gaussian function of widths $\sigma_1 = 3\Delta s_1$ and $\sigma_2 = 3\Delta s_2$, respectively, where Δs denotes the bin width. The superposed

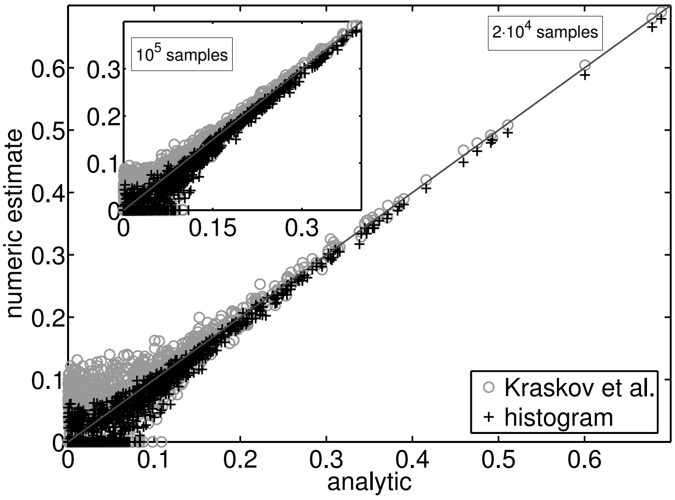


Figure 4.1: Test of estimation of correlation for Gaussian distributed random data sets. r_{MI} estimated with the histogram method (crosses) and r_{MI} estimated with the method of Kraskov et al.[144] (circles) are plotted against analytically computed r_{MI}. The inset shows the same comparison, but with 10^5 sample points used.

trajectories shown in Fig. 4.8 and Fig. 4.9 were obtained by smoothing the projection of the MD trajectory NT1 onto the respective modes with a Gaussian function of width $\sigma = 20\,\mathrm{ps}$.

4.3 Checks of FCA Algorithm

4.3.1 Check of entropy estimation

As described in Sec. 4.1.3, FCA calculates the MI for efficiency reasons with a relatively crude but fast estimator, which is based on histograms. To check the validity of this approach, the estimator was evaluated against the recently devised k-nearest neighbor approach of Kraskov et al., which is unbiased and was found to be more accurate than a number of other methods[144]. To this end, MI estimated from Gaussian distributions with random widths was compared to MI calculated analytically from the same widths. In Figure 4.1 MI estimated with both, the histogram method and Kraskov's method, is plotted against the analytically obtained MI. As seen in the figure, the generalized correlation coefficients estimated with the histogram method are as accurate as those estimated with the alternative method. In particular, in the low correlation regime the histogram estimates deviated less than Kraskov's estimates.

The bandwidth of the histogram estimator is controlled by the number of bins (cf. Sec. 4.1.3). Since the bandwidth optimum usually scales with the number of sample points, we checked the accuracy of our estimator also for the higher boundary of the envisaged range $M = 10^4 \ldots 10^5$ of sample points. Indeed, the inset of Figure 4.1 shows that the chosen number of bins work well also for $M = 10^5$.

So far we have only checked accuracy for Gaussian distributions, whose MI can be obtained analytically. To rule out a significantly lowered accuracy for non-Gaussian distributions, we checked also MIs of distributions obtained from molecular dynamics data of T4 lysozyme using Kraskov's method as a reference. The achieved correlation with the reference (corr. coeff. r=0.98), shows

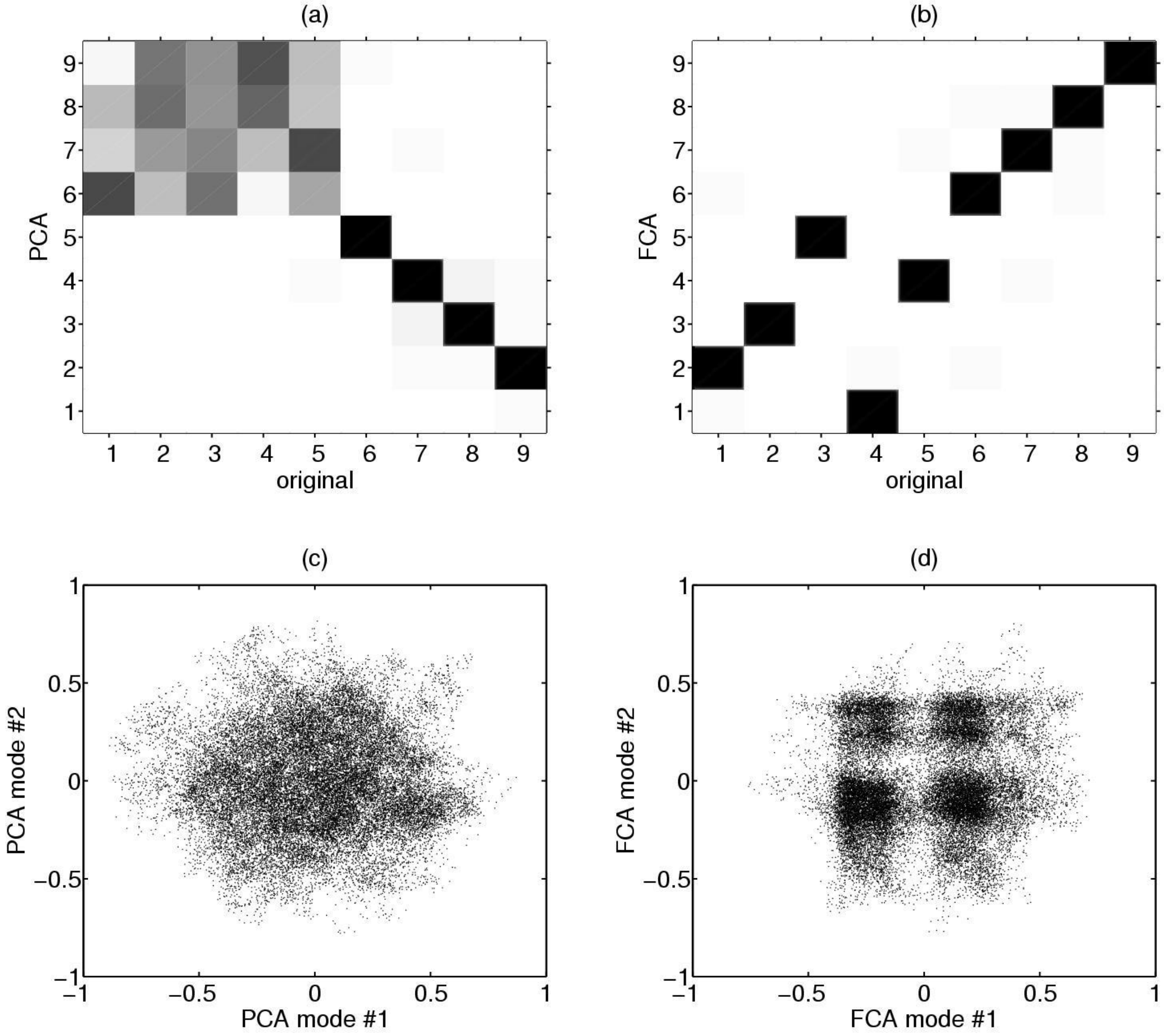


Figure 4.2: **(a-b)** Inner products of PCA (a) and FCA (b) modes, with the directions of the original independent components s_i. The black squares denote entries with a magnitude of near unity. **(c-d)** Projections of the test-ensemble $\mathbf{x}$ onto the first two coordinates of PCA (c) and FCA (d).

that the fast histogram method reaches almost the same accuracy as the computationally much more expensive method, and thus poses a reasonable choice for FCA.

We remark that a recent development[145] might offer increased accuracy without increasing computational costs, but had not yet been available at the time when the presented work was performed.

4.3.2 Application of FCA to test-case with known result

To verify the FCA algorithm, we constructed an example with a known solution, as follows. We started with independent modes, i.e., the solution, and artificially mixed them to generate a mock protein ensemble (cf. Methods). In particular, ten independent modes $s_1(t), s_2(t), \ldots, s_{10}(t)$ were

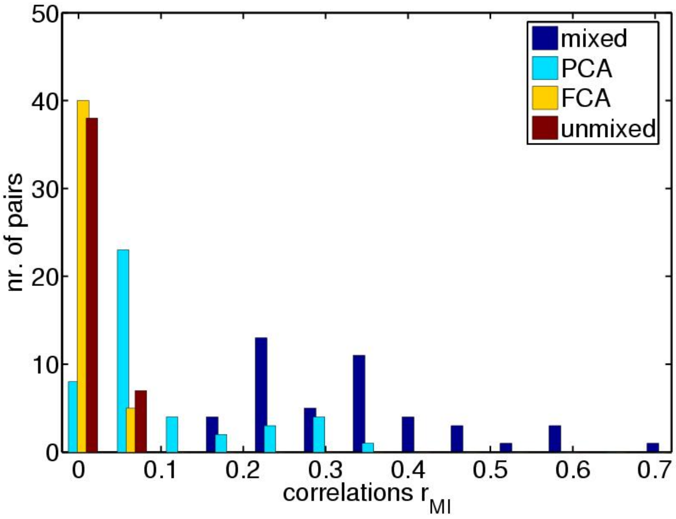


Figure 4.3: Histogram of generalized correlation coefficients between pairs of coordinates. The four histograms count correlations between the s_i (unmixed), the x_i (mixed), and between PCA and FCA modes, respectively.

generated, and FCA was applied to recover them from their mix $\mathbf{x} = \mathbf{A} \cdot (s_1, s_2, \ldots, s_{10})^{\mathrm{T}}$ without any knowledge of the random mixing matrix $\mathbf{A}$. To mimic collective modes of proteins[88], we adopted a one dimensional model of conformational motion (see Methods) to create five pseudo-collective modes $s_1(t), s_2(t), \ldots, s_5(t)$ with a double peaked density, and derived five additional modes $s_6(t), \ldots, s_{10}(t)$ from Gaussian densities with decreasing fluctuational amplitude $\langle s_i^2 \rangle$.

Both, PCA and FCA, were applied to this artificial ensemble. The directions of the original components s_i in the mixed system $\mathbf{x}$ are the columns of $\mathbf{A}$, such that an accurately identified mode would yield an inner product near unity with one of these columns. Fig 4.2a shows the respective inner products for PCA modes. The field of gray boxes in the upper left indicates that PCA was not able to recover the anharmonic modes $s_1, s_2, \ldots, s_5$, whereas the black boxes in the lower right indicate that the quasi-harmonic modes $s_6, \ldots, s_{10}$ were retrieved successfully. On the contrary, all independent components were accurately recovered by FCA, as shown by ten inner products near unity (black boxes, Fig. 4.2b).

As a consequence, the projection to the first FCA modes shown in Fig. 4.2d reveals correctly the peaked structure of the conformational density, whereas this structure is completely blurred in the projection to PCA modes (Fig. 4.2c).

The pairwise correlations, shown as histogram in Fig. 4.3, were drastically reduced between pairs of FCA modes. The small putative correlations remaining between FCA modes are due to the finite number of sample points and statistical inaccuracies in their estimation (cf. Fig. 4.1). Accordingly, these remaining correlations are as high as the estimated correlations between pairs s_i, s_j of the independent modes. As expected, PCA reduced the correlations only incompletely.

Since it is highly unlikely that protein motion can be separated into completely uncoupled modes, we tested whether FCA is able to reverse the mixing also in cases where the known solution contains coupled modes. Indeed, FCA also solved such test examples that were constructed to contain pairs and triples of coupled coordinates $s_i(t)$. (results not shown).

Before applying FCA to selected proteins, we briefly discuss the relation of this FCA algorithm to known algorithms for the methods ICA (and synonymously BSS) from signal processing (cf. Introduction).

The aim of ICA is to recover the underlying independent sources from a recorded multichannel signal of their mixture. Within the context of molecular simulations, the cartesian coordinates represent observation channels, and the collective motions are the putatively independent signals supposed to be recovered. Because the relative amplitudes of different signal sources are meaningless, an ICA algorithm usually simplifies the search problem by applying the so-called pre-whitening, i.e., a scaling which imposes unity on all eigenvalues of the covariance matrix[164, 169, 143].

However, to analyze protein dynamics, the relative amplitudes of all coordinates are meaningful and important. Therefore, the simplification offered by pre-whitening is not applied and FCA is restricted, in analogy to principal component analysis (PCA), to orthogonal transformations. This restriction to rotations conserves the geometry of the conformational ensemble. In particular, volumes are left unchanged enabling, e.g., straightforward computation of free energy surfaces for FCA modes.

Note that ICA algorithms generally do not separate independent quasi-harmonic modes[169] due to the applied pre-whitening. These harmonic modes, however, are a well-known feature of protein dynamics[89], and should be captured. Therefore, it is important to stress that FCA succeeded in their separation, nonetheless.

For brevity we do not present extensive comparison of the performance of the devised algorithm with other available algorithms. Nevertheless, we compare our algorithm to MILCA, which outperforms a large number of ICA algorithms[165]. At first sight both, MILCA and our algorithm, are similar, but they actually differ in important aspects.

The main difference lies in the treatment of MI. Here, the sum of single dimensional entropies, Eq. (4.4), is minimized directly, whereas MILCA minimizes pairwise MI. At first glance, this is equivalent, i.e., in analogy to Eq. (4.4) MILCA uses

$$\Delta_I(\varphi) = I\left[\mathbf{R}_{ij}(\varphi)\mathbf{x}\right] - I\left[\mathbf{x}\right] = I\left[\tilde{x}_i, \tilde{x}_j\right] - I\left[x_i, x_j\right]. \tag{4.6}$$

However, this implicitly involves estimation of two-dimensional entropies $H\left[x_i, x_j\right]$ and $H\left[\tilde{x}_i, \tilde{x}_j\right]$, which cancel out because the rotation $\mathbf{R}_{ij}(\varphi)$ leaves them invariant. Moreover, computed in this way, $\Delta_I(\varphi)$ is prone to statistical errors. As discussed above, MI is quite sensitive to numerical estimation errors due to the near cancellation of large entropies. Because of these inaccuracies, the right hand side of Eq. (4.6) becomes highly rugged, such that identification of the global minimum becomes difficult. To counteract this, MILCA has to evaluate $\Delta_I(\varphi)$ at 150 intermediate angles, and has to apply smoothing by Fourier filtering before it determines the minimum[165]. Such countermeasures were not necessary in a minimization of $\Delta_I(\varphi)$ using Eq. (4.4), which needed typically just 20 evaluations of $H\left[\tilde{x}_i\right] + H\left[\tilde{x}_j\right]$ in a single rotational plane.

A further difference to MILCA is the applied systematic choice of rotational planes, which increases convergence speed.

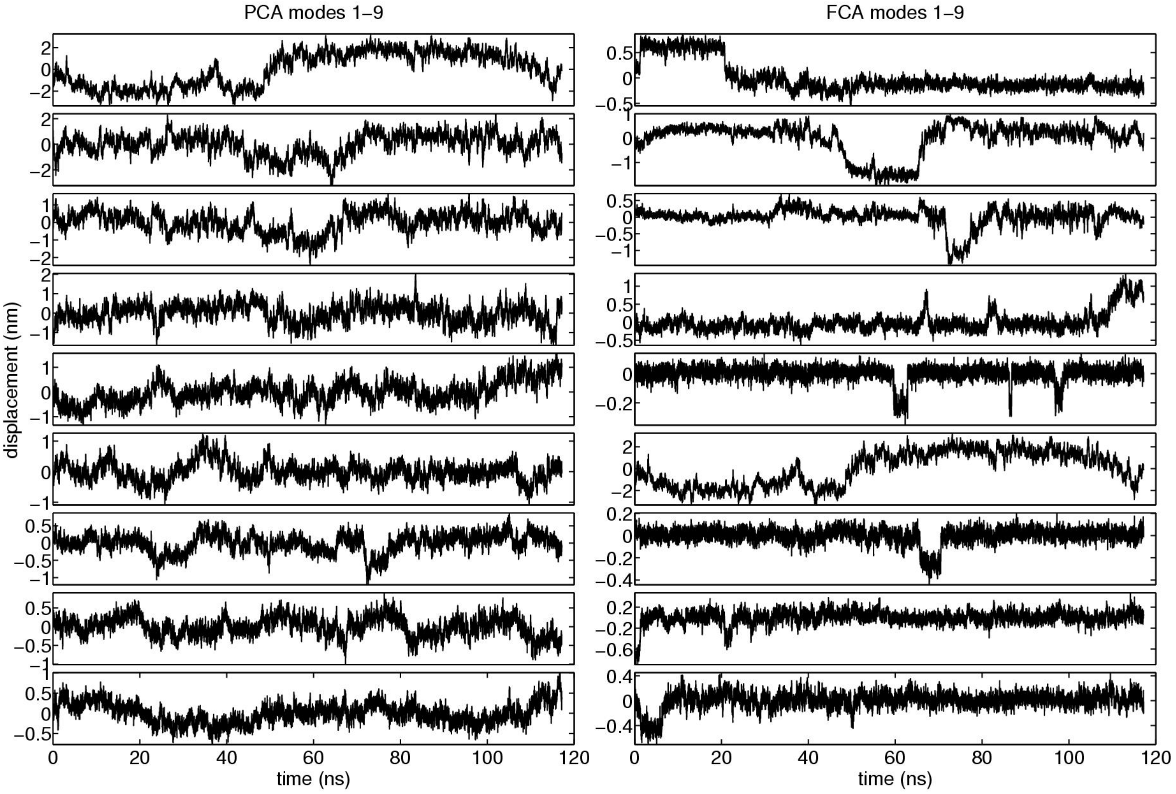


Figure 4.4: Projections of MD simulation of T4L onto the first 9 PCA (left) and FCA modes (right), respectively.

4.4 Extraction of functional motion with FCA

4.4.1 Conformational motion of lysozyme analyzed with FCA

Having established that the FCA algorithm works correctly we now proceed and apply FCA to a real system. To this end we chose T4 lysozyme (T4L). Essential for the function of T4L is a large conformational motion of the two domains, allowing the substrate to enter and the products to leave the active site[153, 154, 156, 155]. The ensemble of T4L structures gained from a 117ns MD simulation was treated with FCA and, for comparison, also with PCA.

Figure 4.4 shows the time-series of projections onto PCA (left) and FCA (right) modes. There are remarkable differences for most of the modes. In contrast to PCA modes, the fluctuations in the FCA modes are most of the time very small until a relatively large transition occurs.

We note that in Fig. 4.4 PCA modes are sorted by fluctuation amplitude, whereas FCA modes are sorted by anharmonicity (cf. Methods). This direct comparison of the differently ranked modes is justified, since the ranking scheme is an essential part of the respective methods. Nevertheless, the most anharmonic PCA modes did not show any FCA-like patterns, either (results not shown).

Now we turn to projections of the ensemble T4L onto pairs of PCA and FCA modes. These kind of projections are often used for exploratory data analysis, because in this presentation cluster of

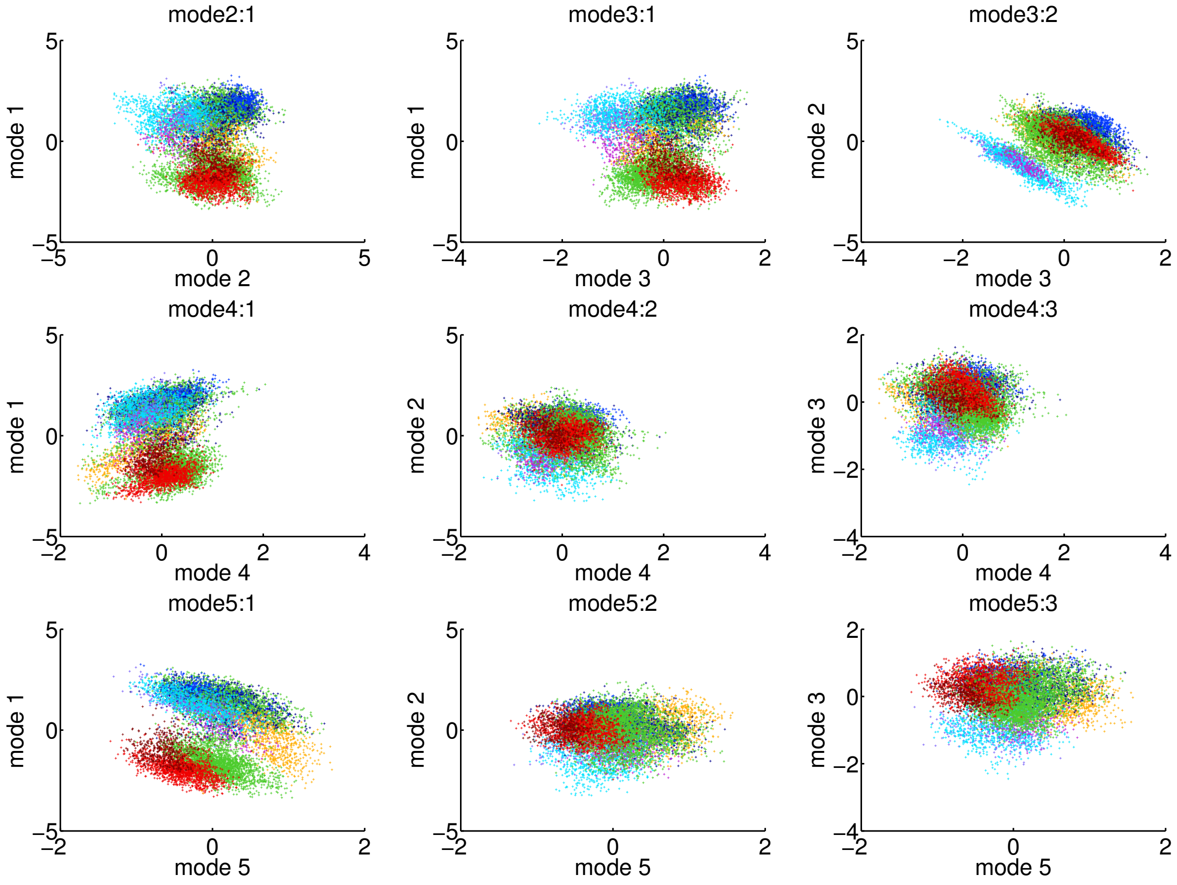


Figure 4.5: Projections of MD simulation of T4L onto pairs of PCA modes. The coloring of the points corresponds to that in Fig. 4.6, i.e., a projection belonging to a certain structure of the protein has the same color in all plots of the two figures.

points indicate conformational substates.

Fig. 4.5 shows projections of the ensemble to pairs of the first PCA-modes. In 5 of the shown projections, such a clustering can be anticipated. Nevertheless, most clusters overlap strongly, such that only in the projection to mode-pairs 3:2 and 5:1 an assignment of points to their respective clusters would be indisputable. In contrary, clusters in the analogous projections to pairs of FCA-modes (Fig. 4.6) are numerous and show a pronounced separation.

Strikingly, the projections to pairs of FCA-modes adopt often the shape of the letter *L*. Thus, FCA tends to describe transitions between two conformational substates with a single FCA mode. For instance, mode 2 describes a transition towards the cyan and purple clusters (visited between 50 ns–70 ns, as seen from Fig. 4.4), and mode 9 describes transitions from the dark red to the light red cluster (2 ns–8 ns).

For PCA modes the situation is very different. For example, the transition along FCA mode 2 is described by PCA modes 2 and 3, and to a lesser extent also by PCA mode 4. This transition is further obfuscated, because motions which do not contribute to it are also mapped onto PCA modes 2 and 3,

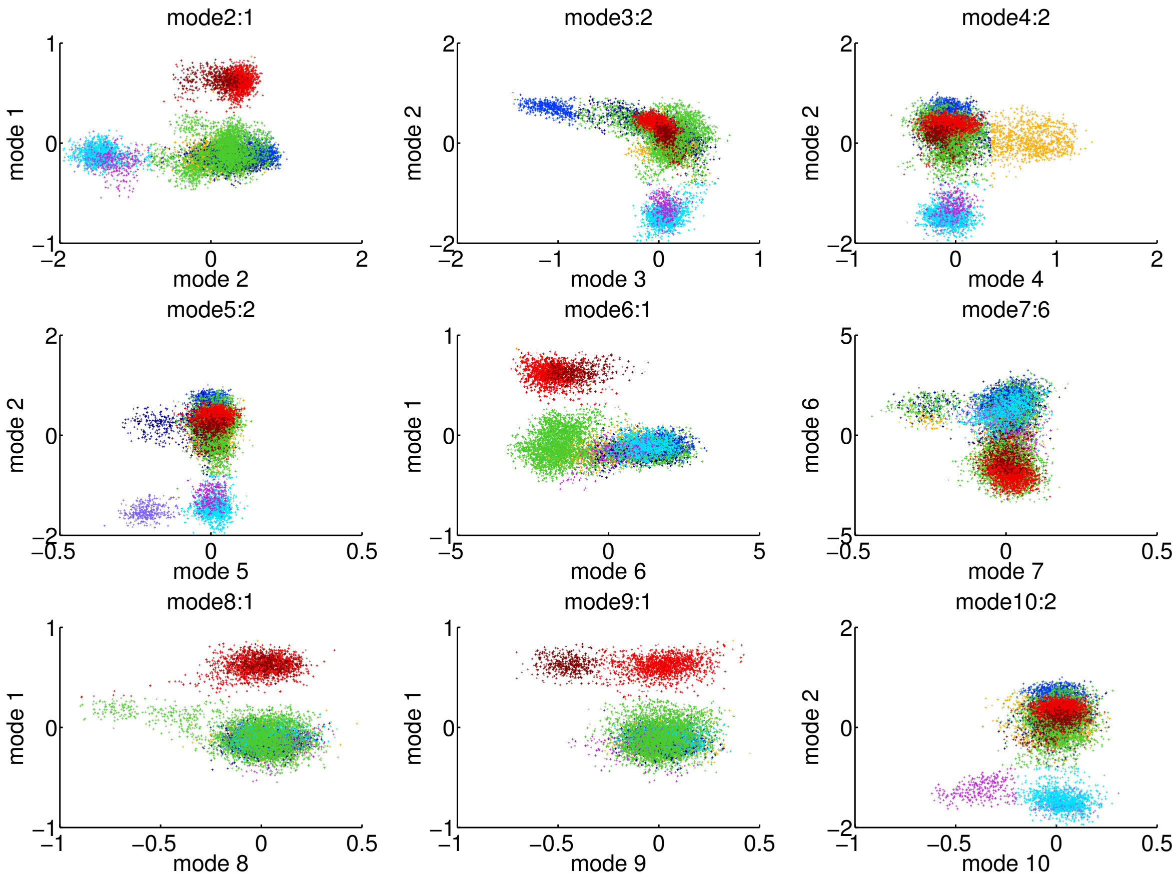


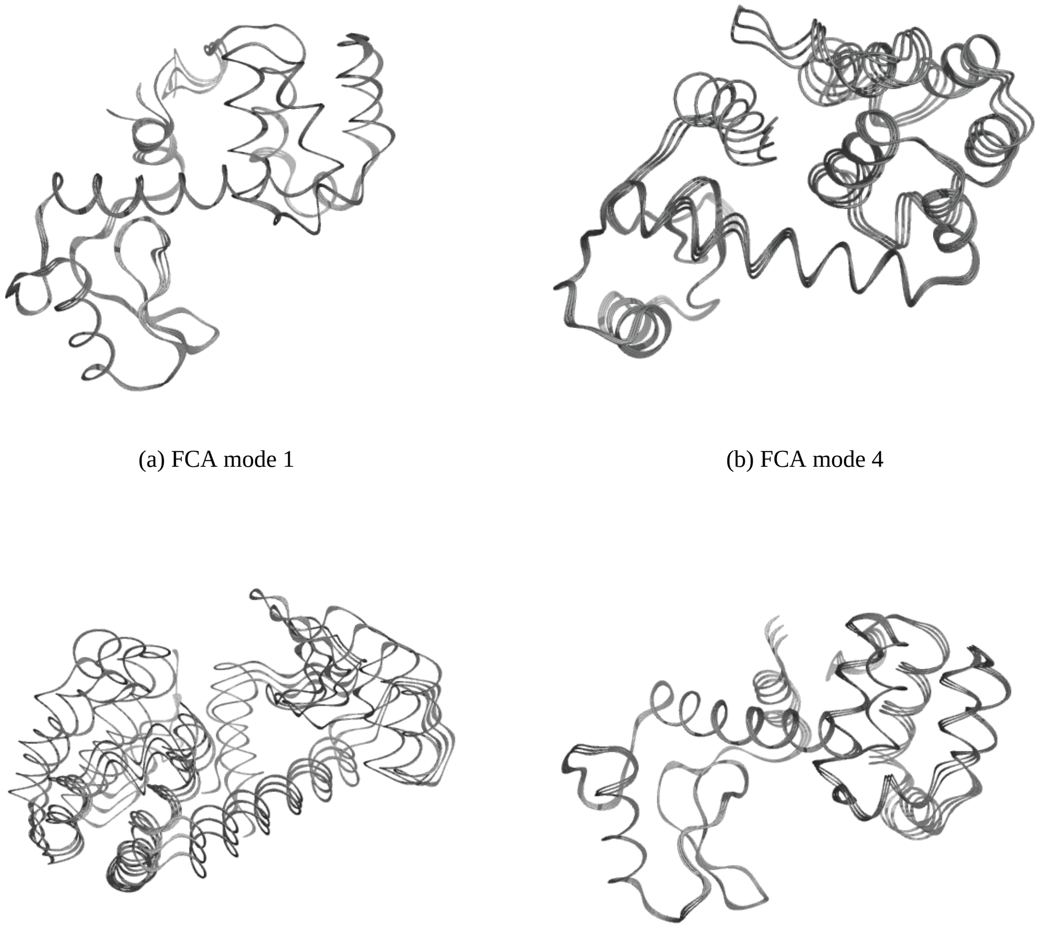
Figure 4.6: Projections of MD simulation of T4L on pairs of FCA modes. The presented pairs of FCA modes were selected by a rule based on pair correlations and anharmonicity (cf. Sec. 4.2.3). The projection belonging to a certain structure of T4L has the same color in all plots. The colors were chosen freely.

causing large background fluctuations during the whole simulation length (cf. Fig. 4.4).

Nevertheless, some FCA modes are very similar in nature to PCA modes, for instance FCA mode 6 corresponds to PCA mode 1.

To visualize which motions are described by selected FCA modes, Fig. 4.7 shows superpositions of 3 structures obtained by projecting the C_α motion of T4 lysozyme on the respective FCA mode. FCA mode 1 corresponds to a local swiveling motion of the 3 N-terminal residues (cf. Fig. 4.7a), whereas FCA modes 2 and 3 described a similar motion of the C-terminus (not shown). FCA mode 4 and FCA mode 9 depicted in Figures 4.7b and 4.7d, respectively, describe collective motions involving the whole C-terminal domain. FCA mode 6 depicted in Figure 4.7c — and the identical PCA mode 1 — yield a highly collective motion of the whole protein.

The presented projections of the T4L ensemble to FCA modes showed a much improved resolution of conformational substates with respect to PCA modes. Furthermore, transitions between substates are described by single FCA modes, which suggests that FCA is particular suitable to yield conformational coordinates, to describe such transitions in a dimension reduced approach.



(a) FCA mode 1

(b) FCA mode 4

(c) FCA mode 6

(d) FCA mode 9

Figure 4.7: Superposition of 3 configurations obtained by projecting the C_α motion of T4 lysozyme onto the respective FCA mode.

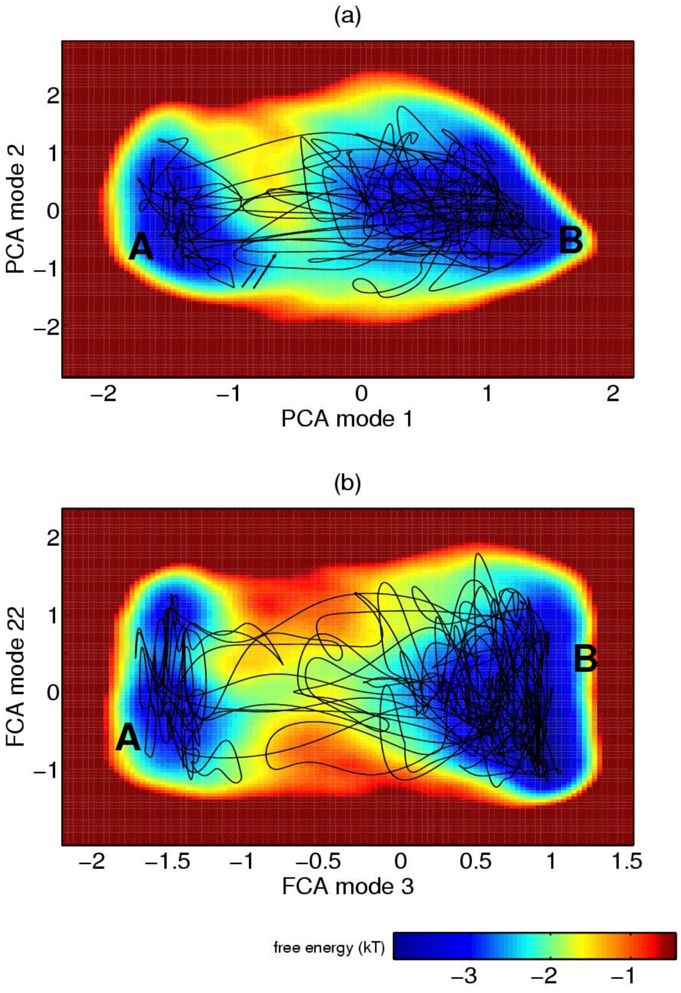


Figure 4.8: Free energy surfaces of neurotensin dynamics: The smoothed trajectory is plotted (black) on top of the free energy surface (colors) in a projection to **(a)** two PCA and **(b)** two FCA modes, respectively. The smoothing by convolution with gaussians of width (10-20ps) pronounces transitions and suppresses intra substate motion. The two arrows in (a) denote two unsuccessful transition attempts, which due to a projection artifact are falsely shown to reach conformational state A, see text.

4.4.2 Reduced description of conformational transitions of neurotensin with FCA

In chapter 8 we will develop a CLD model of the conformational motion of the hexapeptide neurotensin. To this end, we need to define a low-dimensional conformational subspace, which resolves the relevant conformational states. As discussed in Chapter 2 PCA is a good candidate to extract suitable conformational coordinates for this purpose. Nevertheless, as shown above, FCA modes yield an improved resolution of the substates and less modes are needed to describe a transition. Here, we will compare both methods, and focus in particular on the pathways of transitions in relation to the free energy surface estimated for the selected pair of modes.

Figure 4.8a shows the free energy surfaces of two PCA modes estimated from a $100\,$ns MD simulation (NT1) of the peptide. Two major conformational states, denoted A and B, can be identified as two extended basins on the free energy surface. They are connected by a channel of relatively low free energy bearing a putative transition state at $(s_1, s_2) \approx (-0.5, -1)$, which is $\sim 1kT$ lower in energy than the remaining transition region. One would expect to find that most transitions between the

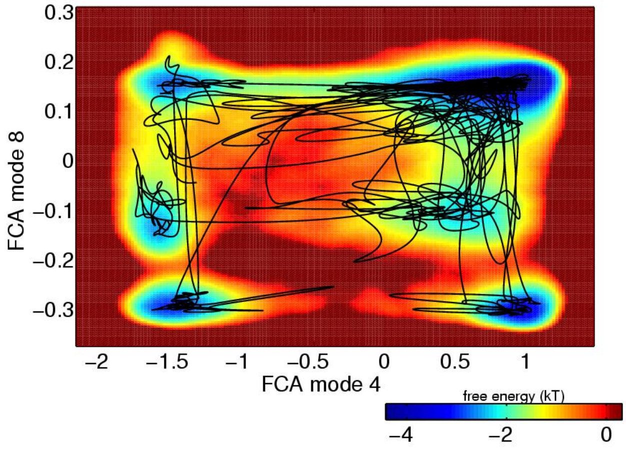


Figure 4.9: In analogy to Figure 4.8, this plot shows the smoothed trajectory on top of the free energy surface of neurotensin for the pair of FCA coordinates that yields the most pronounced separation of conformational states.

two conformational states use this channel. However, the contrary is true. The overlayed smoothed trajectory shows that all *successful* transitions take place in the region between $-0.5 < s_2 < 1.5$, where the energy is about $1kT$ higher than the putative transition channel. In fact, only two crossing attempts (arrows), which are moreover *unsuccessful*, use the putative lowest free energy path. Despite their apparent visit of the low energy region of state A they recrossed the barrier immediately.

Including more PCA modes into the analysis explained this behavior. Instead of reaching state A the system remained in a protuberance of conformational state B, whose projection just happened to overlap with the projection of conformational state A. Obviously, this overlap caused the emerging of a spurious low free energy channel from A to B, as will be discussed in more detail in Sec. 8.4.

We tested, if FCA improves the situation. Indeed, in Fig. 4.8b the two conformational states are well resolved and the channel of lowest free energy agrees with the actual pathways of the observed transitions. Moreover, FCA mode 22 revealed a sub-structure of conformational state A, which was not resolved by the PCA modes. This sub-structure will be resolved in Chapter 8, too, by using a curved conformational coordinate.

To dispel any doubts about the validity of the comparison, we have previously plotted those FCA modes (3 and 22), which were most parallel with PCA modes 1 and 2. However, this pair of modes was not selected by the automatic protocol of FCA described in Sec. 4.2.3. Instead, Figure 3.3 depicts the free energy surface of the fourth of the automatically selected pairs, which was chosen, because from the 9 inspected pairs it displayed the most pronounced clustering. It separates each of the conformational states A and B into 3 sub-clusters. Moreover, in this FCA projection the transition pathways also agree well to the low free energy valleys.

The presented results indicate that FCA modes have a lower tendency than PCA modes to produce artifacts in low dimensional free energy surfaces that result from overlapping projections of conformational states. Thus, FCA extracts from MD simulations relevant collective coordinates, which are well suited to describe conformational dynamics of proteins within the CLD framework.

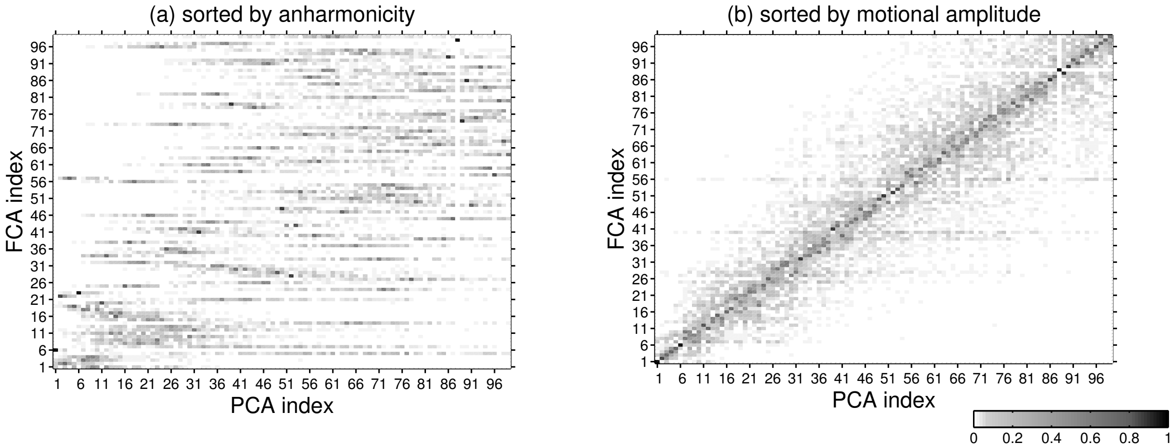


Figure 4.10: Inner product matrices between FCA and PCA modes of T4 lysozyme. In the left plot the FCA modes are sorted by anharmonicity and in the right by motional amplitude.

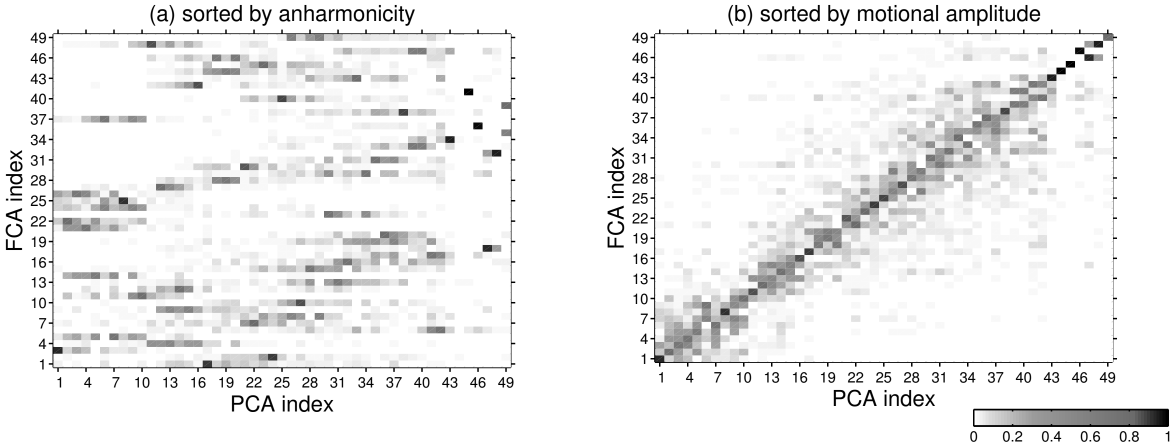


Figure 4.11: Inner products between FCA and PCA modes of neurotensin. In the left plot the FCA modes are sorted by anharmonicity and in the right by motional amplitude.

4.4.3 Comparative analysis of PCA and FCA modes

The previous sections have shown remarkable differences between projections of protein dynamics onto FCA or PCA modes. The results suggested that collective coordinates extracted with FCA pose a superior alternative to principal components. To single out the properties of FCA modes, which are responsible for the observed improvements, we systematically characterized the differences between FCA and PCA modes.

We started by comparing their directions and, therefore, quantified their colinearity by inner products. The inner products between PCA and FCA modes of T4 lysozyme are depicted in Fig. 4.10 and those of neurotensin in Fig. 4.11. On the left hand side the FCA modes are sorted by anharmonicity as it was prevalent in the previous sections, and on the right hand side they are sorted by the motional amplitude. The figures show that anharmonicity-ordering of FCA modes is obviously less suitable for

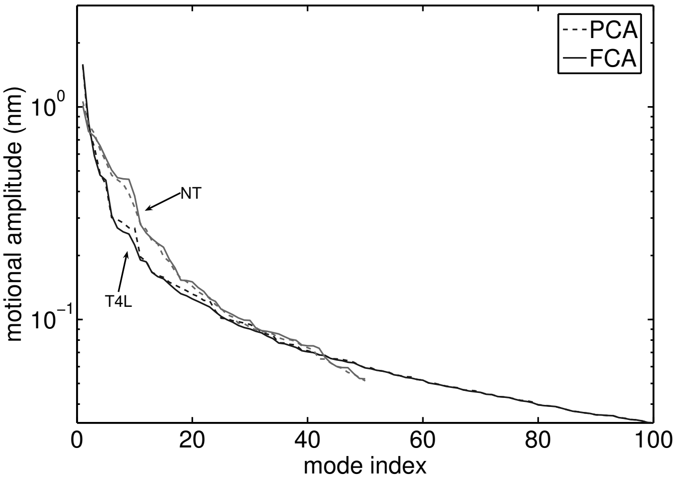


Figure 4.12: Motional amplitude of PCA and FCA modes of T4 lysozyme (T4L) and neurotensin (NT). FCA modes are enumerated by motional amplitude.

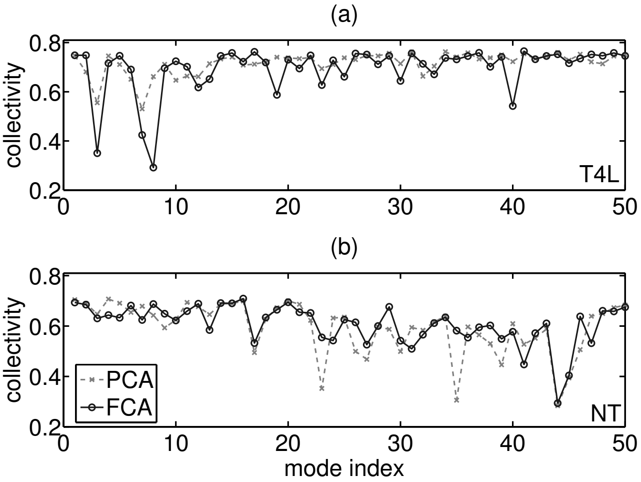


Figure 4.13: Collectivity of the motion described by PCA and FCA modes of (a) T4 lysozyme and (b) neurotensin. FCA modes are enumerated by motional amplitude.

a detailed comparison. Therefore, and to forestall any confusion, we will enumerate in the following both, FCA and PCA modes, by motional amplitude.

For T4 lysozyme, Fig. 4.10b shows that almost all FCA modes have a different direction than PCA modes. In particular, from the low numbered PCA modes only the direction of mode 1 and mode 6 are colinear to FCA modes. Nonetheless, it is evident that specific FCA modes are generally contained in a low-dimensional subspace spanned by PCA modes of similar amplitude. For instance, FCA mode 7 has contributions from PCA modes below 30, whereas FCA mode 50, is a mixture of PCA modes between 30 and 65. Thus, the PCA and FCA subspaces of large amplitude modes overlap to a large extent, although the directions of their respective basis vectors differ strongly.

Also for NT the PCA modes contributing to an FCA mode have a similar amplitude (cf. Fig. 4.11b). In contrast to T4L, no FCA mode is collinear to a PCA mode. Despite this, the PCA contributions to the first 10 FCA modes are evenly divided between the low indexed PCA modes, indicating an overlap of the large amplitude subspaces, as observed already for T4L.

An important and often exploited property of principal components of protein ensembles is the fast

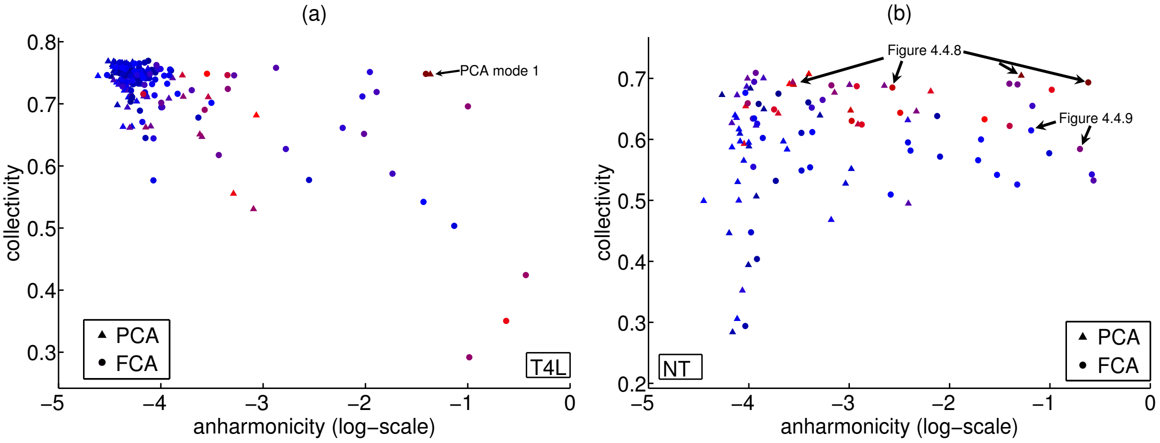


Figure 4.14: The collectivity of PCA and FCA modes is plotted against their anharmonicity. The color gradient from blue to red is in accordance to an increasing motional amplitude of the respective modes. **(a)** T4 lysozyme. **(b)** neurotensin; the arrows mark those modes which have been used in Sec. 4.4.2 to determine the free energy surface in the respective figures.

decrease of their fluctuational amplitude. Figure 4.12 shows that for both test systems the motional amplitude of FCA modes did not differ significantly from that of PCA modes, although FCA optimizes mutual information instead of the motional amplitude. Therefore, similar to PCA, the first few FCA modes describe a major part of the total atomic displacement of the protein ensemble[50].

Aiming for functional relevant motions we are generally not interested to extract modes which describe very local motions, e.g., displacement of single C_α-atoms or the flip of single side chain dihedrals. Hence, we compared the collectivity of FCA and PCA modes. Low-indexed PCA modes are likely to be highly collective modes, since they maximize the motional amplitude, which is generally the larger the more atoms are involved. FCA, in the contrary, has a less direct link to collectivity. Indeed, Fig. 4.13a shows that three FCA modes of T4L have a relatively low collectivity. These modes describe the swiveling motion of either 3 C-terminal or 3 N-terminal residues (cf. Fig. 4.7a). The two rather localized PCA modes describe also such swiveling motions of the terminal residues, but are less dedicated to it, such that their collectivity is slightly higher than that of their FCA counterparts. However, all other FCA modes have a similar collectivity as PCA modes. For NT the collectivities of FCA and PCA are very similar, too (cf. Fig. 4.13b). Unexpectedly, for NT the most localized modes were obtained by PCA.

So far no significant differences between properties of FCA and PCA modes have been observed. On the contrary, their anharmonicity, Eq. (4.5), differs strongly. In particular, Fig. 4.14, a scatter plot of anharmonicity and collectivity of modes, reveals for both test systems a cluster of modes of high collectivity and high anharmonicity, which is almost exclusively populated by FCA modes.

Possibly, the high anharmonicity is responsible for the improved resolution of conformational states obtained by FCA, which was observed in the two previous sections. The arrows in Fig. 4.14b denote modes that have been used in the previous section to obtain free energy surfaces for NT.

Indeed, the two PCA modes, which were unable to resolve the conformational states sufficiently well (cf. Fig. 4.8a), show both a lower anharmonicity than their corresponding FCA modes. Thus, the improvement seen in Fig. 4.8b can be attributed to the increased anharmonicity of FCA modes. The other two labeled FCA modes improved the resolution of conformational substates even further, which is indicated by their nearly maximum anharmonicity.

Note that the amplitude of the modes, which is color coded into the figure, is uncorrelated with both, collectivity and anharmonicity. High amplitude modes are found in any region of the plot, even for those FCA modes with a low collectivity and a high anharmonicity that describe the irrelevant swiveling motion of the terminals. Hence, FCA modes that are likely to describe functional relevant motions can be selected by means of a combination of high collectivity and high anharmonicity. This criterion is superior to a selection based purely on amplitude.

4.4.4 Remaining correlations between modes

At last we investigated to what extent pairs of modes were correlated. Figure 4.15a shows that only the first 10 PCA modes of T4L were highly correlated $(r_{\mathrm{MI}} > 0.2)$. These correlations were not significantly reduced by FCA (cf. Fig. 4.15b). However, the small correlations, which occur sporadically between higher indexed PCA modes were completely removed by FCA.

For NT the situation differs in two aspects (cf. Fig 4.15d-f). First, all PCA modes of NT are correlated considerably, as seen in Fig. 4.15c. Second, correlations between both, high and low indexed modes, were drastically reduced by applying FCA, as shown by the drastic reduction of the *average* pair correlation (solid lines in Fig. 4.15f). However, the *maximal* pair correlations (dashed line) remain high and even increased in some instances.

As seen from the inset in Fig. 4.15e, these remaining correlations constituted small clusters of coupled modes. Thus, FCA successfully identifies uncoupled motions in NT, but these are described by multiple linear FCA modes. The reason for this is either that the motion is non-linear, e.g., rotational, or that a further separation into single-dimensional degrees of freedom is not possible due to strong non-linear coupling.

Why was FCA not able to find a similar separation into uncoupled (multi-dimensional) modes for T4L? An explanation might be that during the 117ns MD simulation its motion was not sampled as thoroughly as that of NT, whose simulation was with $100\,\mathrm{ns}$ comparatively long for the much smaller peptide. Thus, some modes of conformational motion of T4L were excited only once due to the short simulation time. For these modes spurious correlations are likely to be detected, because any coincidental excitation of a different mode yields a correlation of the respective modes in the generated MD ensemble. Only longer trajectories with repeated excitations of the same modes would enable separation of true correlations from coincidental motion. In particular, T4L underwent a slow opening motion of its two domains described by FCA mode 1 (FCA mode 6 in Fig. 4.4a). During half of the simulation T4L was closed and during the other half opened. Thus, all motions which occurred only once seem to be correlated with mode 1, which is also reflected by the high number of strong

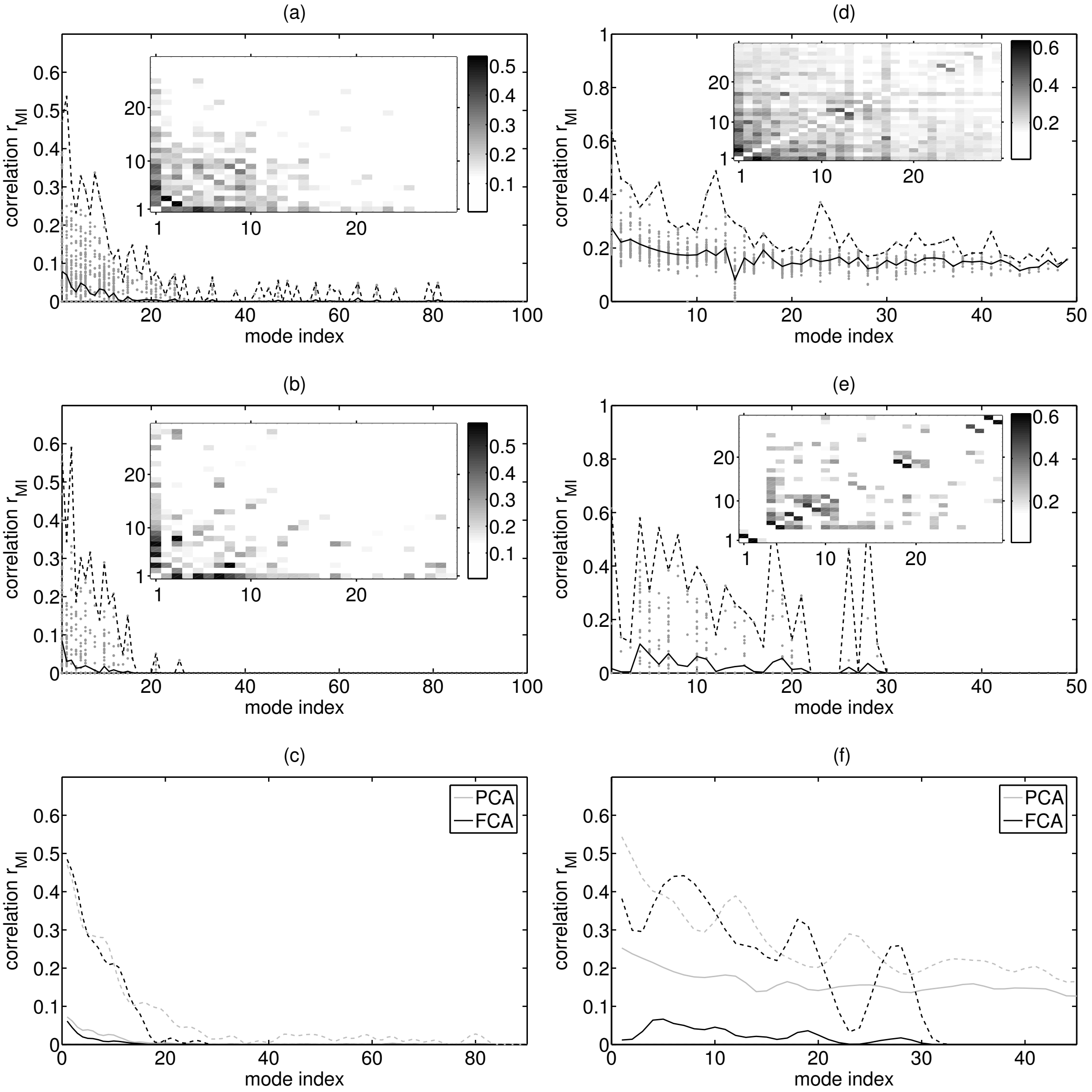


Figure 4.15: Correlation between pairs of FCA and PCA modes, respectively. Correlations between pairs of modes $\mathbf{x}_i, \mathbf{x}_j$ are quantified with the generalized correlation coefficient $r_{\text{MI}}[\mathbf{x}_i, \mathbf{x}_j]$ for **(a-c)** T4 lysozyme and **(d-f)** neurotensin, respectively. **(a,b,d,e)** For PCA (a,d) and FCA (b,e) modes x_i (horiz. axis) the correlations to higher indexed modes $r_{\text{MI}}[\mathbf{x}_i, \mathbf{x}_j]_{j>i}$ were plotted (gray dots) together with the respective average (solid line) and maximum (dashed line). The insets show the mutual pair-correlations of the first 30 modes. **(c,f)** The plots contrast smoothed curves of the average and maximum correlation to higher indexed PCA and FCA modes, respectively.

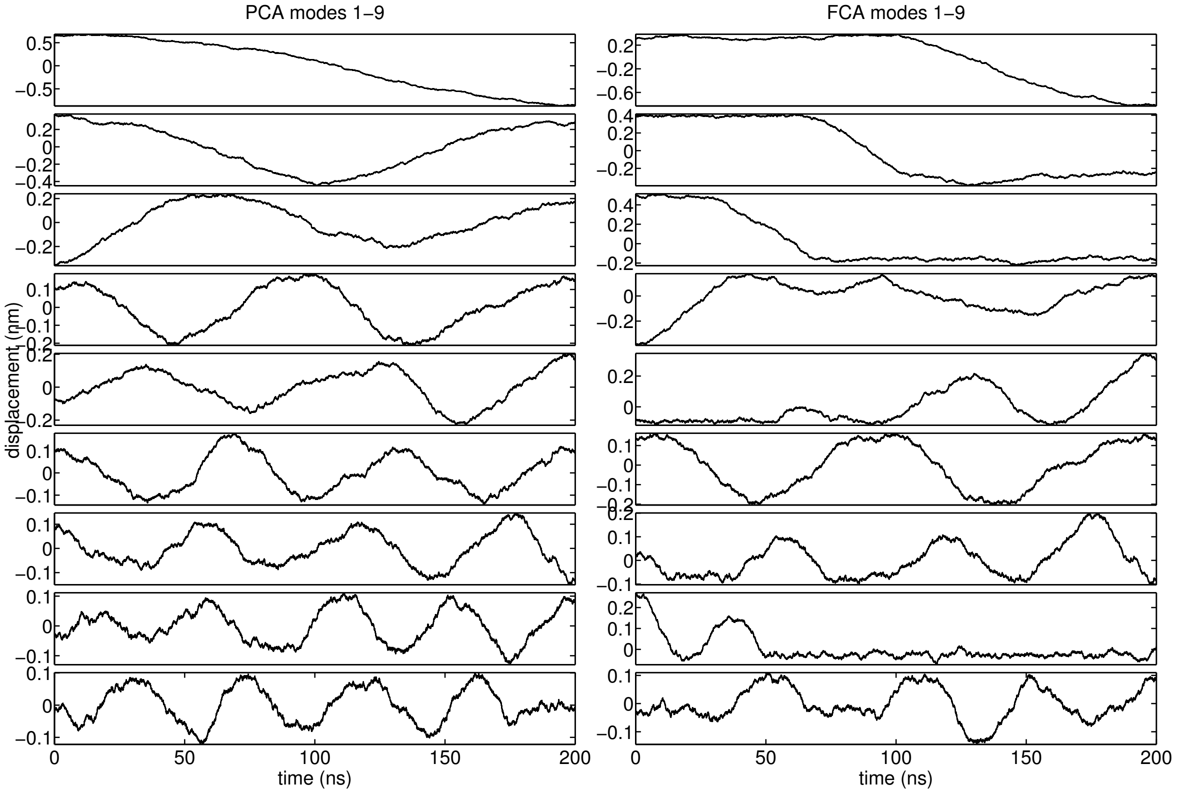


Figure 4.16: Projections of a 120 dimensional random walk to large amplitude PCA and FCA modes.

correlations of mode 1 to others, as seen in the inset of Fig. 4.15a,b.

4.4.5 Convergence of FCA

In respect to convergence, FCA might suffer from the underlying sampling problem of MD simulations in a similar way as PCA. The remarkable and at first surprising effect of an insufficiently sampled protein dynamics on its principal components was illustrated by Hess[170, 129], who showed that projections to principal components obtained from short MD trajectories, as e.g., found in Ref. [50], are very similar to projections of a random-diffusion to its principal components. In particular, the projections to the first PCA modes show sine and cosine shaped curves of large amplitude[170, 129], such as shown in Fig. 4.16. This result allows to identify artificial large amplitude ’features’ in projections onto principal components, which should not be interpreted as functional motions.

Following these lines, we applied FCA to a random diffusion, too. Projections of the random diffusion to the resulting FCA modes are shown in Fig. 4.16. These projections show also large amplitudes and slow transitions, as seen previously for PCA modes. In contrast to FCA modes of the T4L ensemble the FCA modes of a random diffusion show more gradual transitions (cf. Fig. 4.4). This suggests that apart from FCA mode 1 all FCA modes of T4L did converge. On the contrary, such a clear distinction between projections of random diffusion and protein dynamics cannot be established for PCA. In particular, projections of the T4L ensemble onto PCA modes show also gradual transitions (cf. Fig. 4.4) similar to the PCA modes of random diffusion.

4.5 Conclusions

With Full Correlation Analysis (FCA) we have developed a new approach to extract a reduced dimensional description for functionally relevant motion from configurational ensembles. FCA minimizes the coupling, i.e., correlation between the coordinates. In this way it differs from the well-established and widely used principal component analysis (PCA), which maximizes the motional amplitude along the coordinates.

Remarkable differences between the two methods, PCA and FCA, became obvious in a comparative study of MD simulations of two systems, T4 lysozyme and neurotensin.

FCA on the one hand, yields collective coordinates, which are well suited to describe the conformational dynamics of a protein. In particular, FCA aligns the extracted modes along the pathways of conformational transitions, and, thereby, yields an improved resolution of conformational subspaces. Moreover, the transition regions of presented free energy surfaces of pairs of FCA modes were consistent with the observed transitional dynamics.

PCA on the other hand, is less suited for a dimension reduced description of conformational dynamics. Seeking large amplitude modes, PCA chooses directions for the modes that are often skew with the conformational transitions. This has the consequence that the structure of the conformational substates is blurred. Furthermore, due to this skewness more PCA modes are needed to yield a free energy surface that is consistent with the observed dynamics. In particular, in contrast to FCA, two PCA modes did not suffice to describe the conformational motion of neurotensin, because two otherwise separated conformational substates overlapped in the projections to the first two PCA modes. Due to the overlap an artificial low free energy channel between the two conformational states emerged in a region, where no transitions occurred due to a high free energy barrier. To reveal this high free energy barrier a higher number of PCA modes had to be used.

PCA is widely used to extract highly collective modes with a large amplitude of atomic displacement from MD trajectories. We have shown, that these properties are very similar found for FCA modes, but additionally FCA modes are much more anharmonic, than PCA modes. This increased anharmonicity reflects the improved resolution of conformational substates. Since functional motion implies anharmonicity[90], this further suggests that the motion extracted by FCA modes is functionally more relevant than that captured by PCA modes.

One usually selects the PCA modes with largest amplitude to analyze protein motion[50]. This also selects to a large extent for anharmonic modes, since both properties are correlated for PCA modes[89]. However, irrelevant local motions can also have a large amplitude, e.g., the swiveling modes extracted by both, PCA and FCA. Moreover, motional amplitude and anharmonicity of FCA modes were not correlated as it is the case for PCA modes. Therefore, we suggest to rank FCA modes by a combination of the two properties, anharmonicity and collectivity, instead of motional amplitude.

Despite our prevalent interest in collective motions it is an advantageous feature of FCA to separate off local motions, too, e.g., the swiveling motion of the terminal residues of T4 lysozyme. This large amplitude motion, which is functionally irrelevant, would otherwise be distributed over many modes, obscuring thus, the nature of other more relevant motions. PCA also extracted this swiveling motion, but distributed it over more modes than FCA.

We found it helpful to visualize the matrix of mutual correlations of FCA modes as shown in the inset of Fig. 4.15e. This analysis revealed, that for neurotensin FCA was able to separate the dynamics into several uncoupled motions, whereas the same was not possible for the T4 lysozyme. Moreover, based on these pair correlations we proposed a scheme, which selects those pairs of modes that are particular suitable to visualize the structure of the essential subspace (cf. Fig. 4.6).

Based on our results, we suppose that FCA can be used in many applications. As shown, FCA modes are less coupled and allow better separation of conformation substates, but have otherwise similar beneficial characteristics as PCA modes.

Further work needs to address convergence issues. Firstly, the convergence of the minimization of mutual information. In particular, it needs to be addressed whether always the global minimum of mutual information is found, and if similar FCA modes are extracted from a slightly perturbed MD ensemble. Secondly, the slow convergence of correlations in the configurational ensemble due to the sampling problem of MD[171] . We suspect that in this respect the convergence properties are very similar to those of PCA, since we showed that FCA yields, just like PCA, highly anharmonic modes for a random diffusion[129]. Thus, both, PCA and FCA, suffer from insufficient sampling in the same way: two motions which are observed only once but coincide are detected as correlated, whereas further sampling might reveal their independence. Nonetheless, the application of FCA to a random walk indicated that the ’foot-print’ of an unconverged mode is more distinctly identified in projections to FCA modes than it is the case for PCA modes.

It has been suggested to use nonlinear coordinates to do PCA[172]. Similarly, it will be rewarding to combine nonlinear coordinates with the criterion to minimize mutual information.

The great tragedy of science — the slaying of a beautiful hypothesis by an ugly fact.
— Thomas H. Huxley

Chapter 5

Covariation of protein backbone motion: a comparison between NMR relaxation measurements and MD simulations

As mentioned previously, correlated motions in proteins are ubiquitous and often essential for protein function[133]. Collective Langevin dynamics (CLD) directly describes these correlated motions by choosing suitable degrees of freedom, which are extracted from MD trajectories, e.g., with one of the two methods, PCA and FCA, proposed in previous chapters. Thus, the ability of CLD to accurately represent correlated motions crucially depends on the quality of the extracted coordinates, and accordingly on the accuracy of correlated motions in MD simulations used for their extraction.

In this chapter, we investigate means to check the validity of the extracted conformational degrees of freedom. To this end, one ideally compares with experiments which directly probe correlations. Such experiments would allow not only verification of an important observable of CLD, but also offer an experimental access to collective coordinates rendering obsolete extended MD simulations for their extraction.

Few experimental techniques are capable of providing direct information on collective motions in proteins. One reasonable approach probes correlations via the diffusive part of the Xray scattering signal[173, 174]. However, conclusive interpretation of these signals in terms of correlation of individual atoms or residues proved to be difficult so far[175, 176, 54].

NMR relaxation studies, on the contrary, are a powerful and established method to gain experimental access to fast protein dynamics in atomic detail[177, 178, 179]. Model-free analysis of NMR relaxation times in particular, yields generalized order parameters for individual bond vectors [180], which allows to extract information about flexibility and timescales of motions of individual backbone sites [181] and sidechains [182]. However, hitherto NMR relaxation experiments could only probe the flexibility of individual bond vectors, whereas correlated motions could not be probed. The recently proposed method by Mayer et al.[59] promises to overcome this limitation by measuring the covariation of backbone motion by NMR relaxation studies. Specifically, the authors “propose a general approach to the detection of correlated changes in internal protein motions, which are expected to reflect the underlying influence of correlated dynamics" [59]. In this approach NMR relaxation data for a small protein domain of Protein G were obtained for ten mutants of the same residue. The perturbations due to the mutations were reported to cause changes in the measured order parameters, which were recorded for each individual residue. It has been found that for many residue pairs these changes are significantly correlated, (Fig. 5.1a) which led the authors to suggest that the observed covariances reflect underlying correlated atomic motions [59]. Whether and how the measured covariations actually reflect correlated atomic motion, cannot be resolved by experiment alone, which prevents direct atomistic interpretation of these types of measurements. The previously developed generalized correlation measure (cf. Chapter 3) enables us to address this issue. We present sub-microsecond molecular dynamics simulations of the protein G domain, which together with additional NMR data, provide a comprehensive picture of the correlated atomic dynamics of the protein G domain.

5.1 Methods

5.1.1 Generation of structure ensembles from NMR NOE data

An NMR-NOE structure ensemble of 30 structures was generated using the standard simulated annealing protocol in CNS [183], applying the NOE distance bounds as available from the 3GB1 PDB entry [184]. In short, individual structures were generated by slow cooling from high temperature simulations starting from an extended structure and different sets of starting velocities. Each annealing cycle consisted of 15 ps of torsion-angle MD at high temperature (50 000 K), followed by a 15 ps annealing phase to zero temperature using torsion-angle MD, and a 15 ps annealing phase with cartesian dynamics, from 2 000 to 0 K. Finally, each structure was energy minimized by 10 cycles of 200 steepest decent steps. Default parameters for the scaling of the individual energy terms were used, including the NOE energy term.

5.1.2 Root mean square fluctuations

To analyze the data in the molecular coordinate frame, all structures were fitted to the backbone of the crystal structure. Root mean square fluctuations (RMSF) for each ensemble were calculated as $\mathrm{RMSF}_i = \sqrt{\langle r_i^2 \rangle}$, where r_i is the distance of the ith C_α atom from its average position, and the $\langle\,\rangle$ denote the average over the whole trajectory using snapshots recorded every picosecond.

5.1.3 Generalized correlation coefficients

The generalized correlation coefficient r_{MI} introduced in Chapter 3 was used. For MD ensembles the correlation matrices were computed as $r_{\mathrm{MI}}\left[\mathbf{x}_i, \mathbf{x}_j\right]$, where the $\mathbf{x}_i$ and $\mathbf{x}_j$ are atomic displacement vectors between the ith and jth C_α-atom, respectively. The correlation matrices of NMR structural ensembles were computed as linearized correlations $r_{\mathrm{LMI}}\left[\mathbf{x}_i, \mathbf{x}_j\right]$, because the low number of available structures was not sufficient for numerical estimation of mutual information. The correlations in the motion of NH-bond vectors were computed by applying the generalized correlation measure to pairs of the normalized internuclear vectors rather than to pairs of atomic positions. For the correlation of the NH vectors visibility of the differences in the range near zero had to be enhanced. To this end, the correlations were weighted by the sigmoidal function $W(x) = 1/\left[1 + \exp(-\lambda(x - 0.1))\right]$, with $\lambda = 17$ before plotting.

5.1.4 Order parameters

Order parameters S^2 are defined as the asymptotic value of internal correlation functions [180]. The internal correlation function $C_{\text{int}}(t)$ of the NH-bond vector motion is given by C_{int}(t)= $\langle P_2(\cos\chi(t))\rangle$, where $\chi(t)$ is the angle between the internuclear vectors $\mathbf{r}(t)$ and $\mathbf{r}(0)$, and $\mathbf{r}$ is measured in the molecule-fixed frame. $P_2(x) = (3x^2 - 1)/2$ denotes the second Legendre polynomial. Fluctuations in the internuclear separation were not included, because the length of all covalent bonds were fixed by LINCS [28] throughout the simulation. It has been shown previously that the effect of such constraints on order parameters calculated from simulations is negligible [185].

To estimate statistical errors in the obtained order parameters, we divided the MD trajectories into N fragments of length 1ns each. For each fragment s, internal correlation functions $C^{s,i}_{\text{int}}(t)$ were computed for each bond vector i using snapshots taken every picosecond. Parameters $\chi_{s,i}$ were computed as the average from 480 ps to 500 ps of the internal correlation function $C^{s,i}_{\text{int}}(t)$. The order parameter of bond vector k was computed as the mean of $\chi_{s,i}$ over all the fragments, i.e. $S^2_i = \left(\sum_s^N \chi^i_s\right)/N$, with the errorbars given by

$$\Delta S^2_i = \frac{1}{\sqrt{N(N-1)}}\left[\sum_s^N \chi^2_{s,i} - \left(\sum_t^N \chi_{t,i}\right)^2\right]^{1/2}.$$

The results do not change significantly if the fragmentation length is changed from 1ns to longer time intervals (data not shown).

5.2 Comparison of NMR covariance with correlations in MD simulations

We carried out two molecular dynamics (MD) simulations of the B1 domain of protein G, with different lengths of 100 and 200 ns, respectively, referred to as GB1/2 and GB1. Correlations of atomic motions were quantified by correlation matrices calculated from the MD simulations. Figure 5.1c shows correlation matrices obtained from GB1/2 (above the diagonal) and MD2 (below the diagonal), respectively. The similarity of the two matrices shows that the computed correlations are largely well converged. Full convergence is not reached for residues 37-42, which constitute the loop connecting the α-Helix with the β3-strand. Closer inspection revealed long timescale contributions to

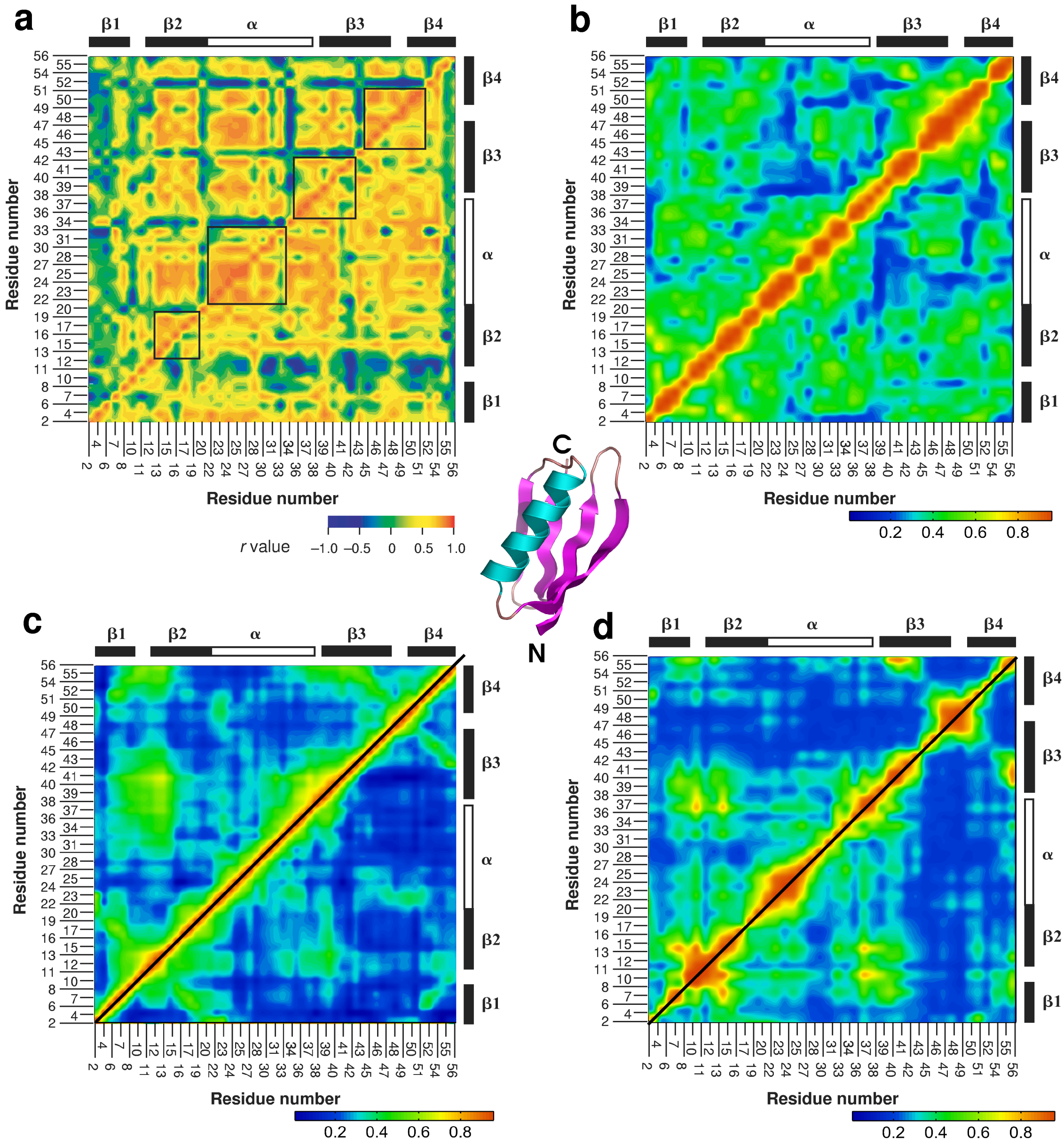


Figure 5.1: Correlation matrices from NMR experiment and MD calculation.
(a) Correlation matrices taken with permission from Mayer et al.[59]. Data for S^2-covariations are shown above the diagonal, whereas those for τ_e-covariations are shown below the diagonal. The black boxes are irrelevant for the present discussion. **(b)** Correlation matrix computed from an NMR-structural ensemble. **(c)** Correlation matrices computed from MD simulations. The correlations computed from GB1/2 are shown below the diagonal, correlations derived from GB1 above the diagonal. **(d)** Correlations computed for NH-vector motions. The center shows the simulated B1 domain of protein G.

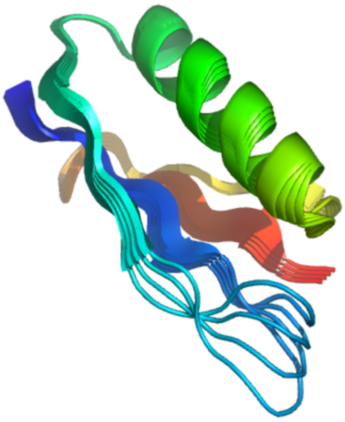


Figure 5.2: Main collective motion as revealed by the principal component analysis described in the text

the loop dynamics. However, the relevant sub-microsecond timescale probed by NMR relaxation experiments [186] is well sampled in our simulations.

The matrix is dominated by strong correlations between neighboring residues which show up as a band along the diagonal. The broad region of high correlation between residues 22 and 38 is caused by the 1-4 contacts of the central α-helix. The two bands of high correlation perpendicular to the diagonal are due to the contacts between different strands of the four-stranded β-sheet. Strong correlations and anticorrelations are also found between residues 10-15 of the first hairpin loop and the rest of the molecule. They are caused by a kinking-out motion of the turn between β-strand $\beta 1$ and $\beta 2$ together with a part of the β-sheet. This motion, hinged around residues 8 and 15, is the main contribution to the principal collective motion (Fig. 5.2), as revealed by principal component analysis [39, 50].

To test if there is a direct connection between the covariation of NMR order parameters and the computed correlated atomic motion observed in the simulation, we compared the computed correlation matrices (Fig. 5.1c) with the S^2 and the τ_e based covariation matrices derived by Mayer et al. [59] (Fig. 5.1a). Overall, the level of correspondence is low. The only feature shared with the MD simulations is the band of anticorrelated motion between residues 10 to 14, which is present in the experimental results derived from covariations in τ_e-order parameters, but is not seen in the S^2 derived correlations. Apart from this detail, the overall lack of similarity is striking. The nature of this discrepancy deserves closer inspection.

5.3 Verification of molecular dynamics simulation through additional experimental data

On the simulation side, the question is whether the MD simulations describe atomic motions with sufficient accuracy. Potential artefacts include force field inaccuracies and convergence problems[171]. Therefore we compared the root mean squared fluctuations (RMSF) of the two MD trajectories (GB1/2

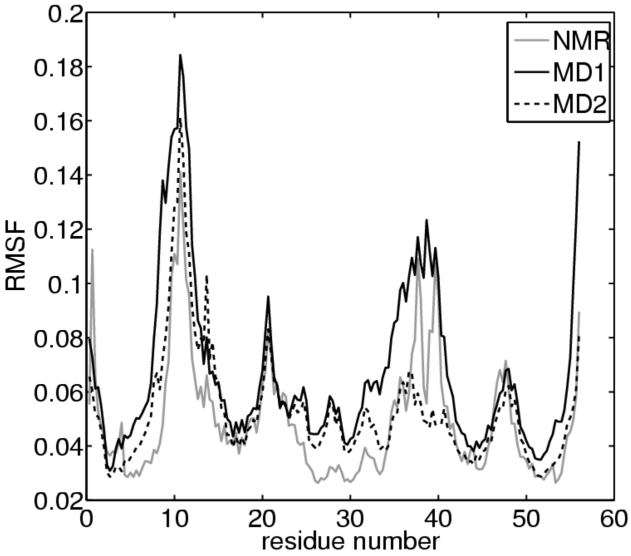


Figure 5.3: Root Mean Square Fluctuations (RMSF) of backbone atoms. The grey line is the RMSF observed in the NMR ensemble, while the black lines correspond to the RMSF computed from the two different MD trajectories. The positions of the major RMSF peaks agree well with the experimental data. The fluctuations of the largest motion, residues 8-15, and of the α-helix are slightly overestimated by the MD simulations, but overall the flexibility of the remaining regions of the molecule is well reproduced.

and GB1) with the RMSF of a structural ensemble of the B1 domain of protein G obtained from NMR NOE data [187] (PDB entry 3gb1, see Methods Section). The RMSF profiles obtained form the MD simulation and the NMR NOE ensemble agree well (Fig. 5.3). Moreover, also the correlation matrices obtained from the simulations (Fig. 5.1c) agree well with the one obtained from the NMR NOE ensemble (Fig. 5.1b). These agreements are quantified by correlation coefficients of 0.6 and 0.74, for GB1/2 and GB1, respectively, between the simulation-derived and the NMR-derived results. In contrast, the correlation coefficients between all of these matrices and the covariance matrix computed from order parameters are much smaller, -0.08 and -0.01 for the two MD matrices, and 0.01 for the NMR NOE matrices, respectively. The good agreement between MD and the NMR NOE ensemble in terms of the atomic fluctuations and in particular of their correlations renders it unlikely that simulation artefacts cause the discrepancy with the covariance matrix obtained from order parameters. That this agreement is not just anecdotal is indicated by a correlation coefficient of 0.6 between NMR NOE and MD covariance matrix obtained for a different protein, ubiquitin (cf. Fig. 5.6), and by a correlation coefficient of 0.72 obtained using a different NMR data set for ProteinG (PDB entry 1gb1, cf. Fig. 5.7).

Because order parameters form the basis for the covariances determined by Mayer et al., it is necessary to check if the motions probed by order parameters are accurately described by the MD simulations as well. To this end, we compared the experimentally obtained order parameters [188] for protein G with order parameters computed from the MD trajectories (Fig. 5.4a). Within the error bars, the simulations agree with each other as well as with the measured order parameters. The inset of Fig. 5.4 shows that most differences between the observed and the computed order parameters are below 0.1 with few outliers below 0.2. Overall, the agreement is good, in line with earlier observations [189, 190, 191, 192].

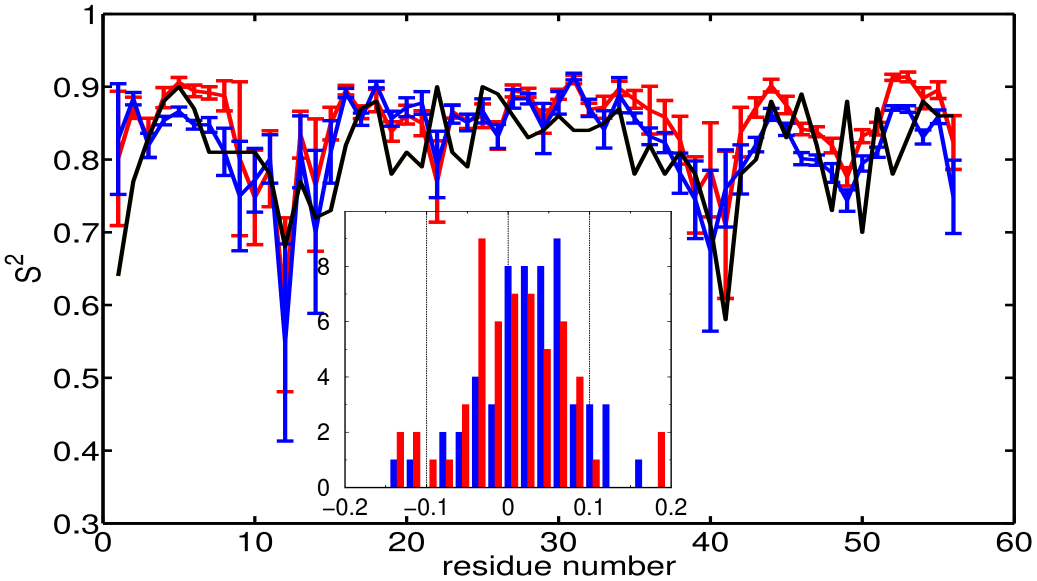


Figure 5.4: Comparison of the generalized order parameters (S^2) between the MD simulations and the NMR data set [188]. Computed order parameters (colored lines) with error bars are compared with experimentally observed order parameters (black line). The inset shows histograms of the order parameter differences between the MD data sets and the experimental results.

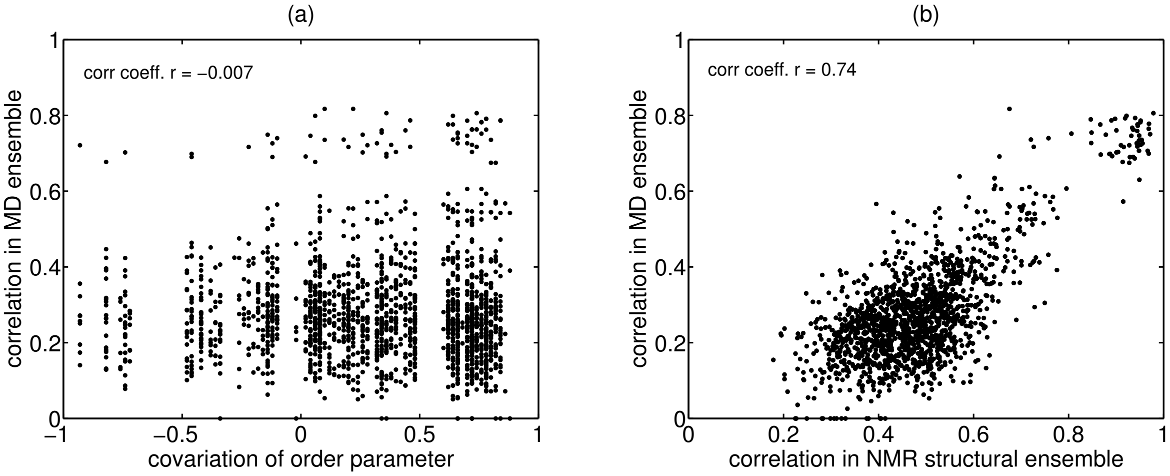


Figure 5.5: Scatter plots of covariation and correlation matrix elements. In each of these plots we compare two of the correlation matrices shown in Fig. 5.1; each point represents the two different correlations obtained with the respective methods for the same residue pair. **(a)** The correlations obtained from MD simulation, as depicted in Fig. 5.1c, are plotted against the covariations of order parameters, shown in Fig. 5.1a. **(b)** The correlations obtained from MD simulation are plotted against correlations observed in the NMR structural ensemble (Fig. 5.1b).

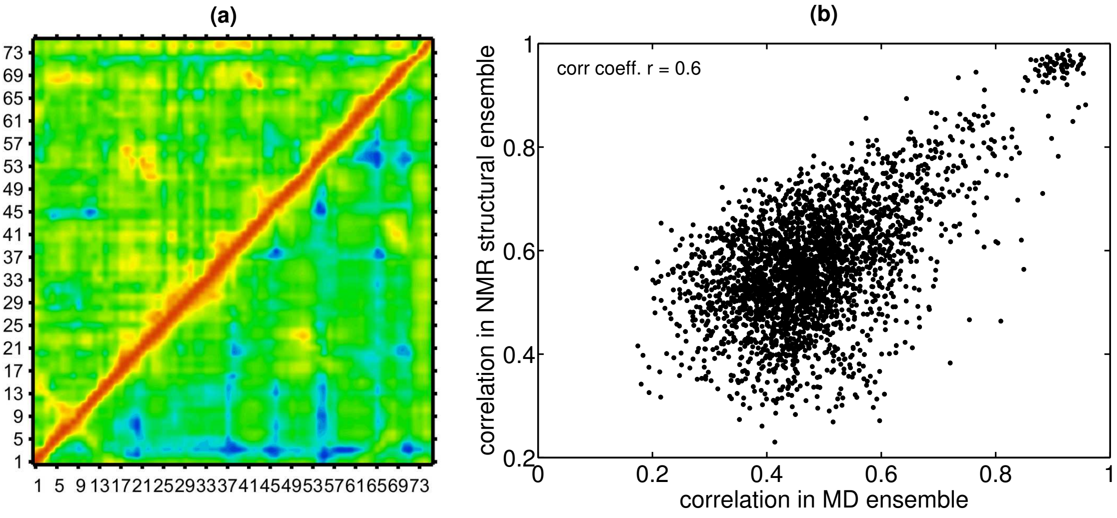


Figure 5.6: Correlated Motions of Ubiquitin. The correlated motions as obtained from MD simulations (28ns/OPLS/gromacs) are shown above the diagonal, whereas the corresponding correlations obtained from the NMR-structural ensemble (1D3Z) are shown below the diagonal. The correspondence between the differently obtained correlations is good, as confirmed by the scatter plot and a correlation coefficient of 0.6.

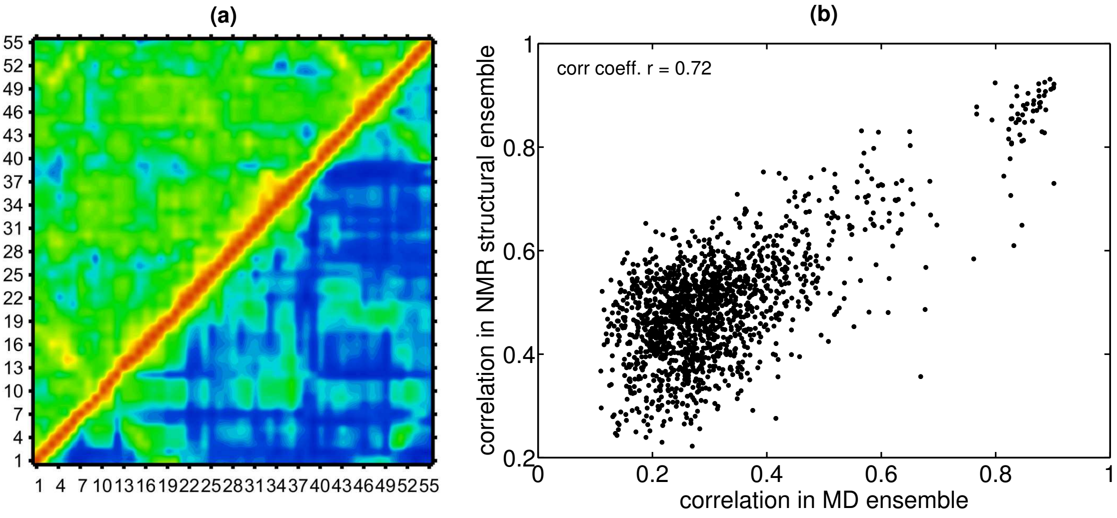


Figure 5.7: Correlated Motions of Protein G. The dataset 3GB1 in the protein data base(PDB) contains only those NOEs which are compatible with additional dipolar coupling data. Here, below the diagonal, we present the correlated motions obtained from an ensemble generated with all NOEs, as stored in the PDB data set 1GB1. For comparison, above the diagonal the correlated motions obtained from MD simulation (MD2) as presented in Fig. 1c. As confirmed by the scatter plot (correlation coefficient 0.72), the good correspondence between MD simulation and NMR NOE data is not thwarted by the additional NOEs stored in 1GB1.

Therefore we conclude that the simulations provide a comprehensive and accurate picture of the correlated atomic motion within the protein G domain that is consistent with the available experimental data. The remaining unexplained discrepancy to the covariances derived from NMR order parameters suggests that these two quantities are in fact not *directly* related. Comparison of all elements of the MD correlation matrix with the respective covariations of NMR order parameters as a scatter plot (Fig. 5.5a) confirms this finding. Furthermore, the absence of any detectable structure in this plot suggests the absence of *any* relation to this measure of correlated atomic motion.

5.4 Rotational correlation

One, finally, might argue that a more direct comparison between MD and NMR data would rest on an analysis of the correlation in the orientational fluctuations of NH bond vectors, which are probed by the NMR order parameters, rather than on the Cartesian coordinates. However, these fluctuations are already included within the generalized correlation measure described above and, hence, similar results are expected. Figure 5.1d (below diagonal) shows that this is indeed the case, as quantified by the low correlation coefficient of 0.05 between NH-vector fluctuations in MD and the covariances derived from the NMR order parameters.

5.5 Conclusions

In summary, we have demonstrated that for two different proteins correlated motions can be accurately extracted from MD simulations, that are compatible with the measured NMR data (NOE and order parameters). However, the correlated atomic motions described by our sub-microsecond molecular dynamics simulations of the B1 domain of protein G are unrelated to covariations derived from order parameters. The obtained agreement with independent NMR NOE data and with NH order parameters provides strong evidence that the simulations accurately describe the atomic motions and their correlations at the experimentally relevant timescale. This is further supported by the good qualitative agreement with recently published residual dipolar coupling experiments [193]. Taken together, the results render it unlikely that the observed covariances in measured order parameters reflect the underlying influence of the correlated atomic dynamics for the wild type protein. Instead, we speculate that the measured covariances are caused by correlated structural changes due to the introduced point mutations, which in turn affect the atomic mobilities. In this framework the experiments would probe remarkably correlated structural plasticities rather than correlated atomic motions. Further simulations of all the ten studied mutants will thus be required to test this hypothesis and to structurally characterize the properties of this proposed non-local plasticity.

Furthermore, the results showed that NMR NOE ensembles might offer a good experimental access to correlated atomic motion. However, the fluctuation observed in those ensembles has two sources. The physical fluctuations of the atoms during the experiment, on the one hand, and an uncertainty of the measurements, on the other hand. Since we are only interested in the former, it is necessary to find means by which they can be separated. Thus, to extract correlated motions from NMR NOE ensemble remains a challenging task. Nonetheless, the approach can be used to falsify correlations observed in MD simulations, and thus might be useful in the framework of CLD.

I would have thought it impossible to investigate
Brownian motion with such precision
– Albert Einstein

Chapter 6

Equations of Motion for Collective Langevin Dynamics

In this chapter the framework of *Collective Langevin Dynamics* (CLD) is introduced. It combines the two concepts, generalized collective degrees of freedom and dimension reduced description using statistical mechanics. The dynamics of the relevant and slow degrees of freedom are actively evolved, whereas the remaining ones are treated in an implicit manner.

Proper treating of the coupling to degrees of freedom which are not explicitly considered, is essential to yield accurate dynamics of the active subsystem, because it allows energy to be exchanged constantly between the two subsystems. Consider, for example, transitions in a double well potential. An isolated degree of freedom in such a potential would either cross the barrier never, if it was low in energy, or, being high in energy, cross the barrier every single oscillation period. On the contrary, the a particle in the same potential but coupled to many other degrees of freedom, would oscillate in one well until it accumulated enough energy from the other degrees of freedom to cross the barrier. On arrival in the second well the coupling enables a dissipation of the excess energy inhibiting an immediate recrossing. Thus, the constant energy accumulation and dissipation of the considered degree of freedom has a significant and qualitative impact on the type of its dynamics.

We use statistical mechanics to treat the many remaining degrees of freedom in form of a stochastic bath, which is coupled to the actively evolved subsystem. The coupling to this bath is modeled by a stochastic process. As discussed in the Introduction, the treatment of Brownian motion is conceptually similar, but in contrast to CLD it rests on a clear separation of timescales. The absence of such a gap in timescales governed by protein dynamics (cf. Chapter 2) has a considerable impact on the statistical properties of the stochastic process, which models the coupling to the bath.

To fathom this qualitative change briefly consider a system where such a gap existed, e.g., a Brownian particle. Under this condition an intermediate time would exist, which suffices for the bath to equilibrate, while its environment determined by the slow degrees of freedom hardly changes. In particular, the state of the bath is fully determined by the instantaneous state of the slow subsystem, such that the effect of the bath can be modeled by a Markovian process. On the contrary, no such gap exists in the context of protein dynamics, such that the state of the bath depends also on the history of the slow subsystem. Thus, the accumulative and dissipative forces are not free of *memory.* The stochastic process describing the coupling to the bath is non-Markovian.

The projection operator formalism developed by Zwanzig[41] and Mori[42] allows a rigorous treatment of the proposed separation of the overall dynamics into slow degrees of freedom and a stochastic bath replacing the fast degrees of freedom. According to this formalism the projection operator is used to split the effect of the fast subsystem onto the slow degree of freedom in two parts. First, the potential force averaged over the canonical ensemble of the fast subsystem yields a *potential of mean force,* and, second, the instantaneous deviation from this average force is captured in a generalized friction term and a term for the accumulative force, called *random force.* In the following we detail how this treatment leads to a Generalized Langevin Equation (GLE). The non-Markovian dynamics of the stochastic bath are fully accounted for by a generalized dissipative term, which is a convolution of the *memory kernel* with past velocities. The treatment also yields a fluctuation-dissipation theorem, which enforces a balance between energy accumulation and dissipation.

In the following treatment we will consider non-linear degrees of freedom in full generality. This enables the use of a curved coordinate in the subsequent application of the CLD model to conformational dynamics of neurotensin.

6.1 Projection Operator Formalism

Let us first consider the conceptual framework, which we sketch here following[194] to clarify notation. We start with the full molecular dynamics, which are described by the Hamiltonian

$$\mathcal{H}(\mathbf{x},\mathbf{p}) = \frac{1}{2}\sum_{i=1}^{n}\frac{p_i^2}{m_i} + \mathcal{V}(\mathbf{x}) \tag{6.1}$$

where $\mathbf{x}$ and $\mathbf{p}$, with components x_i and p_i, respectively, are the n-dimensional position and momentum vectors and m_i their masses. A solution of the corresponding canonical equations is defined through an initial value $(\mathbf{x}_0,\mathbf{p}_0)$; to each initial condition corresponds a trajectory, $\varphi(t) = \varphi(\mathbf{x}_0,\mathbf{p}_0,t)$. Subsequently the subscript 0 is omitted.

In the framework of the projection operator formalism a dynamical variable[42, 41, 195], mechanical property[194], or physical quantity[196] is defined as a mapping on phase space $\mathbb{R}^{3N}\times\mathbb{R}^{3N}$

$$\begin{aligned} A \quad : \quad \mathbb{R}^{3N}\times\mathbb{R}^{3N} &\longrightarrow \mathbb{R}^{2m} \\ (\mathbf{q},\mathbf{p}) &\mapsto A\left(\mathbf{q},\mathbf{p}\right), \end{aligned}$$

with the $2m$ components denoted by A_i, $i = 1,\ldots,2m$. The space $\mathcal{D}$ of all dynamical variables is endowed with the inner products

$$\langle A|\, B\rangle_{ij} = \int A_j(\mathbf{x},\mathbf{p}) B_j(\mathbf{x},\mathbf{p}) \rho(\mathbf{x},\mathbf{p}) d\mathbf{x} d\mathbf{p}.$$

Here ρ is the canonical distribution $\rho(\mathbf{x},\mathbf{p}) = Z^{-1} e^{-\beta \mathcal{H}(\mathbf{x},\mathbf{p})}$ with partition function Z and inverse temperature β. We use the bracket formalism and denote the elements of $\mathcal{D}$ as *ket*-vectors $|A\rangle$.

A dynamical variable varies in time through its argument; a dynamical variable whose value at $t = 0$ was $A(\mathbf{x},\mathbf{p})$ acquires at time t the value $A(\varphi(\mathbf{x},\mathbf{p},t))$. One can also take a “Heisenberg” or “Lagrangian” point of view and introduce a time-dependent dynamical variable $e^{\mathcal{L}t}A$, where $\mathcal{L}$ denotes the Liouville operator defined by the Poisson Bracket

$$\mathcal{L} = \{\cdot, \mathcal{H}\} = \sum_{i=1}^{n} \left(\frac{\partial \mathcal{H}}{\partial p_i} \frac{\partial}{\partial x_i} - \frac{\partial \mathcal{H}}{\partial x_i} \frac{\partial}{\partial p_i} \right). \tag{6.2}$$

The propagator $e^{\mathcal{L}t}$ allows us to define the time-dependent *ket*-vector

$$|A(t)\rangle \equiv \left| e^{\mathcal{L}t} A \right\rangle = A\left(\varphi(\mathbf{x},\mathbf{p},t)\right),$$

which obeys the Liouville equation

$$\frac{d}{dt} |A(t)\rangle = \mathcal{L}\, |A(t)\rangle\,. \tag{6.3}$$

The projection operator[42]

$$\mathcal{P} = \langle A|\, A\rangle^{-1} |A\rangle \langle A| \tag{6.4}$$

allows to separate the time dependence of the dynamical variable into a part within the linear subspace U spanned by the ket-vectors $|A_i\rangle$ and a part within the orthogonal subspace $U^{\perp}$ [197],

$$\frac{d}{dt} |A(t)\rangle = e^{\mathcal{L}t} \mathcal{P}\mathcal{L}\, |A\rangle + \int_0^t d\tau e^{\mathcal{L}(t-\tau)} \mathcal{P}\mathcal{L}\, |F(t)\rangle + |F(t)\rangle\,, \tag{6.5}$$

with the random force $|F(t)\rangle \equiv e^{(1-\mathcal{P})\mathcal{L}t}(1-\mathcal{P})\mathcal{L}\,|A\rangle$. The random force lies within the subspace $U^{\perp}$, i.e., $(1-\mathcal{P})\,|F(t)\rangle = |F(t)\rangle$, which allows to compute

$$\mathcal{P}\mathcal{L}\,|F(t)\rangle = \mathcal{P}\mathcal{L}(1-\mathcal{P})\,|F(t)\rangle = \langle \mathcal{L}(1-\mathcal{P})F(t)|\, A\rangle \langle A|\, A\rangle^{-1} |A\rangle\,.$$

Defining the *memory function* $\Gamma(t) \equiv \langle \mathcal{L}(1-\mathcal{P})F(t)|\, A\rangle \langle A|\, A\rangle^{-1}$[60] Eq. (6.5) becomes the Generalized Langevin Equation (GLE),

$$\frac{d}{dt} |A(t)\rangle = \mathcal{P}\mathcal{L}\, |A(t)\rangle + \int_0^t d\tau \Gamma(\tau)\, |A(t-\tau)\rangle + F(t). \tag{6.6}$$

Thus, the dynamics of $|A\rangle$ are split into the dynamics within U and a correction term which describes the evolution of the system in $U^{\perp}$. $F(t)$ is the *random force*[42] exerted by the uncoupled motion in $U^{\perp}$, i.e., $\langle F(t)|\, A\,(0)\rangle = 0$, with its realization depending on the chosen initial conditions for the orthogonal part of the motion. The energy uptake due to the random force $F(t)$ is counterbalanced by the generalized friction, as expressed formally by the generalized fluctuation-dissipation theorem

$$\left\langle F(t)|\, F(t')\right\rangle = \langle A|\, A\rangle\, \Gamma(t-t'). \tag{6.7}$$

6.2 Definition of Motion along Conformational Coordinate(s) as the Observable

Here we propose to apply the projection-operator formalism rigorously to the dynamics of suitably chosen collective degrees of freedom,

$$c_i = f_i(\mathbf{x}_1, \ldots, \mathbf{x}_N), \quad i = 1 \ldots m, \tag{6.8}$$

and derive equations of motion for them. To be specific, and for simplicity of notation, we consider the dynamics of *one* nonlinear collective variable c ($m = 1$), although the theory can be generalized to more dimensions in a straightforward manner. The dynamics of the collective degree of freedom c are best represented by motion along a suitably chosen one-dimensional submanifold $\mathcal{M} \subset \mathbb{R}^{3N}$ of the configurational space parameterized by c. However, in practice, at first not a f_i but a submanifold $\mathcal{M}$ that is able to represent the motion of interest will be chosen, and in turn the collective degree of freedom is defined as a projection to that submanifold.

To derive the equations of motions for the collective coordinate with the projection-operator formalism, the problem is recast in terms of a dynamical variable A with two components. The first component, A_1, is given by the projection of vector $\mathbf{x}$

$$\begin{array}{rcl} A_1: \quad \Gamma & \longrightarrow & \mathbb{R} \\ (\mathbf{x}, \mathbf{p}) & \mapsto & c = f(\mathbf{x}), \end{array}$$

and the second component by the orthogonal projection of the momentum $\mathbf{p}$ onto the tangential space $T_{f(\mathbf{x})}\mathcal{M}$ of the manifold to the point corresponding to parameter $f(\mathbf{x})$

$$\begin{array}{rcl} A_2: \quad \Gamma & \longrightarrow & T_{f(\mathbf{x})}\mathcal{M} \\ (\mathbf{x}, \mathbf{p}) & \mapsto & \dot{c} = \nabla_{\mathbf{x}} f \cdot \mathbf{M}^{-1}\mathbf{p}, \end{array} \tag{6.9}$$

where $\mathbf{M}$ is a diagonal mass matrix. For a one-dimensional equation of motion, a suitably chosen reduced mass μ is required, which is derived from Eq. (6.9) via the equipartition theorem, $\left\langle \dot{c}^2 \right\rangle =$

$(\beta\mu)^{-1}$. The mean squared velocity

$$\langle \dot{c}^2 \rangle = \int \left(\nabla_{\mathbf{x}} f \cdot \mathbf{M^{-1}p}\right)^2 \rho(\mathbf{x},\mathbf{p})\ d\mathbf{x} d\mathbf{p},$$

consists of a sum of pure terms $\sim p_i^2$ and mixed terms $\sim p_i p_j$. After integration over the momenta the mixed terms vanish, which allows, via $\int p_i^2 d\mathbf{p} = \beta^{-1} m_i$, to define the reduced mass as

$$\mu = \left(\int \sum_{i=1}^{n} \left(\frac{\partial f}{\partial x}_i \right)^2 \frac{1}{m_i} \rho(\mathbf{x}) d\mathbf{x} \right)^{-1}.$$

6.3 Equations of Motion for Conformational Coordinate(s)

The above definitions allows for the application of the projection operator formalism in order to derive the equations of motion for the collective degree of freedom(s). To this aim, and exploiting the fact that the two components of the dynamical variable A are conjugated variables, the system of the two first order GLEs, Eq. (6.6), is cast into one second order GLE. This is possible because the first components of random force and memory function vanish, which can be seen from

$$\mathcal{L} \left| A_1 \right\rangle = \sum_i \frac{\partial \mathcal{H}}{\partial p_i} \frac{\partial f}{\partial x_i} \in T_{f(\mathbf{x})} \mathcal{M},$$

such that the orthogonal part $(1 - \mathcal{P})\mathcal{L} \left| A_1 \right\rangle$ vanishes. Hence, the fluctuation-dissipation theorem, Eq. (6.7), simplifies accordingly to

$$\left\langle F_2(t) \right| \left. F_2(t') \right\rangle = \left\langle A \right| \left. A \right\rangle \gamma(t - t'), \tag{6.10}$$

with $\gamma(t) \equiv \Gamma_2(t)$.

The conventional way to proceed from here is to apply the linear projector $\mathcal{P}$, Eq. (6.4), to the remaining component $\mathcal{P}\mathcal{L} \left| A_2(t) \right\rangle$ of the first term in Eq. (6.6), thus obtaining an effective harmonic force Ω[198, 42]. However, here we avoid this harmonic approximation by adopting the nonlinear projection operator originally introduced by Zwanzig[195] and rediscovered recently[199]. Apart from introducing a dependency of γ on the ket-vector $\left| A \right\rangle$, this generalization does not change the above derivation, Eqs. (6.5-6.6)[199, 197].

To be able to project to a curved conformational coordinate we generalize the operator defined in Ref. [199] to

$$\mathcal{P} \left| . \right\rangle = \frac{1}{\rho^c(c, \dot{c})} \int \left| . \right\rangle \rho(\mathbf{x}, \mathbf{p})\, \mathrm{d}\Omega(c, \dot{c}), \tag{6.11}$$

with $\mathrm{d}\Omega(c, \dot{c}) := \delta\left(f(\mathbf{x}) - c\right) \delta\left(\nabla_{\mathbf{x}} f \cdot \mathbf{M}^{-1}\mathbf{p} - \dot{c}\right) \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{p}$. Here we have defined the *conformational density* $\rho^c(c, \dot{c})$ as the projection of the density in configurational space onto the conformational coordinate,

$$\rho^c(c,\dot{c}) = \int \rho(\mathbf{x},\mathbf{p}) \left\| \nabla_{\mathbf{x}} f \mathbf{M}^{-1/2} \right\|^2 \, \mathrm{d}\Omega(c,\dot{c}). \tag{6.12}$$

The mass matrix $\mathbf{M}$ would vanish if mass-weighted coordinates ($\tilde{\mathbf{x}}$ =$\mathbf{M}^{1/2}\mathbf{x}$ and $\tilde{\mathbf{p}} = \mathbf{M}^{-1/2}\mathbf{p}$) were used. Since the chosen example comprises masses in the range 12 to 16 atomic mass units and, therefore, the difference is small, we have not used mass-weighted coordinates in Chapter 8.

Applying the generalized projector shows that $\mathcal{PL}\,|A_2(t)\rangle$ is the expectation value of the potential force acting tangentially to $\mathcal{M}$ under all possible realizations of the trajectory and a correction term due to the curvature of the chosen parameterization f of $\mathcal{M}$,

$$\mathcal{PL}\,|A_2(t)\rangle = -\frac{1}{\rho^c(c,\dot{c})} \int \left[\nabla_{\mathbf{x}}\mathcal{V} \cdot \nabla_{\mathbf{x}} f \mathbf{M}^{\mathbf{-1}} - \nabla_{\mathbf{p}}\mathcal{H} \cdot \nabla_{\mathbf{x}} \left(\nabla_{\mathbf{x}} f \cdot \mathbf{M}^{-1}\mathbf{p} \right) \right] \rho(\mathbf{x},\mathbf{p}) \mathrm{d}\Omega(c,\dot{c}). \tag{6.13}$$

Defining the potential of mean force $W(c,\dot{c}) = -\beta^{-1}\log\rho^c(c,\dot{c})$, we show in the Appendix that its derivative

$$\frac{\partial W}{\partial c}(c,\dot{c}) \;\; = \;\; -\beta^{-1}\frac{1}{\rho^c(c,\dot{c})}\frac{\partial\rho^c(c,\dot{c})}{\partial c} \tag{6.14}$$

yields the right hand side of Eq. (6.13). In the linear case, integration on the momentum part can be carried out separately, such that the dependence on the velocities, $\dot{c}$, vanishes. This yields the final result

$$\mathcal{PL}\,|A_2(t)\rangle = \frac{1}{\mu}\frac{\partial W}{\partial c}(c). \tag{6.15}$$

For reasons of practicality we approximate in the non-linear case by averaging out the dependence on the velocities.

To cast the resulting equation into the more usual form[43, 200, 201] of a second order GLE, we set $R(t) = \mu F_2(t)$, and from Eq. (6.6) one obtains

$$\mu\ddot{c}(t) = -\frac{d}{dc}W(c(t)) - \int_0^t d\tau\mu\gamma(t-\tau,c)\dot{c}(\tau) + R(t), \tag{6.16}$$

which is, except for the approximation in Eq. (6.15) for non-linear f, the *exact* equation of motion for the projected dynamics. Its right hand side is composed of a potential of mean force W, a generalized friction γ, and a random force R. The latter two obey the corresponding fluctuation-dissipation theorem,

$$\langle R(0)|\; R(t)\rangle = \mu\beta^{-1}\gamma(t). \tag{6.17}$$

The computation of the random force $R(t)$ requires solution of a Liouville equation which is far more complicated than the original unprojected problem. The advantage of the reformulation of the equations of motion in the form of the GLE, Eq. (6.16), is, of course, that the random force can be replaced by a stochastic term, i.e., a randomly generated force with similar statistical properties. In particular, its autocorrelation function has to satisfy the fluctuation-dissipation theorem Eq. (6.7).

Accordingly, we address in the following the task to extract the three components W, γ, and R from atomistic molecular dynamical simulations.

The potential of mean force $W(c) = -\beta^{-1} \log \rho^c(c)$ is obtained from the configurational density projected to the chosen collective coordinate(s). Here, the necessary canonical ensembles will be generated by MD simulations, but, indeed, any method that yields a canonical ensemble can be used, e.g., replica exchange molecular dynamics (REMD)[73], umbrella sampling[202, 75], or metadynamics[203].

The generalized friction is extracted from MD simulations. Here relatively short trajectories contain already sufficient information, because the respective memory kernels typically decay rapidly. The extraction of memory kernels is treated in Chapter 7.

The autocorrelation function of the random force process $R(t)$ is determined by the memory kernel via the fluctuation-dissipation theorem, Eq. (6.17). In Sec. 6.4.1 we present an elegant strategy to generate a sequence of random numbers which is unbiased apart from a predetermined autocorrelation.

6.4 Integration of the generalized Langevin equation

In order to obtain trajectories of the motion in the conformational subspace we need to integrate the GLE, Eq. (6.16). The random force process $R(t)$ does not depend on the trajectory and can, thus, be generated independently of the integration. The most time consuming part of the integration is the evaluation of the convolution of the memory kernel with the past velocities. Thus, we focussed our efforts for speeding up the calculation on an efficient computation of this convolution via FFT.

6.4.1 Generation of a random force

To generate instances of the random force process $R(t)$ for a given memory kernel $\gamma(\tau)$ *via* the fluctuation-dissipation theorem, we follow the method proposed in Ref. [200], which is exact and unbiased in contrast to other methods [204, 201, 205]. Briefly, the Wiener-Khintchin Theorem is exploited, which connects the spectral density

$$J(\omega) = \int_{-\infty}^{\infty} dt \left\langle R(0) \right| R(t) \rangle \, e^{-i\omega t} \tag{6.18}$$

to the power spectrum

$$J(\omega) = \left| \int_{-\infty}^{\infty} dt R(t) e^{-i\omega t} \right|^2 . \tag{6.19}$$

Hence the average amplitude of the Fourier transformed noise $\langle |R(\omega)| \rangle$ is determined by the memory function $\gamma(t) = \mu\beta^{-1} \left\langle R(0) \right| R(t) \rangle$. This is achieved by

$$R(t) = \int_{-\infty}^{\infty} d\omega \sqrt{J_K(\omega)} z(\omega) e^{i\omega t},$$

where $z(\omega) \in \mathbb{C}$ are realizations of a normal distribution with unit variance, and J_K is the spectral density corresponding to the given memory function

$$J_K(\omega) = m\beta \int_{-\infty}^{\infty} dt\gamma(t)e^{-i\omega t}.$$

This method allows generation of noise with arbitrary given autocorrelation.

6.4.2 Integration of the GLE

In order to integrate the GLE we used an algorithm of Tuckerman et al.[206], which is based on the velocity Verlet scheme[207]

$$\begin{aligned} c_{n+1} &= c_n + \Delta t\dot{c}_n + \frac{(\Delta t)^2}{2\mu} f_n, & (6.20) \\ \dot{c}_{n+1} &= \dot{c}_n + \frac{(\Delta t)^2}{2\mu}\left(f_n + f_{n+1}\right), & (6.21) \end{aligned}$$

where $c_n = (n\Delta t)$, etc., and f_n denotes the force at the nth time step. The force is given by the GLE, Eq. (6.16), which in discrete notation is written as

$$f_n = \frac{\partial W}{\partial c}(c_n) - \Delta t \sum_{k=0}^{n} \omega_{n-k}\gamma_{n-k}\dot{c}_k + R_n, \tag{6.22}$$

where ω_k are suitable quadrature weights (e.g., $\omega_0 = \omega_n = 1/2$, $\omega_k = 1$, if the trapezoidal rule is used). Substitution of Eq. (6.22) into Eq. (6.20) gives a direct method for obtaining the positions. The velocity equation, Eq. (6.21), however, requires f_{n+1}, which involves $\dot{c}_{n+1}$. Therefore, Tuckerman et al. separated out the unknown term by writing

$$f'_{n+1} := f_{n+1} + \Delta t\omega_0\gamma_0\dot{c}_{n+1}.$$

Replacing f_{n+1} and f_n in Eq. (6.21) and solving for $\dot{c}_{n+1}$ yields the result

$$\dot{c}_{n+1} = \frac{\dot{c}_n + (\Delta t/2\mu)\left(f_n + f'_{n+1}\right)}{1 + (\Delta t)^2\omega_0\gamma_0/2\mu}.$$

In Chapter 8, we require to integrate the corresponding (non-generalized) Langevin equation. To this end we set $\omega_0\gamma_0 = 2\gamma^c/\Delta t$ and $\gamma_k = 0$ for $k > 0$, and replaced the random forces by $R_n = (2kTm\gamma^c/\Delta t)^{-1/2}\xi_n$, where the ξ_n are independent Gaussian random variables with zero mean and $\langle\xi_n^2\rangle = 1$, and Δt denotes the integration time step.

6.4.3 Treatment of the convolution integral with FFT

The bottleneck of the integration of the GLE is the evaluation of the convolution sum

$$C_n = \sum_{k=0}^{n} \gamma_{n-k} \dot{c}_k \omega_k \tag{6.23}$$

(cf. Eq. (6.22)).

Our wish is to exploit FFT, which allows for the efficient computation of families of discrete convolutions

$$s_m := \sum_{k=0}^{M-1} \gamma_{m-k}^{\#} v_k^{\#}, \tag{6.24}$$

with periodically defined $v_k^{\#} = v_{jM+k}^{\#}$, $\gamma_k^{\#} = \gamma_{jM+k}^{\#}$, where $m,\, j = 0, \pm 1, \pm 2, \ldots$. The role of FFT is defined by the discrete convolution theorem, which shows that the discrete Fourier transform S_l of the sequence s_m is the product of the discrete Fourier transforms V_l and Γ_l of $v_k^{\#}$ and $\gamma_k^{\#}$, respectively. Thereby, the whole family of convolution sums is computed simultaneously reducing the complexity of the computation from $O\left(M^2\right)$ to $O\left(M \log M\right)$[208, 209]. Thus, for a specific hard- and software dependent M^o the FFT method becomes faster than a direct summation. Here, on an AMD 1.8GHZ Opteron with the FFTW Library Ver 3.0[210] this cross-over against a direct evaluation of the sum using the BLITZ++ package[211] was found to be already at $M^o = 8$.

However, a direct application of FFT is hindered, because the family of convolution sums, Eq. (6.23), differs from the definition of the family s_m in two ways. First, the upper limit of the summation varies with time $n\Delta t$, and second, at time $n\Delta t$ velocities $\dot{c}_k,\ k \geq n$ are not yet known. Nonetheless, the convolution theorem can be exploited by defining suitable sub-families of convolution sums[212, 213]

$$F_{\nu,\mu}(n) := \sum_{k=\nu}^{\mu-1} \gamma_{n-k} \dot{c}_k \omega_k.$$

Setting $v_k^{\#} := \dot{c}_{k+\nu} \omega_{k+\nu}$, $\gamma_k^{\#} := \gamma_k$, and $m = n - \nu$ we get

$$F_{\nu,\mu}(m) = \sum_{k=0}^{\mu-\nu-1} \gamma_{m-k}^{\#} v_k^{\#}.$$

Furthermore, the upper limit of the summation is extended to $2\left(\mu - \nu\right) - 1$ without changing $F_{\nu,\mu}(m)$ by defining $v_k^{\#} := 0$ for $k \geq \mu - \nu$. Thus, from the similarity with Eq. (6.24) with $M = 2\left(\mu - \nu\right)$, we know that for $m = 0, \ldots, M-1$ the sub-family of convolutions $F_{\nu,\mu}(m)$ can be computed efficiently with FFT.

This computation is carried out as soon as $n = \mu$ using just those velocities $\dot{c}_k,\ k < \mu$, which are already known. At this time only the latter $M/2$ of the convolutions $F_{\nu,\mu}(n)$ are of interest to us, since C_n for $n < \mu$ has been computed already. In spite of this waste, the simultaneous determination of the parts $F_{\nu,\mu}(n)$ of the $\mu - \nu$ subsequent convolution sums C_n for $\mu \leq n < 2\mu - \nu$ with FFT is more efficient than a direct evaluation of the complete sum for every integration step.

This efficient computation of subfamilies $F_{\nu,\mu}$ is exploited for the sequence of convolutions C_n as follows. The first M^o convolutions are computed by a direct evaluation of the sum, Eq. (6.23). Then the first FFT convolutions $F_{0,M^o}(n)$ are carried out. The following $M^o \leq n < 2M^o$ convolutions C_n are computed by separating C_n into

$$\begin{aligned} C_n &= F_{0,M^o}(n) + \sum_{k=M^o}^{n} \gamma_{n-k}\dot{c}_k\omega_k \\ &= F_{0,M^o}(n) + F_{M^o,n}(n). \end{aligned}$$

For $2M^o \leq n < 4M^o$ we discard $F_{0,M^o}(n)$ and compute instead $F_{0,2M^o}(n)$. As soon as $n > 3M^o$ an additional $F_{2M^o,3M^o}(n)$ yields parts of the C_n for $3M^o \leq n < 4M^o$. For $n \geq 4M^o$ both sub-families, $F_{0,2M^o}(n)$ and $F_{2M^o,3M^o}(n)$, are expired, and are replaced by $F_{0,4M^o}(n)$, which yields values for $4M^o \leq n < 8M^o$, and so on. In summary

$$C_n(n) = \begin{cases} F_{0,n}(n) & n < M^o \\ F_{0,M^o}(n) + F_{M^o,n}(n) & (M^o \leq n < 2M^o) \\ F_{0,2M^o}(n) + F_{2M^o,n}(n) & (2M^o \leq n < 3M^o) \\ F_{0,2M^o}(n) + F_{2M^o,3M^o}(n) + F_{3M^o,n}(n) & (3M^o \leq n < 4M^o) \\ F_{0,4M^o}(n) + F_{4M^o,n}(n) & (4M^o \leq n < 5M^o) \\ F_{0,4M^o}(n) + F_{4M^o,5M^o}(n) + F_{5M^o,n}(n) & (5M^o \leq n < 6M^o) \\ \dots \end{cases}$$

Therefore, at most $M^o - 1$ terms of a single convolution sum are computed by direct evaluation, whereas all the other terms are obtained efficiently with FFT via the sub-families $F_{\nu,\mu}(n)$.

6.5 Appendix

In this appendix we show that Eq. (6.14) evaluates to the force term $\mathcal{PL}\,|A_2(t)\rangle$. To simplify notation, we use mass-weighted coordinates ($\tilde{\mathbf{x}}$ =$\mathbf{M}^{1/2}\mathbf{x}$ and $\tilde{\mathbf{p}} = \mathbf{M}^{-1/2}\mathbf{p}$) and define

$$\begin{aligned} \delta c &:= \delta\left(f(\mathbf{x}) - c\right) \\ \delta v &:= \delta\left(\nabla_{\mathbf{x}} f \cdot \mathbf{p} - \dot{c}\right). \end{aligned}$$

For the proof we will need the relations

$$\begin{aligned} \nabla_{\mathbf{x}} f \cdot \nabla_{\mathbf{x}} \delta c &= \delta'(f(\mathbf{x}) - c) \, \|\nabla_{\mathbf{x}} f\|^2 && (6.25) \\ \nabla_{\mathbf{x}} f \cdot \nabla_{\mathbf{x}} \delta v &= \nabla_{\mathbf{p}} \delta v \cdot \nabla_{\mathbf{x}} \left(\nabla_{\mathbf{x}} f \cdot \mathbf{p} \right), && (6.26) \end{aligned}$$

which are easily shown by applying the chain rule to the delta-functions.

Consider the derivative of $\rho(c, \dot{c})$, which appears in Eq. (6.14). As seen from the definition, Eq. (6.12), its dependence on c is restricted to the delta-function δc. Therefore,

$$\frac{\partial \rho(c, \dot{c})}{\partial c} = - \int \rho(\mathbf{x}, \mathbf{p}) \, \|\nabla_{\mathbf{x}} f\|^2 \; \delta'(f(\mathbf{x}) - c) \, \delta v \, \mathrm{d}\mathbf{x} \mathrm{d}\mathbf{p},$$

which is transformed via relation Eq. (6.25) to

$$\frac{\partial \rho(c, \dot{c})}{\partial c} = - \int \rho(\mathbf{x}, \mathbf{p}) \nabla_{\mathbf{x}} f \nabla_{\mathbf{x}} \delta c \, \delta v \, \mathrm{d}\mathbf{x} \mathrm{d}\mathbf{p}.$$

Having brought the integrand into this suitable form, the derivative of the delta-function δc is easily removed via partial integration, which yields

$$\frac{\partial \rho(c, \dot{c})}{\partial c} = \int \left\{ \left[(-\beta) \nabla_{\mathbf{x}} \mathcal{V} \cdot \nabla_{\mathbf{x}} f + \nabla_{\mathbf{x}} \cdot \nabla_{\mathbf{x}} f \right] \delta v + \nabla_{\mathbf{x}} f \nabla_{\mathbf{x}} \delta v \right\} \rho(\mathbf{x}, \mathbf{p}) \delta c \, \mathrm{d}\mathbf{x} \mathrm{d}\mathbf{p},$$

where we have used that $\rho(\mathbf{x}, \mathbf{p}) = Z^{-1} \exp\left(-\beta \mathcal{H}\right)$. To remove also the newly appeared derivative of δv, we employ Eq. (6.26) and integrate partially over momentum space. This yields

$$\frac{\partial \rho(c, \dot{c})}{\partial c} = \int \left\{ -\beta \nabla_{\mathbf{x}} \mathcal{V} \cdot \nabla_{\mathbf{x}} f + \beta \nabla_{\mathbf{p}} \mathcal{H} \cdot \nabla_{\mathbf{x}} \left(\nabla_{\mathbf{x}} f \cdot \mathbf{p} \right) \right\} \rho(\mathbf{x}, \mathbf{p}) \delta c \, \mathrm{d}\mathbf{x} \mathrm{d}\mathbf{p},$$

since the remaining terms

$$\nabla_{\mathbf{x}} \cdot \nabla_{\mathbf{x}} f - \nabla_{\mathbf{p}} \cdot \nabla_{\mathbf{x}} \left(\nabla_{\mathbf{x}} f \cdot \mathbf{p} \right)$$

cancel each other out. Thus, comparison with Eq. (6.13) shows that

$$\frac{1}{\rho(c, \dot{c})} \frac{\partial \rho(c, \dot{c})}{\partial c} = -\beta \mathcal{P} \mathcal{L} \left| A_2(t) \right\rangle .$$

One must have a good memory to be able to keep the promises one makes.
— Friedrich Nietzsche

Chapter 7

Extraction of Memory Kernels from Molecular Dynamics Simulations

In the previous chapter we have shown that the collective Langevin dynamics (CLD) is governed by a generalized Langevin equation (GLE). This equation accounts for memory effects of the collective dynamics by a convolution of past velocities with a *memory kernel*. The memory kernel decays rapidly, such that it can be extracted from short MD simulations. In this chapter suitable extraction methods are developed and evaluated.

In a first step, we neglect the dependence of the memory kernel $\gamma(t, c)$ on the position c of the conformation coordinate c, hence we extract a spatially averaged $\gamma(t)$. In a second step, we also investigate methods to extract the full $\gamma(t, c)$, since this will allow to improve the CLD model by considering the spatial dependence of $\gamma(t, c)$.

The general strategy is to obtain memory from either a force autocorrelation function (FACF) or a velocity autocorrelation function (VACF) of the relevant degrees of freedom. The first approach either exploits the Kubo relation[196] or directly the fluctuation-dissipation theorem, Eq. (6.17). For the latter one needs to obtain an autocorrelation function of $R(t)$, i.e., the *random force*, which is detailed in Section 7.5. The second approach exploits the Memory equation, a Volterra-type equation, which connects the VACF $\Psi(t)$ to the spatially averaged memory kernel $\gamma(t)$. Solving this equation, however, is not straightforward, hence a detailed investigation is required.

A solution of the Memory equation for a VACF $\Psi(t)$ given a memory $\gamma(t)$ is straightforward. One simply integrates the GLE, Eq. (6.16), (setting $W \equiv 0$), and computes the autocorrelation of the obtained velocities.

However, we have to consider the inverse problem: Solving the Memory equation for $\gamma(t)$ with a given VACF $\Psi(t)$ is very challenging indeed, because it suffers from instabilities arising from a fundamental ill-posedness of the problem. In previous examples, solving the Memory equation for single particle VACFs[62, 64, 65, 67, 68, 69], the effects of this instability were not as severe as encountered here. This finding can be attributed to an earlier observation, that the statistical error of single particle VACFs is much lower than that of VACFs of collective degrees of freedom[214]. Here, the arising instability cannot be counteracted with ad-hoc measures anymore, as done, e.g., with a 6th-order interpolation of the VACF by Berne et al.[60], or by application of infinite-precision arithmetics by Kneller et al.[61].

The instabilities of methods based on the Memory equation have not attracted much attention so far and were — to our knowledge — never explicitly attributed to the underlying ill-posedness of the problem. Here we will analyze the ill-posedness in detail, because a thorough grasp of the difficulties allows to search systematically for solutions. Gaining thus a generalized perspective on the problem should allow to take advantage of the extensive literature on solution of *inverse problems*[215, 216, 217, 218, 219, 220, 221, 222].

The general strategy is to regularize the problem in order to neutralize its ill-posedness. We will give a short summary about the most important regularization methods and discuss the established techniques to solve the Memory equation[60, 61] in light of this new view. Furthermore, we propose new methods to solve the problem, which apply two different regularization techniques.

7.1 The Memory equation

The Memory equation[60, 223, 224], a Volterra-type equation,

$$\frac{d}{dt}\Psi(t) = -\int_0^t d\tau\gamma(t-\tau)\Psi(\tau) \tag{7.1}$$

connects the VACF, $\Psi(t) = \langle \dot{c}(0) | \, \dot{c}(t) \rangle / \langle \dot{c}^2 \rangle$, with the memory kernel, $\gamma(t)$. Eq. (7.1) is obtained from the GLE, Eq. (6.16), without the potential ($W \equiv 0$) by application of $\mu^{-1} \left\langle \dot{c}^2 \right\rangle^{-1} \langle \dot{c}(0)|$ and noting that $\langle F(t) | \, A\,(0) \rangle = 0$.

The VACF $\Psi(t)$ can be computed readily from MD simulations. Thus, the memory kernel $\gamma(t)$ can be extracted by solving Eq. (7.1) given a $\Psi(t)$. Note, however, that this usual form of the Memory equation[60, 223, 224] yields an adulterated $\gamma(t)$, due to the additional velocity correlations, which are caused by the inertial motion of the system within the free energy surface and not by memory effects due to the eliminated degrees of freedom. We, therefore, suggest to consider the contribution from the potential separately: the additional term to Eq. (7.1) takes the form of a velocity/force correlation,

$$\Pi(t) \equiv \mu^{-1} \left\langle \dot{c}^2 \right\rangle^{-1} \left\langle \dot{c}(0) \frac{\partial W}{\partial c}(c(t)) \right\rangle, \tag{7.2}$$

and serves to quantify the accuracy of the usual approximation Eq. (7.1).

Indeed, in the application to neurotensin reported in Chapter 8 $\Pi(t)$ was found to be some magnitudes smaller than the other terms involved and was neglected. Therefore, for simplicity $\Pi(t) \equiv 0$ in the following discussion, although an extension of the presented methods to non-zero $\Pi(t)$ is straightforward.

Note that an unjustified treatment with $\Pi(t) \equiv 0$ and $W \equiv 0$, although consistent, alters the

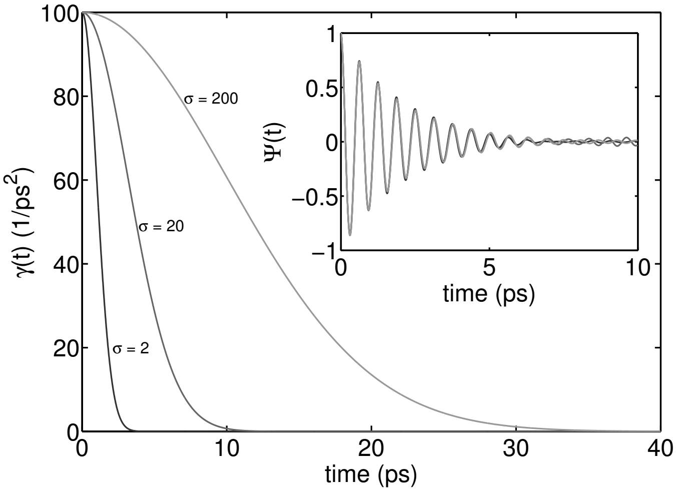


Figure 7.1: The memory equation is ill-posed. The plot shows three memory kernels, as defined by Eq. (7.7). The inset shows their corresponding VACFs, which are indistinguishable.

resulting memory kernel in the impractical and counterintuitive way that it does not decay to zero anymore. E.g., a motion of a mass m in a harmonic potential $W = 0.5\omega c^2$ can be accurately described by a GLE with $W \equiv 0$, but then $\gamma(\infty) = \omega/m$ asymptotically.

For completeness, we note that an alternative Memory equation can be obtained following Berkowitz et al.[66] by applying $\mu^{-1} \langle c(0)|$ to the GLE which leads in our case to

$$\langle \dot{c}(0)| \, \dot{c}(t)\rangle = \mu^{-1} \left\langle c(0)\frac{\partial W}{\partial c}(c(t))\right\rangle - \int_0^t \gamma(t-\tau) \, \langle c(0)| \, \dot{c}(t)\rangle \, . \tag{7.3}$$

However, we do not consider this any further, because it contains slowly converging positional contributions to the autocorrelation functions.

7.2 Ill-posedness of the Memory equation

Solving the Memory equation, Eq. (7.1) for the memory kernel γ given a VACF with statistical noise is challenging, because this type of equation, i.e., a Volterra equation of 1st kind, is known to suffer from various degrees of *ill-posedness*[225]. This means our problem does not fulfill Hadamard's definition of *well-posedness*, i.e., at least one of the following properties does not hold[215]

For all admissible data, a solution exists (7.4)

For all admissible data, the solution is unique (7.5)

The solution depends continuously on the data (7.6)

To illustrate in which way these properties are violated by the considered problem we go through them one at a time.

First, clearly a small perturbation of the velocity autocorrelation function might render $\dot{\Psi}(0) \neq 0$, which means that Eq. (7.1) has *no solution* anymore due to a vanishing integral on the right hand side for $t = 0$.

Uniqueness also poses a problem here, which is illustrated with a simple example in Figure 7.1. The plot shows three memory kernels generated by

$$\gamma(t) = 100e^{-t^2/2\sigma} + 2\delta(t), \tag{7.7}$$

with $\sigma = 2, 20, 200$, respectively. The inset shows that the corresponding three VACFs obtained numerically are identical up to deviations on the order of the statistical error. Seen as an analytical problem the inverse problem is well-posed, however, here it must be considered from a practical point of view. In particular, on finite intervals and with the presence of noese, the inverse problem has *no unique* solution for the VACF shown in the inset of Figure 7.1, which in turn also means that it does *not depend continously* on the data.

The usual remedy for this is a transformation of a Volterra equation of first kind

$$\dot{\Psi}(t) = \int_0^t \gamma(\tau)\Psi(t-\tau)$$

to a Volterra equation of second kind[226]

$$\Psi(0)\gamma(t) = -\ddot{\Psi}(t) - \int_0^t \dot{\Psi}(t-\tau)\gamma(\tau)d\tau,$$

which can be achieved by a simple differentiation. Although the differentiation itself increases the destabilizing effect of the noise in the data this transformation is often advantageous. It usually renders the problem more stable, because in the resulting equation $\gamma(t)$ would depend continuously on the (noisy) data $\ddot{\Psi}(t)$[225]. However, here the integrand $\dot{\Psi}(t-\tau)$ is also effected by noise such that the instability remains vigorous: denoting the noise in the input data $\dot{\Psi}(t)$ and $\ddot{\Psi}(t)$ by δ and ϵ, respectively, we note that in

$$\gamma(t) = -\ddot{\Psi}(t) + \epsilon(t) - \int_0^t \dot{\Psi}(t-\tau)\gamma(\tau)d\tau - \int_0^t \delta(t-\tau)\gamma(\tau)d\tau$$

the last integral might become rather large rendering the problem instable.

Taken together, we have shown that the considered problem is *ill-posed* on finite intervals. In absence of uniqueness a physical interpretation of the resulting memory kernels has to be treated with care. Nonetheless, any solution would have the desired effect to yield accurate dynamics, and thus we go further and regularize the problem.

7.3 Regularization of Inverse Problems

We now briefly introduce the main regularization techniques following[215] and investigate their applicability for the problem at hand. In order to simplify the following discussion, let us first introduce some notation. The Memory equation is reformulated as $\mathbf{Kf} = \mathbf{g}$ by setting

$$\begin{aligned} \mathbf{g} &= -\dot{\Psi}(t) \\ \mathbf{K}(\cdot) &= \int_0^t \Psi(t-\tau)\cdot(\tau)\mathrm{d}\tau \\ \mathbf{f} &= \gamma(t) \end{aligned}$$

The strategy is to define a new but related problem which fulfills Hadamard's definition of well-posedness. As first step we secure the first two conditions, i.e., *uniqueness* by singling out the smoothest solution, and *existence* by requiring only that the equation is solved *optimally.*

Formally, the solution to this new problem is denoted as $\mathbf{f}^\dagger = \mathbf{K}^\dagger \mathbf{g}$. $\mathbf{K}^\dagger$ is called the Moore-Penrose inverse[215] defined to yield the solution $\mathbf{f}^\dagger$ which minimizes the L_2-norm $\left\|\mathbf{K}\mathbf{f}^\dagger - \mathbf{g}\right\|_2$ and a certain side constraint $\Omega(\mathbf{f}) = \left\|\mathbf{L}\left(\mathbf{f}-\mathbf{f}^0\right)\right\|_2$. Since we aim in our application for smooth solutions, we set $\mathbf{f}^0 = 0$ and choose the second derivative operator $\mathbf{L} = \partial^2/\partial t^2$.

However, the Moore-Penrose inverse does not abolish the severe problems of numerical instabilities encountered in attempts of numerical solutions, which are due to the *non-continuous dependence* of the solution $\mathbf{f}^\dagger$ on the data $\mathbf{g}$. One has to go further and regularize the problem, i.e., replace the still ill-posed Moore-Penrose Inverse by a family of well-posed problems, whose solutions approximate the proper result[215].

Regularization aims at approximating $\mathbf{f}^\dagger$ for a specific right hand side $\mathbf{g}$ in the situation that the *exact data* $\mathbf{g}$ is not known precisely, but that only an approximation $\mathbf{g}^\delta$ with

$$\left\|\mathbf{g}^\delta - \mathbf{g}\right\| \leq \delta$$

is available[215]. Because $\mathbf{K}^\dagger\mathbf{g}^\delta$ is not a good approximation of $\mathbf{K}^\dagger\mathbf{g}$ due to the ill-posedness, one seeks approximations $\mathbf{f}_\alpha^\delta$, which, on the one hand, continously depend on the noisy data $\mathbf{g}^\delta$ and can, thus, be computed in a stable way, and, on the other hand, fulfill

$$\mathbf{f}_\alpha^\delta \longrightarrow \mathbf{f}^\dagger,$$

for vanishing noise-level δ and an appropriately chosen regularization parameters $\alpha(\delta, \mathbf{g}^\delta)$.A regularization technique, therefore, has to specify how a solution $\mathbf{f}_\alpha^\delta$ should be obtained in a stable way from the noisy data and how the regularization parameter α is chosen. Note that often the exact noise-level δ is not known, such that α has to be chosen purely on the grounds of $\mathbf{g}^\delta$.

Many different regularization methods fitting this general framework have been developed, some yield the solution directly for a given regularization parameter, whereas others yield the family $\mathbf{f}_\alpha^\delta$ iteratively.

In the following the popular *Tikhonov* and *Landweber* regularizations are evaluated as representatives of a direct and an iterative method, respectively. Additionally, a different approach regularizes by projection to a finite dimensional subspace, i.e., by decreasing the degrees of freedom. This can be achieved by discretization, collocation or approximation through certain models, and is often combined with other regularization methods. In Section 7.3.3 we gain a rather strong regularization by allowing only solutions compatible to a three parameter family of memory functions. Another application of this regularization technique is reviewed in Section 7.3.4; Kneller et al. projected to a finite dimensional solution-space by using auto-regressive models with up to 1000 parameters[61].

7.3.1 Discretization of Memory equation

In this section the discretization of the Memory equation is introduced, which is a technical but necessary step for the following numerical treatment. We explicitly account for a possible delta-function contribution $\gamma(t) = 2\gamma^c\delta(t) + \breve{\gamma}(t)$. The added term yields a memory free contribution to the dissipative forces. As seen from the fluctuation dissipation theorem, Eq. (6.17), this leads also to contribution of random forces with a uniform spectrum, i.e., white-noise. Inserting this in the Memory equation

$$\dot{\Psi}(t) = -\gamma^c\Psi(t) - \int_0^t \Psi(t-\tau)\breve{\gamma}(\tau)d\tau$$

the discretization $\Psi_i = \Psi(i\Delta t)$, $\dot{\Psi}_i = \dot{\Psi}(i\Delta t)$ and $\gamma_i = \breve{\gamma}(i\Delta t)$ yields

$$\dot{\Psi}_i = -\gamma^c\Psi_i - \Delta t\sum_{k=0}^{i-1}\gamma_k\Psi_{i-k}\omega_{ik} - \gamma_i\Psi_0\omega_{ii}\Delta t \tag{7.8}$$

with quadrature weights ω_{ik}, e.g., for the trapezoidal rule

$$\omega_{ik} = \begin{cases} 1/2 & k = 0 \vee k = i \\ 1 & \text{otherwise.} \end{cases} \tag{7.9}$$

Note that $\breve{\mathbf{K}}$ has a Toeplitz structure, i.e., $\breve{\mathbf{K}}_{ij} = \breve{\mathbf{K}}_{i-j,0}$ for $i > j$, if the quadrature weights are chosen accordingly $\omega_{ij} = \omega_{i-j}$. This can be exploited by the algorithms to simplify computations.

Note further that $\mathbf{L}$ is constructed to enforce smoothness on $\breve{\gamma}(t)$ exclusively, i.e., the part of the memory kernel, which does not contain the delta-function contribution.

7.3.2 Straightforward recursion formulas without regularization

Before we turn to regularization techniques we introduce now un-regularized recursion formulas, which exploit the Toeplitz structure of the discretization above. Because such a recursion formula going back to Berne and Harp was used extensively to extract memory kernel's from MD simulations of simple liquids[60, 65, 227, 228, 67], we test this approach for its applicability in the context of CLD.

$$\mathbf{f} = \begin{pmatrix} \gamma^c \\ \gamma_0 \\ \gamma_1 \\ \vdots \\ \gamma_{n-1} \end{pmatrix} \qquad \mathbf{K} = \begin{pmatrix} \Psi_0 & \\ \Psi_1 & \\ \Psi_2 & \breve{\mathbf{K}}\Delta t \\ \vdots & \\ \Psi_{n-1} & \end{pmatrix} \qquad \mathbf{L} = \Delta t^{-2} \begin{pmatrix} 0 & 1 & -2 & 1 & & \\ 0 & & \ddots & \ddots & \ddots & \\ 0 & & & 1 & -2 & 1 \end{pmatrix}$$

$$\mathbf{g} = -\begin{pmatrix} \dot{\Psi}_0 \\ \dot{\Psi}_1 \\ \dot{\Psi}_2 \\ \vdots \\ \dot{\Psi}_{n-1} \end{pmatrix} \quad \breve{\mathbf{K}} = \begin{pmatrix} \Psi_0\omega_{00} & & & & \\ \Psi_1\omega_{10} & \Psi_0\omega_{11} & & & \\ \Psi_2\omega_{20} & \Psi_1\omega_{21} & \ddots & & \\ \vdots & \vdots & & \Psi_1\omega_{n-2,n-2} & \\ \Psi_{n-1}\omega_{n-1,0} & \Psi_{n-2}\omega_{n-1,1} & \cdots & \Psi_0\omega_{n-1,n-2} & \Psi_0\omega_{n-1,n-1} \end{pmatrix}$$

Volterra 1st kind

The discretized Volterra equation of first kind, Eq. (7.8), can be reorganized, keeping in mind that $\Psi_0 = 1$, to obtain the iterative formula

$$\gamma_i = -\frac{1}{\Delta t \omega_{ii}} \left[\gamma^c \Psi_i + \Delta t \sum_{k=0}^{i-1} \gamma_k \Psi_{i-k} \omega_{ik} + \dot{\Psi}_i \right] \quad i \geq 1 \tag{7.10}$$

for numerical solution provided that γ_0 is known.

Volterra 2nd kind

The discretization of the Volterra equation of second kind

$$\ddot{\Psi}(t) = -\gamma^c \dot{\Psi}(t) - \Psi(0)\gamma(t) - \int_0^t \dot{\Psi}(t-\tau)\gamma(\tau) d\tau,$$

which is derived from Eq. (7.8) by differentiating. Exploiting the Toeplitz structure it yields the iterative formula

$$\gamma_i = -\frac{1}{1 + \dot{\Psi}_0 \omega_{ii} \Delta t} \left[\ddot{\Psi}_i + \gamma^c \dot{\Psi}_i + \Delta t \sum_{k=0}^{i-1} \gamma_k \dot{\Psi}_{i-k} \omega_{ik} \right] \quad i \geq 0. \tag{7.11}$$

This equation also defines the starting value $\gamma_0 = -\left(\ddot{\Psi}_0 + \gamma^c \dot{\Psi}_0\right) / \left(1 + \dot{\Psi}_0 \Delta t / 2\right)$ used for both recursion formulas.

Method of Berne and Harp

Berne and Harp used Eq. (7.11) with $\gamma^c \equiv 0$ and assumed $\dot{\Psi}_0 = 0$[60]. They used a second order Gregory formula for quadrature, which requires that the first 4 points are obtained via a dedicated starting method[229]. Furthermore, the first and second derivatives of the VACF are obtained from a fit to the VACF for the first 4 points, and from local 6th order polynomial interpolations otherwise[60].

7.3.3 Strong regularization by projection

A rather strong regularization is gained by allowing only solutions compatible to the three parameter family of memory functions

$$\gamma_{\text{fit}} = 2\gamma^c \delta(t) + Ae^{-at}. \tag{7.12}$$

The Memory equation, Eq. (7.1), can be solved analytically for this memory function, which yields a VACF that can be fitted to the data, thereby determining the parameters of γ_{fit}. Moreover, the pairs of VACF and memory kernel will be used to generate problems with analytically known solution for the evaluation of numerical methods in Section 7.4.

The equation is solved by means of a Laplace transformation, which simplifies the one-sided convolution to a product in Laplace space. In particular, the Laplace transform of the Memory equation

$$z\hat{\Psi}(z) - 1 = -\hat{\gamma}(z)\hat{\Psi}(z)$$

yields, together with the Laplace transform of the memory kernel $\hat{\gamma}(z) = \gamma^c + A/(z+a)$, the transformed VACF $\hat{\Psi}(z) = (z + \gamma^c + A/(z+a))^{-1}$, whose back-transformation is

$$\Psi_{\text{fit}}(t) = \exp\left(-\frac{a+\gamma^c}{2}t\right)\left[\cosh(Rt/2) + \frac{a-\gamma^c}{R}\sinh(Rt/2)\right], \tag{7.13}$$

with $R := \sqrt{a^2 - 2a\gamma^c + \gamma^{c2} - 4A}$. This VACF is fitted to the MD results to obtain the parameters γ^c, a and A

of the model memory function $\gamma_{\text{fit}}(t)$.

7.3.4 Weak regularization by projection to auto-regressive models

The most recently developed approach to compute memory functions from molecular dynamics simulations was proposed by Kneller and Hinsen[61]. Without explicitly mentioning it, their success in outperforming the more established methods, e.g. the method of Berne and Harp, results from regularization by projection (see above). Their strategy is to use an auto-regressive (AR) model for the underlying stochastic process

$$x(t) = \sum_{n=1}^{P} a_n^{(P)} x\left(t - n\Delta t\right),$$

where P denotes the AR-order and Δt the sampling frequency. The coefficients $a_n^{(P)}$ are obtained by fitting the AR-model to the computationally obtained VACF, and the Memory equation is solved via Laplace transformations. As direct consequence of this ansatz the memory function of order P rapidly falls off to zero for $t > P\Delta t$[61].

Their approach regularizes the problem by reducing the number of degrees of freedom to the AR-order P. With high AR orders the regularization is weakened such that the crucial step of their calculation, the inversion of the Laplace transformed expression of the memory function, becomes instable. They counteracted this problem by applying a high-precision arithmetic with floating point numbers having a 150-bit mantissa whenever $P > 85$[61]. They obtained memory functions for liquid argon with $P = 250$[61] and, very recently, for the Fourier transformed particle density of lysozyme with $P = 400$ and $P = 1000$[230].

7.3.5 Regularization by Landweber's iterations

Landweber's iterative method exploits that often the direct problem $\mathbf{g} = \mathbf{K}\mathbf{f}$ is well-posed such that it can be efficiently and stably evaluated. The iterative scheme

$$\mathbf{f}_k = \mathbf{f}_{k-1} + \eta\mathbf{K}^{\mathrm{T}}\left(\mathbf{g} - \mathbf{K}\mathbf{f}_{k-1}\right), \tag{7.14}$$

where $0 < \eta \leq 1$ is a suitable relaxation parameter, converges against the solution $\mathbf{f}_k \longrightarrow \mathbf{K}^{\dagger}\mathbf{g}$, and it has been shown[215] that

$$\left\|\mathbf{f}_k - \mathbf{f}_k^{\delta}\right\| \leq \sqrt{k}\delta,$$

where $\mathbf{f}_k^{\delta}$ denotes the sequence one obtains by replacing $\mathbf{g}$ against $\mathbf{g}^{\delta}$ in Eq. (7.14). Therefore, the total error $\left\|\mathbf{f}_k^{\delta} - \mathbf{K}^{\dagger}\mathbf{g}\right\|$ has two components, an *approximation error* converging to zero, and a *data error* of the order of at most $\sqrt{k}\delta$. Consequently, for small values of k the data error is negligible and the iteration seems to converge against the exact solution. When $\sqrt{k}\delta$ reaches the order of magnitude of the approximation error the available solution $\mathbf{f}_k^{\delta}$ starts to deviate again from the exact solution $\mathbf{K}^{\dagger}\mathbf{g}$.

This behavior is known as *semi-convergence[215]*. The regularizing property of this method ultimately depends on a reliable stopping rule for detecting the transient from convergence to divergence. The stopping rule has, therefore, the role of the regularization parameter α. The convergence, however, is slow, such that the principle was advanced to the *accelerated Landweber iteration* and the ν*-method*[231, 232].

Lets consider whether and how this scheme can be applied to the problem at hand. A direct numerical solution of the Memory equation, Eq. (7.1), if γ is known, suffers from similar instability problems[225] as the solution of the inverse problem, and, therefore, nothing would be gained by applying Landweber's iteration in this way.

Nevertheless, as noted previously, the VACF Ψ corresponding to a given memory kernel γ can be stably obtained by stochastic integration of the GLE, Eq. (6.16). However, this operation $\Psi = \mathbf{F}(\gamma)$ is non-linear, i.e., $\mathbf{F}\left(\gamma_1 + \gamma_2\right) \neq \mathbf{F}\left(\gamma_1\right) + \mathbf{F}\left(\gamma_2\right)$. Thus, the update rule for the linear case, Eq. (7.14), $\eta\mathbf{K}^{\mathrm{T}}\left(\mathbf{g}-\mathbf{K}\mathbf{f}_{k-1}\right)$, which is the negative gradient of the linear error $\|\mathbf{g}-\mathbf{K}\mathbf{f}_{k-1}\|^2$, would have to be replaced by the negative gradient of the non-linear error $\|\Psi_{k-1}-\mathbf{F}\left(\gamma_{k-1}\right)\|^2$, which is disproportionate harder to compute. Furthermore, each evaluation of $\mathbf{F}\left(\gamma\right)$ involves a costly stochastic integration of the GLE. Consequently, this iterative scheme, and any of its accelerated variants could not straightforwardly be applied to the problem at hand.

7.3.6 Tikhonov regularization

The *Tikhonov regularization*[216, 233] technique defines a regularized solution $\mathbf{f}_\alpha^\delta$ as the minimizer of a weighted combination of the residual norm and the side constraint. In this combination

$$\min\left\{\left\|\mathbf{K}\mathbf{f}-\mathbf{g}^\delta\right\|^2+\alpha^2\Omega(\mathbf{f})\right\}, \tag{7.15}$$

called *Tikhonov criterion,* the regularization parameter α controls the weight given to minimization of the side constraint relative to minimization of the residual norm. Clearly, with our choices of $\mathbf{f}^0=0$ and $\mathbf{L}$ the second derivative operator, a large α favors smooth solutions at the cost of a larger residual norm, while a small α has the opposite effect.

The solution is obtained from the equivalent formulation

$$\min\left\{\left\|\begin{pmatrix}\mathbf{K}\\ \lambda\mathbf{L}\end{pmatrix}\mathbf{f}-\begin{pmatrix}\mathbf{g}\\ \alpha\mathbf{L}\mathbf{f_0}\end{pmatrix}\right\|^2\right\},$$

which yields the *normal equation*

$$\left(\mathbf{K}^{\mathrm{T}}\mathbf{K}+\alpha^2\mathbf{L}^{\mathrm{T}}\mathbf{L}\right)\mathbf{f}=\mathbf{K}^{\mathrm{T}}\mathbf{g}+\alpha\mathbf{L}^{\mathrm{T}}\mathbf{f_0}. \tag{7.16}$$

In the case $\mathbf{L}=\mathbf{I}$ and $\mathbf{f_0}\equiv 0$ this can be simplified using singular value decomposition, which yields two orthonormal matrices $\mathbf{U},\mathbf{V}$, such that $\mathbf{K}=\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}}$, with $\mathbf{\Sigma}$ a diagonal matrix. Then the normal equation can be written as

$$\begin{aligned}\left[\left(\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}}\right)^{\mathrm{T}}\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}}+\alpha^2\mathbf{I}\right]\mathbf{f} &= \left(\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}}\right)^{\mathrm{T}}\mathbf{g}\\ \mathbf{V}\left[\mathbf{\Sigma}^2+\alpha^2\mathbf{I}\right]\mathbf{V}^{\mathrm{T}}\mathbf{f} &= \mathbf{V}\mathbf{\Sigma}\mathbf{U}^{\mathrm{T}}\mathbf{g}\\ \mathbf{f} &= \mathbf{V}\left[\mathbf{\Sigma}^2+\alpha^2\mathbf{I}\right]^{-1}\mathbf{\Sigma}\mathbf{U}^{\mathrm{T}}\mathbf{g}.\end{aligned}$$

In the general case $\mathbf{L}\neq\mathbf{I}$ one either transforms the equation into the standard form such that $\mathbf{L}=\mathbf{I}$ or transforms the normal equations using generalized singular value decomposition (GSVD)[218]. Here we employed the latter approach.

GSVD yields for a given $p\times n+1$ matrix $\mathbf{L}$ and a $n+1\times n+1$ matrix $\mathbf{K}$ the orthonormal matrices $\mathbf{U}\in\mathbb{R}^{n\times n+1}$,$\mathbf{X}\in\mathbb{R}^{n+1\times n+1}$ and $\mathbf{V}\in\mathbb{R}^{p\times p}$, such that

$$\begin{aligned}\mathbf{K} &= \mathbf{U}\begin{pmatrix}\mathbf{\Sigma} & 0\\ 0 & \mathbf{I}_{n+1-p}\end{pmatrix}\mathbf{X}^{-1}\\ \mathbf{L} &= \mathbf{V}\left(\mathbf{M},0\right)\mathbf{X}^{-1}.\end{aligned}$$

$\mathbf{\Sigma}$ and $\mathbf{M}$ are $p \times p$ diagonal matrices: $\mathbf{\Sigma} = \text{diag}\left(\sigma_1, \ldots, \sigma_p\right)$, $\mathbf{M} = \text{diag}\left(\mu_1, \ldots, \mu_p\right)$. Then the regularized solution $\mathbf{f}_\alpha^\delta$ is given by

$$\mathbf{f}_\alpha^\delta = \sum_{i=1}^{p} \frac{\sigma_i}{\left(\sigma_i^2 + \mu_i^2\alpha^2\right)}\left(\mathbf{u}_i^{\mathrm{T}}\mathbf{g}^\delta\right)\mathbf{x}_i + \sum_{i=p+1}^{n+1}\left(\mathbf{u}_i^{\mathrm{T}}\mathbf{g}^\delta\right)\mathbf{x}_i,$$

with $\mathbf{u}_i$, $\mathbf{x}_i$ denoting the i-th column of $\mathbf{U}$ and $\mathbf{X}$, respectively[218].

Note that, with a single GSVD suffices to compute solutions for all regularization parameters. An alternative method to solve the normal equations uses Hausholder transformations and Givens rotation and is described elsewhere[215]. A more efficient implementation is gained by exploiting the Toeplitz structure of $\mathbf{K}$[234, 217]. However, we were not concerned with efficiency in the evaluation of this method, and used GSVD because an implementation could be obtained readily for MATLAB(tm)[218].

7.3.7 Sequential Tikhonov regularization

Solution of the Tikhonov criterion, Eq. (7.15), becomes computationally very costly, if large matrices are involved, i.e., the operation count is $O(n^3)$, where n is the dimensionality of the discretized equations. Sequential Tikhonov regularization[219] exploits the Toeplitz structure, i.e., $\mathbf{K}_{ij} = \mathbf{K}_{i-j}$, by separating the problem into smaller overlapping Tikhonov problems. Applying this approach to the Memory Equation, we took advantage of this idea, and enhanced the method to have smooth solutions across the separation boundaries.

Assuming the equation has been solved for $\gamma^c, \gamma_0, \ldots, \gamma_{i-1}$, $i > 0$, we separate from the sum in the l-th equation

$$\dot{\Psi}_l \quad = \quad -\gamma^c\Psi_l - \sum_{k=0}^{l}\gamma_k\Psi_{l-k}\omega_{lk}\Delta t$$

the part, which is already determined by the previously solved values of γ

$$\dot{\Psi}_l = -\gamma^c\Psi_l - \sum_{k=0}^{i-1}\gamma_k\Psi_{l-k}\omega_{lk}\Delta t - \sum_{j=i}^{l}\gamma_j\Psi_{l-j}\omega_{lj}\Delta t.$$

This allows formulation of a new Volterra equation

$$\dot{\Psi}_l + \gamma^c \Psi_l + \sum_{k=0}^{i-1} \gamma_k \Psi_{l-k} \omega_{lk} \Delta t = -\sum_{j=i}^{l} \gamma_j \Psi_{l-j} \omega_{lj} \Delta t,$$

which becomes clearer (sic) after reassigning indices. Set $t := j - i + 1$ and $p := l - i + 1$, then

$$\dot{\Psi}_{i+p-1} + \gamma^c \Psi_{p+i-1} + \sum_{k=0}^{i-1} \gamma_k \Psi_{p+i-1-k} \omega_{p+i-1,k} \Delta t = -\sum_{t=1}^{p} \gamma_{t+i-1} \Psi_{p-t} \omega_{(p+i-1)(t+i-1)} \Delta t.$$

adopts the typical form of a Volterra equation by definition of a new unknown $\beta_t^{(i)} := \gamma_{t+i-1}$ and replacing the known left hand side by $h_p^{(i)}$

$$h_p^{(i)} = \sum_{t=1}^{p} \beta_t^{(i)} \Psi_{p-t} \hat{\omega}_{pt} \Delta t, \tag{7.17}$$

where $\hat{\omega}_{pt} := \omega_{(p+i-1)(t+i-1)}$. For trapezoidal rule

$$\hat{\omega}_{pt} = \begin{cases} 1/2 & p = t \\ 1 & \text{sonst} \end{cases}.$$

This new Volterra equation, Eq. (7.17), with the operator matrix

$$\mathbf{K}_{\text{seq}} = \begin{pmatrix} \Psi_0 \hat{\omega}_{11} & & & & \\ \Psi_1 \hat{\omega}_{21} & \Psi_0 \hat{\omega}_{22} & & & \\ \Psi_2 \hat{\omega}_{31} & \Psi_1 \hat{\omega}_{21} & \ddots & & \\ \vdots & \vdots & & \Psi_1 \hat{\omega}_{r-1,r-1} & \\ \Psi_{r-1} \hat{\omega}_{r,1} & \Psi_{r-1} \hat{\omega}_{r,2} & \cdots & \Psi_0 \hat{\omega}_{r,r-1} & \Psi_0 \hat{\omega}_{r,r} \end{pmatrix}$$

can be solved using standard Tikhonov regularization for $p = 1, \ldots, r$. Note that $\mathbf{K}_{\text{seq}}$ is independent of i such that a single GSVD suffices for all overlapping subproblems. For every sequential step the first $m < r$ new results are stored, $\gamma_{p+i-1} = \beta_p^{(i)} \quad p = 1, \ldots, m$, and the others discarded. Then a new left hand side, $h_p^{(i+m)}$, of Equation 7.17 is computed, and the process is repeated with $i \longrightarrow i + m$. The overlap m and the look-ahead r can be chosen freely.

To allow the regularization criterion to act also upon the connection points, we extended the seq. Volterra equation, Eq. (7.17), such that the two last points of the previous step are included and kept fixed. This is achieved by using for all but the first subproblems the operator matrix

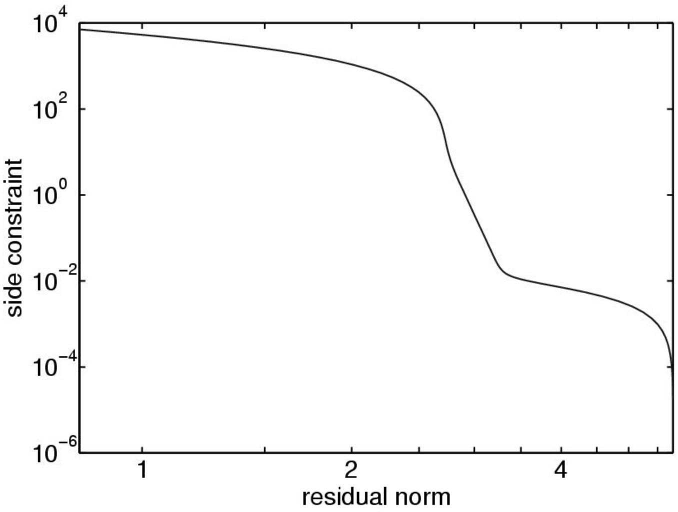


Figure 7.2: An example of an L-curve. The plot shows the residual norm and the norm of the side-constraint for a wide range of regularization parameters. (This L-curve was obtained for solutions of the test-problem 'Realistic' in Sec. 7.4.2).

$$\tilde{\mathbf{K}}_{\text{seq}} = \begin{pmatrix} C & 0 & 0 \\ 0 & C & 0 \\ 0 & 0 & \mathbf{K}_{\text{seq}} \end{pmatrix}$$

and the vector $\tilde{\mathbf{h}}^{(i)} = \left(C\beta_{m-1}^{(i-1)}, C\beta_{m}^{(i)}, h_1^{(i)}, ..., h_r^{(i)} \right)$, where $C = 1000\alpha$ guarantees that the first two points of the subproblem are not altered.

In summary, this scheme allows to break down an n-dimensional minimization problem into several pieces of r dimensional problems. Due to the $O(n^3)$ complexity of the GSVD the sequential computation is much faster, although in every step $r - m$ of the computed values are discarded.

7.3.8 Choice of regularization parameter by means of L-curve

In Figure *7.2* we show an example of a popular graphical tool for determination of the optimal regularization parameter α. For a wide range of α the so-called *L-curve* plots the residual norm $\left|\left|\mathbf{Kf} - \mathbf{g}^{\delta}\right|\right|^2$ of the solution against its side constraint $\Omega(\mathbf{f})$ in double logarithmic form[235, 236].

At the corner of the L-curve, in the illustration at $\left(3.6, 10^{-2}\right)$, is the optimal balance between residual norm and the regularizing side constraint.

The vertical part of the L-curve corresponds to solutions where $\Omega\left(\mathbf{f}\right)$ is sensitive to changes in α, whereas the horizontal part correspond to a region where the changing α affects the residual norm $\left|\left|\mathbf{Kf} - \mathbf{g}^{\delta}\right|\right|^2$ more strongly. The absence of such a corner would indicate that the regularization criterion should be chosen differently. Automatic rules for identification of that corner exist[236, 237].

7.4 Evaluation of methods to solve the Memory equation

In the previous sections several methods to extract memory functions from VACFs were introduced. Here, we evaluate these methods by means of VACFs whose corresponding memory kernel $\gamma(t)$ is

Table 7.1: Parameters to define velocity autocorrelation functions by Eq. (7.13) for testing.

	Slow	Slow-White	Fast	Fast-White	Realistic
a	1	0.1	0.5	0.1	1
γ^c	0	2	0	0.2	13
A	1	1	3	3	30

known analytically.

To this end, we generated five memory functions using Eq. (7.12) and computed the corresponding VACFs via Eq. (7.13). This choice of examples is justified, because it will be shown in Chapter 8 that Eq. (7.13) fits the VACF of collective motion of neurotensin remarkably well, and that a CLD model based on this fit predicts conformational transition rates accurately. Thus, basic features of VACFs obtained for collective motion that are relevant to CLD are captured by this family of functions.

The first four example VACFs, that are, ’Slow’, ’Slow-White’, ’Fast’, and ’Fast-White’, were chosen to sample a spectrum of fast and slow decay, whereas the last example ’Realistic’ was motivated by the typical set of parameters one obtains from fitting Eq. (7.13) to VACFs obtained from MD simulations (cf. Table 8.1).

One of the parameters, shown in Table 7.1, i.e., γ^c, is crucial. It controls a fast initial drop of the memory function, which will be shown to be rather pronounced in the MD results. To elucidate whether this fast initial drop was responsible for problems with the evaluated algorithms γ^c was set to zero in two test examples (Slow and Fast).

7.4.1 Method of Evaluation

The parameter sets shown in Table 7.1 were used to generate memory functions with a sampling time step of $\Delta t = 10\,\text{fs}$. The GLE, Eq. (6.16), was integrated numerically with $W \equiv 0$, reduced mass $\mu = 1$ and the inverse temperature $\beta = 1$ to generate GLE trajectories of $5\,\text{ns}$ length, from which VACFs were computed. Note that the resulting VACF does not depend on μ and β, because $W \equiv 0$.

The recursion formulas of first and second kind, Eq. (7.10) and Eq. (7.11), respectively, were solved iteratively with quadrature weights of the trapezoidal rule, Eq. (7.9). The starting values were determined from numerical first and second derivatives of the VACF (cf. Sec. 7.3.2), and the parameter γ^c by fitting, Eq. (7.13), to the data.

For evaluation of the autoregressive method we used the respective routines of the program nMoldyn[238] with $P = 250$ and two different time-steps, $\Delta t = 0.1\text{ps}$ and $\Delta t = 0.01\text{ps}$.

Our method based on sequential Tikhonov regularization was applied with look-ahead $r = 600$ and overlap $m = 350$ (cf. Sec. 7.3.7). The parameter γ^c and the starting value via $\gamma(0) = -\left(\ddot{\Psi}(0) + (\gamma^c)^2\right) / (1 + \gamma^c \Delta t/2)$ were determined by fitting Eq. (7.13) to the data. These two parameters were subsequently held constant during the optimization of the Tikhonov criterion, Eq. (7.15). In a second application of sequential Tikhonov regularization to the example ’Realistic’ γ^c and $\gamma(0)$ was not predetermined by fitting. The regularization parameter $\alpha = 54$ was determined from the L-curve (cf. Sec. 7.3.8) of the ’Realistic’ example, and was used also for all other test

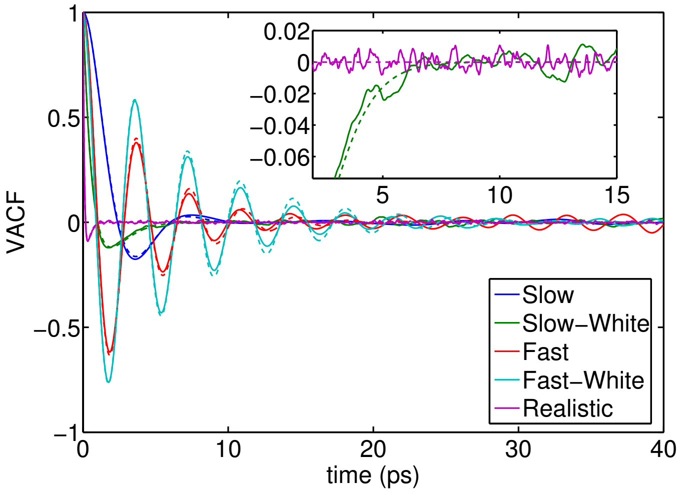


Figure 7.3: Velocity autocorrelation functions of test examples. The **solid** lines are statistically perturbed VACF (see text). The **dashed** lines or the respective color denote the corresponding unperturbed VACF. The **inset** shows selected curves enlarged.

examples.

7.4.2 Results

The generated example VACFs are shown in Figure 7.3. They contained the statistical noise typical in applications to MD data. This is made obvious in the inset, which shows small fluctuations of the generated VACF, whereas the analytical VACF (dashed), Eq. (7.13), was zero. These perturbed example VACFs will challenge the memory extraction methods in a realistic manner.

The four panels of Figure 7.4 show the memory functions computed for the first four VACFs with the following four different methods:

A first kind recursion, Eq. (7.10)

B second kind recursion, Eq. (7.11)

C autoregressive (AR) model by Kneller et al., Sec. 7.3.4

D sequential Tikhonov regularization, Sec. 7.3.7

The overview shows that all methods yielded accurate memory kernels from the VACFs ’Slow’ and ’Fast’. However, the examples with a fast initial drop, i.e., ’Slow-White’ and ’Fast-White’, were only solved satisfactorily by sequential Tikhonov regularization (cf. Fig 7.4d). In the following the results shown in Figure 7.4a-d are detailed for each method separately:

Method **A**, which does not regularize, yields solutions for examples with $\gamma^c > 0$, which fluctuated around the analytical solution. Moreover, the memory for ’Slow’ deviated from the correct solution in form of a small dip $t < 0.05$ ps.

Method **B** relies also on a recursion formula without regularization, but was based on the second kind Volterra equation, which is often considered to be more stable (cf. discussion above, or Refs. [60, 225]). Nonetheless, this method was not able to solve any of the examples with fast initial drop. In

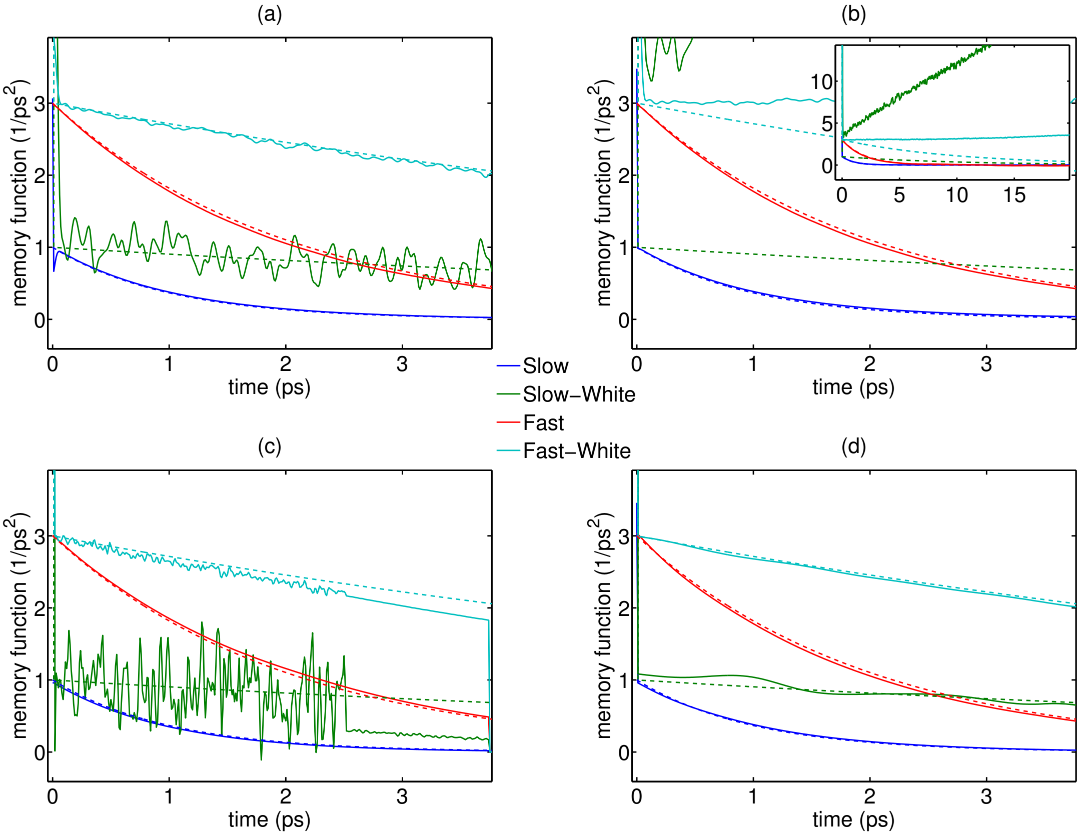


Figure 7.4: Memory functions computed for the first four test-examples (see legend). The dashed lines denote the analytical solution for the unperturbed VACF, whereas the solid lines denote the numerical solution obtained from the statistically perturbed VACF. The evaluated methods are as follows (see also text): **(a)** recursion formula first kind **(b)** recursion formula second kind **(c)** autoregressive models **(d)** sequential Tikhonov regularization. The inset in (b) shows the same data on a larger scale.

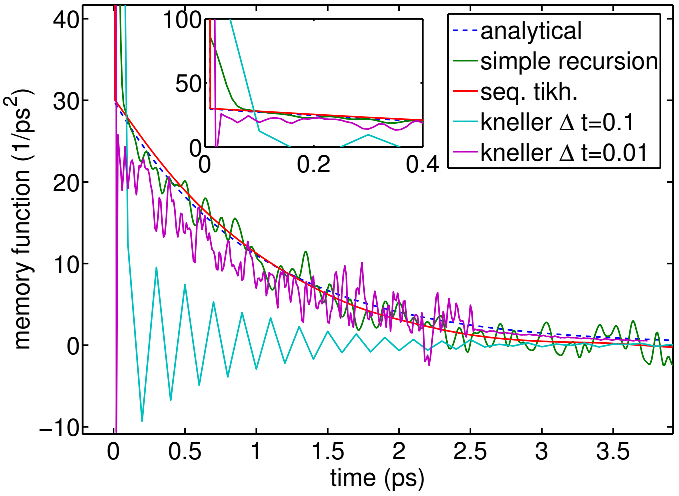


Figure 7.5: Memory functions of test example 'Realistic' computed with different methods (see legend). The dashed curve indicates the analytical solution of the unperturbed data. The inset shows the same data with enlarged time axis.

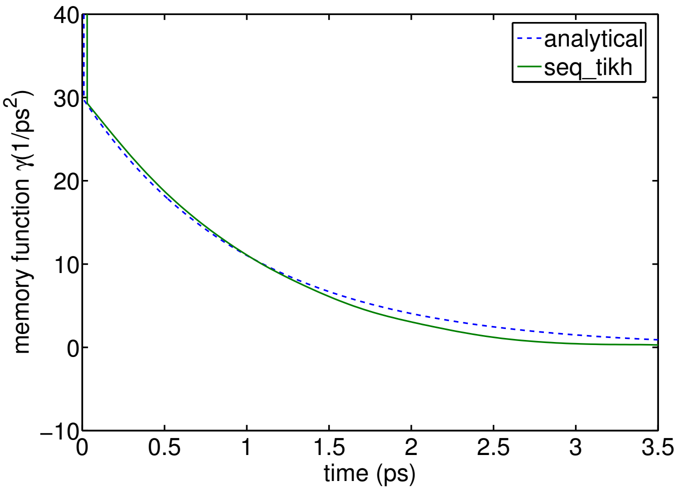


Figure 7.6: Memory function of test example 'Realistic' computed with sequential Tikhonov regularization without fit to determine $\gamma(0)$ and γ^c. The dashed curve indicates the analytical solution of the unperturbed data.

these cases the numerical solutions blew up very fast (cf. inset Fig. 7.4b) due to the noise in the data. Accordingly, the same algorithm accurately solved the Memory equation, provided the unperturbed VACF, Eq. (7.13), and its analytical derivatives were used as input (not shown).

Both of the previous recursion formulas were also tested with quadrature weights derived from the midpoint rule, which did not improve the results. We also tested the established method of Berne and Harp[60], which is based on the same recursion formula as method B (cf. Sec. 7.3.2), but uses a higher order quadrature to increase numerical accuracy. Despite the increased effort solutions of the notorious cases blew up, too (results not shown).

Figure 7.4c shows the solutions obtained with the AR method (**C**). They are more accurate than those of the second kind recursion formula B, but did not improve upon the first kind recursion A. Moreover, the character of the solution abruptly changes at $t = 2.5$ns. This is caused by the inability of this approach to capture correlations in the VACF, which go beyond $t = P\Delta t$. This early cut-off can be avoided by increasing either Δt or the AR-order P. The computationally effort increases strongly with P, however. Already with $P = 250$ the computation was drastically slower than all the other methods tested, hence a further test with a higher AR-order was not deemed necessary. Instead we will solve the 'Realistic' example also with a larger time-step (see below).

Figure 7.4d shows the memory kernels obtained with method **D,** sequential Tikhonov regularization. All curves match the analytical solution, although small deviations are seen for the example 'Slow-White'. In particular the fast oscillations, observed in all solutions of alternative methods, do not emerge here.

Finally, all methods were applied to the data set 'Realistic' that resembled the VACFs from MD simulations most closely. Fig. 7.5 shows the memory kernels obtained with the respective method. Also for this example, the most accurate solution was obtained with sequential Tikhonov regularization. As seen from the enlargement in the inset, it accurately resolves the fast initial decay and starts the slower decay at the correct value.

The simple recursion formula (method A) also yields a fairly accurate solution. However, it is superposed with fast oscillations and deviates from the analytical solution for very small times (see inset).

The AR memory kernel is not only superposed with fast oscillations, but is also too low on average and deviates for small times with high amplitude aberration towards negative values. Also in this example, the AR-solution abruptly changes its character at $t = 2.5$ ns, due to the cut-off of the AR-model. To avoid this cut-off without increasing the computational effort, a second AR-solution was obtained with a ten times larger sampling time-step, that is $\Delta t = 0.1$ ps. However, the coarse sampling drastically decreased the performance (cf. Fig. 7.5).

The sequential Tikhonov method determined the starting value $\gamma(0)$ and γ^c by fitting to Eq. (7.13). Since the examples were generated with the same equation, this method might be over-adapted to the used examples. Therefore, we also evaluated an alternative implementation of this method, which does not rely on this fit. Instead, all values $\gamma(0)$,γ^c and the $\gamma(t_i)$ were directly determined by minimization of the Tikhonov criterion. Indeed, as seen in Figure 7.6, this yields an accurate memory function, too.

7.4.3 Discussion and Outlook

We showed that our method, based on sequential Tikhonov regularization, robustly retrieved all analytically known solutions of all test examples including the one that closely resembled the realistic VACF obtained from MD. In particular, it was robust against the statistical perturbations present in the input data.

On the contrary, none of the established methods was able to accurately solve all problems. These methods failed exactly in those cases where the analytical solution initially dropped to a small fraction of its starting value $\gamma(0)$, i.e., $\gamma^c > 0$. This drop causes a large discretization error in the convolution part of the Memory equation, since its decay time is smaller than the used discretization ($\Delta t = 10$ fs). An explicit treatment of the drop as a delta-function contribution to the memory improved the numerical stability but did not remove it. Thus, the solution cannot be represented sufficiently well in the discretized form. This explains why only regularization techniques, which look for an *optimal* solution were successful. Accordingly, the effect of the discretization error, which is equivalent to a considerable perturbance of the input data, explains the emergence of strong oscillations in the unregularized results. Note that the initial drop, and thus the concomitant numerical problem, does not arise in single-particle memory kernels[62, 64, 65, 67, 68, 69].

The AR-method did not cope with the notorious problems, although it applies a regularization controlled by its AR-order parameter. To increase the strength of its regularization would require to lower the AR-order P. This, however, would decrease the cut-off time $t = \Delta P$, which was already to low with $P = 250$. Furthermore, the method was computationally very slow due to its application of infinite-precision arithmetics. Thus, this method does not pose a valuable approach in the framework of CLD.

The method based on sequential Tikhonov regularization allows to use a fit of Eq. (7.13) to the given VACF to determine the starting value $\gamma(0)$ and the friction constant γ^c. These two parameters are excluded from the subsequent co-optimization of the solution and the regularization criterion. In this way one can exploit that the VACFs of MD simulations are well fitted by Eq. (7.13). In Chapter 8 this method is, therefore, used in this form to determine the memory kernels for the CLD model of the conformational motion of neurotensin. In cases, where Eq. (7.13) badly fits the data, the Tikhonov criterion is instantaneously optimized for $\gamma(0)$ and γ^c together with the other values $\gamma(t_i)$. For the presented test examples both approaches worked equally well.

We propose to test further criteria for regularization. Asymptotically vanishing memory functions can be favored by weighting with a suitable function, e.g., minimizing a side constraint $\Omega\left(\gamma\right) = \int_0^t \gamma(t)\left(1 - \exp\left(-t/\tau\right)\right) \mathrm{d}t$. Moreover, the accuracy of the fast initial decay of the memory kernel might suffer from a strong smoothing. Therefore, one could relax the enforcement of smoothness for small times, while keeping it up for larger times. Moreover, further knowledge about memory functions, e.g., that it has only positive Fourier-coefficients or that $\int_0^\infty \gamma(t)\mathrm{d}t = \left(\int_0^\infty \Psi(t)\mathrm{d}t\right)^{-1}$[223], can be used.

Finally, accuracy might be improved by simultaneously minimizing the Tikhonov criterion for several independently obtained VACFs, instead of applying the method to their average.

7.5 Determination of memory via force autocorrelation functions

We now turn to the alternative strategy extracting a memory kernel from an MD simulation via a force autocorrelation function (FACF). The FACF of the *random* force $R(t)$ is fundamentally related to the memory function $\gamma(t)$ via the fluctuation-dissipation theorem, Eq. (6.17). However, only the total force $f(t)$ can be directly computed from MD simulations.

One possible approach uses the Kubo-relation[196] to relate the FACF of the total force with the FACF of the random force. We do not pursue this approach though, because results were found to be unconvincing for collective motions in a previous study[239].

Instead, we employ a *fixed-particle* or *infinite-mass* approximation[63, 70], which allows to separate the random force $R(t)$ from the total force $f(t)$. To this end, $f(t)$ acting in direction of a conformational coordinate c frozen at $c \equiv c_0$ is recorded during a dedicated MD simulation. The employed constraint renders $\dot{c} \equiv 0$, such that the frictional forces vanish from the GLE, Eq. (6.16). Thus, $f(t) = -W'(c_0) + R(t)$, which allows to compute the random force as $R(t) = f(t) - \langle f(t) \rangle$.

We tested this approach at the example of the conformational motion of the peptide neurotensin, which will be discussed in the next Chapter. A one-dimensional conformational coordinate will be developed as active subspace for a CLD model, which is used here to test the extraction of memory via FACFs.

Note that this extraction method is particular interesting, because it allows to compute the dependence of the memory function $\gamma(t, c)$ on the position c, which was neglected so far. This approach enables us to either justify this approximation or to improve the accuracy at a later stage by using

$\gamma(t, c)$ for the CLD model. Here we perform a preliminary investigation of the dependence of the memory function on the position of the conformational coordinate.

7.5.1 Methods

To compute FACFs in the fixed particle approximations the collective coordinate was constrained using the Essential Dynamics SAMpling (EDSAM) module of GROMACS[97, 114]. We extended its implementation to correct the projected forces for translational and rotational motion, as explained in Sec. 2.3.2.

The curved conformational coordinate c of the one dimensional CLD model of neurotensin, introduced in Chapter 8 was fixed at 60 evenly spaced positions $c\left(\mathbf{z}_i\right)$ $i = 1, 21, 41 \ldots, 1181$, where $\mathbf{z}$ and i refer to the discretization chosen for the coordinate in Sec. 8.2.1.

The MD simulations of neurotensin were set up as described in Sec. 2.3.1 with all but the collective degree of freedom free to move. At every integration step ($\Delta t = 2\,\text{fs}$) the MD forces in cartesian space corrected for translational and rotational motion were projected onto tangents $\mathbf{t}_i$ to the conformational coordinate in points $c\left(\mathbf{z}_i\right)$. The tangents $\mathbf{t}_i$ were described with mass-weighted coordinates, hence the corresponding reduced mass was $\mu = 1$. For every fixed position, 5 trajectories of 2 ns were generated. Starting structures were randomly chosen from those structures $\mathbf{x}_j$ of NT1, whose projection was close to the root point $\mathbf{z}_i$ of the respective tangent $\mathbf{t}_i$, i.e., $|\mathbf{t}_i \cdot (\mathbf{x}_j - \mathbf{z}_i)| < 0.1\,\text{nm}\,\text{u}^{-1/2}$.

FACFs $C^{(i)}_{\text{RR}}(t) = \langle R(0)|\, R(t)\rangle$ were computed from the projected and centered forces $R(t) = f(t) - \langle f(t)\rangle$ and averaged over the five trajectories. They determined the memory function via the fluctuation-dissipation theorem, Eq. (6.17), as $\gamma^{(i)}_{\text{FACF}}(t) = \mu^{-1}\beta C^{(i)}_{\text{RR}}(t)$.

Decay times $\tau^{(i)}_{\text{env}}$ of the memory function were computed from its negative part as

$$\tau^{(i)}_{\text{env}} = 2\frac{\int_{\gamma<0}\left(\gamma^{(i)}_{\text{FACF}}(t)\right)^2 t\,\mathrm{d}t}{\int_{\gamma<0}\left(\gamma^{(i)}_{\text{FACF}}(t)\right)^2 \mathrm{d}t},$$

and a corresponding ’enveloping’ curve, was plotted as $e^{(i)}_{\gamma} = \exp\left(-t/\tau_i\right)$.

Running averages $\overline{g}(t)$ of a function $g(t)$ were computed by convolution with a Gaussian kernel, i.e., $\overline{g}(t) = \int_0^\infty g(t-\tau)k_\sigma(\tau)\mathrm{d}\tau$, where $k_\sigma = (2\pi)^{-1/2}\,\sigma^{-1}\exp(-t^2/\sigma^2)$.

7.5.2 Memory kernel from constrained particle force autocorrelation function

Now we proceed and extract memory kernels for the curved collective coordinate of the conformational motion of neurotensin. Numerous ACFs of forces acting on different fixed positions along the conformational coordinate were obtained (see Methods). A typical FACF is shown in Fig. 7.7, which depicts $\gamma^{(691)}_{\text{FACF}}(t)$ obtained at position $c = 0.58$ of the coordinate.

After a fast initial drop from $6000\,\text{ps}^{-1}$ to $2500\,\text{ps}^{-1}$ the FACF oscillated strongly. The oscillations decayed on a $\tau_{\text{env}} = 0.23\,\text{ps}$ time-scale. All FACFs obtained at other positions showed the same

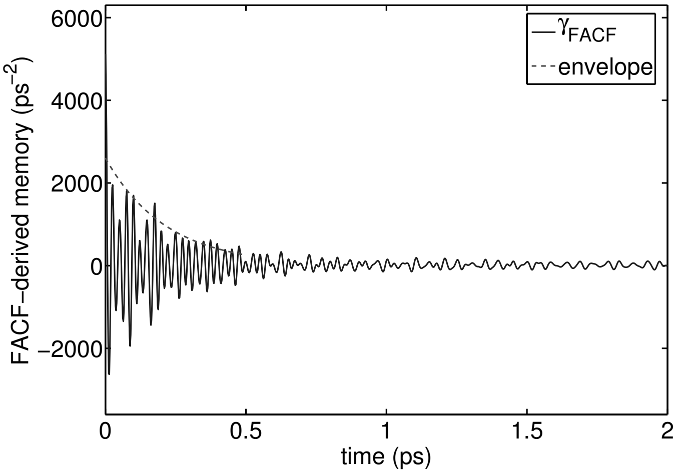


Figure 7.7: The memory kernel $\gamma_{\mathrm{FACF}}^{(691)}(t)$ derived from a force autocorrelation function (FACF)

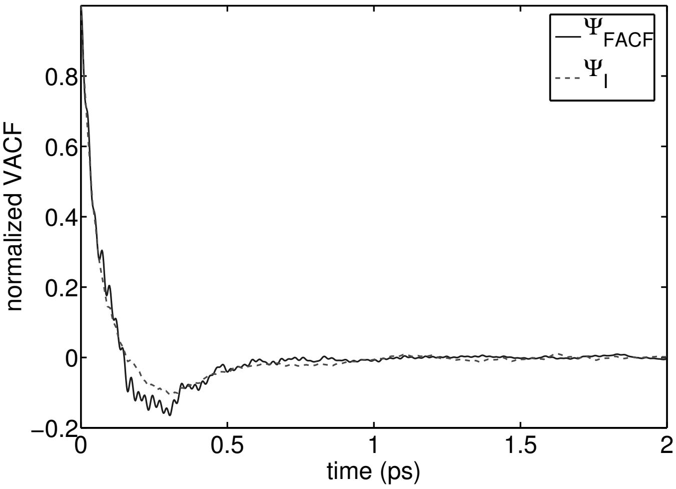


Figure 7.8: Comparison of velocity autocorrelation functions (VACF). The VACF Ψ_{FACF} (solid line) corresponds to the memory function $\gamma_{\mathrm{FACF}}^{(691)}(t)$, and the VACF Ψ_{I} corresponds to the VACF-derived memory function γ_{VACF} (γ_{I} in Chapter 8).

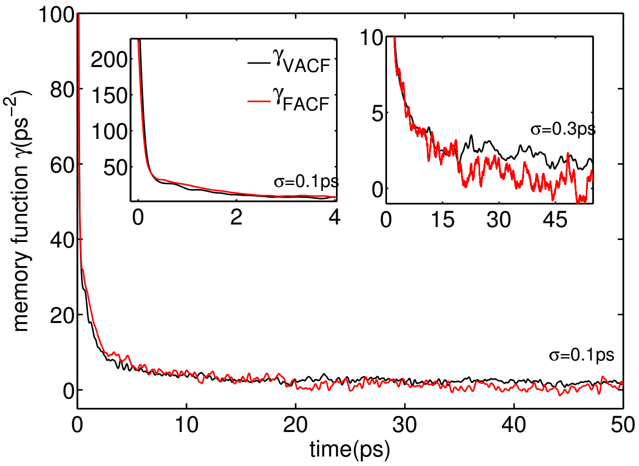


Figure 7.9: Gaussian smoothed memory kernels. This figure shows running averages of $\gamma_{\mathrm{FACF}}^{(691)}(t)$ and $\gamma_{\mathrm{VACF}}(t)$. The width of the Gaussian used for averaging is denoted in the figure ($\sigma = 0.1\,\mathrm{ps}/\sigma = 0.3\,\mathrm{ps}$).

Table 7.2: Results of fit of Eq. (7.18) to smoothed memory functions $\overline{\gamma}(t)(\sigma_{\text{gauss}} = 0.1\,\text{ps})$. The standard deviation σ indicates the range of values obtained for the 60 memory functions derived from FACFs.

	γ_{VACF}	$\gamma_{\text{FACF}} \pm \sigma$
τ_1	0.09 ps	(0.09 ± 0.008) ps
τ_2	1.75 ps	(1.5 ± 0.3) ps
τ_3	64 ps	(21 ± 13) ps
a_1	242 ps^{-1}	(200 ± 44) ps^{-1}
a_2	24 ps^{-1}	(40 ± 10) ps^{-1}
a_3	3.4 ps^{-1}	(10 ± 7) ps^{-1}

oscillating behavior and a similar decay time (not shown).

For comparison consider a VACF-derived memory kernel of the same conformational motion, e.g., γ_{I}, which is extracted from MD in Chapter 8 and re-plotted here in Fig. 7.9 as γ_{VACF}. Apparently, FACF-derived memory functions are strikingly dissimilar to VACF-derived memory functions. In particular, the strong oscillations are not present in VACF derived memory kernels.

To check if the FACF-derived memory is consistent with the conformational dynamics, nonetheless, we generated a CLD trajectory with this memory and computed its VACF. The result was quite unexpected. Fig. 7.8 contrasts the VACF, which corresponds to the oscillating memory, to the VACF of the free MD (dashed line), which corresponds to γ_{VACF}. Despite the enormous differences of the memory kernels the two VACFs are very similar.

The consistency of the resulting CLD dynamics suggests that both kinds of memory kernels share common features. Indeed, these were found by eradicating the strong oscillations of the FACF via smoothing with a Gaussian kernel of width $\sigma = 0.1\,\text{ps}$. As shown in Fig. 7.9, the running average $\overline{\gamma}_{\text{FACF}}(t)$ closely resembled the running average of the VACF-derived memory function $\bar{\gamma}_{\text{VACF}}(t)$. In particular, fitting both functions to the sum of exponentials

$$y(t) = \sum_{j=1}^{3} a_j \exp\left(-t/\tau_j\right) \tag{7.18}$$

revealed that the timescales of both, the fast and the medium decay, τ_1 and τ_2, respectively, agreed remarkably well (cf. Tab. 7.2). Only, for long times $t > 30\,\text{ps}$ both curves deviated significantly. The inset of Figure 7.9 shows that $\overline{\gamma}_{\text{FACF}}(t)$ slowly approached zero, whereas the VACF-derived memory function γ_{I} did not.

Furthermore, the area under both curves, i.e., the effective friction constants $\gamma_{\text{eff}} = \int \gamma(t)\text{d}t$ were with (250 ± 100) ps^{-1} for the FACF-derived similar to $249\,\text{ps}^{-1}$ and $187\,\text{ps}^{-1}$ for the VACF derived kernels γ_{I} and γ_{II}, respectively (cf. Sec. 8.6).

Thus, properties of the smoothed memory functions are comparable between FACF- and VACF-derived memory kernels. Furthermore, a spatial dependence of such properties observed for FACF-derived memory kernels is likely to hold true for memory functions in general.

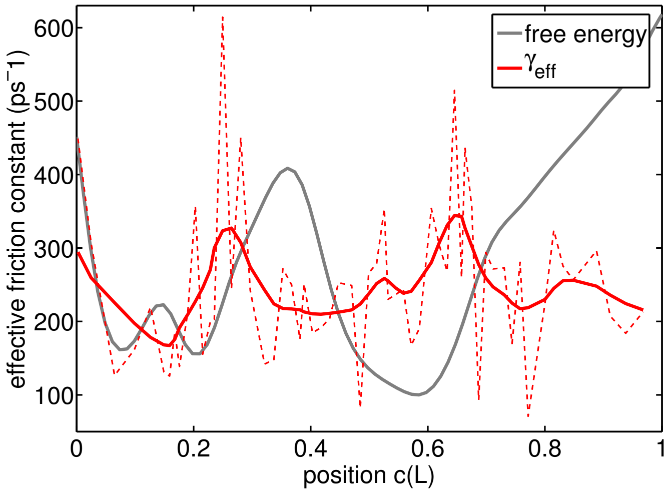


Figure 7.10: Spatial dependence of effective friction constant γ_{eff} (dashed line), together with its running average ($\sigma = 0.05$) (solid line). To facilitate orientation the free energy profile along the conformational coordinate was plotted in gray.

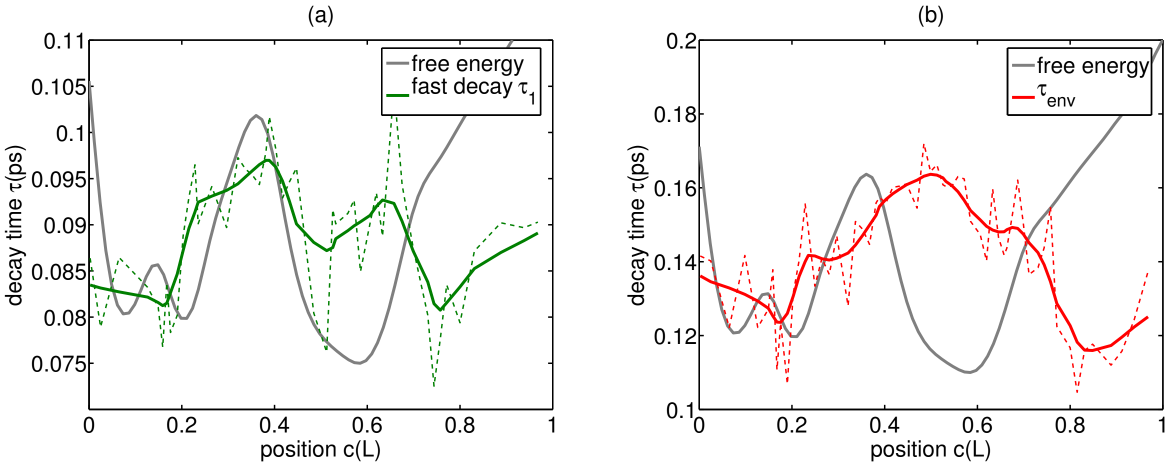


Figure 7.11: Spatial dependence of fast decay time. **(a)** fast decay time $\tau_1(c)$ **(b)** decay of the oscillations τ_{env}. Solid lines see caption Fig. 7.10.

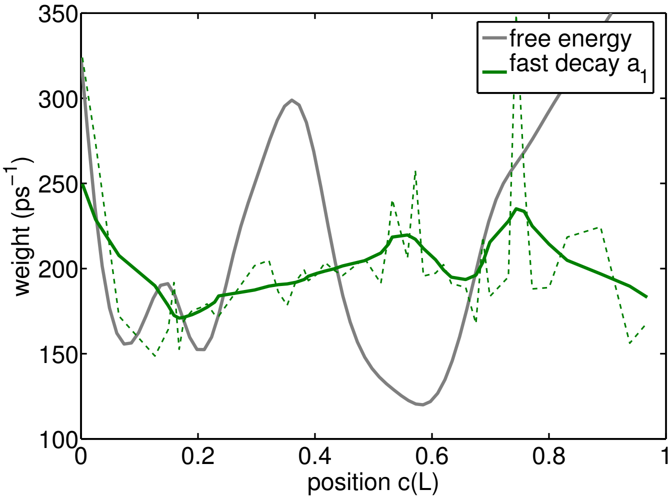


Figure 7.12: Spatial dependence of fast decay contribution a_1 in fit of memory kernel to Eq. (7.18). Solid lines see caption Fig. 7.10.

7.5.3 Spatial dependence of memory functions

To elucidate the dependence of memory functions on the position of the collective coordinate, FACFs were obtained at 60 different positions.

The effective friction constant $\gamma_{\text{eff}}(c)$ fluctuated strongly, as shown in Figure 7.10. To allow a judgment about its spatial dependence, a relative volatility of the effective friction was computed as the fraction of its standard deviation over its mean. The relative volatility of 40% suggests a moderate spatial dependence. However, the large deviations from the running average, seen in the figure, indicate that the effective friction constants are not converged despite a $10\,$ns sampling for each position.

The time constants for the fast decay, obtained from a fit to Eq. (7.18), are shown in Figure 7.11a. They fluctuated much less around their running average, such that its changes can be considered as significant. The low relative volatility of 8% indicates a small spatial dependence of this property. Interestingly, a similar spatial dependence is observed for the decay time $\tau_{\text{env}}(c)$ of the strong oscillations of the non-smoothed γ_{FACF} (cf. Fig. 7.11b). This suggests that the latter probes a similar property of the memory function as τ_1.

The amplitude a_1 of the fast decaying term in the fit function, Eq. (7.18), is shown in Fig. 7.12. The minor deviations from the running average indicate a good convergence and the relative volatility of 15% a small spatial dependence.

The relative volatilities of 22% and 60% for the time constants of medium and slow decay, respectively, suggest an increased spatial dependence for the long time memory effects. However the strong deviations from their running averages (not shown) indicate a bad convergence, and thus an overestimation of the spatial dependence.

7.5.4 Discussion

We obtained memory kernels of a collective coordinate from MD simulations via the FACF of the constrained dynamics. The results were consistent with those of the alternative approach to derive memory kernels from the VACF of the free dynamics. In particular, the smoothed FACF-derived memory kernels gave rise to the same effective friction constant, and agreed in their decay-times.

The FACF-derived memory function was superposed with fast and strong oscillations absent in the VACF-derived memory functions. Whereas the nature of these oscillations was not completely resolved, the results indicated that they were due to the constraint rather than reflecting a true property of the unconstrained dynamics of our interest.

Due to the fixed-particle approximation this method provides a unique possibility to assess the spatial dependence of memory functions. Exploiting this we analyzed how memory kernels vary at different positions of the conformational coordinate. The effective friction constant $\gamma_{\text{eff}} = \int_0^\infty \gamma(t)\mathrm{d}t$ changed its value considerably upon moving along the conformational coordinate. This spatial dependence might strongly affect the dynamics, since the effective friction has a dramatic influence on the observed transition rates, i.e., on long time dynamics of the CLD model, as will be shown in Chapter 8.

A relatively smooth dependence of the decay time on the position along the conformational coordinate indicated that the values were converged, whereas such a conclusion could not be drawn for the erratic changes of the effective friction constant, and the two slower decay times. For these, it needs to be established whether they become smooth at a smaller length scale on the collective coordinate or if more sampling at a single position is needed for a full convergence. Note, however, that we sampled already $10\,$ns at a single position.

7.6 Summary and Conclusions

For both possible strategies to obtain memory kernels from MD simulations, i.e., exploiting the fluctuation-dissipation theorem and solving the Memory equation, we presented solutions. Established techniques, on the contrary, were not applicable due to the collectivity of the dynamics, which gave rise to challenges to the extraction methods not encountered for single particle dynamics.

Two methods (cf. Sec. 7.4) allow to derive the memory kernel from a VACF of a free MD simulation. For this the Memory equation, Eq. (7.1), needs to be solved, which is, however, a challenging inverse problem requiring regularization. The first method imposes a certain functional form on the memory kernel, and thus, regularizing strongly, is not able to accurately solve the Memory equation, but has the advantage of an evident robustness. The second method was devised to solve the Memory equation accurately by regularizing more softly based on sequential Tikhonov regularization. It accurately computed the memory kernel of a realistic example with analytical known solution in a stable way. None of the established methods, was stable enough to accurately determine the memory kernel.

The alternative method, discussed in Sec. 7.5, uses the fixed-particle approximation to obtain the random forces $R(t)$ directly from an MD simulation constrained at a specific position in the conformational subspace. This has the advantage to allow assessment of the spatial dependence of the memory kernel, on the one hand, but implies a huge computational effort, on the other hand. To obtain memory kernels with this approach, separate MD trajectories need to be computed at numerous positions in the conformational subspace. Especially for high dimensional conformational subspaces this becomes intractable.

Our illustrative example demonstrated that the Memory equation does not uniquely define memory kernels. That this problem is also encountered with realistic memory kernels, became obvious when we compared a VACF-derived and a FACF-derived memory kernel. Both correspond to a similar VACF (cf. Fig. 7.8), but are drastically different, i.e., the latter memory function displayed fast oscillations, which were not visible in the former.

On which grounds should one decide which is the ’true’ memory kernel? We argue that this is the wrong question. A characterization of a memory kernel is only physically meaningful with respect to its impact on the dynamics. Therefore, the question is which properties of a memory kernel determine the dynamics and which do not. This suggests to define an equivalence class of memory kernels as the set of all those memory kernels that yield the same dynamics. The challenge to be addressed is then the characterization of these equivalence classes without resorting to an integration of the GLE to yield the dynamics.

It will never be possible to predict conformational transitions in proteins
— Prof. Robert Huber, 4.7.1995

Chapter 8

Collective Langevin Dynamics (CLD) of a Conformational Transition in Neurotensin

Having described all parts of the CLD framework, we now apply it to a specific system and test it by comparing suitable observables between the CLD results and reference MD simulations.

As discussed in the introduction such observables are, e.g., velocity autocorrelation functions (VACFs), positional autocorrelation functions (PACFs) and transition rates. Because VACFs are used to extract the memory kernel, we focus on the latter two observables, which were not used as input and, therefore, pose a hard test for the CLD model.

In Section 8.1 we review several possibilities to compute transition rates from a CLD model. In order to obtain a good estimate for the reference transition rates, however, extensive sampling of the respective conformational transition with standard MD is necessary. This restricts this type of test to much smaller systems, e.g., peptides, than CLD aims for. For instance, repeated conformational transitions were not observed in a 450 ns MD simulation of crambin (cf. Chapter 2). Instead, we chose the hexapeptide neurotensin, because we expected it to undergo sufficiently many conformational transitions at the MD timescale to allow comparisons of transition rates.

As a first step we modeled the CLD of neurotensin by means of a one-dimensional coordinate. Whereas from the methodological point of view this appears to be the simplest case, reduction from $3N$ coordinates to a single one is of course the most drastical case possible and, hence, represents a hard test. To this end we had to use a curved coordinate, as will be described in subsection 8.4. Subsequently, free energy and a memory function will be extracted for the chosen coordinate from explicit MD simulation.

As discussed in Chapter 7, extraction of memory kernels is not straightforward. We proposed two methods to derive memory kernels from a VACF of short MD simulations by solving the Memory equation. Since this necessitates solution of an inverse problem, both methods are based on regularization techniques. They differ, however, in the strength of the applied regularization. The first method, FIT, applies strong regularization by imposing a model function with only 3 free parameters. The second method, DIR, allows to tune the level of regularization by a parameter that controls the smoothness of the resulting memory kernels. In this chapter, we test and compare the performance of the two proposed methods in the context of a real application. To this end, transition rates and positional autocorrelation functions will be compared between the obtained CLD models and a reference MD simulation of neurotensin in Sections 8.8 and 8.9.

8.1 Transition rates

To check how accurately the dimension reduced GLE approximates the fully atomistic dynamics, we will compare conformational transition rates of the CLD model with rates obtained from explicit MD simulations.

The Arrhenius equation[240]

$$k = \eta e^{-\beta \Delta G^{\dagger}},$$

where β denotes the inverse temperature, clarifies in which way transition rates are influenced by the CLD parameters. The importance of the height of the free energy barrier $\Delta G^{\dagger}$ is evident immediately. Important too, however, is the pre-factor η, which accounts for attempt frequency, recrossing events and non-equilibrium effects. The height of the free energy depends on the choice of the conformational coordinate only, whereas the pre-factor depends on the correct description of memory effects by the CLD model. Therefore, the check of the transition rates was also used to evaluate the relative performance of the different approaches to extract memory kernels.

The transition rates for the conformational dynamics governed by the GLE of the CLD model can be obtained in two ways. Either, the GLE is integrated numerically, which yields a trajectory whose transitions can be counted (cf. Sec. 8.2.5), or transition rates are estimated directly from the GLE using Kramers' Theory[240]. For the latter we follow Kramers' approach and approximate the potential of mean force with parabolas at the minima $W_{\alpha}(x) \approx \mu \omega_{\alpha}^{2} x^{2}$ and at the barrier top $W_{\dagger}(x) \approx -\mu \omega_{\dagger}^{2} x^{2}$[240]. Then the escape rate is

$$k_{\alpha} = \left(\sqrt{\frac{\hat{\gamma}^{2}(\xi)}{4} + \omega_{\dagger}^{2}} - \frac{\hat{\gamma}(\xi)}{2} \right) \frac{\omega_{\alpha}}{2\pi \omega_{\dagger}} \exp\left(-\beta \Delta W_{\alpha}^{\dagger} \right) \tag{8.1}$$

with index $i = A, B$ for state A and B, respectively and $\Delta W_{\alpha}^{\dagger} = W^{\dagger} - W_{\alpha}$ the height of the barrier. Here $\hat{\gamma}(z)$ denotes the Laplace-transform of the memory kernel $\gamma(t)$, and ξ is subject to the condition

$$\xi = -\frac{\widehat{\gamma(\xi)}}{2} + \sqrt{\frac{\widehat{\gamma(\xi)}}{4} + \omega_{\dagger}}. \tag{8.2}$$

In the case of memory free friction, $\gamma(t) = 2\gamma_{\text{eff}}\delta(t)$, Eq. (8.1) simplifies due to $\hat{\gamma}(\xi) \equiv \gamma_{\text{eff}}$, and adopts the widely known form[240]. For a comprehensive Review, we refer to Ref. [240].

8.2 Methods

8.2.1 Definition of a one-dimensional curved conformational coordinate

By visual inspection of the projection of trajectory NT1 onto the first three principal components, 8 snapshots $\{\mathbf{x}_{sel,i}\}_{i=1...8}$ were selected evenly spaced along the observed trace of high conformational density (Fig. 8.2). To remove any bias introduced by choosing single snapshots out of a large number of equally reasonable alternatives, averages over all n_i snapshots $\left\{\mathbf{x}^j_{sel,i}\right\}_{j=1...n_i}$ within a sphere of radius 0.1 nm (in the 3D projection) around $\mathbf{x}_{\text{sel,i}}$ were used. The conformational coordinate was then constructed by cubic spline interpolation between these averages in the full $3N$ dimensional space. Subsequent discretization yielded $N = 1200$ points $\{\mathbf{z}_i\}_{i=1...N}$.

8.2.2 Projection onto the conformational coordinate

The conformational coordinate defined by the discretized submanifold $\mathcal{M} = \{\mathbf{z}_i\}_{i=1...N}$ was parameterized by a mapping function f (cf. Eq. (6.8)), such that $c = f(\mathbf{z}_1) = 0$ and $c = f(\mathbf{z}_{1200}) = 1$, respectively. All intermediate values were defined via the contour length $s_j = \sum_{k=2}^{j} \|\mathbf{z}_k - \mathbf{z}_{k-1}\|$ as $c = f(\mathbf{z}_j) := s_j/s_N$. Thus, the length unit, L, of the projected coordinate is $L = s_N$, and the metric of the configurational space is preserved upon projection.

Unfortunately, the straightforward approach to project a point $\mathbf{x}$ onto the point of $\mathcal{M}$ which is closest in space

$$P(\mathbf{x}) = \underset{z\in\mathcal{M}}{\arg\min} \|x - z\| \tag{8.3}$$

led to several “wrong” projections due to the U-shape of the coordinate. In particular, and as will be discussed in the Results Section, this simple projection scheme, therefore, resulted in unphysical discontinuities.

This problem was resolved by additionally considering the time information of the trajectory. Specifically, snapshots close in time were enforced to yield projections close to each other. To determine the projection $P\left(\varphi\left(\mathbf{x}, \mathbf{p}, t_i\right)\right)$, we proceeded as follows. First, both, the discretized conformational coordinate $\mathbf{z}_i$ and the trajectory $\varphi\left(\mathbf{x}, \mathbf{p}, t_i\right)$, were projected preliminary onto the first 100 principal modes (obtained as above) yielding $\overline{\mathbf{z}_i}$ and $\overline{\varphi}\left(\mathbf{x}, \mathbf{p}, t_i\right)$, respectively. Second, the final projection of the trajectory to the curved coordinate was determined via

$$P\left(\varphi\left(\mathbf{x}, \mathbf{p}, t_{i+1}\right)\right) = \underset{z\in I\{c(t_i),r\}}{\arg\min} \left\|\overline{\varphi}\left(\mathbf{x}, \mathbf{p}, t_{i+1}\right) - \bar{z}\right\|, \tag{8.4}$$

where the interval of the conformational coordinate

$$I\left\{c(t_i), r\right\} = \left\{z \in \mathcal{M} | c(t_i) - r \leq f(z) \geq c(t_i) + r\right\}$$

defines a window of width $2r$ centered around the previous result of the projection $c(t_i) = f\left[P\left(\varphi\left(\mathbf{x}, \mathbf{p}, t_i\right)\right)\right]$. Parameter values below 0 or greater than 1 were allowed by extending the conformational coordinate linearly at both ends. The window size r was chosen to trade off sufficient fast response of the projection with robustness against unphysical jumps; for the 10fs sampling, $r = 1/1200$ and for the 1ps sampling $r = 1/12$. Velocities were projected onto the first 100 principal modes as described above, and subsequently onto the tangent to the conformational coordinate at the point $P\left(\mathbf{x}(t_i)\right)$.

8.2.3 Solution of the memory equation

Memory kernels $\gamma(t)$ were obtained by solving the Volterra equation of the first kind, Eq. (7.1),

$$\frac{d}{dt}\Psi(t) = -\int_0^t d\tau \gamma(t-\tau)\Psi(\tau). \tag{8.5}$$

with the velocity autocorrelation function $\Psi(t) = \langle \dot{c}(0) |\, \dot{c}(t)\rangle / \langle \dot{c}^2 \rangle$ computed from the MD simulation.

Integration of this equation is notoriously unstable, as discussed extensively in Chapter 7. To find physically meaningful solutions we therefore resorted to regularizations.

As described in Section 7.3.3, the first method, FIT, achieves strong regularization by imposing a model function, Eq. (7.12), with only 3 free parameters, γ^c, a and A. To obtain these parameters, the corresponding VACF, Eq. (7.13), was least-squares fitted to the numerically obtained VACF, $\Psi(t)$, with the curve fitting tool of MATLAB(tm).

The second method, DIR, regularizes weakly by applying sequential Tikhonov regularization to favor smooth solutions (cf. Sec. 7.3.7). The regularized solution is the minimum of the Tikhonov criterion, Eq. 7.15, whose regularization parameter α can be chosen from an analysis of the *L-curve* (cf. Sec. 7.3.8). However, here the *L-curve* optimum of roughly $\alpha = 20$ was not very pronounced. In order to contrast the strong regularization method above with a method, which does not bias the result too much, we chose with the help of the L-curve the relatively low regularization parameter $\alpha = 0.14$. For illustration purposes we also obtained memory kernels $\gamma_{\text{I-reg}}$ and $\gamma_{\text{II-reg}}$ with $\alpha = 20$, but these memory kernels were not used for the CLD model.

8.2.4 Potential of mean force

To compute the potential of mean force W along the conformational coordinate, $W(c) = -kT \log \rho(c)$, the density $\rho(c)$ was obtained from the MD ensemble projected to the conformational coordinate. For this purpose, histograms with 100 bins were determined and smoothed by convolution with a Gaussian function of width $\sigma = 0.025\,\mathrm{L}$, where L denotes the length scale of the conformation coordinate. Outside the sampled range of c the potential was continued by a harmonic potential $W_{\text{harm}}(c) = 11.5(c - 0.5\,\mathrm{L})^2$ as

$$W_{\text{extend}}(c) = \left[1 - S(c)\right] W(c) + S(c) W_{\text{harm}}(c),$$

where the switching function is defined by the sigmoidal function $S(c) = \{1 + \exp[-50(c-1)]\}^{-1} + 1 - \{1 + \exp[-50c]\}^{-1}$. Forces were computed by linear interpolation between the numerically obtained derivatives at neighboring discretization points.

8.2.5 Statistics of conformational transitions

Transition rates were determined from the one-dimensional projection of molecular dynamics trajectories to the conformational coordinate or from the one dimensional CLD trajectories by counting. First, every snapshot $c(t_i)$ was assigned to one of the two conformational states $s(t_i) = A, B$ and then the number of changes of s was evaluated. To account for non-thermalized re-crossings[240] a variable threshold was applied to the low-pass filtered projection $\tilde{c}(t_i)$, which depended on the previous conformation $s(t_{i-1})$

$$s(t_i) = \begin{cases} A & s(t_{i-1}) = A \ \wedge \ \tilde{c}(t_i) < 0.66 \\ B & s(t_{i-1}) = B \ \wedge \ \tilde{c}(t_i) > 0.36 \end{cases}$$

As low-pass filter we used smoothing with a Gaussian function of width $\sigma = 40\,\text{ps}$. The transition rate $k_{\alpha\beta}$ for the transition $\alpha \longrightarrow \beta$ was given by $k_{\alpha\beta} = n_{\alpha\beta} / \left(N_\alpha \Delta t\right)$, where $n_{\alpha\beta}$ denotes the changes of $s(t_i)$ from state α to state β and N_α is the number of snapshots for which $s(t_i) = \alpha$.

The threshold value and the bandwidth of the low-pass filter were chosen manually and introduces clearly a bias into the obtained rates. However, here we only need to compare rates obtained with the same method, such that this bias was canceled out. Moreover, other sets of parameters tested did not change the relative differences between CLD and reference transition rates.

Confidence intervals were determined via the Poisson-statistic $P_\lambda(n) = e^{-\lambda}\lambda^n/n!$, since transitions were rare events. Via $\left\langle n^2 \right\rangle = \lambda = \langle n \rangle$ the number of observed transitions n determined the Poisson-parameter λ and with that an estimate of the error of the transition rate. A 95% confidence interval in the logarithmic representation was computed by choosing its width d, such that $P_\lambda(k \in [n\exp(-d), n\exp(d)]) = 0.95$. In the case of a large number of observed transitions $n > 60$ the Poisson statistics was approximated by the error function via $P_\lambda(a \leq k \leq b) = \Phi(b/\gamma - 1) - \Phi(a/\lambda - 1)$, with $\Phi(x) = \int_{-\infty}^{x} (2\pi)^{-1/2} \exp(-x^2/2) = \frac{1}{2}\left[\text{erf}\left(x/\sqrt{2}\right) + 1\right]$.

8.3 Conformational dynamics of reference MD simulation

Before we start, we have to check whether the conformational dynamics of neurotensin pose a suitable test case. This system has been chosen because, in contrast to larger proteins, e.g. crambin or lysozyme, we expected it to undergo sufficiently many conformational transitions at the MD timescale to allow comparisons of transition rates.

Indeed, as can be seen in Fig. 8.1, neurotensin underwent several main conformational transitions $A \longrightarrow B$ during the 90ns MD simulation, NT1. The Figure shows the matrix of the root mean square deviation (RMSD) of the C_α-atoms for each pair of snapshots of the trajectory NT1. Conformational states were defined as almost invariant subsets of the configurational space[241]. They are visible in

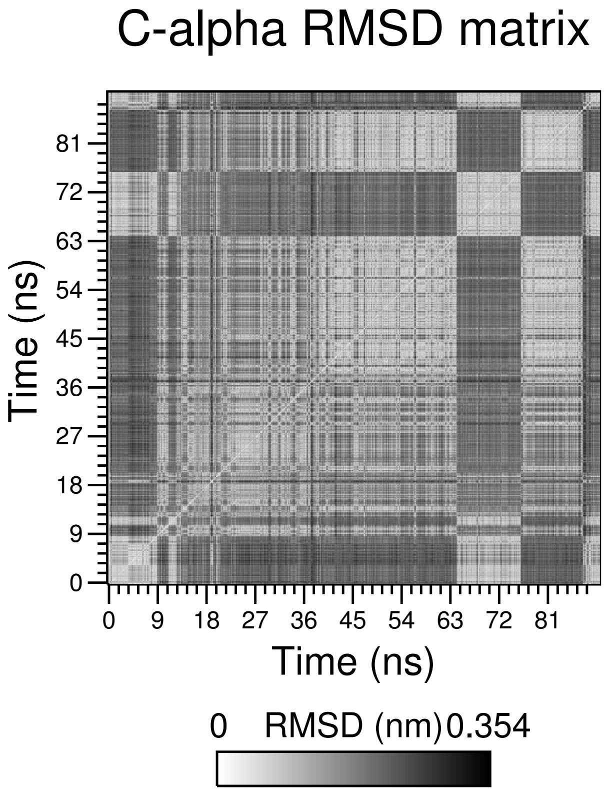
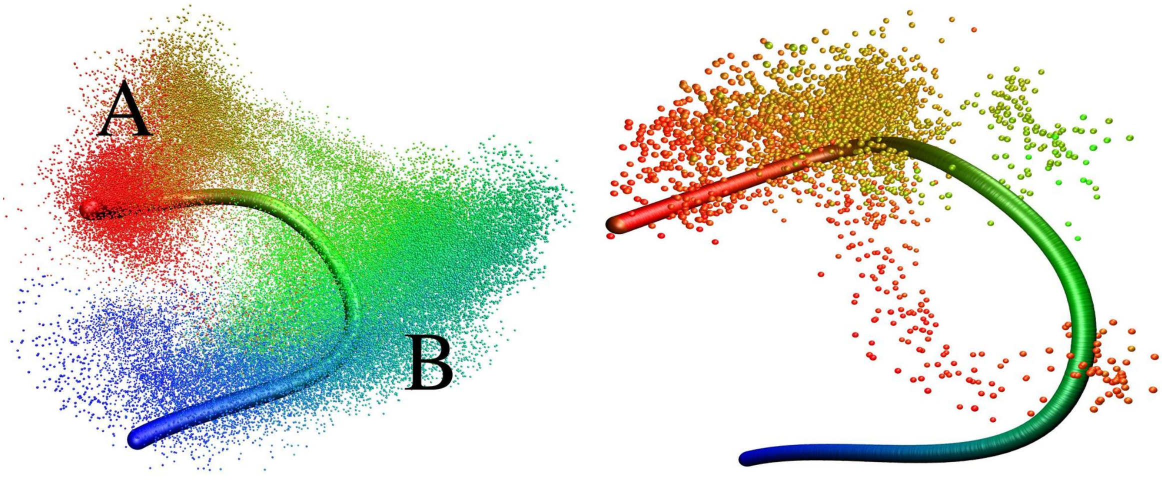


Figure 8.1: Root mean square deviation (RMSD) of the C_α atoms for each pair of snapshots of trajectory NT1. The RMSD ranges from zero (white) to 0.382nm(dark). The labels indicate conformational substates, see text.

the RMSD matrix as distinct bright blocks on the diagonal. Bright off-diagonal blocks indicate that a certain conformational substate was revisited. Interestingly, as can also be seen in the Figure, the two main conformational states subdivide further into substates (denoted by primes) as is typical for proteins[160], thus giving rise to a complex conformational dynamics also within the main states. In this sense, the system represents a particularly harsh test system for CLD: The CLD model has to predict correct first passage times without knowledge of the substate dynamics. This lack of knowledge is of course intrinsic to a reduced dimension approach and it is important to find out how well CLD can cope with it.

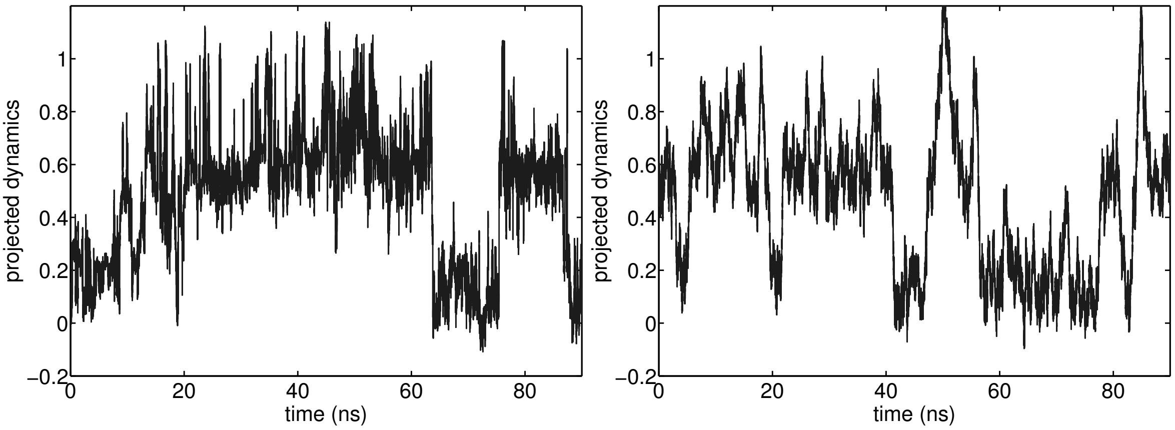
8.4 Construction of a curved conformational coordinate

As a first task, we need to construct a collective coordinate, which resolves both states A and B. We start by analyzing the MD ensemble, as projected onto the first three principal components, shown in Figure 8.2a. Red points represent structures belonging to conformation A, blue points belong to conformation B, and green points belong to transitions between both states. The shape of this ensemble was such that no conceivable *linear* coordinate would resolve the two conformational states. In particular, the close blue and red points are separated by a free energy barrier, which cannot be resolved by a linear coordinate. We therefore constructed the curved coordinate shown in Figure 8.2a (see Methods), which clearly resolves states A and B. The projection of trajectory NT1 onto



(a) (b)

Figure 8.2: Projection of the conformational coordinate (thick line) and the configurational ensemble (dots) onto the first three PCA modes. The colors denote the resulting mapping of snapshots $\mathbf{x}_i$ to position on the coordinate $c = f(\mathbf{x}_i)$, from red, $c \approx 0$, to blue, $c \approx 1$. **a)** whole configurational ensemble NT1 **b)** interval (70ns-75ns) of NT1, where a substate of A is visited.



(a) (b)

Figure 8.3: **(a)** Projection of neurotensin internal motion onto conformational coordinate. This shows the projection of NT1 onto the conformational coordinate. A number of transitions occur between the clearly distinguishable two states centered at 0.2 and 0.6, respectively. **(b)** an example of a CLD trajectory. The plot shows a 90 ns of a trajectory generated by the model $\mathrm{CLD}_{\mathrm{I-fit}}$ (cf. Sec. 8.7).

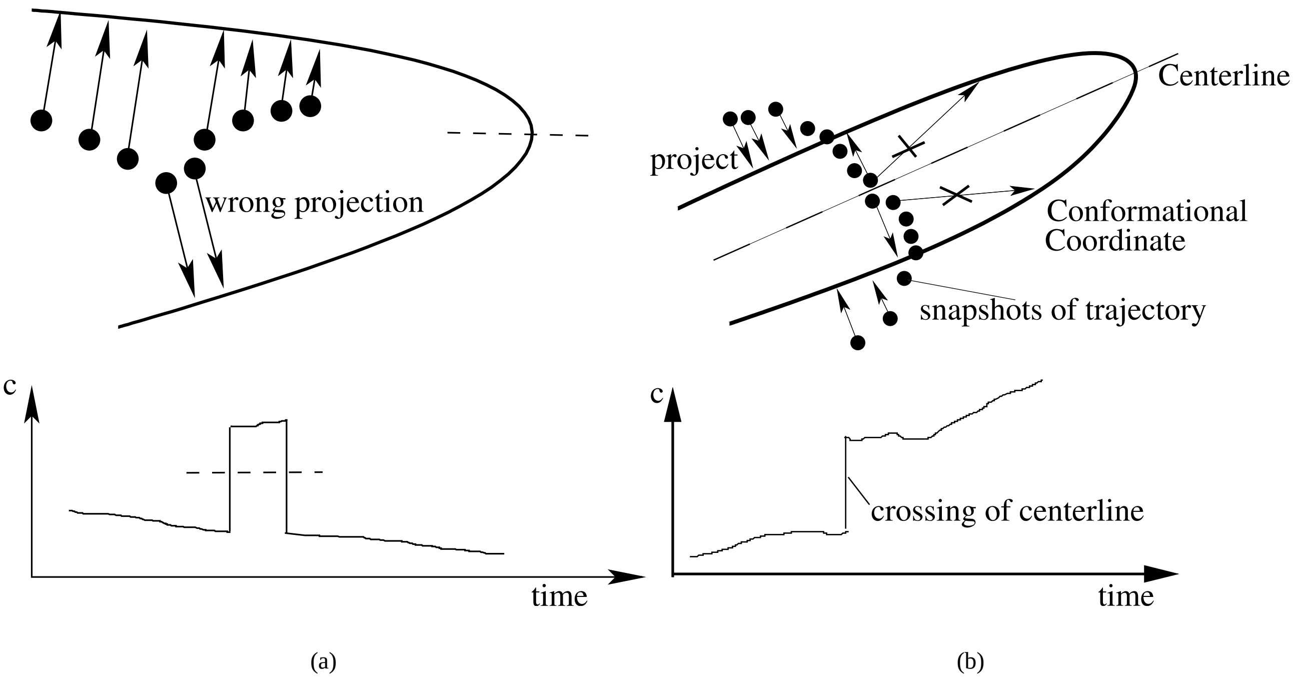


Figure 8.4: Illustration of main sources for artifacts in the projection to a curved coordinate. Snapshots in the configurational space (circles) are projected (arrows) to a curved coordinate by a pure distance criterion. The resulting projection is plotted against time under the pictures. **(a)** A trajectory moves from right to left along one arm of the coordinate, i.e., the projection is decreasing (cf. plot). However, two snapshots are slightly closer to the other side of the curved coordinate and, hence, the projection erroneously jumps to large values and back, although no real conformational transition occurred. **(b)** In contrast to (a), here a real transition from the low projection part to the high projection part of the coordinate occurs. However, the trajectory crosses to far away from the curved coordinate and, therefore, shortcuts the bulge drastically, which results in an artifactual large jump in the projection, in the moment of crossing of the centerline.

the coordinate c (cf. Fig. 8.3) revealed several well resolved transitions between the conformational substates A and B, centered around $c \approx 0.2$ and $c \approx 0.6$, respectively. This projection turned out not to be straightforward, and care had to be taken to avoid possible artifacts.

The more technical aspects of this projection described below are not of direct relevance for the CLD model; we have included a brief description, nonetheless, to illustrate problems , which typically arise from the use of curved coordinates as well as their solutions.

The main problem arose from the fact that no *direct* transitions between the two main states, A and B, were seen in the vicinity of the red and blue points (cf. Fig. 8.2a), but only *indirect* ones in the region of the green points. Therefore, straightforward assignment of each MD structure to the nearest point of the conformational coordinate would fabricate transitions, as illustrated in Fig. 8.4a. These spurious transitions would adulterate the reference transition rates used for confirmation of the CLD model. This problem was solved by taking time-information into account (cf. Methods). Careful inspection showed that the spurious transitions were indeed eliminated.

Figure 8.2b shows an extreme example. Here, several structures seem to approach state B in the

	a (ps^{-1})	γ^c (ps^{-2})	A (ps^{-2})	mass μ (u)
NT2	0.78	14.8	49.2	7.3
NT2$_{\mathrm{II}}$	1.1	11.5	27.0	8.9
NTS$_4$	1.06	11.7	28.1	13.33
NTS$_3$	1.45	12.5	43.9	12.94
NTS$_1$ + NTS$_5$	1.04	12.6	33.6	11.1
NTS$_2$ + NTS$_6$	1.32	13.2	35.5	9.84
NTS$_3$ + NTS$_7$	1.24	13.1	35.0	12.52
NTS$_4$ + NTS$_8$	1.07	12.2	30.7	8.78
NTS$_{\mathrm{uneven}}$	1.13	12.9	34.3	11.82
NTS$_{\mathrm{even}}$	1.18	12.7	32.9	9.31
NTS$_{\mathrm{all}}$	1.16	12.8	33.6	10.56

Table 8.1: Fitting parameters of velocity autocorrelation function, Eq. (7.13), obtained for different trajectories. The presented parameters for NTS$_3$ and NTS$_4$ were the most extreme of all 500ps trajectories. The parameters for unions, e.g., NTS$_1$ + NTS$_5$, were obtained by fitting to an overlay of the VACFs of the respective trajectories. The labels NTS$_{\mathrm{uneven}}$, NTS$_{\mathrm{even}}$ and NTS$_{\mathrm{all}}$ denote NTS$_1$ + NTS$_3$ + NTS$_5$ + NTS$_7$, NTS$_2$ + NTS$_4$ + NTS$_6$ + NTS$_8$, and NTS$_1 + \cdots +$ NTS$_8$, respectively.

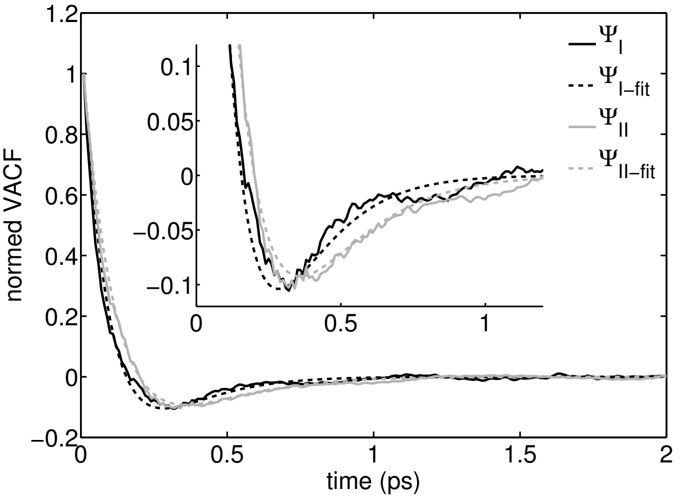


Figure 8.5: Velocity autocorrelations (VACF) of MD trajectories (solid) and their respective fits to Eq. (7.13) (dashed). The inset shows the same data enlarged.

projection onto the first three principal components. Accordingly, these structures would be assigned to state B in any purely distance based projection onto the shown curved coordinate. However, as can also be seen in Fig. 8.1, the RMSD to state A remains small for all shown structures and large to state B, such that assignment to B would be wrong. Indeed, as seen from the coloring in Fig. 8.2b, all snapshots of the substate of A were correctly allocated to conformation A.

Fig. 8.4b illustrates and explains a second problem (see caption), resulting in discontinuities in the projected motion. This problem was solved by careful placement of the curved coordinate.

8.5 Velocity autocorrelation function of collective motion

The velocity autocorrelation function (VACF) is required to derive the memory kernels for the CLD model and, therefore, needs to be extracted from the MD trajectories. Further below we will analyze how well this observable is reproduced by the CLD model.

Two VACFs, Ψ_{I} and Ψ_{II}, were obtained from trajectories NT2 and NT2$_{\mathrm{II}}$, respectively (see Methods). NT2$_{\mathrm{II}}$ refers to the interval interval $10\,\mathrm{ns} - 19\,\mathrm{ns}$ of the $63\,\mathrm{ns}$ trajectory NT2. Both VACFs are shown in Fig. 8.5, together with their respective fits to Eq. (7.13), $\Psi_{\mathrm{I\text{-}fit}}$ and $\Psi_{\mathrm{II\text{-}fit}}$. Both, Ψ_{I} and Ψ_{II}, are very similar, which indicates good convergence, and are well approximated by the fits.

Their rapid decay shows that most correlations occur at a picosecond timescale. Furthermore, the pronounced negative dip in intermediate timescales ($0.3\,\mathrm{ps} < \tau < 0.5\,\mathrm{ps}$) indicates resonant behavior or memory effects in the system. The similarity to the dip in VACFs of simple liquids caused by caging of the tagged molecule by its immediate neighbors[223] is suggestive.

The medium scale oscillations of the VACFs seen in the inset of Fig. 8.5, however, indicate more complex dynamics than typically observed for simple liquids. For example, the slowly decaying oscillatory contributions to the VACF are clearly above the noise threshold seen for larger times $\tau > 5\,\mathrm{ps}$, although the difference between Ψ_{I} and Ψ_{II} indicates that this feature may not be fully converged. These longer correlations are not captured by our simple fit.

8.6 Extraction of memory kernels

From the VACF, we can now proceed and compute the memory kernel as the essential quantity that captures the influence of the many degrees of freedom excluded from explicit treatment in the CLD model. To this aim the Memory equation was here solved using two different methods, FIT and DIR (cf. Methods 8.2.3).

Figure 8.6a compares the memory kernels γ_{I} and γ_{II} computed with DIR with the respective memory kernels computed with FIT, $\gamma_{\mathrm{I\text{-}fit}}$ and $\gamma_{\mathrm{II\text{-}fit}}$. As described in Methods, method FIT admits only a certain type of functions for the memory kernel, and hence involves stronger regularization constraints than DIR, which allows any sufficiently smooth function.

All memory functions drop rapidly to approx. 5% of their initial values at $\tau \approx 0$, followed by a decay with a $1\,\mathrm{ps}$ time constant (a in Table 8.1). Significant differences are also seen. In particular, the memory kernels γ_{I} and γ_{II} — obtained with the less regularizing method DIR — show strong oscillations, a second slower decay component, and do not approach zero. None of these features is seen in the memory kernels $\gamma_{\mathrm{I\text{-}fit}}$ and $\gamma_{\mathrm{II\text{-}fit}}$. These features, therefore, deserve closer analysis.

As can be seen from the left inset in Fig. 8.6a, most details of the fast oscillations differ for the different trajectories. Rather, they are due to the unconverged medium scale oscillations and the small scale statistical noise of the VACF, both strongly amplified by the inherent instability of the Memory equation. Accordingly, they should not be attributed to a physical basis. To aid the remaining discussion, Fig 8.6b also shows memory kernels, whose oscillations were removed by increasing the regularization parameter α as defined in Eq. (7.15).

In contrast to the oscillations, both remaining features not seen in $\gamma_{\mathrm{I\text{-}fit}}$ and $\gamma_{\mathrm{II\text{-}fit}}$, the slower decay component and the lack of complete decay to zero for very long times, are shown in Fig. 8.6b to be

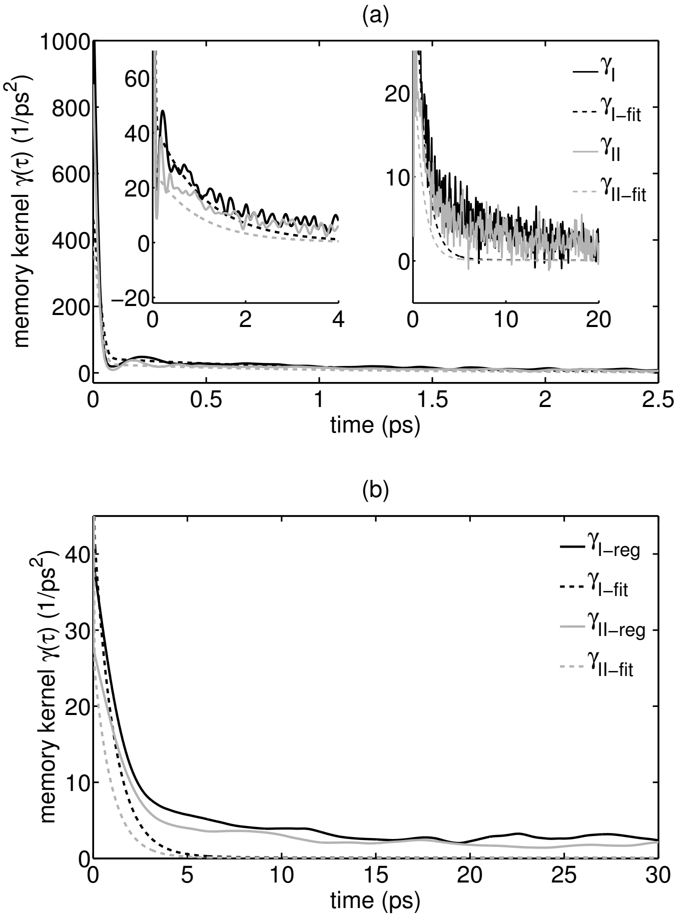


Figure 8.6: Memory Kernel functions computed from the VACFs shown in Fig. 8.5. **(a)** Memory computed with DIR (solid) and with FIT (dashed). The insets show the same data in different zooms. **(b)** Memory kernels from the same VACFs, but method DIR was used with a higher regularization parameter to eradicate oscillations.

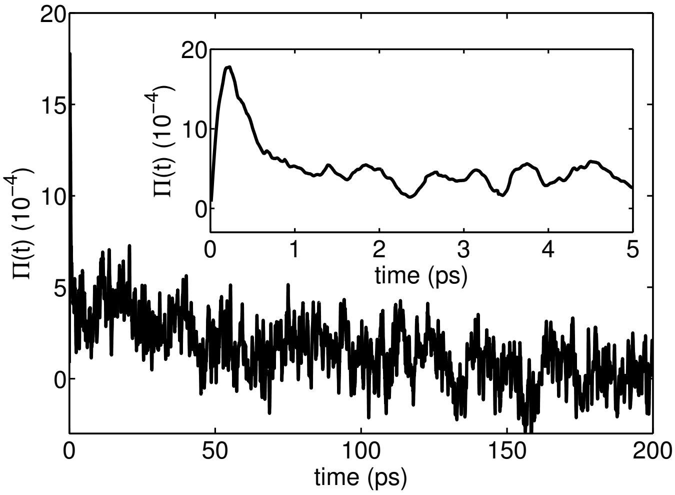


Figure 8.7: Potential term $\Pi(t)$ of Memory equation. The inset shows the same data enlarged.

comparable for both memory functions $\gamma_{\text{I-reg}}$ and $\gamma_{\text{II-reg}}$. Does this mean that the slow and incomplete decay, rather than being due to the amplified noise, actually reflects genuine long time memory effects? Some insight can be obtained by testing the relation for the effective friction constants $\left(\int_0^\infty \Psi(\tau)d\tau\right)^{-1} = \int_0^\infty \gamma(\tau)d\tau$ [223]. First of all, this shows that γ has to approach zero sufficiently fast, such that the effective friction constant is finite. Furthermore, we note that for $\tau \gg 5\,\text{ps}$ $\gamma(\tau)$ is not well-defined by the Memory equation, because $\Psi(\tau)$ is dominated by noise for these longer times. We here assume that with optimal statistics $\Psi(\tau)$ will vanish, thus neglecting possible long-time correlations. One consequence is that, setting $\gamma(\tau) \equiv 0$ for these long times satisfies the Memory equation equally well. Indeed, the effective friction constants estimated from the shown interval of γ_{I} and γ_{II} , respectively, were significantly higher than those derived from the corresponding VACFs (cf. Tab. 8.2), suggesting a spuriously slow decay of the memory kernels. In particular, utilizing the relation of the effective friction constant in addition to the Memory equation shows that it is more reasonable to assume that $\gamma(\tau) \approx 0$ beyond 35ps and 25ps for γ_{I} and γ_{II}, respectively, such that the friction constants correspond to those of the VACFs in Table 8.2. This assumption is supported by the alternative method to extract memory kernels via force autocorrelation functions, which yielded memory functions decaying significantly faster than γ_{I} and γ_{II} (cf. Sec. 7.5).

These considerations justify the following manipulation. We obtained a new set of memory kernels $\gamma_{\text{I-tail}}$ and $\gamma_{\text{II-tail}}$ by manually damping the tail to zero beyond 35ps and 25ps, respectively.

We finally note that in the present context the term $\Pi(\tau)$, as defined by Eq. (7.2), of the Memory equation was neglected. This term, derived from a correlation function between mean force and velocity, corrects for those velocity correlations, which are caused by the inertial motion of the system within a non-zero free energy surface rather than by memory effects due to the eliminated degrees of freedom. Due to the highly diffusive nature of the conformational dynamics of the system at hand, here the influence of the free energy on the velocities is small and, therefore, also the term $\Pi(\tau)$ is expected to be small, implying that it can be neglected to good approximation. Indeed, as shown in Fig. 8.7, $\Pi(\tau)$ is three orders of magnitude smaller than the VACF-term for small τ, and for larger times $\tau > 5\,\text{ps}$ it is one magnitude smaller than the noise in the VACF, which justifies our approximation.

8.7 Conformational dynamics by CLD

In the following three sections we test how well the dynamics along the conformational coordinate is actually described by the CLD model. Additionally to the memory kernels obtained above, a reduced mass, and a free energy is required.

The free energy (Figure 8.8a) along the conformational coordinate c was obtained from the conformational density (Fig. 8.8b) as potential of mean force, averaged over both available MD ensembles, NT1 and NT2. The reduced mass μ was obtained via the equipartition theorem $< \dot{c}^2 >= (\beta\mu)^{-1}$ from the amplitude of the velocity fluctuations (cf. Table 8.1).

Having thus obtained all parameters directly from MD simulations, Collective Langevin Dy-

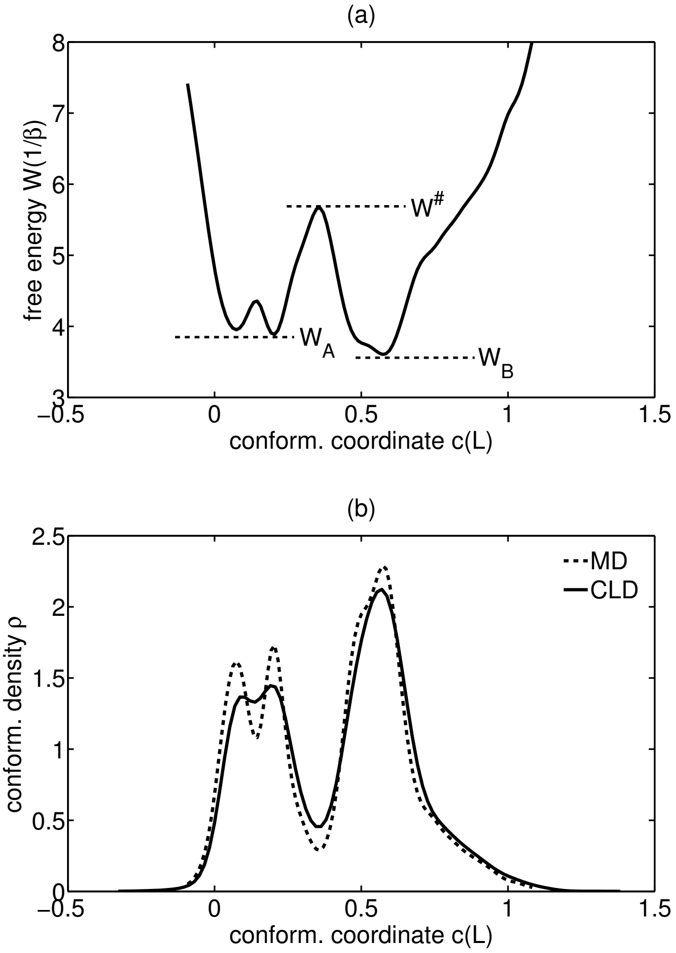


Figure 8.8: **(a)** Potential of mean force along the conformational coordinate. The energy levels depicted as W_A, W_B and $W^{\#}$ were used for calculation of barrier heights in Kramers' theory **(b)** Comparison of conformational density off a CLD ensemble with that of the reference MD ensemble.

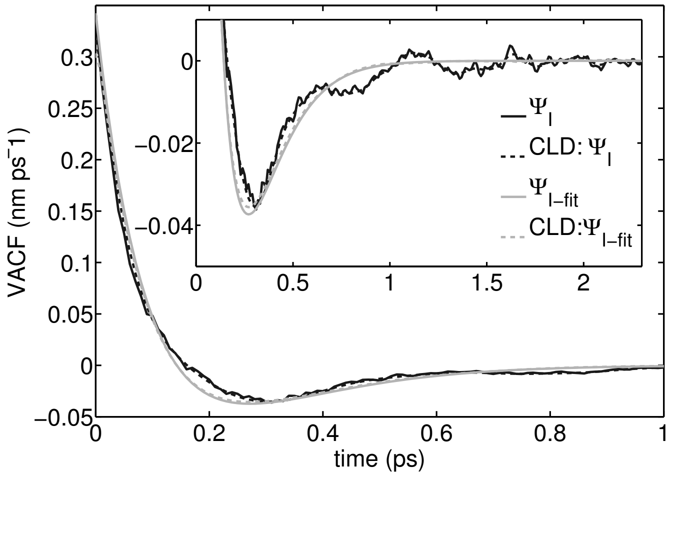


Figure 8.9: Comparison of CLD generated (dashed lines) with reference MD (solid lines) velocity autocorrelation functions. Note that the dashed lines are hardly seen, since the respective curves match very good.

namics trajectories were obtained by numerical integration of the generalized Langevin equation, Eq. (6.16). A single trajectory of 300ns took about 6 minutes on a desktop computer (AMD 1.8GHZ Opteron), as compared to nearly 5 months for the atomistic MD trajectory of the same length computed on the same hardware.

In the following we analyze the accuracy of the CLD model in terms of suitable dynamical and thermodynamical observables of the CLD model.

Firstly, we compare the thermodynamic properties to those obtained from the reference MD simulation. Since all thermodynamic observables of this CLD model can be obtained from its one-dimensional partition function, it suffices to compare the conformational density ρ with that of the MD ensembles, projected to the conformational coordinate (cf. Fig. 8.8b). As can be seen, the densities agreed well with each other, although that of the CLD model was slightly smoother. This result confirms that the used friction kernel and random forces generated from it satisfy the fluctuation-dissipation theorem.

Secondly, the dynamics was checked by comparison of the VACF with references from the MD simulations.

We focus on the evaluation of the CLD models based on Ψ_{I}, because the results for the CLD models obtained from Ψ_{II} were similar. Figure 8.9 shows the VACFs of the two CLD models using γ_{I} and $\gamma_{\text{I-fit}}$ together with the reference VACF of the MD. All VACFs agree well. In particular, the initial decay and the position of the dip were well reproduced.

As was expected, the VACF obtained with $\gamma_{\text{I-fit}}$ is nearly identical with the fit to the MD VACF $\Psi_{\text{I-fit}}$. Therefore, for this model the quality of the fit determines the accuracy. This restriction is gone if the method DIR is used to obtain the memory kernels. As shown in the inset of Figure 8.9 the resulting VACF reproduces the reference very closely. This was quantified by the deviation $\left(\int (\Psi - \Psi_{\text{CLD}})^2 \, dt\right)^{1/2}$ between CLD-VACF and reference, which was smaller for DIR$\left(4.8 \cdot 10^{-3}\text{nm/ps}\right)$ than for FIT $\left(8.2 \cdot 10^{-3}\text{nm/ps}\right)$.

8.8 Prediction of Transition Rates by CLD

It was shown that CLD yields trajectories with accurate conformational densities and VACFs. Although these properties provide a useful consistency check, we do not consider them as a a rigorous test of CLD, because they were also used as input for the CLD model. In Contrast, transition rates were not used for the parameterization. As the most rigorous test, we therefore finally check forward and backward transition rates against references obtained from a long MD simulation.

In the following, we label the results from the different approaches as CLD_{I}, CLD_{II}, $\text{CLD}_{\text{I-fit}}$ and $\text{CLD}_{\text{II-fit}}$. The first two denote the CLD model whose memory was obtained with DIR from Ψ_{I} and Ψ_{II}, respectively, and the latter two denote the corresponding CLD models whose memory was obtained with FIT. In Figure 8.10 transition rates observed from 300ns CLD trajectories are shown (squares) with errorbars indicating their 95% confidence interval (cf. Methods). For comparison, the

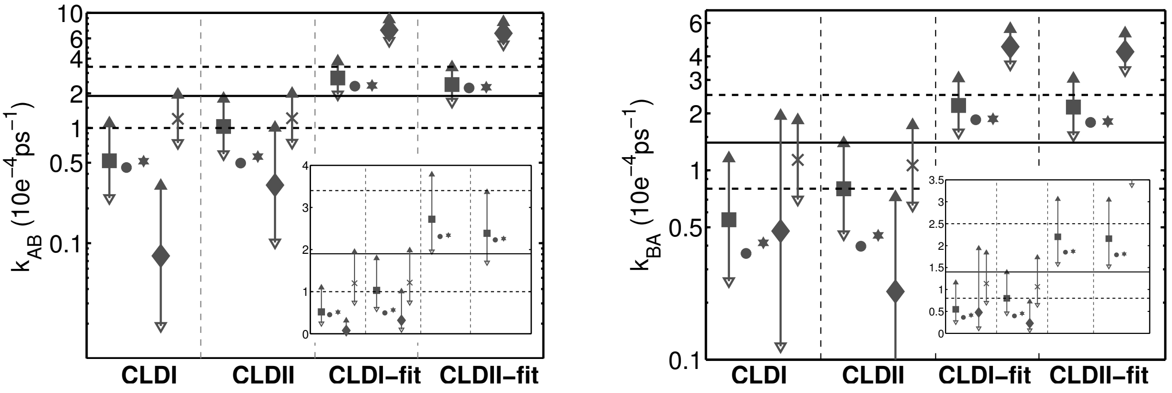


Figure 8.10: Comparison of transition rates. The reference transition rate with its 95% confidence interval is shown by the solid and slashed horizontal lines. The CLD transition rates with errorbars denoting 95% confidence intervals (if available) are grouped in the order $\mathrm{CLD_I}$, $\mathrm{CLD_{II}}$, $\mathrm{CLD_{I-fit}}$ and $\mathrm{CLD_{II-fit}}$. For every model four rates were obtained: stochastic simulation (squares), memory free and full memory Kramers' theory (circles and stars, respectively), and stochastic simulation of the memory free Langevin equation (diamonds). Additionally, rates for $\mathrm{CLD_I}$ and $\mathrm{CLD_{II}}$ obtained by stochastic simulation with the shortened memory functions $\gamma_{\text{I-tail}}$ and $\gamma_{\text{II-tail}}$, respectively, are denoted by crosses.

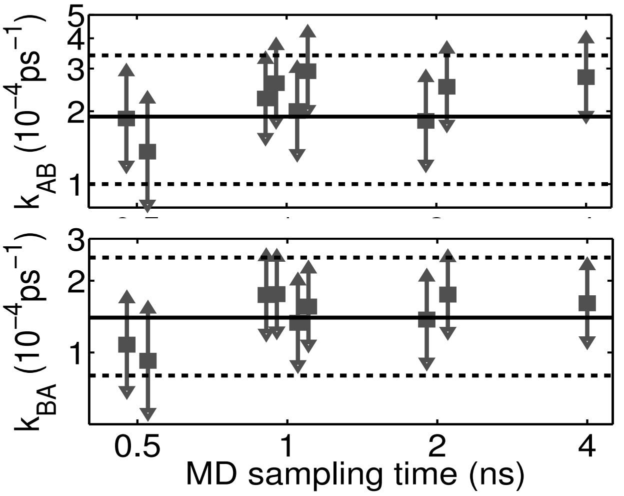


Figure 8.11: Forward (top) and backward (bottom) transition rates predicted by fitted CLD-models in dependence of the sampling time used to obtain the VACF from MD simulations. For the shortest sampling time of 0.5ns only the rates obtained from memory kernel's with the most extreme parameter values were computed. The horizontal lines depict the reference transition rate and it's confidence interval. The boxes and their errorbars depict predicted transition rates and their confidence intervals.

reference transition rates obtained from MD simulations NT1 and NT2 with a total simulation time of 153ns are shown as horizontal lines.

Additionally, transition rates corresponding to the respective CLD models were estimated from Kramers' theory. This estimate relies on the generalized Langevin equation of CLD in harmonic approximation to the free energy. The respective curvatures were determined at the minima as $c_\mathrm{A} = 65\left(\beta \mathrm{L}^2\right)^{-1}$, $c_\mathrm{B} = 76\left(\beta \mathrm{L}^2\right)^{-1}$ and at the barrier as $c_\ddagger = 210\left(\beta \mathrm{L}^2\right)^{-1}$ by fitting parabolas to the free energy profile (cf. Fig. 8.8). The barrier heights were $W^\ddagger - W_A = 1.5\beta^{-1}$ for the forward transition and $W^\ddagger - W_B = 1.8\beta^{-1}$ for the backward transition, respectively. For all four CLD models two Kramers-rates were obtained (cf. Theory), one via memory free Kramers' theory (circles in Fig. 8.10) and the other by full inclusion of memory effects (stars in Fig. 8.10).

	$\int \gamma dt\ (ps^{-1})$	$1/\int \Psi\ (ps^{-1})$	$k_{A\rightarrow B}\ \left(10^{-4}ps^{-1}\right)$	$k_{B\rightarrow A}\ \left(10^{-4}ps^{-1}\right)$
MD			$1.9\quad +1.5/-0.9$	$1.4\quad +1.1/-0.6$
$\mathrm{CLD_I}$	249	147	$0.3\quad +0.7/-0.2$	$0.2\quad +0.5/-0.2$
$\mathrm{CLD_{II}}$	187	120	$0.7\quad +0.7/-0.4$	$0.5\quad +0.5/-0.2$
$\mathrm{CLD_{I-fit}}$	49	56	$2.2\quad +1.1/-0.7$	$1.4\quad +0.7/-0.5$
$\mathrm{CLD_{II-fit}}$	42	30	$2.1\quad +1.2/-0.8$	$1.0\quad +0.6/-0.4$

Table 8.2: The first two columns show effective friction constants estimated from input VACFs (Ψ_{I}, Ψ_{II}, $\Psi_{\text{I-fit}}$ or $\Psi_{\text{II-fit}}$) or from corresponding memory functions (γ_{I}, γ_{II}, $\gamma_{\text{I-fit}}$ or $\gamma_{\text{II-fit}}$). The second two columns show forward and backward transition rates observed in trajectories of the respective CLD models. The reference transition rates from the MD trajectory are provided in the first line.

As can be seen from the figure, the transition rates of simulations $\mathrm{CLD_{I-fit}}$ and $\mathrm{CLD_{II-fit}}$ fall well into the range set by the reference trajectory (cf. horizontal lines), whereas the rates of $\mathrm{CLD_I}$ and $\mathrm{CLD_{II}}$ fall somewhat outside. The rates obtained with Kramers' theory did not differ significantly from the numerical results. Remarkably, all models yielded very similar rates with the memory-free and the full-memory version of Kramers' theory. This could suggest that memory-effects do not influence transition rates significantly for the case at hand, and that integration of equations of motion could be simplified by replacing the generalized friction by a constant friction, $\gamma_{\text{eff}} = \int_0^\infty \gamma(t)dt$.

The transition rates obtained with constant friction, however, show that the opposite is true (cf. diamonds in Fig. 8.10). Integration with a constant friction significantly overestimates the rates for the models $\mathrm{CLD_{I-fit}}$ and $\mathrm{CLD_{II-fit}}$, and underestimates those obtained from $\mathrm{CLD_I}$ and $\mathrm{CLD_{II}}$. Therefore, memory effects *do* play an important role.

It is somewhat surprising that the models $\mathrm{CLD_I}$ and $\mathrm{CLD_{II}}$ underestimated the transition rates despite the fact that their VACF is more accurate. However, the effective friction $\int_0^\infty \gamma(t)dt$ implied by the memory kernels γ_{I} and γ_{II} is too large ($\approx 220\,\text{ps}^{-1}$, as can be seen by comparison with the estimate of $120-150\,\text{ps}^{-1}$ obtained directly for the VACFs (Table 8.2). Indeed, the memory kernels $\gamma_{\text{I-tail}}$ and $\gamma_{\text{II-tail}}$, whose tail was damped down to zero to correct for this mismatch (see Sec. 8.6) yielded improved transition rates (crosses in Figure 8.10). Moreover, for the models $\mathrm{CLD_{I-fit}}$ and $\mathrm{CLD_{II-fit}}$ the transition rates were slightly too high, in agreement with the effective friction being lower than the estimated range of $120-150\,\text{ps}^{-1}$ (cf. Table 8.2). This suggests to use the friction integral as an additional and important regularization criterion for the memory kernel.

The CLD models discussed so far were based on VACFs obtained from simulations NT1 and NT2, with $T > 9$ns simulation time. To check if such a long simulation time is actually necessary to obtain sufficiently accurate memory kernels, we systematically assessed the amount of molecular dynamics sampling needed. To this end, eight $500\,$ps trajectories, NTS_i $i = 1, \ldots, 8$, were generated from different starting positions (cf. Methods) and used to compute memory kernels via the FIT method for parameters see Table 8.1. Memory kernels were computed from single trajectories, or from combinations of two, four or all eight trajectories NTS_i, constituting sampling times of $500\,$ps, 1ns, 2ns and 4ns, respectively. In Fig. 8.11 the transition rates predicted by the CLD model with these

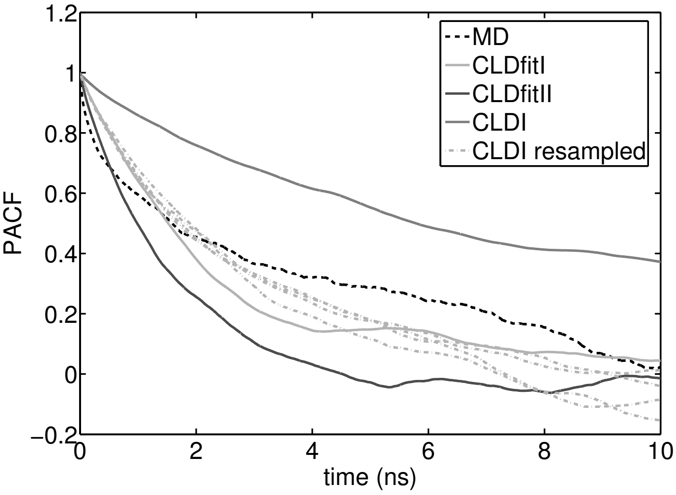


Figure 8.12: Comparison of CLD generated (gray) PACFs with that computed from both MD trajectories NT1 and NT2. The curves of $\mathrm{CLD_I}$ and $\mathrm{CLD_{II-fit}}$ indicate the most extreme PACFs obtained from the discussed CLD models. The PACF of $\mathrm{CLD_{I-fit}}$ agrees best with the MD results. From the scatter of the 5 PACFs for $\mathrm{CLD_{I-fit}}$ (gray, dash-dotted) the statistical error can be estimated.

memory kernels are plotted against used sampling time. All obtained rates were within the range of and are centered at that of the reference MD simulation. The only exception are the rates obtained with memory kernels from the shortest sampling time (0.5ns), which are systematically smaller than the reference MD rate (only the highest and lowest rate were shown). Already for sampling times $t \geq 1$ns the reference rates of the models $\mathrm{CLD_{I-fit}}$ and $\mathrm{CLD_{II-fit}}$ were reproduced. Thus, sampling as short as 1 ns is sufficient to correctly predict transition rates.

8.9 Prediction of positional autocorrelation functions by CLD

The last observable of the CLD dynamics we compared to the reference MD is the positional autocorrelation function (PACF). Figure 8.12 shows the PACF obtained from MD simulations NT1 and NT2 covering a total simulation time of 153 ns in comparison to PACFs obtained from 300 ns CLD trajectories. We plotted the PACFs of model $\mathrm{CLD_I}$ and $\mathrm{CLD_{II-fit}}$ with slowest and fastest decay, respectively, as well as the PACF of $\mathrm{CLD_{I-fit}}$, which best agrees with the MD result.

The overall decay of all CLD-derived PACFs corresponds to that of the reference PACF from the MD simulation. Fits to single-exponential decays yield decay times ranging from 1.35 ns to 9 ns for the CLD-derived PACFs, which are on the same order of magnitude as that obtained from the MD-derived PACF (3.3 ns). Remarkably, the decay of the CLD-derived PACFs is systematically too slow for short times $\tau < 0.5$ ns, whereas on long times some decays are faster and others slower than the reference. Moreover, the CLD-derived PACFs are well described by a single exponential decay, whereas the MD-derived PACF shows two significantly different timescales.

The large spread of the CLD-derived PACFs is striking. In order to rule out that this is due to unconverged correlations we obtained several independent trajectories for each CLD model and computed their PACFs. For $\mathrm{CLD_{I-fit}}$, these are shown in the figure, and their much smaller statistical variation confirms that the spread of the PACFs is indeed significant.

Furthermore, we compared the decay times of the CLD models with their respective transition rates. Correlation coefficients of $r = 0.69$ and $r = 0.72$ for the forward and backward rate, respectively, indicate a weak connection. However, the relatively low value also shows that not all dynamical properties that are relevant for the transition rates are captured by the PACF. Vice versa, other dynamical properties, which are described by the PACF, are not reflected in the transition rates.

The large differences between the PACFs are at first sight unexpected because they are uniquely defined by the corresponding VACFs, which vary much less for the different CLD models (cf. Fig. 8.9). Note, however, that the PACF is dominated by low frequency components, i.e., long time correlations, whereas the VACF is dominated by the high frequency components. The fact that the memory kernels computed from VACFs, therefore, cannot capture the long time correlations explains the observed spread of the CLD-derived PACFs.

Nevertheless, the large spread of the PACFs indicates a tremendous influence of the memory kernels on the long-time dynamics. In order to achieve better accuracy the PACF could be used in addition to the VACF to determine the memory kernel, e.g., solving the alternative Memory equation, Eq. (7.3). However, here one needs to trade-off the accuracy of the CLD model with the sampling time to gain the slowly converging PACF.

8.10 Discussion and Conclusions

The presented results show that the CLD model is capable of accurately predicting transition rates of complex systems. Remarkably, already sampling as short as 1ns proved to be sufficient to obtain a good prediction of transition events occurring on timescales of 50ns.

Furthermore, we found that the transition rate was mainly influenced by the effective friction $\int_0^\infty \gamma(t)dt$. In those cases where the effective friction was accurate, both methods to extract memory kernels, FIT and DIR (with removed tails), performed equally well. Thus, the reproduction of the small oscillations of the VACF, which only the method DIR is capable of, was not important for the transition rate. However, an accurate effective friction alone, i.e., a memory free description, did not suffice for accurate rates (cf. diamonds in Fig. 8.10). Proper treatment of memory is thus important on the timescale of the chosen integration step.

Whereas memory effects were important on the short timescales of the integration steps, they were irrelevant on timescales probed by Kramers' theory. Indeed, the memory, decaying with ($\tau \approx 1$ps), influenced the dynamics over 100 integration steps, whereas the fastest timescale seen by Kramers' theory, i.e., $T \approx 2\pi/\omega = 9$ps is much slower and, therefore, not affected.

To decide which memory extraction method to chose, two different objectives need to be distinguished. The aim to predict transition rates computationally efficient is optimally achieved with the robust method FIT. In particular, VACFs computed from MD trajectories as short as 1 ns have sufficed, here. Anyhow, putative features of the VACF that cannot be captured by FIT do not raise significantly above noise level before sampling time reaches well above 10 ns.

The objective to understand the physical nature of the system via its memory kernel, in contrast, is optimally approached with the method DIR, which was able to accurately reproduce all features of the VACF. A straightforward interpretation of the memory kernel is hindered, however, by artifacts, such as spurious oscillations and a too slow decay. In Chapter 7 several strategies to improve the regularization were discussed, which might be able to remove these artifacts.

Positional autocorrelation functions (PACFs), which are dominated by long time correlations, were not accurately reproduced by the CLD model. This indicates that long time correlations need to be considered more accurately for determination of memory kernels. However, they cannot be extracted from the VACF, because it is dominated by the short time correlations. The most promising strategy, therefore, is to determine the memory kernel via the alternative Memory equation, Eq. (7.3), which includes positional correlations. An adaption of the method DIR to this Memory equation will be straightforward.

Science... never solves a problem
without creating ten more.
— George B. Shaw

Chapter 9

Summary and Conclusions

We have developed Collective Langevin dynamics (CLD) as a consistent framework to describe and simulate slow collective motions of proteins in an approach with drastically reduced number of degrees of freedom and, hence, reduced dimensionality. In this framework the dynamics are separated into slow and fast degrees of freedom. The dynamics in the slow coordinates are evolved explicitly, whereas the fast degrees of freedom are treated in an implicit manner.

CLD is a bottom up approach based on first principles in the sense that all relevant information is extracted from the well validated description of protein dynamics by molecular dynamics (MD) simulations. Furthermore, it is a systematic approach because the level of coarse graining can be tuned by the number of degrees of freedom which are explicitly considered. The extreme case of a one dimensional description is presented here; the other extreme is explicit consideration of all degrees of freedom and in the CLD framework would trivially reproduce the MD model.

It was shown that suitable slow coordinates can be systematically obtained with principal component analysis (PCA) from short (nanoseconds) explicit MD simulations and are stable enough to also properly describe protein dynamics at much longer time scales. In particular, for crambin ten percent of the principal components obtained from only a 5ns MD trajectory were shown to describe over 85% of the atom displacement observed in a 450 ns MD simulation. Furthermore, PCA, if based on the covariance matrix of the displacement of all *heavy* atoms (as opposed to C_α-atoms only), proved able to separate timescales to a large extent. In particular, those modes which describe the slow degrees of freedom are nearly free of contributions from the fast vibrational dynamics.

This partial separation of timescales motivated and justified the application of the projection operator formalism by Mori and Zwanzig to derive equations of motions for the dynamics of the collective coordinates. Both, linear (e.g., principal components) and curved coordinates were considered in full generality. The resulting exact equations of motions take the form of a generalized Langevin equation with a potential of mean force. Here, we approximate this exact equation by replacing its *noise term* with a non-Markovian stochastic process that obeys the fluctuation-dissipation theorem. The memory effects are found to be not negligible and thus are fully accounted for by a generalized frictional force, whose specific memory kernel is obtained for any dynamical system individually.

We proposed three methods to extract memory kernels from short (few nanoseconds) MD trajectories deriving it from either a velocity autocorrelation function or a force autocorrelation function. In order to obtain memory kernels from the former we solved a Volterra-type equation. Because this inverse problem is notoriously difficult to solve and suffers from numerical instabilities, we tested different levels of regularization. The method FIT applied rather strong regularization, and hence was very robust against the inherent statistical noise in the VACF. In contrast, the second method, DIR, regularized only weakly, such that it allowed to capture more details of the VACF. The results indicated that for an accurate description of transition rates, the trade-off should be struck on the side of stronger regularization, i.e., increased robustness.

The third approach is conceptually simpler, because it exploits the more direct link of the force autocorrelation function to the memory kernel via the fluctuation-dissipation theorem, but involves a higher computational effort. Its strength, though, is to probe the spatial dependence of the memory kernel, which is neglected in the currently used CLD model. Our preliminary investigation of memory kernels obtained with this method at numerous positions in the conformational subspace of neurotensin, suggested a substantial positional dependence of the memory. Whether and how strong this affects the collective dynamics needs to be established in further research.

CLD is complementary and rests upon the many existing enhanced sampling methods to calculate free energy surfaces such as, REMD[73], umbrella sampling[75, 202] or SMC[74]. All these methods, by construction, sacrifice dynamics to speed up sampling. We have proposed to reconstruct the conformational dynamics from the obtained free energy surfaces via CLD. Alternatively, ensembles obtained from experimental sources like NMR might also be used to estimate a free energy surface.

As a test system, the hexapeptide neurotensin was considered. Explicit treatment in CLD was restricted to a one-dimensional (curved) conformational coordinate. Comparison of transition rates obtained from this extremely dimension reduced and, hence very efficient, description with those obtained from a 150 ns MD simulation showed excellent agreement.

Remarkably, this good agreement for the neurotensin peptide was achieved by the most extreme conceivable dimension reduction, i.e., to only one dimension. A generalized curved coordinate was required to achieve such a drastic reduction; more than one but less than five linear degrees of freedom would likely allow to achieve similar accuracy.

We note that similar tests for much larger protein systems would of course be called for to further evaluate our approach. However, the requirement of converged reference transition rates from long MD simulations, severely restricts the size of the test system. For instance, the presented 450 ns simulation of crambin did not contain enough recurring transitions to reliably estimate reference transition rates, whereas enough transitions occurred in the presented 150ns simulation of neurotensin. Nevertheless, our results indicate that CLD is also capable of accurately describing conformational dynamics of soluble proteins at μs time scales.

The generality of the treatment allows to apply the CLD framework to model also a water file in a protein pore, e.g., aquaporin, gramicidin, etc. The water flux can be measured experimentally, and is also accessible by sufficiently long all-atom MD simulations. Therefore, such a system is a good candidate for the next test of the CLD approach. A single collective degree of freedom was able to describe the motion of the whole chain of water molecules along the pore axis. In this way, elegantly accounting for the collectivity of the water motion, the CLD model might be able to accurately predict the water flux.

Our test simulations also demonstrated a large reduction of computational effort by the CLD method. Here transition rates were accurately predicted for much longer $(\sim 50\text{ns})$ timescales than needed for extraction of memory kernels (∼1ns). A $300\,$ns CLD trajectory was obtained in 6 minutes on a desktop computer (AMD 1.8GHZ Opteron), whereas a comparable explicit MD simulation requires 5 months on the same hardware.

CLD yields trajectories with accurate thermodynamical and dynamical behavior, in particular accurate free energies and velocity autocorrelation functions. By focusing on relevant quantities, our CLD approach also provides new physical insights into the high-dimensional protein dynamics. The relative fast decay of the memory kernel of neurotensin agrees with previous findings. For a similarly sized peptide an upper limit for a time scale on which no memory effect influenced transition rates was determined to be $1\,$ns[100]. This limit agrees with and is improved by our finding that memory effects did not play a significant role for transition rates at time scales above $10\,$ps. In focusing at accurate velocity autocorrelation functions, CLD might be particularly useful for the interpretation of neutron scattering experiments, which probe these.

The observed deviations of the CLD-derived *positional autocorrelation functions* indicate that for this observable memory effects on longer time-scales are important. We further suggest to improve the accuracy of the required memory kernel by combining positional and velocity autocorrelation functions for its extraction, because the former probe long time scales and the latter short ones.

We demonstrated that different memory functions can lead to the same dynamics, which renders their direct physical interpretation problematic. Further research is required to find the 'invariant properties' hidden in the memory functions, i.e., those properties that cannot be altered without changing the dynamics. As possible candidates we suggested effective friction constants and decay times. Following this line of investigation one would find out which kind of knowledge about the physical system can be extracted from a memory function.

With the generalized correlation coefficient devised in Chapter 3 we provided a measure to quantify any correlated motion in MD simulations. Thereby, we removed long standing obstacles for a quantitative comparison of correlations between MD simulations and experiment. As shown, the hitherto used method suffered from a purely geometrical artifact, such that more than 50% of the correlations remained undetected. The enhanced characterization of the collective motion provided by the generalized correlation matrix also complements the analysis of collective motions with PCA.

The generalized correlation coefficient was applied to shed light on an experiment that attempted to measure correlated backbone motion in the B1 domain of Protein G. Recently order parameters for ten different mutants of this protein were obtained with NMR relaxation experiments. It was proposed to interprete the observed covariations of the order parameters as a probe for correlated motion of the protein backbone. However, our results cast strong doubts on this interpretation. We speculate that the observed covariations are rather due to remarkably correlated structural plasticities. Further simulations of all the ten studied mutants will thus be required to test this hypothesis and to structurally characterize the properties of this proposed non-local plasticity.

As alternative to PCA we developed *Full Correlation Analysis (FCA)* to gain maximally uncoupled collective modes, which shall prove particularly suitable for use within the CLD framework. We applied this method to extract collective modes for neurotensin, which successfully removed an inconsistency previously observed in the free energy surface of the first two PCA modes. Moreover, FCA aligns the extracted modes along the pathways of conformational transitions. Both results suggest that FCA is promising for application within CLD.

Besides its use for CLD, we suppose that FCA can be used in many other applications. As shown, FCA modes are less coupled and allow better separation of conformational substates, but have otherwise similar beneficial characteristics as PCA modes.

One such application, for instance, could address the long standing problem to compute the configurational entropy of a macromolecule from an MD trajectory[242]. Although the entropy is accurately described by MD simulations, it is infeasible to perform the required non-parametric density estimation in the high-dimensional configurational space. As a first step in the framework of PCA, the entropy is estimated by treating all PCA modes as independent oscillators[81, 161]. Obviously this approximation will improve if we use FCA to extract maximally decoupled coordinates, and treat those as independent oscillators. However, FCA might enable a more considerable increase of accuracy. As shown, the FCA modes of neurotensin separated into relatively uncoupled clusters of less than ten modes. Since it is possible to accurately estimate the entropy with non-parametric methods in such low-dimensional subspaces[147, 148], a more accurate estimate of the configurational entropy might be gained as sum of the non-parametric estimations of the entropy in these subspaces.

Another route to future developments for CLD concerns the number of degrees of freedom that are explicitly considered. Whereas two or three explicit degrees of freedom can be treated within the presented CLD framework in a straightforward manner, inclusion of more explicit coordinates will become impractical due to the high dimensionality of the free energy landscapes, which would render the non-parametric free energy estimation used here infeasible. As an alternative, weighted sums of multivariate Gaussians could be used to approximate the ensemble density. A CLD model based on a similar parametric approximation was already used in this work in the Kramers' approach, and its rates agreed well with those obtained from the non-parametric free energy surface. Preliminary work of the author indicates that a maximum-likelihood estimate of weights, positions and widths of such Gaussians yields an accurate fit to high-dimensional densities, as checked by a newly devised high-dimensional goodness-of-fit method.

We finally suggest that the extension to large conformational subspaces might allow on-the-fly computations of small regions of the free energy landscape, thereby, alleviating the sampling problem. In particular, the higher frequency PCA modes behave quasi-harmonically, and are much more efficiently sampled by MD then the low frequency modes. Thus, a two layered approach for CLD might be considered, which switches to explicit MD to probe entropic contributions to the free energy, whenever new, previously unvisited, regions of the conformational subspace are encountered.

Chapter 10
Acknowledgments

At last i want to express my gratitude to everybody who encouraged and supported the work for this thesis.

First of all, i would like to thank PD. Dr. Helmut Grubmüller, with the reasons for my acknowledgment being manyfold. He at all times had an open ear for any questions and had an enormous capacity to provide important impulses to my work. His steady support, at the same time, came along with the freedom to follow own ideas, and the liberty to find a balance of order and chaos in my scientific work that suited me. His unbelievable patience and dedication for training of my scientific writing and his polishing of my presentation skills deserves special praise. Moreover, I am happy that he convinced me to start the PhD work immediately after finishing my master degree in maths instead of spending at least a further year on the remaining part of my chemistry degree.

I also thank Prof. Dr. Annette Zippelius for her willingness to take over the position of the primary referee for this thesis.

I very much thank the whole Theoretical Molecular Biophysics department, its former and current members, for all the discussions, help with Gromacs and all the fun. I especially would like to name Bert de Groot, who, as my room mate in the first years, endured the freshman's barrage of mostly technical questions with memorable patience, and had always a suitable MD trajectory at hand, which could be used to test my methods. His continuous interest in the comparison of correlations between MD simulations and NMR saved me from abolishing this line of research too early. Of my group members I also would like to name Gunnar Schröder for many ideas and discussions, Lars V. Schäfer for his strong friendship, an open ear for problems and, together with Jürgen Haas, Frauke Meyer and Guillem Portella, for reading major parts of this thesis, Daniel Seeliger for help with Python, Martin Stumpe for exchange of MATLAB-lore, and Eveline Heinemann, Oliver Slawik and Ingo Hoffmann for providing and maintaining always a superb working environment. I would like to thank Frauke Meyer also for inviting me to Shanghai as assistant for the practical lessons during an exciting EMBO workshop in 2004.

I am glad that my talk in Ringberg 2003 inspired a collaboration with Jeremy Smith, and i am thankful for the many useful discussions with him and his post-doc Kei Moritsugo.

I thank Gert Vriend for the opportunity to create with OLGA something useful in times of bleak outlook for my thesis project.

I would like to express my thanks to the Max-Planck-Society and the Volkswagen Foundation for financial support. Stimulating visits of major conferences in Europe and overseas were enabled amongst others by the German Research Foundation.

This thesis would not be on hand in this form without those persons, my family and friends, who supportively accompanied me before and during this work. I am indebted to my parents for encouraging and backing me in all years, and for granting unrestricted freedom to follow my interests. Thanks to my sisters Heike and Katrin, for all good advice and enjoyable times.

Bibliography

[1] L. Stryer. *Biochemistry*. W. H. Freeman and Co., New York, 3rd edition, 1988.

[2] W. Hoppe, W. Lohmann, H. Markl, and H. Ziegler, editors. *Biophysik*. Springer, 1982.

[3] J. W. Jung and W. Lee. Structure-based functional discovery of proteins: Structural proteomics. *J. Biochem. Mol. Biol.*, 37:28–34, 2004.

[4] A. T. Brunger and M. Nilges. Computational challenges for macromolecular structure determination by Xray crystallography and solution NMR spectroscopy. *Q. Rev. Biophys.*, 26:49–125, 1993.

[5] M. Nilges. Structure calculation from NMR data. *Curr. Opin. Struct. Biol.*, 6:617–623, 1996.

[6] J. G. Kempf and J.P. Loria. Protein dynamics from solution NMR theory and applications. *Cell Biochem. Biophys.*, 37:187–211, 2003.

[7] H. J. Steinhoff. Methods for study of protein dynamics and protein-protein interaction in protein-ubiquitination by electron paramagnetic resonance spectroscopy. *Frontiers in Biosicience*, 7:C97–C110, 2002.

[8] J. C. Smith. Protein dynamics — comparison of simulations with inelastic neutron-scattering experiments. *Q. Rev. Biophys.*, 24:227–291, 1991.

[9] F. Gabel, D. Bicout, U. Lehnert, M. Tehei, M. Weik, and G. Zaccai. Protein dynamics studied by neutron scattering. *Q. Rev. Biophys.*, 35:327–367, 2002.

[10] S. Weiss. Fluorescence spectroscopy of single biomolecules. *Science*, 283:1676–1683, 1999.

[11] M. Gerstein, A. M. Lesk, and C. Chothia. Structural mechanisms for domain movements in proteins. *Biochemistry*, 33(22):6739–6749, 1994.

[12] V. Srajer, T. Y. Teng, T. Ursby, C. Pradervand, Z. Ren, S. Adachi, W. Schildkamp, D. Bourgeois, M. Wulff, and K. Moffat. Photolysis of the carbon monoxide complex of myoglobin: Nanosecond time-resolved crystallography. *Science*, 274(5293):1726–1729, 1996.

[13] V. Srajer, Z. Ren, T. Y. Teng, M. Schmidt, T. Ursby, D. Bourgeois, C. Pradervand, W. Schildkamp, M. Wulff, and K. Moffat. Protein conformational relaxation and ligand migration in myoglobin: A nanosecond to millisecond molecular movie from time-resolved laue X-ray diffraction. *Biochemistry*, 40(46):13802–13815, 2001.

[14] J. Norberg and L. Nilsson. Advances in biomolecular simulations: Methodology and recent applications. *Q. Rev. Biophys.*, 36(3):257–306, 2003.

[15] R. H. Zhou, E. Harder, H. F. Xu, and B. J. Berne. Efficient multiple time step method for use with Ewald and particle mesh Ewald for large biomolecular systems. *J. Chem. Phys.*, 115(5):2348–2358, 2001.

[16] M. E. Tuckerman, B. J. Berne, and G. J. Martyna. Molecular-dynamics algorithm for multiple time scales - systems with long-range forces. *J. Chem. Phys.*, 94(10):6811–6815, 1991.

[17] M. Tuckerman, B. J. Berne, and G. J. Martyna. Reversible multiple time scale molecular-dynamics. *J. Chem. Phys.*, 97(3):1990–2001, 1992.

[18] P. Minary, M. E. Tuckerman, and G. J. Martyna. Long time molecular dynamics for enhanced conformational sampling in biomolecular systems. *Phys. Rev. Lett.*, 93(15):–, 2004.

[19] J. L. Scully and J. Hermans. Multiple time steps - limits on the speedup of molecular-dynamics simulations of aqueous systems. *Mol. Simul.*, 11(1):67–77, 1993.

[20] F. Zhang. Operator-splitting integrators for constant-temperature molecular dynamics. *J. Chem. Phys.*, 106(14):6102–6106, 1997.

[21] J. A. Board, J. W. Csey, J. F. Leathrum, A. Windemuth, and K. Schulten. Accelerated molecular-dynamics simulation with the parallel fast multipole algorithm. *Chem. Phys. Lett.*, 198(1-2):89–94, 1992.

[22] A. M. Mathiowetz, A. Jain, N. Karasawa, and W. A. Goddard. Protein simulations using techniques suitable for very large systems - the cell multipole method for nonbond interactions and the Newton-Euler inverse mass operator method for internal coordinate dynamics. *Proteins*, 20(3):227–247, 1994.

[23] M. Eichinger, H. Grubmüller, H. Heller, and P. Tavan. FAMUSAMM: An algorithm for rapid evaluation of electrostatic interactions in molecular dynamics simulations. *J. Comput. Chem.*, 18(14):1729–1749, 1997.

[24] L. Greengard and V. Rokhlin. On the evaluation of electrostatic interactions in molecular modeling. *Chem. Scr.*, 29A:139–144, 1989.

[25] A. Y. Toukmaji and J. A. Board. Ewald summation techniques in perspective: A survey. *Comput. Phys. Commun.*, 95(2-3):73–92, 1996.

[26] J. P. Ryckaert, G. Ciccotti, and H. J. C. Berendsen. Numerical-integration of cartesian equations of motion of a system with constraints - molecular-dynamics of n-alkanes. *J. Comput. Phys.*, 23(3):327–341, 1977.

[27] S. Miyamoto and P. A. Kollman. SETTLE: An analytical version of the SHAKE and RATTLE algorithms for rigid water models. *J. Comp. Chem.*, 13:952–962, 1992.

[28] B. Hess, H. Bekker, H. J. C. Berendsen, and J. G. E. M. Fraaije. LINCS: A linear constraint solver for molecular simulations. *J. Comp. Chem.*, 18:1463–1472, 1997.

[29] K. Tai. Conformational sampling for the impatient. *Biophys. Chem.*, 107(3):213–220, 2004.

[30] N. Gō and H. A. Scheraga. Analysis of contribution of internal vibrations to statistical weights of equilibrium conformations of macromolecules. *J. Chem. Phys.*, 51(11):4751, 1969.

[31] B. Roux and T. Simonson. Implicit solvent models. *Biophys. Chem.*, 78:1–20, 1999.

[32] S. J. Marrink and D. P. Tieleman. Molecular dynamics simulation of spontaneous membrane fusion during a cubic-hexagonal phase transition. *Biophys. J.*, 83(5):2386–2392, 2002.

[33] G. Ayton and G. A. Voth. Bridging microscopic and mesoscopic simulations of lipid bilayers. *Biophys. J.*, 83(6):3357–3370, 2002.

[34] T. Head-Gordon and S. Brown. Minimalist models for protein folding and design. *Curr. Opin. Struct. Biol.*, 13(2):160–167, 2003.

[35] A. Liwo, M. Khalili, and H. A. Scheraga. Ab initio simulations of protein-folding pathways by molecular dynamics with the united-residue model of polypeptide chains. *Proc. Natl. Acad. Sci. U. S. A.*, 102(7):2362–2367, 2005.

[36] J. P. Ulmschneider and W. L. Jorgensen. Monte Carlo backbone sampling for polypeptides with variable bond angles and dihedral angles using concerted rotations and a gaussian bias. *J. Chem. Phys.*, 118(9):4261–4271, 2003.

[37] F. Sartori, B. Melchers, H. Bottcher, and E. W. Knapp. An energy function for dynamics simulations of polypeptides in torsion angle space. *J. Chem. Phys.*, 108(19):8264–8276, 1998.

[38] A. Kloczkowski, J. E. Mark, and B. Erman. Chain dimensions and fluctuations in random elastomeric networks. 1. Phantom gaussian networks in the undeformed state. *Macromolecules*, 22(3):1423–1432, 1989.

[39] A. E. Garcia. Large-amplitude nonlinear motions in proteins. *Phys. Rev. Lett.*, 68(17):2696–2699, 1992.

[40] H. J. C. Berendsen and S. Hayward. Collective protein dynamics in relation to function. *Curr. Opin. Struct. Biol.*, 10(2):165–169, 2000.

[41] R. Zwanzig. Memory effects in irreversible thermodynamics. *Phys. Rev.*, 124(4), 1961.

[42] Hazime Mori. Transport, collective motion, and brownian motion. *Prog. Theoret. Phys.*, 33(3):423, 1965.

[43] I. Benjamin, L. L. Lee, Y. S. Li, A. Liu, and K. R. Wilson. Generalized Langevin model for molecular-dynamics of an activated reaction in solution. *Chem. Phys.*, 152(1-2):1–12, 1991.

[44] R. Ferrando, R. Spadacini, and G. E. Tommei. Jump rate and jump lengths in periodic systems with memory. *Chem. Phys. Lett.*, 347(4-6):487–492, 2001.

[45] C. C. Martens. Qualitative dynamics of generalized Langevin equations and the theory of chemical reaction rates. *J. Chem. Phys.*, 116(6):2516–2528, 2002.

[46] W. K. Park and S. C. Park. 3d generalized Langevin equation approach to gas-surface reactive scattering: Model H+H -> H-2/Si(100)-(2x1). *Theochem-J. Mol. Struct.*, 630:215–223, 2003.

[47] P. Romiszowski and R. Yaris. A dynamic simulation method suppressing uninteresting degrees of freedom. *J. Chem. Phys.*, 94(10):6751–6761, 1991.

[48] Y. X. Zhang and S. C. Park. 3d generalized Langevin equation (GLE) approach to gas-surface energy transfer: Model H+H -> H-2/Si(100)-(2x1). *Bull. Korean Chem. Soc.*, 21(11):1095–1100, 2000.

[49] A. Kitao, F. Hirata, and N. Gō. The effects of solvent on the conformation and the collective motions of protein - normal mode analysis and molecular-dynamics simulations of melittin in water and in vacuum. *Chem. Phys.*, 158(2-3):447–472, 1991.

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

[51] A. Kitao and N. Gō. Investigating protein dynamics in collective coordinate space. *Curr. Opin. Struct. Biol.*, 9(2):164–169, 1999.

[52] M. A. Balsera, W. Wriggers, Y. Oono, and K. Schulten. Principal component analysis and long time protein dynamics. *J. Phys. Chem.*, 100(7):2567–2572, 1996.

[53] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, New York, 1991.

[54] S. Hery, D. Genest, and J. C. Smith. X-ray diffuse scattering and rigid-body motion in crystalline lysozyme probed by molecular dynamics simulation. *J. Mol. Biol.*, 279(1):303–319, 1998.

[55] D. Ringe and G. A. Petsko. Mapping protein dynamics by X-ray-diffraction. *Prog. Biophys. Mol. Biol.*, 45(3):197–235, 1985.

[56] G. A. Petsko and D. Ringe. Fluctuations in protein-structure from X-ray-diffraction. *Ann. Rev. Biophys. Bioeng.*, 13:331–371, 1984.

[57] T. Ichiye and M. Karplus. Collective motions in proteins - a covariance analysis of atomic fluctuations in molecular-dynamics and normal mode simulations. *Proteins*, 11(3):205–217, 1991.

[58] P. H. Hünenberger, A. E. Mark, and W. F. van Gunsteren. Fluctuation and cross-correlation analysis of protein motions observed in nanosecond molecular dynamics simulations. *JMB*, 252(3):492–503, 1995.

[59] K. L. Mayer, M. R. Earley, S. Gupta, K. Pichumani, L. Regan, and M. J. Stone. Covariation of backbone motion throughout a small protein domain. *Nat. Struct. Biol.*, 10(11):962–965, 2003.

[60] Bruce J. Berne and G. D. Harp. On the calculation of time correlation functions. *Adv. Chem. Phys.*, 17:63–227, 1970.

[61] G. R. Kneller and K. Hinsen. Computing memory functions from molecular dynamics simulations. *J. Chem. Phys.*, 115(24):11097–11105, 2001.

[62] J. P. Boon and S. A. Rice. Memory effects and autocorrelation function of a dynamical variable. *J. Chem. Phys.*, 47(7):2480, 1967.

[63] P. A. Egelstaff. Collective modes and many-body forces in fluids — an experimental-study. *Phys. Chem. Liq.*, 16(4):293–305, 1987.

[64] P. S. Damle and A. D. Tillu. Memory function of velocity auto-correlation in liquid argon. *Indian J. Pure Appl. Phys.*, 7(8):539, 1969.

[65] S. C. Jain and R. C. Bhandari. Memory effects and dynamical correlations in liquid argon and sodium. *Physica A*, 52(3):393–, 1971.

[66] M. Berkowitz, J. D. Morgan, D. J. Kouri, and J. A. McCammon. Memory kernels from molecular-dynamics. *J. Chem. Phys.*, 75(5):2462–2463, 1981.

[67] F. Shimojo, K. Hoshino, and M. Watabe. Dynamical correlation-functions and memory functions of liquid-sodium. 2. A mode-coupling analysis. *J. Phys. Soc. Jpn.*, 63(5):1821–1827, 1994.

[68] T. Yamaguchi, Y. Kimura, and N. Hirota. Molecular dynamics simulation of solute diffusion in lennard-jones fluids. *Mol. Phys.*, 94(3):527–537, 1998.

[69] S. L. Yang and J. S. Cao. Direct measurements of memory effects in single-molecule kinetics. *J. Chem. Phys.*, 117(24):10996–11009, 2002.

[70] B. J. Gertner, K. R. Wilson, and J. T. Hynes. nonequilibrium solvation effects on reaction-rates for model SN_2 reactions in water. *J. Chem. Phys.*, 90(7):3537–3558, 1989.

[71] B. D. Bursulaya and C. L. Brooks. Folding free energy surface of a three-stranded beta-sheet protein. *J. Am. Chem. Soc.*, 121(43):9947–9951, 1999.

[72] P. E. Smith. The alanine dipeptide free energy surface in solution. *J. Chem. Phys.*, 111(12):5568–5579, 1999.

[73] Y. Sugita and Y. Okamoto. Replica-exchange molecular dynamics method for protein folding. *Chem. Phys. Lett.*, 314(1-2):141–151, 1999.

[74] A. Kidera. Smart Monte Carlo simulation of a globular protein. *Int. J. Quantum Chem.*, 75(3):207–214, 1999.

[75] G. M. Torrie and J. P. Valle. Monte-Carlo study of a phase-separating liquid-mixture by umbrella sampling. *J. Chem. Phys.*, 66(4):1402–1408, 1977.

[76] H. Grubmüller and P. Tavan. Multiple time step algorithms for molecular dynamics simulations of proteins: How good are they? *J. Comp. Chem.*, 19(13):1534–1552, 1998.

[77] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, New York, 2001.

[78] B. Brooks and M. Karplus. Harmonic dynamics of proteins - normal-modes and fluctuations in bovine pancreatic trypsin-inhibitor. *Proc. Natl. Acad. Sci. USA*, 80(21):6571–6575, 1983.

[79] N. Gō, T. Noguti, and T. Nishikawa. Dynamics of a small globular protein in terms of low-frequency vibrational-modes. *Proc. Natl. Acad. Sci. USA*, 80(12):3696–3700, 1983.

[80] M. Levitt, C. Sander, and P. S. Stern. The normal-modes of a protein - native bovine pancreatic trypsin-inhibitor. *Int. J. Quantum Chem.*, pages 181–199, 1983.

[81] M. Karplus and J. N. Kushick. Method for estimating the configurational entropy of macromolecules. *Macromolecules*, 14(2):325–332, 1981.

[82] R. M. Levy, M. Karplus, J. Kushick, and D. Perahia. Evaluation of the configurational entropy for proteins - application to molecular-dynamics simulations of an alpha-helix. *Macromolecules*, 17(7):1370–1374, 1984.

[83] R. M. Levy, A. R. Srinivasan, W. K. Olson, and J. A. McCammon. quasi-harmonic method for studying very low-frequency modes in proteins. *Biopolymers*, 23(6):1099–1112, 1984.

[84] M. M. Teeter and D. A. Case. Harmonic and quasiharmonic descriptions of crambin. *J. Phys. Chem.*, 94(21):8091–8097, 1990.

[85] I. Bahar, B. Erman, T. Haliloglu, and R. L. Jernigan. Efficient characterization of collective motions and interresidue correlations in proteins by low-resolution simulations. *Biochemistry*, 36(44):13512–13523, 1997.

[86] T. D. Romo, J. B. Clarage, D. C. Sorensen, and G. N. Phillips. Automatic identification of discrete substates in proteins - singular-value decomposition analysis of time-averaged crystallographic refinements. *Proteins*, 22(4):311–321, 1995.

[87] S. Hayward, A. Kitao, F. Hirata, and N. Gō. effect of solvent on collective motions in globular protein. *J. Mol. Biol.*, 234(4):1207–1217, 1993.

[88] S. Hayward and N. Gō. Collective variable description of native protein dynamics. *Annu. Rev. Phys. Chem*, 46:223–250, 1995.

[89] S. Hayward, A. Kitao, and N. Gō. Harmonicity and anharmonicity in protein dynamics - a normal-mode analysis and principal component analysis. *Proteins*, 23(2):177–186, 1995.

[90] A. Amadei, B. L. de Groot, M. A. Ceruso, M. Paci, A. Di Nola, and H. J. C. Berendsen. A kinetic model for the internal motions of proteins: Diffusion between multiple harmonic wells. *Proteins: Struct. Funct. Genet.*, 35:283–292, 1999.

[91] I. Daidone, A. Amadei, D. Roccatano, and A. DiNola. Molecular dynamics simulation of protein folding by essential dynamics sampling: Folding landscape of horse heart cytochrome c. *Biophys. J.*, 85(5):2865–2871, 2003.

[92] R. A. Bockmann and H. Grubmüller. Nanoseconds molecular dynamics simulation of primary mechanical energy transfer steps in f-1-atp synthase. *Nat. Struct. Biol.*, 9(3):198–202, 2002.

[93] M. C. Lee, J. X. Deng, J. M. Briggs, and Y. Duan. Large-scale conformational dynamics of the HIV-1 integrase core domain and its catalytic loop mutants. *Biophys. J.*, 88(5):3133–3146, 2005.

[94] C. J. Chen, Y. Xiao, and L. S. Zhang. A directed essential dynamics simulation of peptide folding. *Biophys. J.*, 88(5):3276–3285, 2005.

[95] B. L. de Groot, G. Vriend, and H. J. C. Berendsen. Conformational changes in the chaperonin GroEL: New insights into the allosteric mechanism. *J. Mol. Biol.*, 286(4):1241–1249, 1999.

[96] D. Mustard and D. W. Ritchie. Docking essential dynamics eigenstructures. *Proteins*, 60(2):269–274, 2005.

[97] A. Amadei, A. B. M. Linssen, B. L. de Groot, D. M. F. vanAalten, and H. J. C. Berendsen. An efficient method for sampling the essential subspace of proteins. *J. Biomol. Struct. Dyn.*, 13(4):615–625, 1996.

[98] R. Abseher and M. Nilges. Efficient sampling in collective coordinate space. *Proteins*, 39(1):82–88, 2000.

[99] A. L. Tournier and J. C. Smith. Principal components of the protein dynamical transition. *Phys. Rev. Lett.*, 91(20):–, 2003.

[100] B. L. de Groot, X. Daura, A. E. Mark, and H. Grubmüller. Essential dynamics of reversible peptide folding: Memory-free conformational dynamics governed by internal hydrogen bonds. *J. Mol. Biol.*, 309(1):299–313, 2001.

[101] A. Kitao, S. Hayward, and N. Gō. Energy landscape of a native protein: Jumping-among-minima model. *Proteins: Struct. Funct. Genet.*, 33:496–517, 1998.

[102] A. Amadei, M. A. Ceruso, and A. DiNola. On the convergence of the conformational coordinates basis set obtained by the essential dynamics analysis of proteins' molecular dynamics simulations. *Proteins*, 36(4):419–424, 1999.

[103] W. F. vanGunsteren and H. J. C. Berendsen. Computer-simulation of molecular-dynamics - methodology, applications, and perspectives in chemistry. *Angew. Chem. Int. Ed.*, 29(9):992–1023, 1990.

[104] C. L. III Brooks, M. Karplus, and B. M. Pettitt. *A theoretical perspective of Dynamics, Structure and Thermodynamics*, volume 71. John Wiley & Sons, New York, 1988.

[105] A. R. Leach. *Molecular Modelling, Principles and Applications*. Prentice-Hall, 2001.

[106] J. M. Haile. *Molecular Dynamics Simulation. Elementary Methods*. John Wiley & Sons, New York, 1992.

[107] W. L. Jorgensen, D. S. Maxwell, and Tirado-Rives. Development and testing of the OPLS all-atom force field on conformational energetics and properties of organic liquids. *J. Am. Chem. Soc.*, 118:11225–11236, 1996.

[108] B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan, and M. Karplus. CHARMM: A program for macromolecular energy, minimization, and dynamics calculations. *J. Comp. Chem.*, 4:187–217, 1983.

[109] W. F. van Gunsteren and H. J. C. Berendsen. *Groningen Molecular Simulation (GROMOS) Library Manual*. Biomos, Groningen, 1987.

[110] S. J. Weiner, P. A. Kollman, D. T. Nguyen, and D. A. Case. An all atom force field for simulations of proteins and nucleic acids. *J. Comp. Chem.*, 7:230–252, 1986.

[111] N. L. Allinger, Y. H. Yuh, and J.H. Lii. Molecular mechanics. The MM3 force fields for hydrocarbons. *J. Am. Chem. Soc.*, 111:8551–8566, 1989.

[112] K. Nam, X. Prat-Resina, M. Garcia-Viloca, L. S. Devi-Kesavan, and J. L. Gao. Dynamics of an enzymatic substitution reaction in haloalkane dehalogenase. *J. Am. Chem. Soc.*, 126(5):1369–1376, 2004.

[113] B. L. deGroot, T. Frigato, V. Helms, and H. Grubmüller. The mechanism of proton exclusion in the aquaporin-1 water channel. *J. Mol. Biol.*, 333(2):279–293, 2003.

[114] E. Lindahl, B. Hess, and D. Van der Spoel. GROMACS 3.0: A package for molecular simulation and trajectory analysis. *J. Mol. Model.*, 7:306–317, 2001. Internet: `http://www.gromacs.org`.

[115] R. W. Hockney, S. P. Goel, and J. W. Eastwood. 10000 particle molecular dynamics model with long-range forces. *Chem. Phys. Lett.*, 21(3):589–591, 1973.

[116] S. Nose. A molecular-dynamics method for simulations in the canonical ensemble. *Mol. Phys.*, 52(2):255–268, 1984.

[117] H. J. C. Berendsen, J. P. M. Postma, A. DiNola, and J. R. Haak. Molecular dynamics with coupling to an external bath. *J. Chem. Phys.*, 81:3684–3690, 1984.

[118] W. G. Hoover. Canonical dynamics - equilibrium phase-space distributions. *Phys. Rev. A*, 31(3):1695–1697, 1985.

[119] Tom Darden, Darrin York, and Lee Pedersen. Particle mesh Ewald: An N·log(N) method for Ewald sums in large systems. *J. Chem. Phys.*, 98:10089–10092, 1993.

[120] U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. G. Pedersen. A smooth particle mesh Ewald method. *J. Chem. Phys.*, 103(19):8577–8593, 1995.

[121] N. H. Heo M. M. Teeter, S. M. Roe. Atomic resolution (0.83 a) crystal structure of the hydrophobic protein crambin at 130 k. *J. Mol. Biol.*, 230:292–, 1993.

[122] W.F. Van Gunsteren, S.R. Billeter, A.A. Eising, P.H. Hünenberger, P. Krüger, A.E. Mark, W.R.P. Scott, and I.G. Tironi. *Biomolecular simulation: the GROMOS96 manual and user guide*. Biomos b.v., Zürich, Groningen, 1996.

[123] J. Hermans, H. J. C. Berendsen, W. F. van Gunsteren, and J. P. M. Postma. A consistent empirical potential for water-protein interactions. *Biopolymers*, 23(8):1513–1518, 1984.

[124] S. Luca, J. F. White, A. K. Sohal, D. V. Filippov, J. H. vanBoom, R. Grisshammer, and M. Baldus. The conformation of neurotensin bound to its G protein-coupled receptor. *Proc. Natl. Acad. Sci. U. S. A.*, 100(19):10706–10711, 2003.

[125] W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.*, 79:926–935, 1983.

[126] X. J. Zhang and B. W. Matthews. Conservation of solvent-binding sites in 10 crystal forms of T4 lysozyme. *Protein Sci.*, 3:1031–, 1994.

[127] T. Gallagher, P Alexander, P. Bryan, and G. L. Gilliland. Two crystal structures of the B1 immunoglobulin-binding domain of streptococcal protein G and comparison with NMR. *Biochemistry*, 33:4721–, 1994.

[128] Michael Unser. Sampling - 50 years after shannon. *Proceedings of the IEEE*, 88(4):569–587, 2000.

[129] B. Hess. Convergence of sampling in protein simulations. *Phys. Rev. E*, 65(3):031910, 2002.

[130] J. D. Faraldo-Gomez, L. R. Forrest, M. Baaden, P. J. Bond, C. Domene, G. Patargias, J. Cuthbertson, and M. S. P. Sansom. Conformational sampling and dynamics of membrane proteins from 10-nanosecond computer simulations. *Proteins*, 57(4):783–791, 2004.

[131] B. L. de Groot, S. Hayward, D. M. F. van Aalten, A. Amadei, and H. J. C. Berendsen. Domain motions in bacteriophage T4 lysozyme: A comparison between molecular dynamics and crystallographic data. *Proteins*, 31(2):116–127, 1998.

[132] T. Pöhlmann, R. A. Böckmann, H. Grubmüller, B. Uchanska-Ziegler, A. Ziegler, and U. Alexiev. Differential peptide dynamics is linked to major histocompatibility complex polymorphism. *J. Biol. Chem.*, 279:28197–28201, 2004.

[133] P. K. Agarwal, S. R. Billeter, P. T. R. Rajagopalan, S. J. Benkovic, and S. Hammes-Schiffer. Network of coupled promoting motions in enzyme catalysis. *Proc. Natl. Acad. Sci. U. S. A.*, 99(5):2794–2799, 2002.

[134] A. Scheer and S. Cotecchia. Constitutively active G protein-coupled receptors: Potential mechanisms of receptor activation. *J. Recept. Signal Transduction Res.*, 17:57–73, 1997.

[135] Richard L. Cross. Our primary source of ATP. *Nature*, 370:594–595, 1994.

[136] J. N. Onuchic, Z. Luthey-Schulten, and P. G. Wolynes. Theory of protein folding: The energy landscape perspective. *Annu. Rev. Phys. Chem.*, 48:545–600, 1997.

[137] J. D. Forman-Kay. The 'dynamics' in the thermodynamics of binding. *Nat. Struct. Biol.*, 6(12):1086–1087, 1999.

[138] A. L. Lee and A. J. Wand. Microscopic origins of entropy, heat capacity and the glass transition in proteins. *Nature*, 411(6836):501–504, 2001.

[139] F. Briki and D. Genest. Canonical-analysis of correlated atomic motions in DNA from molecular-dynamics simulation. *Biophys. Chem.*, 52(1):35–43, 1994.

[140] L. Meinhold and J. C. Smith. Fluctuations and correlations in crystalline protein dynamics: A simulation analysis of staphylococcal nuclease. *Biophys. J.*, 88(4):2554–2563, 2005.

[141] A. M. Fraser and H. L. Swinney. Independent coordinates for strange attractors from mutual information. *Phys. Rev. A*, 33(2):1134–1140, 1986.

[142] H. Matsuda. Physical nature of higher-order mutual information: Intrinsic correlations and frustration. *Phys. Rev. E*, 62(3):3096–3102, 2000.

[143] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. Wiley, New York, 2001.

[144] A. Kraskov, H. Stogbauer, and P. Grassberger. Estimating mutual information. *Phys. Rev. E*, 69(6):066138, 2004.

[145] S. Shwartz, M. Zibulevsky, and Y. Y. Schechner. Fast kernel entropy estimation and optimization. *Signal Process.*, 85(5):1045–1058, 2005.

[146] P. Grassberger. Finite-sample corrections to entropy and dimension estimates. *Phys. Lett. A*, 128(6-7):369–373, 1988.

[147] Y. I. Moon, B. Rajagopalan, and U. Lall. Estimation of mutual information using kernel density estimators. *Phys. Rev. E*, 52(3):2318–2321, 1995.

[148] M. S. Roulston. Estimating the errors on measured entropy and mutual information. *Physica D*, 125(3-4):285–294, 1999.

[149] G. A. Darbellay and I. Vajda. Entropy expressions for multivariate continuous distributions. *IEEE Trans. Inf. Theory*, 46(2):709–712, 2000.

[150] E. G. Learned-Miller and J. W. Fisher. ICA using spacings estimates of entropy. *J. Mach. Learn. Res.*, 4(7-8):1271–1295, 2004.

[151] B. W. Matthews and S. J. Remington. The three-dimensional structure of lysozyme from bacteriophage T4. *Proc. Natl. Acad. Sci. USA*, 71:4178–4182, 1974.

[152] J. A. McCammon, B. Gelin, M. Karplus, and P. G. Wolynes. The hinge bending mode in lysozyme. *Nature*, 262:325–326, 1976.

[153] R. Kuroki, L. H. Weaver, and B. W. Mathhews. A covalent enzyme-substrate intermediate with saccharide distortion in a mutant T4 lysozyme. *Science*, 262:2030, 2033 1993.

[154] H. R. Faber and B. W. Matthews. A mutant T4 lysozyme displays five different crystal conformations. *Nature*, 348:263–266, 1990.

[155] H. S. Mchaourab, K. J. Oh, C. J. Fang, and W. L. Hubell. Conformation of T4 lysozyme in solution. Hinge-bending motion and the substrate-induced conformational transition studied by site-directed spin labeling. *Biochemistry*, 36(2):307–316, 1997.

[156] H. P. Lu. Single-molecule spectroscopy studies of conformational change dynamics in enzymatic reactions. *Curr. Pharm. Biotechnol.*, 5(3):261–269, 2004.

[157] S. Hayward and H. J. C. Berendsen. Systematic analysis of domain motions in proteins from conformational change: New results on citrate synthase and T4 lysozyme. *Proteins*, 30(2):144–154, 1998.

[158] B. L. de Groot, D. M. F. van Aalten, R. M. Scheek, A. Amadei, G. Vriend, and H. J. C. Berendsen. Prediction of protein conformational freedom from distance constraints. *Proteins*, 29:240–251, 1997.

[159] W. Qian, J. Bandekar, and S. Krimm. Vibrational analysis of peptides, polypeptides, and proteins. 41. vibrational analysis of crystalline tri-L-alanine. *Biopolymers*, 31(2):193–210, 1991.

[160] H. Frauenfelder and P. G. Wolynes. Rate theories and puzzles of hemeprotein kinetics. *Science*, 229(4711):337–345, 1985.

[161] J. Schlitter. Estimation of absolute and relative entropies of macromolecules using the covariance matrix. *Chem. Phys. Lett.*, 215(6):617–621, 1993.

[162] J. F. Cardoso. Blind signal separation: Statistical principles. *Proc. IEEE*, 86(10):2009–2025, 1998.

[163] P. Comon. Independent component analysis, a new concept. *Signal Process.*, 36(3):287–314, 1994.

[164] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. Neural Netw.*, 10(3):626–634, 1999.

[165] H. Stogbauer, A. Kraskov, S. A. Astakhov, and P. Grassberger. Least-dependent-component analysis based on mutual information. *Phys. Rev. E*, 70(6):066123, 2004.

[166] L. B. Almeida. MISEP - linear and nonlinear ICA based on mutual information. *J. Mach. Learn. Res.*, 4(7-8):1297–1318, 2004.

[167] G. E. Forsythe, M. A. Malcolm, and C. B. moler. *Computer Methods for Mathematical Computations*. Prentice-Hall, New York, 1976.

[168] M. P. Wand. Data-based choice of histogram bin width. *Am. Stat.*, 51(1):59–64, 1997.

[169] A. Hyvärinen and E. Oja. Independent component analysis: Algorithms and applications. *Neural Netw.*, 13(4-5):411–430, 2000.

[170] B. Hess. Similarities between principal components of protein dynamics and random diffusion. *Phys. Rev. E*, 62(6):8438–8448, 2000.

[171] J. B. Clarage, T. Romo, B. K. Andrews, B. M. Pettitt, and G. N. Phillips. A sampling problem in molecular-dynamics simulations of macromolecules. *Proc. Natl. Acad. Sci. U. S. A.*, 92(8):3288–3292, 1995.

[172] J. B. Tenenbaum, V. deSilva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[173] D. S. Moss and G. W. Harris. Diffuse-x-ray scattering from macromolecular crystals using synchrotron-radiation. *Radiat. Phys. Chem.*, 45(3):523–535, 1995.

[174] J. P. Benoit and J. Doucet. Diffuse-scattering in protein crystallography. *Q. Rev. Biophys.*, 28(2):131–169, 1995.

[175] J. Perez, P. Faure, and J. P. Benoit. Molecular rigid-body displacements in a tetragonal lysozyme crystal confirmed by x-ray diffuse scattering. *Acta Crystallogr. Sect. D-Biol. Crystallogr.*, 52:722–729, 1996.

[176] M. E. Wall, J. B. Clarage, and G. N. Phillips. Motions of calmodulin characterized using both Bragg and diffuse X-ray scattering. *Structure*, 5(12):1599–1612, 1997.

[177] R. Ishima and D. A. Torchia. Protein dynamics from NMR. *Nat. Struct. Biol.*, 7(9):740–743, 2000.

[178] A. G. Palmer. NMR probes of molecular dynamics: Overview and comparison with other techniques. *Annu. Rev. Biophys. Biomolec. Struct.*, 30:129–155, 2001.

[179] V. A. Daragan and K. H. Mayo. Motional model analyses of protein and peptide dynamics using C-13 and N-15 NMR relaxation. *Prog. Nucl. Magn. Reson. Spectrosc.*, 31:63–105, 1997.

[180] G. Lipari and A. Szabo. Model-free approach to the interpretation of nuclear magnetic-resonance relaxation in macromolecules. 1. theory and range of validity. *J. Am. Chem. Soc.*, 104(17):4546–4559, 1982.

[181] L. E. Kay, D. A. Torchia, and A. Bax. Backbone dynamics of proteins as studied by N-15 inverse detected heteronuclear NMR-spectroscopy - application to staphylococcal nuclease. *Biochemistry*, 28(23):8972–8979, 1989.

[182] A. G. Palmer, M. Rance, and P. E. Wright. Intramolecular motions of a zinc finger DNA-binding domain from xfin characterized by proton-detected natural abundance C-12 heteronuclear NMR-spectroscopy. *J. Am. Chem. Soc.*, 113(12):4371–4380, 1991.

[183] A. T. Brünger, P. D. Adams, G. M. Clore, W. L. DeLano, P. Gros, R. W. Grosse-Kunstleve, J. -S Jiang, J. Kuszewski, N. Nilges, N. S. Pannu, R. J. Read, L. M. Rice, T. Simonson, and G. L. Warren. Crystallography and NMR system (CNS): A new software system for macromolecular structure determination. *Acta. Cryst. D*, 54:905–921, 1998.

[184] J. Kuszewski, A. M. Gronenborn, and G. M. Clore. Improving the packing and accuracy of nmr structures with a pseudopotential for the radius of gyration. *J. Am. Chem. Soc.*, 121:2337–2338, 1999.

[185] S. Pfeiffer, D. Fushman, and D. Cowburn. Simulated and NMR-derived backbone dynamics of a protein with significant flexibility: A comparison of spectral densities for the beta ARK PH domain. *J. Am. Chem. Soc.*, 123(13):3021–3036, 2001.

[186] A. G. Palmer. Nmr characterization of the dynamics of biomacromolecules. *Chem. Rev.*, 104(8):3623–3640, 2004.

[187] A. M. Gronenborn, D. R. Filipula, N. Z. Essig, A. Achari, M. Whitlow, P. T. Wingfield, and G. M. Clore. A novel, highly stable fold of the immunoglobulin binding domain of streptococcal protein G. *Science*, 253:657–661, 1991.

[188] D. Idiyatullin, V. A. Daragan, and K. H. Mayo. (NH)-N-15 backbone dynamics of protein GB1: Comparison of order parameters and correlation times derived using various "model-free" approaches. *J. Phys. Chem. B*, 107(11):2602–2609, 2003.

[189] M. Philippopoulos and C. Lim. Molecular-dynamics simulation of escherichia-coli ribonuclease H-1 in solution - correlation with NMR and X-ray data and insights into biological function. *J. Mol. Biol.*, 254(4):771–792, 1995.

[190] D. C. Chatfield, A. Szabo, and B. R. Brooks. Molecular dynamics of staphylococcal nuclease: Comparison of simulation with N-15 and C-13 NMR relaxation data. *J. Am. Chem. Soc.*, 120(21):5301–5311, 1998.

[191] M. Philippopoulos, A. M. Mandel, A. G. Palmer, and C. Lim. Accuracy and precision of NMR relaxation experiments and MD simulations for characterizing protein dynamics. *Proteins*, 28(4):481–493, 1997.

[192] D. A. Case. Molecular dynamics and NMR spin relaxation in proteins. *Accounts Chem. Res.*, 35(6):325–331, 2002.

[193] G. M. Clore and C. D. Schwieters. Amplitudes of protein backbone dynamics and correlated motions in a small alpha/beta protein: Correspondence of dipolar coupling and heteronuclear relaxation measurements. *Biochemistry*, 43(33):10678–10691, 2004.

[194] Bruce J. Berne. *Dynamic Light Scattering*. Wiley, 1976.

[195] R. Zwanzig. Problems in nonlinear transport theory. In L. Garrido, editor, *Systems far from equilibrium*, pages 198–225. Springer, New York, 1980.

[196] R. Kubo. Fluctuation-dissipation theorem. *Rep. Prog. Phys.*, 29:255–, 1966.

[197] A. J. Chorin, O. H. Hald, and R. Kupferman. Optimal prediction and the mori-zwanzig representation of irreversible processes. *Proc. Natl. Acad. Sci. U. S. A.*, 97(7):2968–2973, 2000.

[198] G. D. Harp and B. J. Berne. Time-correlation functions, memory functions, and molecular dynamics. *Phys. Rev. A*, 2(3):975, 1970.

[199] A. Chorin, A. Kast, and R. Kupferman. Optimal prediction of underresolved dynamcis. *Proc. Natl. Acad. Sci. USA*, 95(8):4094–4098, 1998.

[200] M. Berkowitz, J. D. Morgan, and J. A. McCammon. Generalized Langevin dynamics simulations with arbitrary time-dependent memory kernels. *J. Chem. Phys.*, 78(6), 1983.

[201] T. Srokowski. Stochastic processes with finite correlation time: Modeling and application to the generalized Langevin equation. *Phys. Rev. E*, 64(3):031102, 2001.

[202] M. Souaille and B. Roux. Extension to the weighted histogram analysis method: combining umbrella sampling with free energy calculations. *Comput. Phys. Commun.*, 135(1):40–57, 2001.

[203] M. Iannuzzi, A. Laio, and M. Parrinello. Efficient exploration of reactive potential energy surfaces using Car-Parrinello molecular dynamics. *Phys. Rev. Lett.*, 90(23):238302, 2003.

[204] D. E. Smith and C. B. Harris. Generalized brownian dynamics. 1. numerical-integration of the generalized Langevin equation through autoregressive modeling of the memory function. *J. Chem. Phys.*, 92(2):1304–1311, 1990.

[205] T. Shimizu. Relaxation and bifurcation in brownian-motion driven by a chaotic force. *Physica A*, 164(1):123–146, 1990.

[206] Mark E. Tuckerman and Bruce J. Berne. Stochastic molecular dynamics in systems with multiple time scales and memory friction. *J. Chem. Phys.*, 95(6):4389–4396, 1991.

[207] M. P. Allen and D. J. Tildesley. *Computer simulation of liquids*. Clarendon Press, 1987.

[208] J. W. Cooley and J. W. Tukey. An algorithm for machine calculation of complex fourier series. *Math. Comput.*, 19(90), 1965.

[209] H. J. Nussbaumer. *Fast Fourier Transform and Convolution Algorithms*. Springer, New York, 1982.

[210] Matteo Frigo and Steven G. Johnson. The design and implementation of FFTW3. *Proceedings of the IEEE*, 93(2):216–231, 2005. Special issue on "Program Generation, Optimization, and Platform Adaptation".

[211] Todd L. Veldhuizen. Arrays in Blitz++. In *Proceedings of the 2nd International Scientific Computing in Object-Oriented Parallel Environments (ISCOPE'98)*, Lecture Notes in Computer Science. Springer-Verlag, 1998.

[212] E. Hairer, C. Lubich, and M. Schlichte. Fast numerical-solution of nonlinear Volterra convolution equations. *SJSSC*, 6(3):532–541, 1985.

[213] C. T. H. Baker and M. S. Derakhshan. FFT techniques in the numerical-solution of convolution equations. *J. Comput. Appl. Math.*, 20:5–24, 1987.

[214] C. Hoheisel. Memory functions and the calculation of dynamical properties of atomic liquids. *Comp. Phys. Rep.*, 12:29–66, 1990.

[215] Heinz W. Engl, Martin Hanke, and Anreas Neubauer. *Regularization of Inverse Problems*. Kluwer Academic Publishers, 2000.

[216] A. N. Tikhonov. Solution of incorrectly formulated problems and regularization method. *Dokl. Akad. Nauk SSSR*, 151(3):501–, 1963.

[217] P. C. Hansen. Deconvolution and regularization with toeplitz matrices. *Numer. Algorithms*, 29(4):323–378, 2002.

[218] P. C. Hansen. Regularization tools, a MATLAB package for analysis of discrete regularization problems. *Numer. Algorithms*, 6:1–35, 1984.

[219] P. K. Lamm and L. Elden. Numerical solution of first-kind volterra equations by sequential tikhonov regularization. *SIAM J. Numer. Anal.*, 34(4):1432–1450, 1997.

[220] P. K. Lamm and T. L. Scofield. Sequential predictor-corrector methods for the variable regularization of volterra inverse problems. *Inverse Probl.*, 16(2):373–399, 2000.

[221] P. K. Lamm. Variable-smoothing local regularization methods for first-kind integral equations. *Inverse Probl.*, 19(1):195–216, 2003.

[222] U. Schmitt and A. K. Louis. Efficient algorithms for the regularization of dynamic inverse problems: I. theory. *Inverse Probl.*, 18(3):645–658, 2002.

[223] J. P. Boon and S. Yip. *Molecular Hydrodynamics*. McGraw-Hill, New York, 1980.

[224] F. Shimojo, K. Hoshino, and M. Watabe. Dynamical correlation-functions and memory functions of liquid-sodium - a molecular-dynamics simulation. *J. Phys. Soc. Jpn.*, 63(1):141–155, 1994.

[225] C. T. H. Baker. A perspective on the numerical treatment of volterra equations. *J. Comput. Appl. Math.*, 125(1-2):217–249, 2000.

[226] Patricia K. Lamm. *Surveys on Solution Methods for Inverse Problems*, pages 53–82. Springer, New York, 2000.

[227] J. S. Chaturve and Baijal. Memory effects and dynamical correlations in liquid argon. *J. Phys. Soc. Jpn.*, 33(2):389–, 1972.

[228] S. K. Mitra, N. Varshney.nc, and N. Dass. Memory function of velocity autocorrelation of classical liquids. *Phys. Rev. A*, 6(3):1214–, 1972.

[229] J. T. Day. A starting method for solving nonlinear volterra integral equations. *Bit*, 7:179–188, 1967.

[230] G. R. Kneller and K. Hinsen. Fractional brownian dynamics in proteins. *J. Chem. Phys.*, 121(20):10278–10283, 2004.

[231] M. Hanke. Accelerated Landweber iterations for the solution of ill-posed equations. *Numer. Math.*, 60:341–373, 1991.

[232] Eberhard Schock. Semi-iterative methods for the approximate solution of ill-posed problems. *Numer. Math.*, 50:263–271, 1987.

[233] D. L. Phillips. A technique for numerical solution of certain integral equations of first kind. *J. ACM*, 9(1):84–, 1962.

[234] L. Élden. An efficient algorithm for the regularization of ill-conditioned least-squares problems with triangular toeplitz matrix. *SIAM J. Sci. Statist. Comput.*, 5(1):229–236, 1984.

[235] P. C. Hansen. Analysis of discrete ill-posed problems by means of the L-curve. *SIAM Rev.*, 34(4):561–580, 1992.

[236] P. C. Hansen and D. P. O'Leary. The use of the L-curve in the regularization of discrete ill-posed problems. *SIAM J. Sci. Comput.*, 14(6):1487–1503, 1993.

[237] T. Reginska. A regularization parameter in discrete ill-posed problems. *SIAM J. Sci. Comput.*, 17(3):740–749, 1996.

[238] T. Rog, K. Murzyn, K. Hinsen, and G. R. Kneller. nMoldyn: A program package for a neutron scattering oriented analysis of molecular dynamics simulations. *J. Comput. Chem.*, 24(5):657–667, 2003.

[239] A. Briese. Effektive Modelle der Proteindynamik. Master's thesis, Ludwig-Maximilians-Universität München, 1996.

[240] P. Hänggi, P. Talkner, and M. Borkovec. Reaction-rate theory - 50 years after kramers. *Rev. Mod. Phys.*, 62(2):251–341, 1990.

[241] C. Schütte, 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, 1999.

[242] D. C. Sullivan and C. M. Lim. Configurational entropy of proteins: Covariance matrix versus cumulative distribution calculations. *J. Chin. Chem. Soc.*, 51(5B):1209–1219, 2004.