12-2012

Towards a Unification of Supercomputing, Molecular Dynamics Simulation and Experimental Neutron and X-ray Scattering Techniques

Benjamin Lindner
blindne1@utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Benjamin Lindner entitled "Towards a Unification of Supercomputing, Molecular Dynamics Simulation and Experimental Neutron and X-ray Scattering Techniques." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Life Sciences.

Jeremy C. Smith, Major Professor

We have read this dissertation and recommend its acceptance:

Jerome Baudry, Hong Guo, Xiaolin Cheng, Tongye Shen

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Towards a Unification of Supercomputing, Molecular Dynamics Simulation and Experimental Neutron and X-ray Scattering Techniques

A Dissertation
Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

Benjamin Lindner
December 2012

To all which makes life worth living for

Acknowledgements

My gratitude stretches out to all people who made my work at UTK/ORNL enjoyable and possible. In particular, I thank:

Jeremy C. Smith for putting enough trust into me to handle multi-million dollar computational equipment and investments, paying for all the bills I had to cover in the meantime and for being so forthcoming with support when needed.

Roland Schulz for closely sharing the Ph.D. experience from the start and our continued collaboration and friendship.

Loukas Petridis for being a friend and colleague and keeping me on my toes.

Angelina Pfeuffer for being such a great daughter and making me already feel obsolete.

Andrea Pfeuffer for stepping back for me.

For all his bluster, it is the sad province of Man that he cannot choose his triumph. He can only choose how he will stand when the call of destiny comes... hoping that he'll have the courage to answer.
Tim Kring, *Don't Look Back, Heroes*

Abstract

Molecular dynamics simulation has become an essential tool for scientific discovery and investigation. The ability to evaluate every atomic coordinate for each time instant sets it apart from other methodologies, which can only access experimental observables as an outcome of the atomic coordinates. Here, the utility of molecular dynamics is illustrated by investigating the structure and dynamics of fundamental models of cellulose fibers. For that, a highly parallel code has been developed to compute static and dynamical scattering functions efficiently on modern supercomputing architectures. Using state of the art supercomputing facilities, molecular dynamics code and parallelization strategies, this work also provides insight into the relationship between cellulose crystallinity and cellulose-lignin aggregation by performing multi-million atom simulations. Finally, this work introduces concepts to augment the ability of molecular dynamics to interpret experimental observables with the help of Markov modeling, which allows for a convenient description of complex molecule dynamics as transitions between well defined conformations. The work presented here suggests that molecular dynamics will continue to evolve and integrate with experimental techniques, like neutron and X-ray scattering, and stochastic models, like Markov modeling, to yield unmatched descriptions of molecule dynamics and interpretations of experimental data, facilitated by the growing computational power available to scientists.

Contents

List of Tables

List of Figures

1. Introduction

The fundamental premise of structural biology is that biological function arises from the interactions and properties of molecules. The power to manipulate biological function in a controlled manner is thus closely related to the capacity to understand and study molecules and to exert change on the molecular level. With recent advances in drug design [1, 2] and genetic engineering [3, 4], structural biology is one of the most successful domains of applied nanotechnology to date [5, 6, 7, 8].

The advancement of structural biology was and still is greatly facilitated by progresses in experimental techniques which allow the characterization of molecules on many different structural levels. Mass spectrometry is frequently used to discover molecular chemical composition and structural motives [9], X-ray and neutron crystallography are used to determine molecular shape and configuration [10], NMR is used to detect the specific chemical environment within molecules [11], microscopic methods are used to localize molecules within a given environment or detect their association with each other [12], and many more techniques exist. It is fair to say that it is the combination of techniques which gives the experimental toolkit such a versatility.

A subgroup of experimental techniques not only allows the determination of molecular structure but also dynamics. The time resolution of the technique is closely related to the type of physical process it is exploiting. E.g. NMR uses spin polarization to study diffusion processes and changes in chemical environments [13] and probes the millisecond time scale, while the scattering of neutrons and X-rays is sensitive to time-dependent changes in the positions of atoms [14] and probes the nanosecond to picosecond and picosecond to femtosecond time scale.

The interpretation of experimental data and the study of molecular dynamics was traditionally (prior to the establishment of molecular simulation techniques) constrained to simple, yet powerful, analytical theories. A prominent example is the inverse scattering problem in small angle neutron scattering (SANS), where the one dimensional scattering vector length dependent scattering intensity, *S(q)*, can be matched against a library of intensity profiles computed from idealized geometric shapes [15]. While this approach works fine whenever the particular molecule is static and resembles an idealized shape, it falls short for most realistic molecules, because they may adopt a variety of shapes with different probabilities. Molecular dynamics (MD) simulation assisted SANS solves this problem by computing the molecular shapes and their probability distribution a priori. However, it requires knowledge of the structural composition of the molecule which may not be accessible. Another example where the experimental data suffers from underdetermination is in dynamic neutron scattering in which a relatively simple signal may result from the superposition of many relaxation processes [16]. Here, MD simulation can be used as a deductive tool, matching experimental spectral fingerprints to the corresponding fingerprint calculated for various simulation conditions.

The establishment of Petascale supercomputing facilities [17, 18] and the shift towards massively parallel computing platforms is rapidly changing the landscape of scientific data analysis. On the one hand the increase in computational power generates more raw data for the interpretation of scientific experiments, on the other hand the data analysis can be performed at a much higher fidelity. Especially molecular dynamics simulation assisted interpretation of neutron and X-ray scattering experiments is an example of a highly CPU-intensive analysis, which considerably benefits from massive sampling [19, 20].

2. Theory

The convergence between supercomputing, molecular dynamics simulation and experimental scattering techniques brings together a diverse set of aspects, including computational performance, approximation procedures, and data analysis strategies and interpretation. It is the intent of the following sections to cover some of those aspects and put them into perspective.

2.1. Molecular Dynamics Simulation

The objective of molecular dynamics simulation is to compute trajectories in time and space by solving Newton's equation of motion for each atom within the simulated system. From that, all thermodynamic and statistical properties of the comprised molecules can be derived. Another approach to derive these properties usually is a variation of a Monte Carlo based simulation, where the configurational space of the molecular system is explored by evaluating the value of the potential energy and creating a set of configurations with a maximum of likelihood. Monte Carlo is computationally less demanding and offers better task parallelism properties than molecular dynamics. However, Monte Carlo is a probabilistic scheme and cannot be used whenever the type of molecular motion and the transitional dynamics between configurations is of interest.

The need for massive sampling has motivated a variety of enhancements to plain molecular dynamics. Path-sampling methods usually employ artificial potentials to overcome kinetic barriers by increasing the likelihood of the transition state [21], Replica-exchange methods take advantage of a general increase in kinetic barrier crossing at higher temperatures and perform configurational swapping between parallel simulations at different temperatures based on energetic overlap [22], and multiscale coupled dynamics slaves the slow transitional dynamics of an atomic-detailed simulation to the smoother and kinetically enhanced dynamics of a coarse-grained simulation [23]. The list of enhancement protocols is long and continually evolving. However, each enhancement is build around the basic properties of the plain molecular dynamics methodology, which is to solve the equations of motions based on a set of well described forces.

Even though this work makes exclusive use of an all-atom description of the molecules, the methodology itself is independent of the granularity of the involved particles. Hence coarse graining, i.e. the combination of certain atoms to a single bead with its own properties, is commonly used to reduce the computational demand for larger scale systems by reducing the degree of freedoms and removing fast motions, e.g. the hydrogen bond vibrations. However, coarse grained descriptions are very limited to the specific chemical and molecular environment they were created for, which is why they frequently rely to a degree on all-atomic simulations for validation.

2.1.1. Force Field

The methodology of molecular dynamics (MD) simulation is based on the premise that the motions of molecules are governed by a set of clearly distinguishable forces. On a basic quantum mechanical level every force is based on the electrostatic and spin coupled interactions between electrons and nuclei, and thus quantum mechanical calculations are frequently used to derive parameters for the set of forces used in MD simulation [24]. The most common form of a molecular force field is given by the total energy potential, $V(\vec{r}^N)$:

$$\begin{aligned} V\left(\vec{r}^N\right) &= \sum_{bonds} \frac{k_i}{2}\left(l_i - l_{i,0}\right)^2 + \sum_{angles} \frac{k_i}{2}\left(\theta_i - \theta_{i,0}\right)^2 + \sum_{torsions} \frac{V_n}{2}(1 + cos(n\omega - \gamma)) \\ &+ \sum_{i=1}^{N} \sum_{j=i+1}^{N} \left(4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right] + \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}}\right) \end{aligned} \tag{2.1}$$

where $\sum_{bonds}$, $\sum_{angles}$, and $\sum_{torsions}$ are the strong bonded interactions which require chemical cleavage to be resolved and the last term comprises the non-bonded interactions which describe van der Waals short ranged repulsion and attraction with a Lennard-Jones potential and long-ranged coulombic interactions. The simple form of this molecular force field does not allow the simulation of chemical cleavage or dynamic electronic polarization, which can be important under some special circumstances [25]. Both limitations can be alleviated by using special purpose MD force fields [26, 27] and hybrid QM/MM simulation software [28]. However, the common form of the MD force field is a satisfactory approximation for simulation studies which are interested in the configurational sampling of molecules and their interactions. Additional variety exists for the set of force field parameters available, because the derivation of the force field coefficients is neither unique nor constrained to a particular methodology [29, 30]. The main difference between fixed-charged force fields, like CHARMM and AMBER, tends to be their convergence point for the static polarization terms in the modeled molecules [31].

Not considered part of the actual force field, but equally important is the treatment of non-bonded interactions during the calculation, because they need to be truncated for computational efficiency [32]. While the truncation of the Lennard-Jones term usually just offsets the effective pressure of the simulation system, artifacts in the electrostatic field calculation can severely alter the resulting molecular dynamics [33]. Two computationally efficient electrostatic treatments are Particle Mesh Ewald (PME) [34, 32] and Reaction Field (RF) [35, 36, 37, 38]. PME splits the computation of the electrostatic field into two terms, one for short-ranged electrostatic with a cut-off, and a long-ranged term which is solved using the Fourier transform of the global charge distribution. RF instead avoids the discontinuity of the electrostatic force at the cut-off distance by assuming each charge to be surrounded by a dielectric material with a significantly higher charge conductivity, leading to a strong dampening of the electrostatic force at the cut-off distance. RF was originally derived for neutral molecules, but has been shown to give promising results for locally charged systems as well [39].

2.1.2. Equations of Motion

The force field is used to compute the acceleration, $\vec{a}(t)$, for each atom from its coordinates, $\vec{x}(t)$, and the positions of all other atoms. The molecular dynamics software then solves the differential equation for the equation of motion by integrating with a sufficiently small time step, δt. A commonly used algorithm to update coordinates, $\vec{x}(t)$, and velocities, $\vec{v}(t)$, is known as Velocity Verlet [40]:

$$\vec{x}(t + \delta t) = \vec{x}(t) + \vec{v}(t) \cdot \delta t + \frac{1}{2}\vec{a}(t) \cdot \delta t^2 \tag{2.2}$$

$$\vec{v}(t + \delta t) = \vec{v}(t) + \frac{\vec{a}(t) + \vec{a}(t + \delta t)}{2} \cdot \delta t \tag{2.3}$$

For fully atomistic systems with no motional constrains the typical time step needs to be $\delta t \lesssim 1fs$ to achieve numerical stability and avoid errors in the calculation of the forces. Integration errors lead to a loss of energy conservation and affects the apparent temperature of each atom, due to inaccurate velocities.

Because the stability and the numerical accuracy of the algorithm is mainly related to the maximum velocities of the particles, the motions of light atoms, like hydrogens, are usually the limiting factor for the time step. A larger time step of $\delta t = 2fs$ can be achieved by removing the hydrogen bond vibration by constraining their position to the equilibrium distance to their bonded heavy atoms. Even larger time steps are possible by additionally removing their angle vibrations. Different numerical procedures have been developed to implement these constrains. The first, which is still in use today, is SETTLE [41], which only works for molecules with 3 atoms and is thus mainly used for water molecules. Two other constrain algorithms commonly used are SHAKE [42] and LINCS [43] and are applicable to larger molecules.

A relatively newly developed scheme to retain stability and accuracy for larger time steps, are virtual sites [44]. They are superior to traditional constrain algorithms in cases which involve angle constains, e.g. the methyl group rotation dynamics, where SHAKE and LINCS have been shown to severely alter the macromolecular flexiblity and reduce the number of torsional and angle transitions [45].

2.2. Scattering Techniques

The theory of X-ray and neutron scattering is well known [14, 15, 46]. In the classical description of scattering theory the atomic particles are sources of plane waves, which may interfere in different ways dependent on the type of scattering experiment. The scattering patterns arise from the constructive interference of atomic scattering amplitudes $a(\vec{q},t)$:

$$a_n(\vec{q},t) = b_n(\vec{q}) \cdot e^{i \cdot \vec{q} \cdot \vec{r}_n(t)} \tag{2.4}$$

where $\vec{q}$ is the scattering vector, $b_n(\vec{q})$ is the atomic prefactor and $\vec{r_n}(t)$ is the time dependent Cartesian position vector of atom n. The total scattering amplitude for all atoms, $A(\vec{q},t)$, and the associated scattering intensity, $F(\vec{q},t)$, is given by:

$$F(\vec{q},t) = A(\vec{q},t) \cdot A^*(\vec{q},t) = |A(\vec{q},t)|^2 \tag{2.5}$$

$$\text{where } \; A(\vec{q},t) = \sum_n a_n(\vec{q},t) \tag{2.6}$$

and $A^*(\vec{q},t)$ is the complex conjugate of $A(\vec{q},t)$. $A(\vec{q},t)$ may be associated with a higher organizational structure in which the individual atoms are incorporated.

The atomic prefactor b_n is different for X-ray and neutron scattering. In X-ray scattering it represents the form factor of the electronic shell of the respective atom and is approximated as a series of Gaussians [47]:

$$b_n(\vec{q}) = \sum_i c_i \cdot e^{-d_i \cdot |\vec{q}|^2} \tag{2.7}$$

where the set of c_i and d_i are empirically derived constants. In neutron scattering, b_n is the atomic scattering length, which is different for each isotope and independent of $|\vec{q}|$. The variation of the atomic scattering length due to isotopic distribution and random nucleic spin orientations gives rise to coherent and incoherent scattering, which is described by two distinct scattering lengths for each isotope type: b^{coh} and b^{inc} [48]. The resulting total scattering function for neutron scattering is therefore split into a coherent part F_{coh} and an incoherent part, F_{inc} :

$$F(\vec{q},t) = F_{coh}(\vec{q},t) + F_{inc}(\vec{q},t) \tag{2.8}$$

$$F_{coh}(\vec{q},t) = |\sum_n a_n^{coh}(\vec{q},t)|^2 \tag{2.9}$$

$$F_{inc}(\vec{q},t) = \sum_n |a_n^{inc}(\vec{q},t)|^2 = \sum_n (b_n^{inc})^2 \tag{2.10}$$

The coherent scattering function is based on the total scattering amplitude $A(\vec{q},t)$ and uses b^{coh}, while the incoherent function is associated with atomic scattering amplitudes $a_n(\vec{q},t)$ and uses b^{inc}. In dynamic scattering experiments the scattering amplitudes at times t and $t+\tau$ are superimposed, resulting in a correlated time signal of the form:

$$F(\vec{q},t,\tau) = A(\vec{q},t) \cdot A^*(\vec{q},t+\tau) \tag{2.11}$$

or

$$f_n(\vec{q},t,\tau) = a_n(\vec{q},t) \cdot a_n^*(\vec{q},t+\tau) \tag{2.12}$$

where $F(\vec{q},t,\tau)$ and $f_n(\vec{q},t,\tau)$ are the total and atom-correlated time signals, respectively.

2.2.1. Solution Scattering

In solution scattering a large number of identical solute particles are in a solvent permitting solute rotational and translational motion. Thus, there is no preferred orientation and the scattering signal becomes isotropic, *i.e.,* independent of the sample orientation: $F(\vec{q},t) \rightarrow F(q,t)$. Also, the system can be in any of its thermodynamically allowed states, resulting the scattering representing the ensemble average of the individual particle dynamics, which is taken as the MD time average: $F(q,t) \rightarrow \langle F(q,t)\rangle_t$. The scattering expression associated with solution scattering conditions is therefore:

$$F(q) = N_P \left\langle \langle F(\vec{q},t)\rangle_t \right\rangle_\Omega \tag{2.13}$$

where N_P is the number of scattering particles and $\langle\rangle_\Omega$ performs the orientational averaging for all orientations of the scattering vector $\vec{q}$. For solution scattering, the solute atomic scattering factors can be adjusted to represent the proper scattering length contrast, as is outlined in Ref. [49].

2.2.2. Crystal Diffraction

In a crystal diffraction experiment the scattering particles are orientationally aligned. The resulting scattering pattern is a convolution of the scattering due to the lattice and the scattering unit. The scattering expression associated with diffraction is:

$$F(\vec{q}) = N_P \langle F(\vec{q},t)\rangle_t \tag{2.14}$$

where N_P is the number of scattering particles. The scattering function due to the crystal lattice is neglected for simplicity, but can be modeled explicitly in MD simulations.

2.2.3. Dynamic Scattering

Dynamic scattering experiments allow the measurement of either the intermediate scattering function $I(\vec{q},\tau)$ or the dynamic structure factor $S(\vec{q},\omega)$. Since $S(\vec{q},\omega)$ is simply the Fourier transform of $I(\vec{q},\tau)$, only the intermediate scattering function $I(\vec{q},\tau)$ is discussed in the following. In dynamic X-ray and coherent neutron scattering the scattering intensities arise from a superposition of the total scattering amplitude $A(\vec{q},t)$ at two times t and $t+\tau$, given by:

$$I_{coh}(q,\tau) = \langle\langle F(\vec{q},t,\tau)\rangle_t\rangle_\Omega \tag{2.15}$$

which probes structural-temporal correlations. Dynamic incoherent neutron scattering measures the superposition of the individual atomic scattering amplitudes $a_n(\vec{q},t)$ at t and $t+\tau$, and is given by

$$I_{inc}(q,\tau) = \sum_n \langle\langle f_n(\vec{q},t,\tau)\rangle_t\rangle_\Omega \tag{2.16}$$

which probes correlations in the displacements of individual atoms. For large correlation times $\tau\to\infty$, the coherent and incoherent dynamic scattering function converge, allowing the function to be split into time-dependent and -independent parts:

$$I_{coh,inc}(q,\tau) = I^*_{coh,inc}(q,\tau) + I_{coh,inc}(q,\infty) \tag{2.17}$$

where $I_{coh,inc}(q,\infty)$ are the elastic coherent and incoherent structure factors (EISF) and can be approximated without the need to compute the autocorrelation:

$$I_{coh}(q,\infty) = \left\langle |\langle A(\vec{q},t)\rangle_t|^2 \right\rangle_\Omega \tag{2.18}$$

$$I_{inc}(q,\infty) = EISF(q) = \sum_n \left\langle |\langle a_n(\vec{q},t)\rangle_t|^2 \right\rangle_\Omega \tag{2.19}$$

2.3. Markov State Modeling

Molecular dynamics simulations produce trajectories, describing the Cartesian coordinates of atoms in time and space. For molecular systems, the motion can be decomposed into translational, rotational, and internal dynamics, where the latter can be conveniently described in the framework of MSM [50]. MSM describes the internal molecular configuration space as a set of conformational substates $s = \{1,\dots,m\}$. The transitional dynamics between these states then follows the characteristic $m\times m$ matrix $T(\tau)$, which contains the conditional probabilities, T_{ij}, of finding the system in state j at time $t+\tau$ given that it was in state i at time t:

$$T_{ij} = P(s_{t+\tau} = j\,|\,s_t = i) \tag{2.20}$$

The transition matrix, $T(\tau)$, is based on the chosen lag time τ (the time step for building the MSM). The time evolution of the system is governed by the equation

$$p(t+\tau) = p(t)\cdot T(\tau) \tag{2.21}$$

where $p(t)$ is a m-dimensional row vector containing the probability to find the system in each of its m states at time t. If the time evolution of the probabilities follows the Markov property, then for longer time increments, $n \cdot \tau$, $T(n \cdot \tau) = T(\tau)^n$ holds and equation 2.21 becomes:

$$p(t + n \cdot \tau) = p(t) \cdot T(n \cdot \tau) = p(t)\,[T(\tau)]^n \tag{2.22}$$

The approximation of the underlying dynamics with a Markov model requires appropriate choices for the state space variables and the lag time and needs to be systematically validated [50]. Using the assumption of detailed balance, which holds true for a system under equilibrium conditions, the eigenvalue decomposition (EVD) of the transition matrix yields:

$$T(\tau) = \Phi \cdot \Lambda(\tau) \cdot \Phi^{-1} = R \cdot \Lambda(\tau) \cdot R^{-1} = R \cdot \Lambda(\tau) \cdot L \tag{2.23}$$

where $\Phi = (\phi_1, \ldots, \phi_m)$ and $\Lambda = diag(\lambda_1, \ldots, \lambda_m)$ are an arbitrary eigenvector decomposition and the eigenvalue matrix, respectively, and $L = (l_1, \ldots, l_m)$ and $R = (r_1, \ldots, r_m)$ are the left and right eigenvector matrices, which are normalized against the equilibrium distribution $\Pi = diag\,(\pi_1, \ldots, \pi_m)$:

$$r_k = \frac{1}{\sqrt{\phi_k^T \cdot \Pi \cdot \phi_k}} \cdot \phi_k \tag{2.24}$$

The relationship $L = R^{-1}$ can be used to interconvert left and right eigenvectors. The eigenvectors l_k provide information about the structural change, while the corresponding eigenvalues λ_k only describe the relaxation time. Transitions l_k with $\lambda_k \ll 1$ correspond to fast processes, while transitions with $\lambda_k \approx 1$ describe slow ones. Since $T(\tau)$ is a stochastic matrix, the first process, l_1, has, by definition, an eigenvalue of $\lambda_1 = 1$, and thus corresponds to the equilibrium distribution of the system $\Pi = (\pi_1, \ldots, \pi_m)$. The characteristic relaxation time, t_k, of each process, l_k, can be calculated from the eigenvalues, λ_k:

$$t_k = -\frac{\tau}{\ln \lambda_k(\tau)} \tag{2.25}$$

Ideally, the deduced relaxation time, t_k, is independent on the choice of the lagtime τ. However, due to different sources of errors, this is usually not the case and finding a suitable τ requires an implied time scale analysis [51].

2.3.1. Correlation Functions

Any time correlation function between an observable a and b can be expressed as transitions between states i and j, given their correlation matrix, $C\,(\tau)$ [16]:

$$\langle a(t)b(t+\tau)\rangle_t = \sum_{ij} C_{ij} a_i b_j \tag{2.26}$$

Each time instant, t, is associated with one of a finite set of states, s, that the system can be in. In this case a_i and b_j become the expectation values of the observables for states i and j, and the coefficients C_{ij} represent the absolute probabilities of a transition from state i to state j:

$$C_{ij} = P\,(s(t+\tau) = j, s(t) = i) \tag{2.27}$$

Using a_i and b_j as expectation values introduces the assumption, that the averaging is mainly performed over processes which exist on a time scale significantly shorter than the lagtime τ. A consequence of performing this mapping is that any dynamics within a discrete state s cannot be resolved and only transitions between states contribute to the time-dependence of the correlation function. Eq. 2.26 can be rewritten by using C as the correlation matrix and e_a and e_b as the column vectors containing the expectation values of a and b , i.e.,

$$e_{a,i} = \langle a(t) \rangle_{s(t)=i} \quad e_{b,j} = \langle b(t) \rangle_{s(t)=j} \tag{2.28}$$

which yields:

$$\langle a(t) b(t+\tau) \rangle_t = e_a^T \cdot C \cdot e_b \tag{2.29}$$

The absolute probabilities, C_{ij}, and conditional probabilities, T_{ij}, are closely related:

$$T_{ij} = P(s_{t+\tau} = j \,|\, s_t = i) = \frac{P(s_{t+\tau} = j,\, s_t = i)}{P(s_t = i)} = \frac{P(s_{t+\tau} = j,\, s_t = i)}{\pi_i} = \frac{C_{ij}}{\pi_i} \tag{2.30}$$

and the correlation matrix, $C(\tau)$, can be expressed in terms of $T(\tau)$ and the probability distribution, Π:

$$C(\tau) = \Pi \cdot T(\tau) = L^T \cdot \Lambda(\tau) \cdot L \tag{2.31}$$

which leads to an expression connecting the correlation function with the Markov state model:

$$\langle a(t) \cdot a(t+\tau) \rangle_t = e_a^T \cdot C(\tau) \cdot e_b = e_a^T \cdot L^T \cdot \Lambda(\tau) \cdot L \cdot e_b \tag{2.32}$$

By using the eigenvector representation of $C(\tau)$:

$$L^T \cdot \Lambda(\tau) \cdot L = \sum_k l_k^T \cdot \lambda_k(\tau) \cdot l_k \tag{2.33}$$

the equation 2.32 can be split into a sum over relaxation processes, k:

$$\langle a(t) \cdot b(t+\tau) \rangle_t = e_a^T \cdot \left(\sum_k^m l_k^T \cdot \lambda_k(\tau) \cdot l_k \right) \cdot e_b = \sum_k^m \left(e_a^T \cdot l_k^T \right) \cdot \lambda_k(\tau) \cdot (l_k \cdot e_b) = \sum_k^m \lambda_k(\tau) \cdot A_k \tag{2.34}$$

where

$$A_k = (e_a^T \cdot l_k^T) \cdot (l_k \cdot e_b) \tag{2.35}$$

are the time independent and process dependent amplitudes of the correlation function. The decomposition of the transition kernel into m eigenvectors, leads to a decomposition of the correlation function into m processes. Each process represents a single exponential decay function with a relaxation time determined by $\lambda_k(\tau) = exp(-\tau/t_k)$ and an amplitude of $A_k = (e_a^T \cdot l_k^T) \cdot (l_k \cdot e_b)$.

2.3.2. X-ray and Neutron Scattering Observables

Both the incoherent and coherent intermediate scattering function, $F(\vec{q}, t)$, are time-autocorrelation functions, which makes it possible to express them in terms of Eq. 2.32. The generalized observable $a(t)$ for coherent scattering is

$$a(t) := A(\vec{q}, t) = \sum_\alpha b_\alpha \cdot e^{-i \cdot \vec{q} \cdot \vec{r}_\alpha(t)} \tag{2.36}$$

and for incoherent scattering:

$$a(t) := a(\alpha, \vec{q}, t) = b_\alpha \cdot e^{-i \cdot \vec{q} \cdot \vec{r}_\alpha(t)} \tag{2.37}$$

Within the framework of the MSM the observable itself becomes independent of time and is represented as a multicomponent vector $a = (a_1, \ldots, a_m)$, where each component is the expectancy value of the generalized observable for the particular Markov state i. This means that for coherent scattering the components of the observable are

$$a_i := \langle A(\vec{q}, t)\rangle_i = \left\langle \sum_\alpha b_\alpha \cdot e^{-i\cdot\vec{q}\cdot\vec{r}_\alpha(t)} \right\rangle_i \tag{2.38}$$

and for incoherent scattering:

$$a_i := \langle a(\alpha, \vec{q}, t)\rangle_i = \left\langle b_\alpha \cdot e^{-i\cdot\vec{q}\cdot\vec{r}_\alpha(t)} \right\rangle_i \tag{2.39}$$

The average is carried out for all time instances t for which the system occupies Markov state i. By applying the decomposition into m processes and making use of the relation $t_k = -\tau/\ln\lambda_k(\tau)$, both the coherent and incoherent intermediate scattering function assume the form:

$$F(\vec{q}, \tau) = \sum_k \lambda_k \cdot A_k(\vec{q}) = \sum_k \exp(-\frac{\tau}{t_k}) \cdot A_k(\vec{q}) \tag{2.40}$$

For coherent scattering the amplitudes are determined by

$$A_{k,coh}(\vec{q}) = \left| \sum_i \left\langle \sum_\alpha b_{\alpha,coh} \cdot e^{-i\cdot\vec{q}\cdot\vec{r}_\alpha(t)} \right\rangle_i \cdot L_{ik} \right|^2 \tag{2.41}$$

where L_{ik} is the left eigenvector component i of eigenvector l_k, corresponding to process k.

For incoherent scattering, the amplitudes are calculated according to

$$A_{k,inc}(\vec{q}) = \sum_\alpha \left| \sum_i \left\langle b_{\alpha,coh} \cdot e^{-i\cdot\vec{q}\cdot\vec{r}_\alpha(t)} \right\rangle_i \cdot L_{ik} \right|^2 \tag{2.42}$$

The following features of this representation are interesting:

- $F(\vec{q}, t)$ is completely described by the sum of individual exponential decay functions with individual relaxation times t_k
- The contribution of each atom to a particular process for incoherent scattering can be identified
- Since $k = 1$ corresponds to the equilibrium distribution, $A_{1,coh}$ and $A_{1,inc}$ instantly provide information about the elastic structure factor (ESF)

If the scattering particles in the experimental sample have no preferred orientation, the intermediate scattering function has to be orientationally averaged: $F(q, t) =< F(\vec{q}, t) >_{\vec{q}}$. The orientational averaging can be carried out at the level of the individual scattering amplitudes:

$$A_{k,coh}(q) = \left\langle \left| \sum_i \left\langle \sum_\alpha b_{\alpha,coh} \cdot e^{-i\cdot\vec{q}\cdot\vec{r}_\alpha(t)} \right\rangle_i \cdot L_{ik} \right|^2 \right\rangle_{\vec{q}} \tag{2.43}$$

$$A_{k,inc}(q) = \sum_\alpha \left\langle \left| \sum_i \left\langle b_{\alpha,coh} \cdot e^{-i\cdot\vec{q}\cdot\vec{r}_\alpha(t)} \right\rangle_i \cdot L_{ik} \right|^2 \right\rangle_{\vec{q}} \tag{2.44}$$

2.4. Supercomputing

Supercomputing is a very general term which can be applied to a variety of computational problems and computational resources. For instance, the Jaguar Petaflop* machine and the Folding@Home† network are both considered supercomputing platforms, even though their hardware and network characteristics are very different. The specific hardware layout in turn strongly determines the performance for scientific applications, because algorithms have to make distinct choices for data access and inter-node communication patterns. While it is possible that a particular scientific application may be able to choose from a set of algorithms to exploit different hardware specifications, most applications are inherently limited to only a few algorithms which are biased towards a specific hardware layout.

The tight coupling between the hardware specification of the supercomputer and the design of the algorithm has implications on the preference of scientific applications towards a particular supercomputing platform. The purpose of this section is to clarify some of the dependence of scientific applications on hardware requirements. Two relevant scientific applications here are molecular dynamics and simulation data analysis, which are distinctly different in their parallel performance characteristics. While molecular dynamics usually needs to make heavy use of message passing and only occasionally writes results to disk, simulation data analysis may only make occasional use of message passing and depends heavily on the available IO and network performance for reading the data into local memory and subsequent processing.

2.4.1. Software and Hardware Characteristics

The introduction and commercialization of personal computers had a profound impact on the architecture of today's supercomputers [52]. Early machines were filled with a massive amount of individual processing units connected through a high performance bus, with all processing units having access to a shared random access memory (RAM). However, with time it became much more cost effective to create massively parallel computing platforms by acquiring a large quantity personal computer hardware and connect them through a high performance network. This created a paradigm shift for scientific application programming from a shared memory environment towards a distributed memory environment, where each processing unit holds ownership of only part of the total memory. One consequence of this paradigm shift is that data access patterns became the defining criterium for application performance, because processing units need to request data from other processing units through message passing [53].

Parallel programming in a distributed memory environment requires scientific applications to express their algorithms as a set of tasks which may need to communicate data with each other. This puts scientific applications in either of two categories: *Embarrassingly parallel*, where the individual tasks do not have to communicate with the exception of initialization and finalization, and *delightfully parallel*, were the tasks need to exchange data within the core algorithm either synchronously or asynchronously. The strategy of task creation, i.e. the partitioning scheme, is a defining criterium, both for the scalability and the type of parallelism involved. A finely grained partitioning scheme usually results in an algorithm which is scalable to a large number of processing units, but may introduce interdependency between the tasks and downgrades the algorithm to delightfully parallel. For many scientific applications scalability can be distinguished by its strong and weak performance. *Strong scalability* denotes the proportion by which the execution time of an algorithm is reduced as a function of increasing number of processing units for a fixed problem

*NCCS: http://www.nccs.gov

†Folding@home: http://folding.stanford.edu

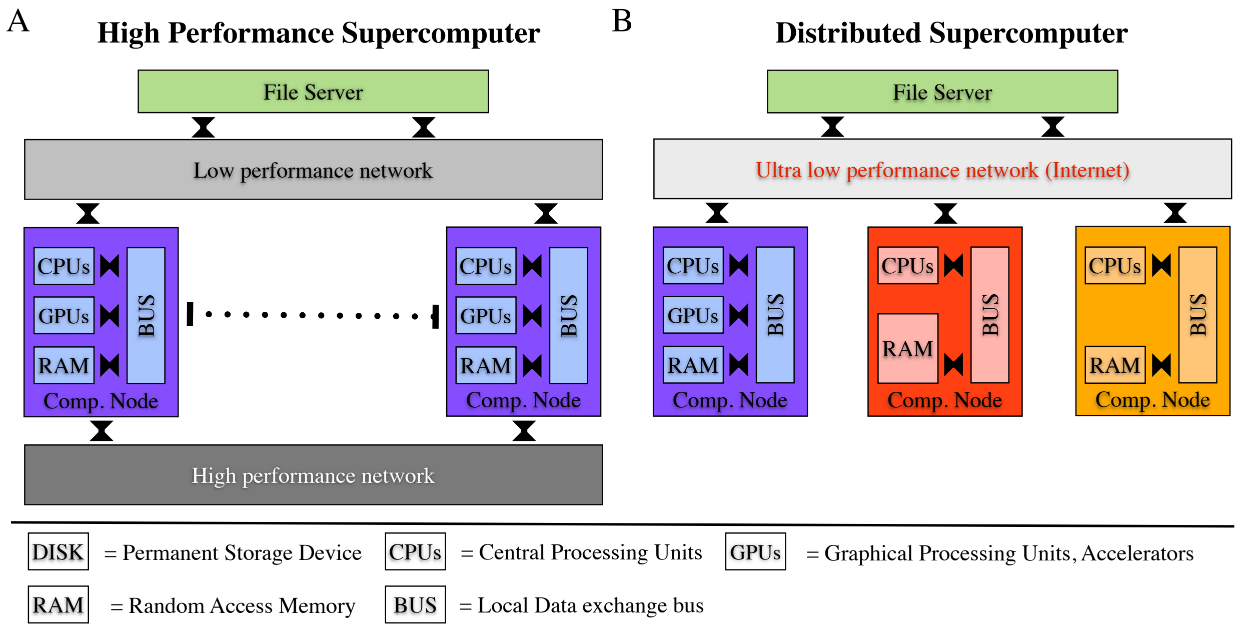

A
High Performance Supercomputer
File Server
Low performance network
CPUs
GPUs
RAM
BUS
Comp. Node
High performance network
B
Distributed Supercomputer
File Server
Ultra low performance network (Internet)
DISK = Permanent Storage Device
CPUs = Central Processing Units
GPUs = Graphical Processing Units, Accelerators
RAM = Random Access Memory
BUS = Local Data exchange bus

Figure 2.1.: The generic hardware layout of modern supercomputers. A) High performance supercomputer consist of a high performance network between equal compute nodes for message passing and a lower performance network for data access, while B) distributed supercomputer contain a variety of compute nodes with different hardware characteristics with a low performance connection to a data server.

size, while *weak scalability* describes the variation in execution time as a function of the number of processing units for a fixed problem size per unit.

Figure 2.1 shows the generic layout of two characteristic classes of modern supercomputers. Fig. 2.1A is comparable to the Jaguar PetaFlop machine, which currently consists of 18,688 compute nodes connected through a SeaStar type high performance network, where each node contains two hex-core 2.6GHz AMD Opteron 2435 processors and 16GB of DDR2-800 RAM. Although the permanent storage data access is usually carried out by a lower performance network, on Jaguar it is managed by the high performance network as well[‡§]. Fig. 2.1B is comparable to the Folding@Home supercomputing project platform[¶], in which owner's of personal computer voluntarily participate in scientific computing projects. Participants are generally not able to communicate data between each other, which restricts this type of computing platform to algorithms which are embarrassingly parallelizable. An additional defining criteria of a scientific application is the amount of data which has to be retrieved from the file server: even if the algorithm itself is embarrassingly parallel, the low network performance may make the distributed supercomputer unpractical.

2.4.2. Scaling of Molecular Dynamics

The aim of molecular dynamics simulation is to compute the time evolution of a model structure in order to explore the thermodynamically allowed configurations. The most simplistic approach to this is to iterate the differential equation of motion and calculate the forces on each atom for

[‡] http://www.theregister.co.uk/2010/04/19/cray_third_gen_linux/

[§] http://en.wikipedia.org/wiki/Portals_network_programming_api

[¶] http://folding.stanford.edu/

every time step. The implementation of this algorithm within a distributed memory environment can be done in a variety of ways which have significant impact on the attainable performance and scalability [54, 55**?**]. E.g. atomistic partitioning schemes assign atoms permanently to compute nodes based on the initial structure, while domain decomposition assigns atoms temporarily to compute nodes based on their momentary position in space. The use of domain decomposition increases the inter-node communication by the number of trafficking atoms, but minimizes the number of interactions which have to be computed between atoms on different nodes. This is especially true for model systems in which atoms are allowed to diffuse in space, in which case an atomic decomposition scheme leads to an overall increase in the number of interaction calculations between compute nodes.

In many cases, the number of interactions, N_I, is significantly larger than the number of atoms, N_A, because each atom is subject to a variety of forces at the same time. Dependent on the type of the force, N_I scales differently as a function of N_A: the number of bonded interactions, $N_{I,bond}$, scale with $O(N_A)$ while non-bonded, $N_{I,nonbond}$, principally scale with $O(N_A^2)$. The scaling for $N_{I,nonbond}$ can be reduced to $O(N_A)$ by introducing distance dependent cutoffs.

Simulation protocols designed for good scalability usually exploit the physical features of the simulated system and the forces. E.g. non-bonded interactions can be calculated less frequently than bonded interactions (usually a factor 2), because they are less sensitive to the distance variations of atoms than bonded interactions. Also, if the system consists of overall neutral molecules, the electrostatic interactions can be computed using the Reaction Field approximation, instead of Particle Mesh Ewald (PME) [56, 39].

Other strategies to increase the scalability of molecular dynamics simulations usually involve minimizing the number of global synchronization points during the simulation, e.g. the frequency for updating the neighbor atom exchange list, the frequency at which the atomic coordinates are saved to disk, or the communication of energies (thermostat control).

2.4.3. Scaling of Simulation Data Analysis Algorithms

The analysis of molecular dynamics simulation data involves loading the atomic coordinates into computer memory and subsequently applying an algorithm to derive the property of interest. Dependent on the type of property, the simulation data can either be processed in small chunks or has to be read wholly into computer memory. Because disk access is usually several orders of magnitude slower than direct memory access or message passing, algorithms must *avoid* disk access at any cost, especially for large trajectory files. However, since the data has to be read at least once, the optimal solution for an algorithm is to request the data only once with the highest performance achievable and then keep the data available in computer memory.

The calculation of a property from molecular dynamics simulation assumes the generic form

$$f(n,t,\lambda_i) \mapsto F \tag{2.45}$$

where the macroscopic property F is computed from the set of microscopic properties $f(n,t,\lambda_i)$, which describes a function for atoms n at times t with respect to a set of parameters λ_i. Trajectory files in native format contain atomic coordinates, $\vec{r}(n,t)$, as a series of snapshots of system configurations, where each frame contains the coordinates for all atoms at a particular time. An efficient algorithm needs to implement an optimal strategy to compute the macroscopic property through microscopic properties from the atomic coordinates, i.e.

$$\vec{r}(n,t) \mapsto f(n,t,\lambda_i) \mapsto F \tag{2.46}$$

Equation 2.46 makes two aspects apparent which are important for the performance on a parallel computer with distributed memory:

data locality The operation which computes $f(n, t, \lambda_i)$ from $\vec{r}(n, t)$ iterates over the set of parameters λ_i. Whenever this set is large, the latency for retrieving $\vec{r}(n, t)$ adds up and can outweigh the time to compute $f(n, t, \lambda_i)$. Putting data close to the processing unit can reduce latency by several orders of magnitude. The latency itself is dependent on the location of the data within the memory hierarchy in increasing order: processor cache, RAM, disk/network.

intermediate results The iteration over the set of parameters λ_i generates a large amount of intermediate data which has to be reduced to derive the macroscopic property. However, intermediate data which belongs together may be scattered throughout the parallel computer and need to be collected on one node.

Fulfilling data locality and eliminating intermediate data exchange, with the exception of a final communication step, is possible but can be mutually exclusive for some algorithms and mainly depends on the size of the total data. Good examples, which are also of relevance here, are the incoherent and coherent dynamic neutron scattering functions. An algorithm for computing incoherent scattering can easily fulfill both data locality and eliminate exchange of intermediate results because the principal microscopic properties, $f(n, t, \lambda_i)$, are specific to individual atoms and require only a part of the total simulation data (see Equation 2.16):

$$f\left(n, t, \lambda_i\right) := f_n(\vec{q}, \tau) = \langle a_n(\vec{q}, t) \cdot a_n^*(\vec{q}, t + \tau)\rangle_t \tag{2.47}$$

$$F := F(\vec{q}, \tau) = \sum_n \langle f_n\left(\vec{q}, \tau\right)\rangle_{\vec{q}} \tag{2.48}$$

Thus, putting the complete data for atom n on a single node allows to compute partial results, $f_n(\vec{q}, \tau)$, which only need a final reduction step. Task parallelization is then achieved by assignment for atom n and scattering vector length q. On the contrary, the coherent scattering function has to trade between data locality and exchange of intermediate data. Splitting the computation into its two steps, calculation of total scattering amplitudes, $A(\vec{q}, t) = \sum_n a_n(\vec{q}, t)$, and subsequent autocorrelation $F(\vec{q}, \tau) = \langle A(\vec{q}, t) \cdot A^*(\vec{q}, t + \tau)\rangle_t$, makes this apparent: The individual $A(\vec{q}, t)$ require the atomic coordinates for all atoms at a particular time, while the autocorrelation requires the series of $A(\vec{q}, t)$ for all times. The only way to avoid the exchange of $A(\vec{q}, t)$ between compute nodes is to store a full copy of the simulation data on each node and compute all $A(\vec{q}, t)$ locally. However, this approach prevents the time parameter from being used as a decomposition parameter for parallelization, which has negative effects on scalability.

3. Scattering and Simulation Studies

This work contains both published and unpublished results. The first section, “Cellulose”, contains mainly unpublished data and is intented to illustrate the ability to perform extensive structural and dynamical characterization of simulated biomolecules using experimental scattering techniques. Section “Lignocellulose” and “Alanine Dipeptide” contain excerpts from manuscripts which are currently being reviewed or prepared for publication, respectively. The work presented in the last section, “High Performance Calculation of Scattering Profiles” has been successfully published and is available in the Journal of Computer Physics Communications [19].

This work brings together applied scattering theory and molecular dynamics simulation to derive comprehensive structural and dynamical fingerprints of cellulose fibers (Section 3.1, Cellulose), it provides strategies to simulate and analyze highly heterogeneous molecular systems on a massive scale (Section 3.2, Lignocellulose), it shows how advanced statistical tools like Markov Chain Modeling can be used to decompose and interpret the dynamical scattering functions of molecules (Section 3.3, Alanine Dipeptide), and it explains how the scattering functions can be calculated at high performance by exploiting the architecture of modern supercomputers (Section 3.4, High Performance Calculation of Scattering Profiles). The common aspects of these projects point towards a general theme: The convergence between supercomputing, molecular dynamics simulation and experimental scattering techniques.

3.1. Cellulose

Cellulose is a natural biopolymer and an integral part of the cell wall of plants. It is synthesized within the plasma membrane by multi-protein complexes, called cellulose-synthesizing complexes [57, 58]. The particular structural arrangement of the protein components within the complex has a strong influence on the molecular structure and shape of cellulose. In terrestrial plants the most common form of these complexes are rosettes, which consists of 36 individual proteins organized in 6 domains, as illustrated in Figure 3.1. During synthesis, each protein produces a single glucose chain by successively adding D-glucose molecules to the existing polymer, which results in the creation of a microfibril containing 36 glucose chains. The resulting glucose chain, shown in Figure 3.2, consists of repeating units of cellobiose, which is comprised of two screw symmetrical glucose molecules. The tight packing of the individual chains induces crystallization, with the most common forms being I-α and I-β. The crystal symmetry has been determined experimentally using a combination of fiber-aligned X-ray and neutron scattering [59] (a=7.784Å^{-1},b=8.201Å^{-1},c=10.380Å^{-1},$\measuredangle(a,b) = 96.5°$,$\measuredangle(a,c) = \measuredangle(b,c) = 90°$). The structure of a I-$\beta$ crystalline fiber is provided in Figure 3.3.

Plants produce a significant amount of cellulose during their lifetime leading to a large natural abundance and the largest source for sugar molecules on earth, making it an attractive feedstock for the production of biofuels. However, since cellulose acts as a structural component in the anatomy of plant cells [60], its evolution has caused it to be highly resilient to thermal degradation or enzymatic digestion [62]. Additionally, cellulose tends to be incorporated into a network of noncovalently and covalently bonded biomolecules, mainly comprised of hemicellulose and lignin [63], which protects cellulose even further from enzymatic digestion. To increase the efficiency of hydrolytic enzymes,

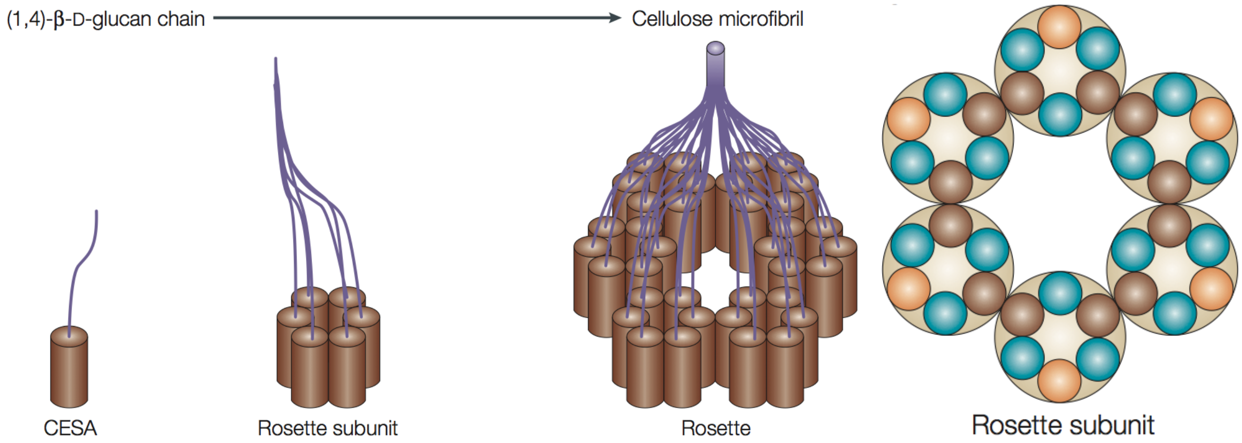


Figure 3.1.: The Cellulose Synthesis Complex is a particle embedded in the plasma membrane of plant cells and facilitates the production of cellulose chains. Graphic taken from Ref. [60]. Each subunit of the hexameric rosette structure is composed of 3 different cellulose synthase (CESA) proteins which is indicated by different colors.

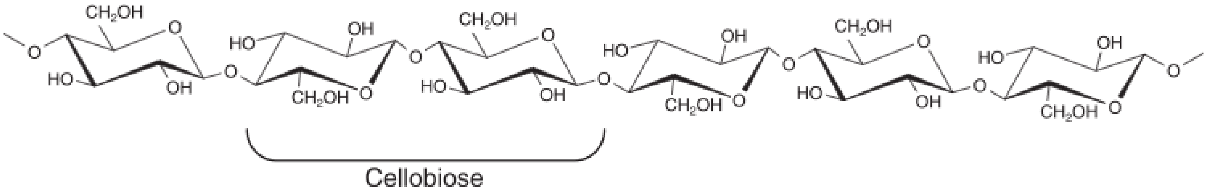


Figure 3.2.: The molecular structure of a single cellulose chain (β-1,4-glucan). The cellobiose subunit contains two glucose molecules linked via a β-1,4 bond, where the second unit is rotated 180 degrees around the chain axis. Graphic taken from Ref. [58].

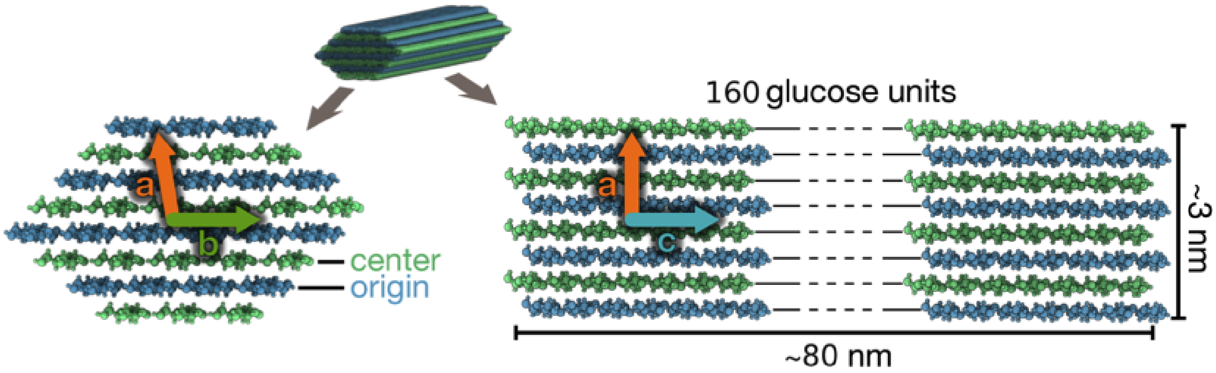


Figure 3.3.: Molecular structure of I-β crystalline cellulose. The unit cell vectors indicate the extend of the crystallographic unit cell. The interaction between neighboring chains are is non-covalent. Cellulose is organized in a sheet-like structure. The interaction within a sheet is dominated by hydrogen bond, while the interaction between sheets is governed by van der Waals forces. The graphic illustrates a consensus model described in Ref. [61].

this biomass is commonly pretreated using one of the major pretreatment technologies, e.g. dilute acid pretreatment, steam explosion or ammonia fiber expansion [64]. While each pretreatment technique has a distinct effect on the structural features of biomass, two aspects are considered the determining factor of the remaining recalcitrance to enzymatic digestibility: Cellulose crystallinity and the amount of remaining lignin [65].

3.1.1. Crystallinity of Cellulose

The crystallinity of cellulose within a biomass sample is commonly assessed in terms of a crystallinity index CrI by measuring the wide angle X-ray scattering (WAXS) spectrum and estimating the intensity of amorphous background, I_{am}, and the 200 Bragg peak, I_{200}, at 1.6 Å^{-1}, which is also known as the Segal method [66]:

$$CrI = \frac{I_{200} - I_{am}}{I_{200}} \tag{3.1}$$

The finite size of the cellulose crystallites leads to errors in the estimation of the crystallinity index because a significant amount of the scattering intensity falls in between the theoretical Bragg peaks. Different methods have been explored previously to yield better estimates for the amount of crystallinity, including Fourier-Transform Infrared Spectroscopy, Nuclear Magnetic Resonance and Fourier-Transform Raman [67]. However, WAXS remains the most commonly used method, likely because of its apparent simplicity and availability.

The reason for the existence of crystallinity within cellulose, i.e. the strong translational symmetry of the D-glucose molecules, is attributed to the formation of a regular hydrogen bonding pattern within chains and between neighboring chains. While the intra-chain hydrogen bonding contributes to the persistence length of individual cellulose chains, i.e. their tendency to remain rigid, the hydrogen bonds between chains prevent the cellulose fibril from disintegrating into individual chains. The dominant hydrogen bonding patterns have been determined previously for the I-α and I-β crystal symmetry with the help of neutron scattering [59, 68]. They revealed that inter-chain hydrogen bonds within cellulose are organized into a sheet pattern, where cellulose chains strongly hydrogen bond with neighboring chains along the cyclic plane of the glucose molecules, while the interaction between the stacked sheets is mostly van der Waals. The combination of the intra- and inter-chain hydrogen bonding network together with the high degree of polymerization is what protects cellulose efficiently from thermal degradation.

The crystallinity of cellulose elementary fibrils can be investigated directly by computing the scattering intensities for all possible scattering vectors q. This has been done for crystalline cellulose and the 2D directional X-ray scattering diagram perpendicular to the elementary fiber axis is shown in Figure 3.4. Each Bragg peak is tightly connected to an existing symmetry within the cellulose fiber. One of the three dominant peaks in the cellulose fiber is the 200 peak, which describes the nearest distance between a sheet comprised of origin chains and one consisting of center chains. The two other peaks, 110 and 1-10, originate from the nearest distance between planes comprised of an alternating mix of origin and center chains.

Experimentally, directional scattering experiments on cellulose are challenging, because the fibers need to be perfectly aligned to yield a diffraction diagram similar to Figure 3.4. However, most of the time scattering analysis is performed on cellulose samples which contain fibers pointing in random directions, which yields orientationally averaged scattering diagrams which can be reduced to one dimension (WAXS). The WAXS of the I-β cellulose model fiber is shown in Figure 3.5. The WAXS pattern has been decomposed into the contributions from the major peaks (200,110,1-10) by fitting with gaussians and the background has been estimated from the intensity minimum between the 110 and 200 peak. The fitting parameters are provided in Table 3.1. Using the Segal

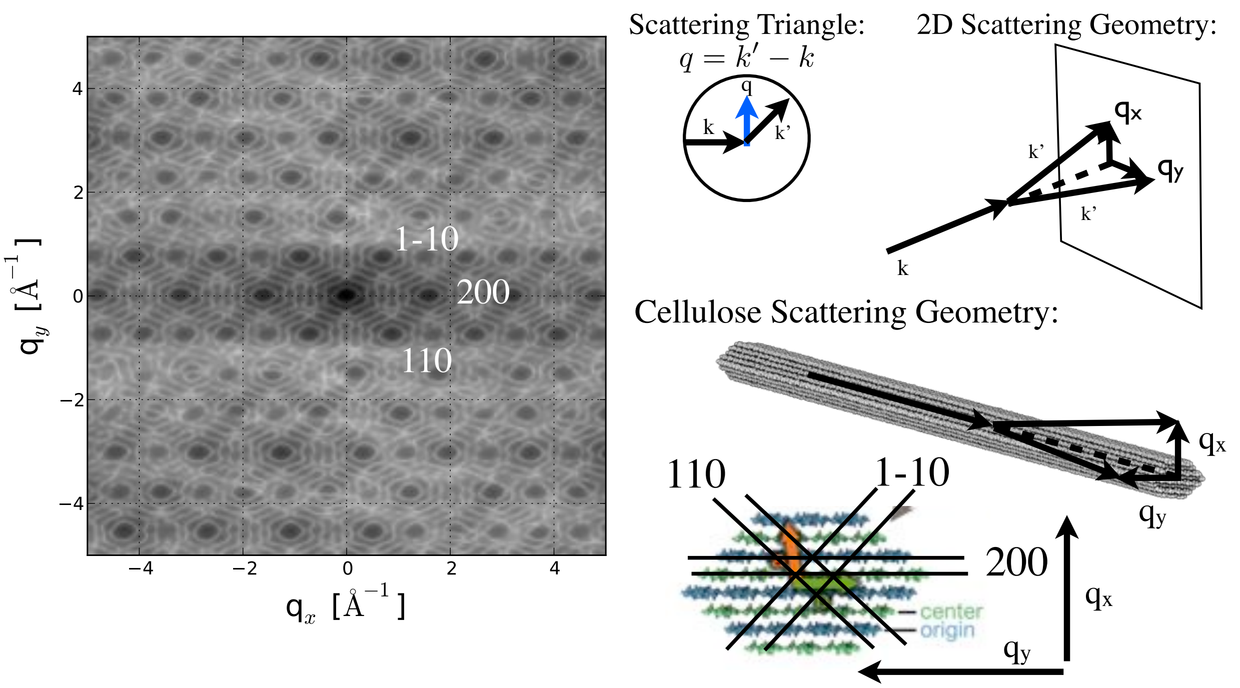


Figure 3.4.: Directional X-ray scattering diagram for I-β crystalline cellulose (left) and illustrations of the scattering geometries (right).

WAXS diagram peak fitting parameters for crystalline cellulose:

	peak location	width	height
1-10	0.96468	0.056318	7314.9
110	1.0978	0.095805	13037
200	1.5748	0.10558	25899

Table 3.1.: Fitting parameters for the WAXS diagram of a I-β cellulose elementary microfibril. The baseline has a value of 9593.4. A standard gaussian function was used for fitting each peak.

equation 3.1, the estimated crystallinity index for a I-β crystalline cellulose elementary fibril is 75%. The underestimation of the crystallinity index due to the Segal method is caused by the finite size of the cellulose crystal. In an infinite crystal, the scattering intensity between the 110 and 200 peak would resolve to the individual peaks other than 110 and 200 which fall into the interval and the width of the 110 and 200 would approach zero.

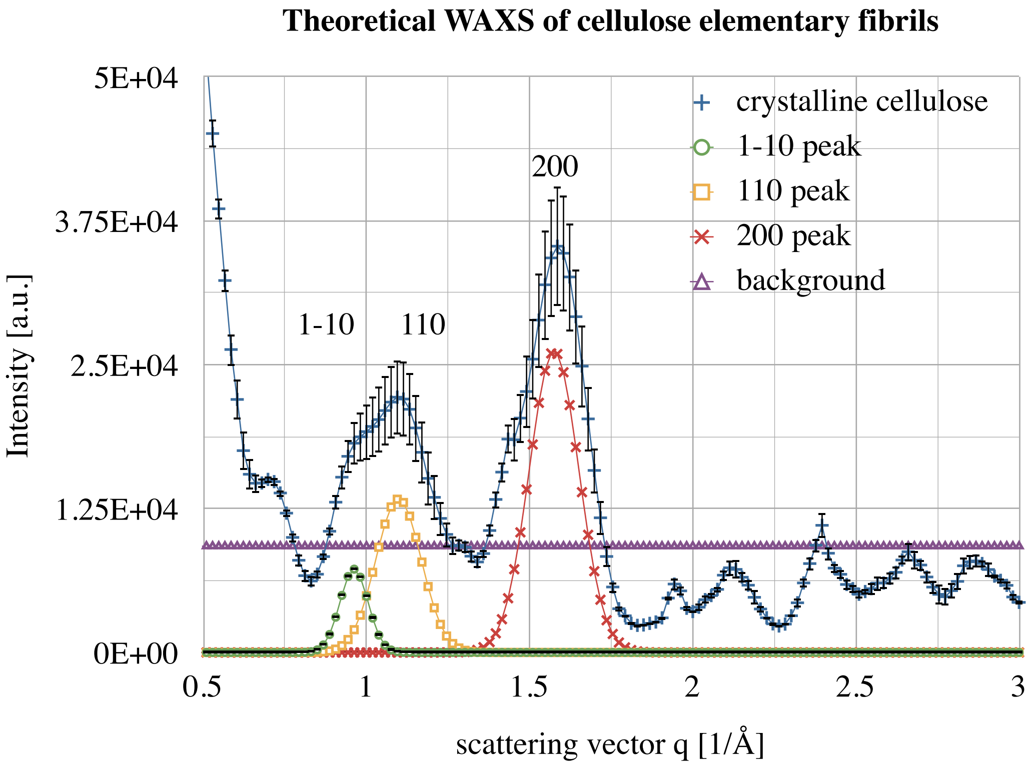


Figure 3.5.: WAXS diagram of the (perfectly) crystalline I-β cellulose elementary fiber model. The pattern is decomposed into the contributions of the 1-10,110 and 200 peak. Other peaks are ignored for simplicity. The choice for the background level is consistent with the experimental practice to estimate the amorphous contribution from the intensity at the minimum between the 110 and 200 peak.

Hydrogen bond statistic for cellulose models:

Model	# Hydrogen Bonds	# Hydrogen Bonds/Glucose
n-0	20180	3.5
n-10	19790	3.4
n-20	19256	3.3
n-30	18441	3.2
n-40	18168	3.2
p-40	17627	3.1
p-30	16693	2.9
p-20	16276	2.8
p-10	15661	2.7
p-0	14939	2.6
np-inner	16209	2.8
np-outer	18132	3.1

Table 3.2.: Hydrogen bond statistic for the cellulose models featuring different crystallinities. Used a distance cutoff of 3.2 Å and an donor acceptor angle of larger than 110 degree. Hydrogen bonds were counted using the VMD software. The number for hydrogen bonds per glucose was rounded to ±0.1.

3.1.2. Disorder in Cellulose

Even though cellulose is capable of forming regular hydrogen bonding patterns which stabilize the crystalline form, a significant portion of biomass contains cellulose which is in a noncrystalline state [66]. The amount of noncrystalline cellulose present in a cellulose sample is commonly determined indirectly through the crystallinity index [67]. However, these measurements do not provide information on how the crystallinity is distributed within a cellulose fiber and where noncrystalline cellulose is located. In particular, it is not known how different forms of disorder contribute to the measurement of the crystallinity index. This question is explored here with a help of a set of artificial cellulose fiber models, shown in Figure 3.6.

The models contain different amounts of crystalline cellulose, with two models exhibiting disorder along the cylindrical axis and the remaining ones as segments embedded in the fiber. The number of internal hydrogen bonds for the different cellulose models are listed in Table 3.2, which shows that with increasing amount of noncrystalline cellulose, the average number of hydrogen bond per glucose molecule decreases. This is consistent with the experimental observation that noncrystalline is more readily hydrolyzable [69].

The WAXS diagram for selected cellulose fiber models are shown in Figure 3.7, which reveals two striking features: Noncrystalline cellulose also produces identifiable peaks in the WAXS diagram, albeit shifted to smaller q values, and WAXS diagrams for different models makes can be very similar. For example, model n-20 and np-outer show nearly an identical WAXS profile, even though their models are qualitatively different. On the other hand, the composite WAXS pattern of crystalline and noncrystalline cellulose can not perfectly reconstitute patterns calculated for the mixed phase models. This is illustrated in Figure 3.8, which compares the WAXS patterns of the n-40 mixed phase model to a superposition of 40% crystalline (n-0) and 60% noncrystalline (p-0) cellulose. The intensities are slightly off, which can be attributed to the fact that n-40 features structural regions between the crystalline and noncrystalline phase, which are only partially disordered.

The crystallinity of each model in Figure 3.7 was analyzed using the Segal method. Due to the peak shifts for the noncrystalline models the method was reinterpreted in the spirit of the original

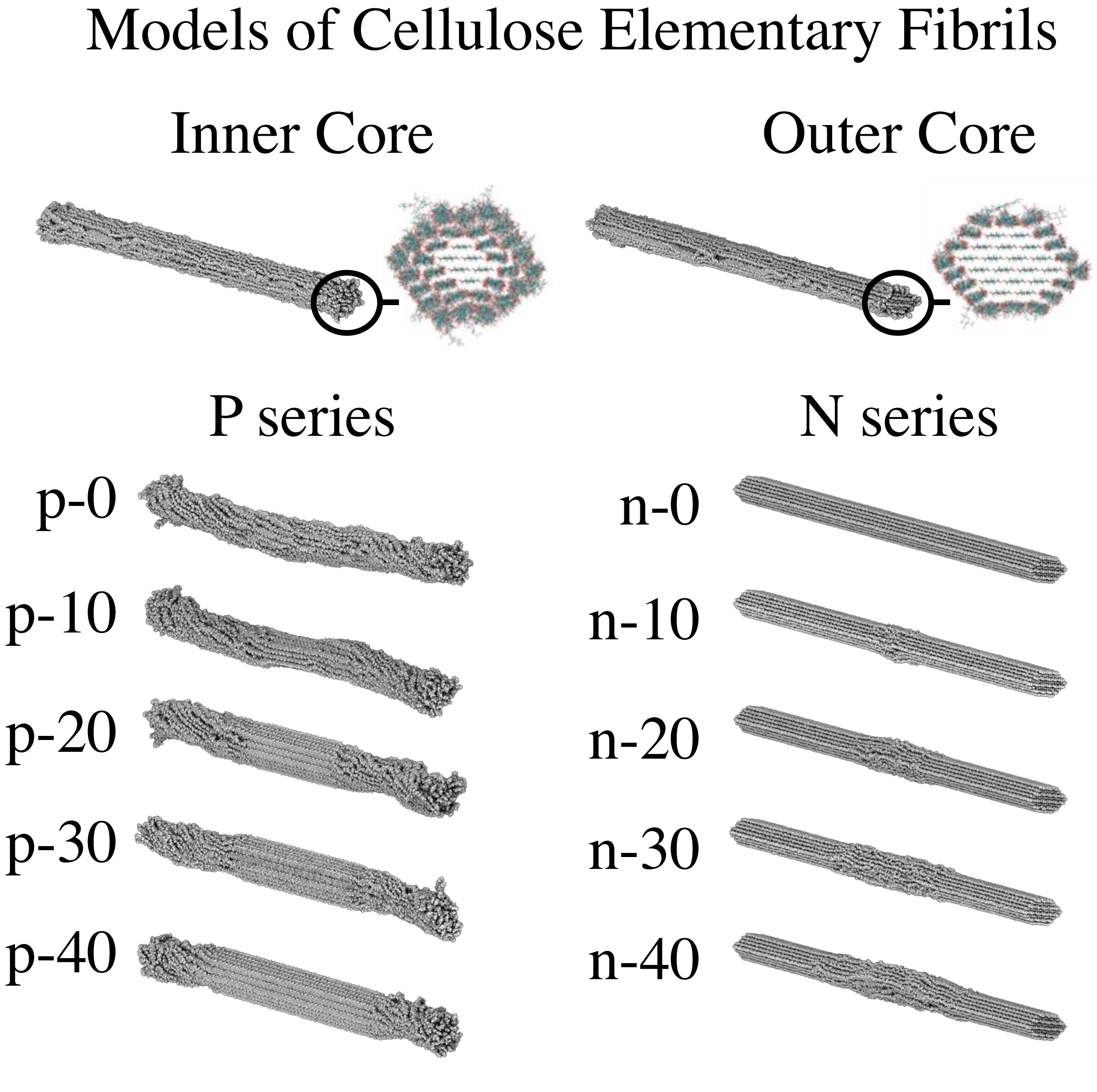


Figure 3.6.: A structural library of I-β cellulose models, with varying degree of crystallinity.

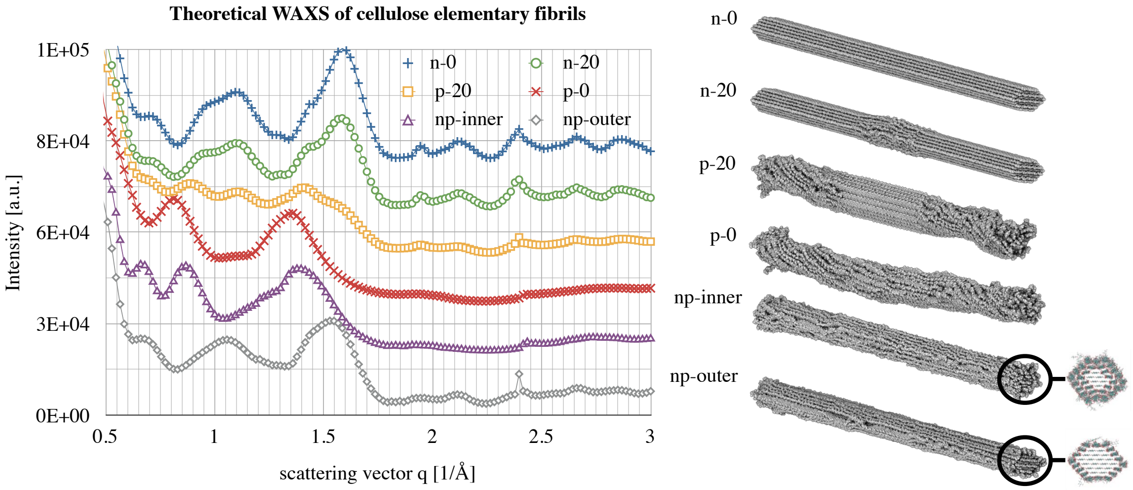


Figure 3.7.: WAXS diagrams for chosen candidates of the cellulose fiber structures library. The scattering intensities were offset by a multiple of 15000 dependent on the model for the sake of better visualization.

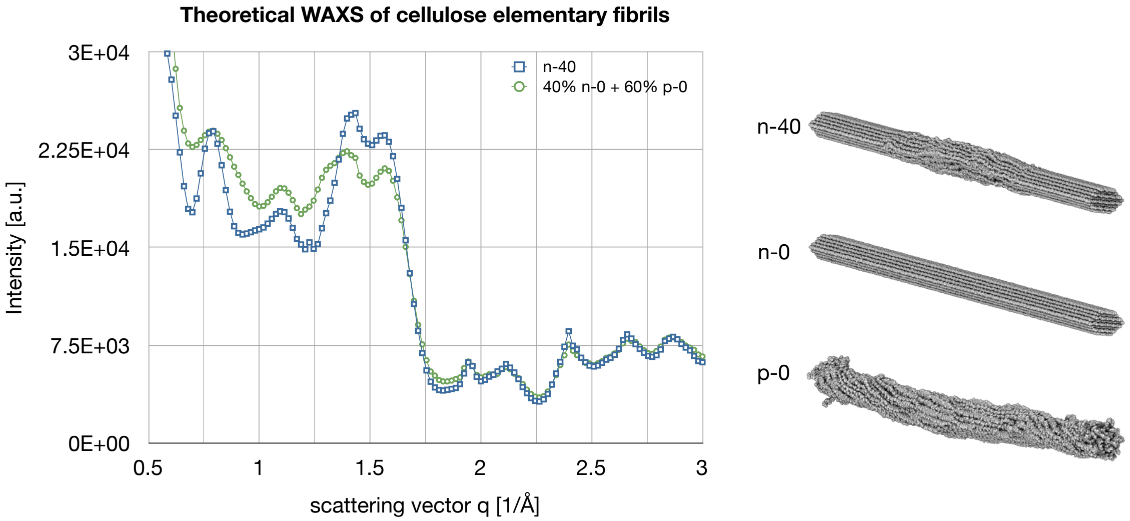


Figure 3.8.: WAXS pattern comparison between two models for mixed cellulose crystallinities. The difference in the scattering intensity between $q = 0.5$ Å^{-1} and $q = 1.6$ Å^{-1} illustrates that the particular manifestation of crystallinity within cellulose has a strong influence on scattering profile.

Crystallinity indexes for selected cellulose fiber models:

Model	200 Peak Height	200 Peak	Background	Minimum	Crystallinity
n-0	34700	1.6	8600	1.35	0.75
n-20	28500	1.6	12400	1.25	0.56
p-20	23300	1.4	18400	1.25	0.21
p-0	30700	1.35	17400	1.0	0.43
np-inner	28900	1.4	14000	1.05	0.51
np-outer	28300	1.55	15600	1.35	0.45

Table 3.3.: Crystallinity indexes and fitting parameters for the different cellulose models. The values were rounded to ±100, ±0.05, ±0.01, ±100, ±0.05 and ±0.01, respective to listed column.

method by identifying the highest peak and the minimal intensity between the peaks originating from the 110 and 200 peak. This leads to fitting parameters and crystallinity indexes listed in Table 3.3. The Segal method leads to the counter intuitive result that the crystallinity index for p-20 is lowest, instead of p-0, even though p-20 still has a portion of crystalline cellulose incorporated into the elementary fibril. With the exception of p-20, the range of crystallinity indexes is between 43% for noncrystalline and 75% for crystalline cellulose. Thus even an experimental sample with purely noncrystalline cellulose may yield a significant crystallinity index.

The particular features of the WAXS diagrams in Figure 3.7 can be understood by investigating the corresponding directional scattering diagrams perpendicular to the cellulose fiber axis and are shown in Figure 3.9. The rigorous peak assignment is possible by inspecting model n-0. The peak assignment for the np-inner and np-outer model can be inferred from the n-0 model. Also apparent from Figure 3.9 is the loss of distinct Bragg peaks for the p-0 model, which is symptomatic for noncrystalline structures.

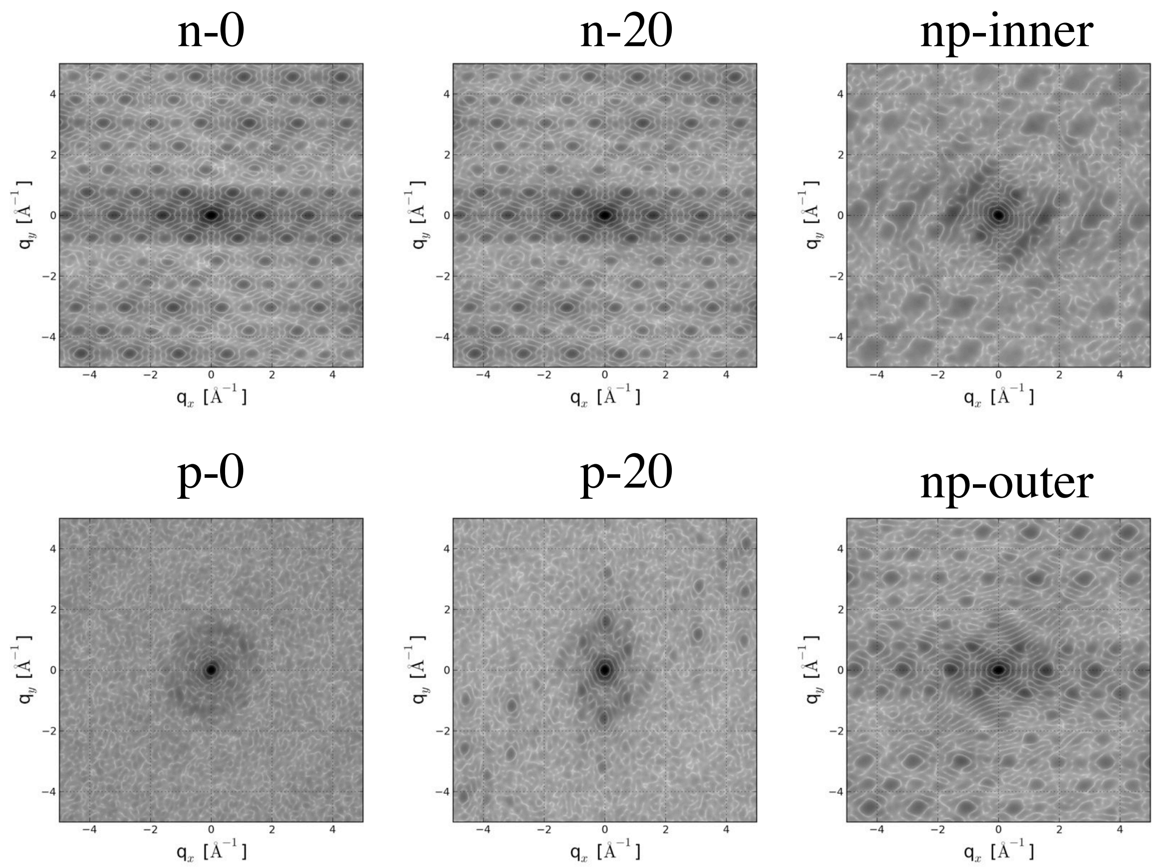


Figure 3.9.: Directional (2D) scattering diagrams for selected cellulose fiber models.

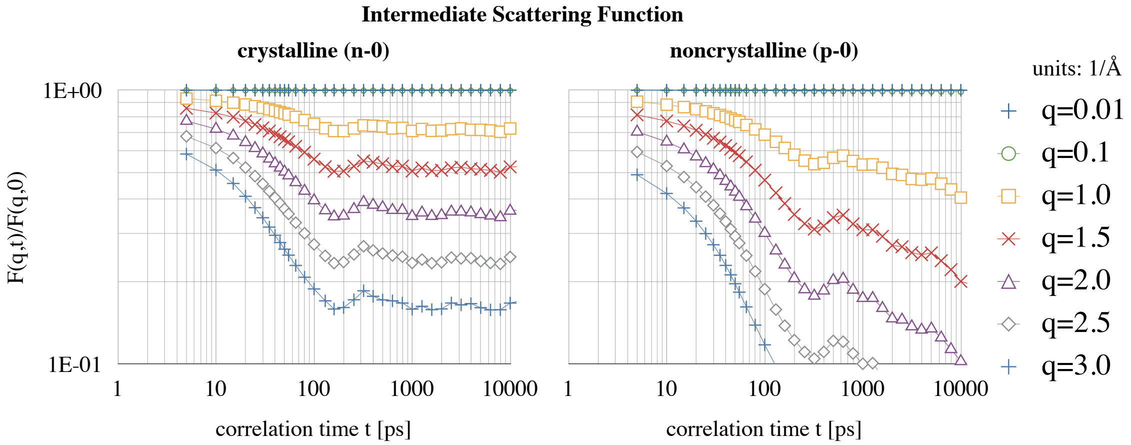


Figure 3.10.: Intermediate scattering function for crystalline and noncrystalline cellulose at different values for the scattering length density. Scattering analysis was performed using incoherent neutron scattering lengths for the atomic scattering amplitudes.

3.1.3. Cellulose Dynamics

The analysis of molecular dynamics simulation using scattering theory does not only provide information about structural symmetries, but also about spatio-temporal correlations within molecules. This is particularly interesting for cellulose, because its crystallinity is expected to influence its intra-molecular diffusional and vibrational characteristics.

Here, two dynamic scattering techniques are applied to the question how cellulose crystallinity affects the scattering signal: Incoherent inelastic neutron scattering (INS) and inelastic X-ray scattering (IXS) (which is coherent). INS probes the self diffusion of individual atoms and is a direct measure for the phase-space volume explored by the atoms, and is defined in Equation 2.16. IXS on the other hand is defined as the autocorrelation of the total scattering signal and thus probes fluctuations in the scattering density of the molecule and is formally described by equation 2.15.

3.1.3.1. Incoherent Inelastic Neutron Scattering

Two cellulose models were simulated: n-0, which is fully crystalline, and p-0, which is noncrystalline (simulation details are provided in the supporting information). Figure 3.10 shows the intermediate scattering function (derived using neutron scattering density lengths) for long time scales up to 10 ns which probe slow relaxations and is based on a trajectory with a total of 25 ns and 5 ps sampling time. It shows that the dynamics is qualitatively independent of q and that values above 1.0 Å^{-1} are needed to achieve a significant decay of the intermediate scattering function (60% decay at q =1.5 Å^{-1}). This indicates that the phase-space volume explored by cellulose is small compared to the size of the molecule, which is a signature of a rigid molecule.

The INS between crystalline and noncrystalline has been directly compared in Figure 3.11 for q =1.5 Å^{-1}. The differences have been grouped into three different time regimes: ps, sub ns, and ns. In the ps regime, both crystalline and noncrystalline cellulose exhibit a decay in the intermediate scattering function. However, the relative difference in the absolute value doesn't vary much as a function of correlation time. This means that the crystalline and noncrystalline cellulose exhibit the same relaxation processes in the *ps* regime, and the initial difference can be attributed to additional

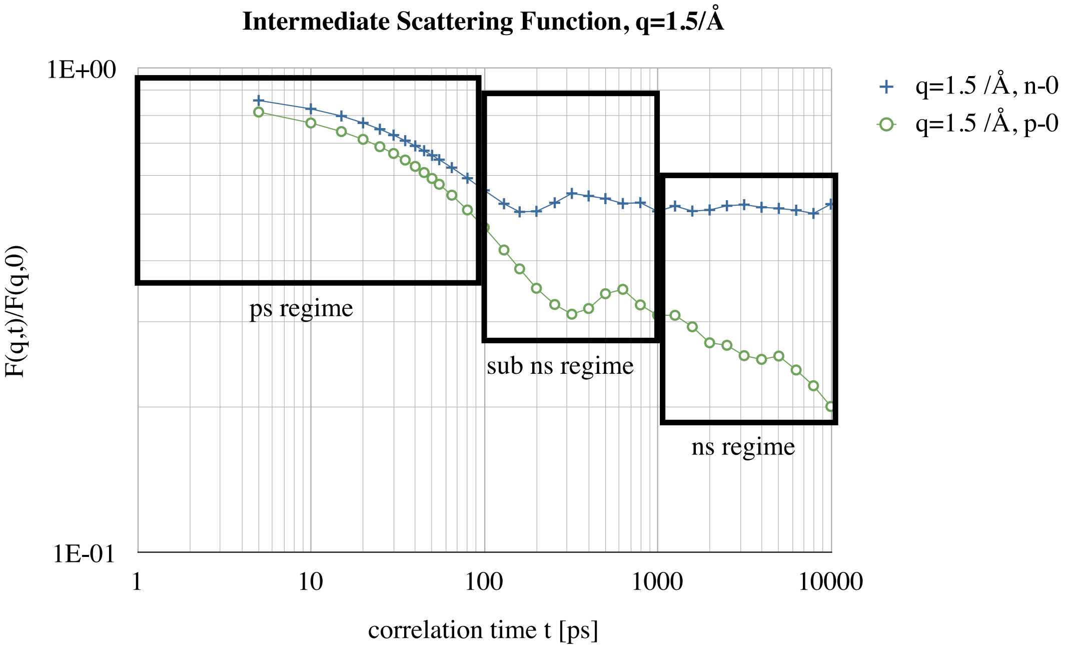


Figure 3.11.: Intermediate scattering function for crystalline and noncrystalline cellulose at q=1.5 Å^{-1} and their subdivision into several characteristic time regimes.

relaxation for noncrystalline on the sub ps time scale. On the sub ns time scale, both models exhibit a reconstitution of the intermediate scattering function (increase), which is typical for well defined oscillations. However, the oscillation peak for crystalline and noncrystalline cellulose are located at different times: 300 ps and 600 ps, respectively. This indicates that crystalline cellulose is stiffer and the corresponding oscillation is associated with a higher force constant and frequency than noncrystalline cellulose.

Crystalline and noncrystalline cellulose also differ in their longtime (sub ns, ns) relaxation: While the INS for crystalline cellulose flattens out above 100 ps, noncrystalline cellulose keeps decaying well above the 5 ns mark. This indicates that noncyrstalline cellulose features an additional diffusion process, allowing its atoms to achieve larger displacements over time.

The origin of the oscillation in the sub ns time regime was investigated by computing the local intermediate scattering function for different segments along the fiber axis. From visual inspection the fiber is observed to exhibit a strong normal mode perpendicular to the fiber axis, forming a standing wave with two knots near the end of the fiber. Figure 3.12 illustrates how this mode dominates the sub ns time regime by causing strong local differences between the intermediate scattering function at the different locations along the fiber axis. Those segments which are located at the knots of the standing wave have a reduced relaxation and do not show the characteristic oscillation peak, while segments located in the middle and at the end of the fiber clearly show the oscillation in the intermediate scattering function.

The ability to detect dynamical differences between surface and core cellulose chains with the help of the intermediate scattering function was also investigated and is shown in Figure 3.13, for both crystalline and noncrystalline cellulose. As expected the core chains exhibit less decay than the

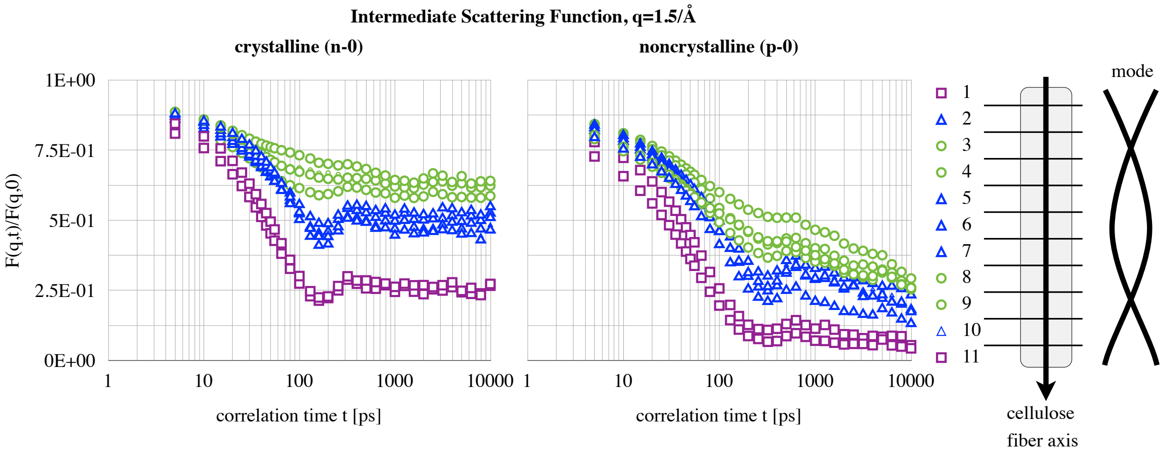


Figure 3.12.: Spatially decomposed intermediate scattering function of crystalline and noncrystalline cellulose for $q = 1.5$ Å^{-1}. The functions are grouped into three categories based on their spacing. The mapping onto the position on the fiber axis reveals that the decay of the scattering function corresponds to a standing wave along the fiber axis, with two knots, i.e. locations with minimal amplitude.

surface chains throughout the whole time scale. However, a significant part of the difference already exists at a correlation time of 5 ps and increases slightly for larger time scales. This indicates that the relaxation processes which distinguish surface from core cellulose chains are rooted in the sub ps time scale. Interestingly, crystalline and noncrystalline cellulose exhibit the same trend for their surface and core chains, which indicates that the processes which inhibit the relaxation within the core of the cellulose are the same for crystalline and noncrystalline cellulose.

The fast time scale dynamics for relaxation times up to 10 ps is shown in Figure 3.14 and is based on a trajectory with a total of 100 ps and a 1 fs sampling time. As in Fig. 3.13, the data is decomposed into contributions from cellulose chains at the fiber surface and the core region. The data reveals that any differences arise only above a relaxation time of 0.2 ps and the slopes become similar again for times above 2 ps. Also two distinct differences arise at about the same time scale: crystalline core cellulose chains start to deviate at about 200 fs (less relaxation) which indicates the loss or a modification of a relaxation process with respect to surface chains and any chain within noncrystalline cellulose. Another fork happens at about 300 fs when the core chains of noncrystalline and the surface chains of crystalline cellulose experience a loss of decay (change in slope). This means that the origin of the dynamical difference between surface and core cellulose chains is associated with an additional relaxation process for surface chains in the 0.2-2 ps time regime.

The origin of the split between core and surface chains for the crystalline fiber only was further clarified by investigating the fast time scale for different locations along the fiber axis, similar to Figure 3.12. The variation along the fiber axis leads to differences in the decay of the intermediate scattering function even below the 100 fs time regime, which is shown in Figure 3.15. However, significant differences between core and surface cellulose chains only arise at time scales above 100 fs (Fig. 3.16) and coincides with the fork between core and surface chains observed earlier (Fig. 3.15). This leads to the conclusion that the whole body motions experience by the cellulose fiber significantly affects the relaxation of the intermediate scattering function on all time scales, and

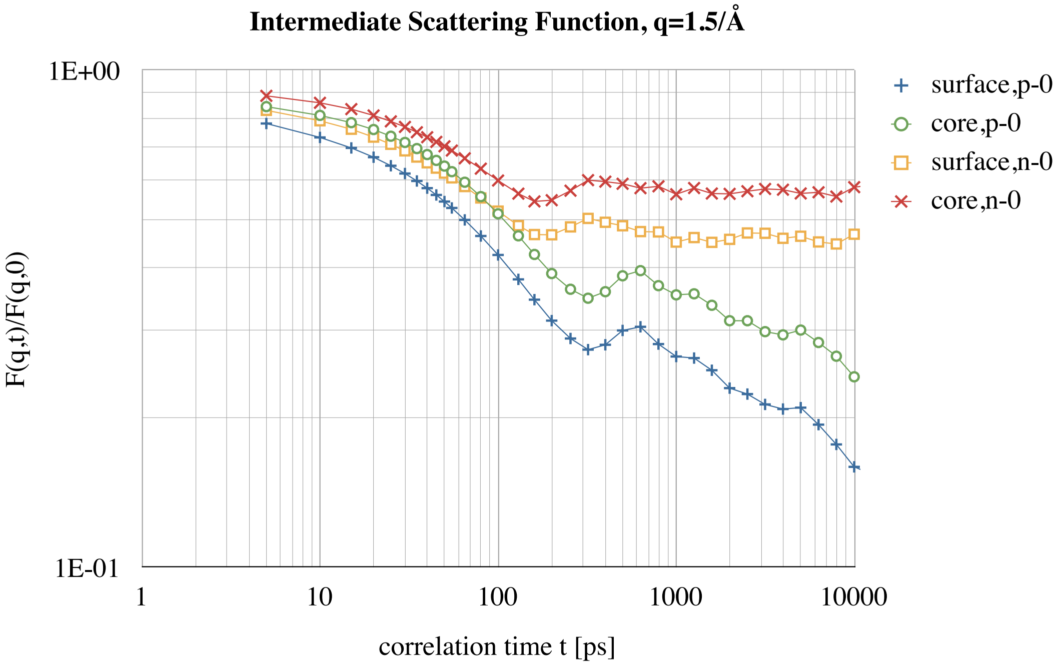


Figure 3.13.: Axially decomposed intermediate scattering function of cellulose into surface and core cellulose chains for crystalline (n-0) and noncrystalline (p-0) cellulose at $q = 1.5$ Å^{-1}.

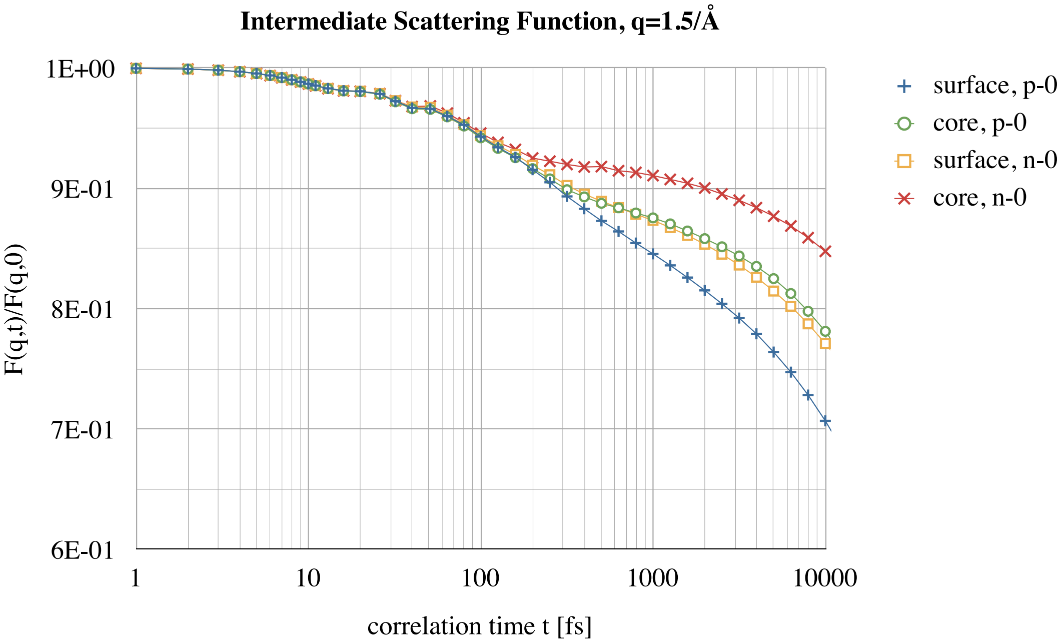


Figure 3.14.: Fast time scale intermediate scattering function for cellulose, decomposed into contributions from surface and core chains.

that the dynamical difference between surface and core cellulose chains is due to larger displacements of surface chains by global motions.

3.1.3.2. Inelastic X-ray Scattering

IXS is a scattering technique which resolves X-ray photon energies after the scattering event and allows the vibrational density of states of the sample to be deduced. In crystals the q vector dependence of the transferred energy of the X-ray photons are quantized solutions to the vibrational spectrum of the lattice and follow an anisotropic dispersion relation. The IXS signal $S(q,E)$ can be calculated as the Fourier transform of the coherent intermediate scattering function and a subsequent renormalization with the energy-frequency relationship $E = hv$, where h is Plank's constant. The characteristic function is:

$$S(q,v) = FT\left\{\langle\langle F(q,t,\tau)\rangle_t\rangle_\Omega\right\} \tag{3.2}$$

IXS experiments on aligned cellulose fiber samples were used to deduce the velocity of sound along the fiber axis [70]. However, the experimental IXS data are very noisy and preclude the identification of different phonon branches. Here the corresponding IXS signals have been calculated for crystalline and noncrystalline cellulose for three relevant directions and are shown in Figure 3.17. The IXS data along the fiber axis clearly shows the imprints of acoustic phonons around the 002 and 004 peaks. Also visible for the crystalline cellulose is an acoustic branch in the direction perpendicular the fiber axis centered around the 010 peak, albeit much weaker. The remaining spectra do not reveal any significant phonon branches. The weak and noisy dispersion along the 010 and 100

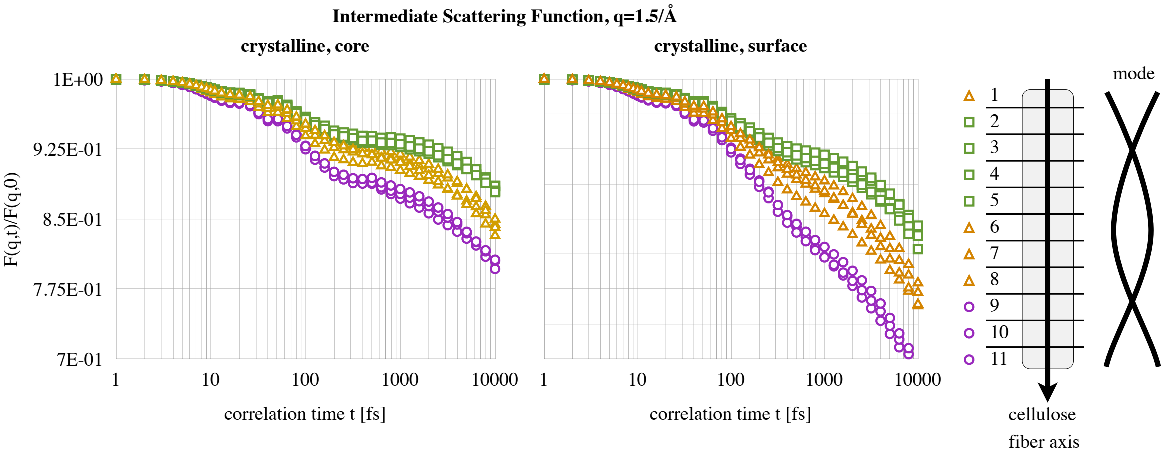


Figure 3.15.: Surface/core and spatially decomposed intermediate scattering function of crystalline cellulose.

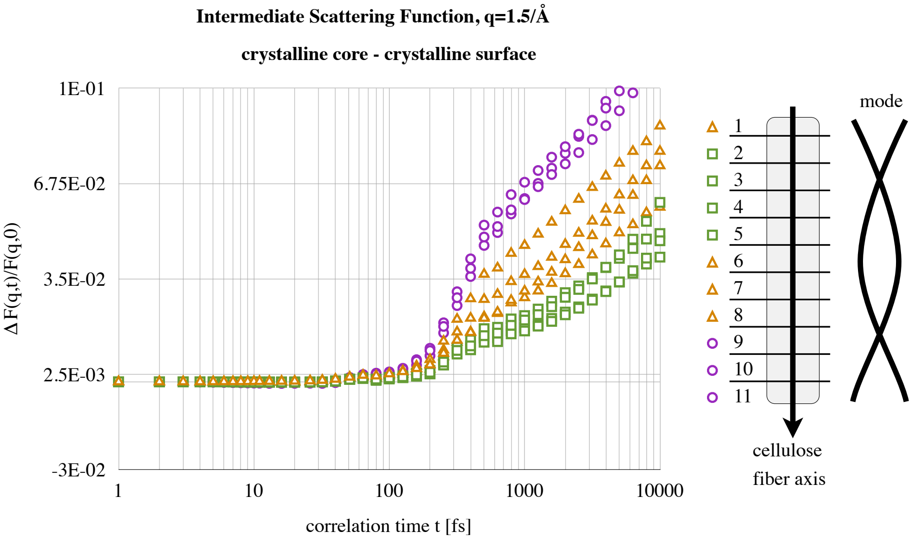


Figure 3.16.: Difference spectrum for surface/core and spatially decomposed intermediate scattering function of crystalline cellulose.

directions for crystalline cellulose may be attributed to the relatively small diameter of the cellulose fiber: the surface chains do not see an homogeneous lattice and thus experience a multitude of effective force constants which give rise to a broad spectrum of vibrational frequencies. The same argument explains the absence of any characteristic phonon branches in the 010 and 100 spectra for noncrystalline cellulose: the additional disorder perpendicular to the fiber axis gives rise to a broader spectrum of effective force constants. Also in the noncrystalline fiber the distinction between the 010 and 100 direction vanishes, which causes their corresponding IXS spectra to become similar. The phonon dispersion along the 001 direction for crystalline and noncrystalline cellulose reveal an interesting difference: The additional phonon branch located at the midpoint between 002 and 004 in crystalline cellulose is absent in noncrystalline cellulose. Also the spectra for crystalline cellulose feature a fine structure at the peak 002 and 004 positions which is absent in noncrystalline cellulose.

The identified phonon branches in cellulose can be used to infer the sound velocities and the elastic modulus along the given crystalline directions. The sound velocities can be deduced by using the dispersion relation for acoustic phonons [70]:

$$E(q) = \frac{2\hbar}{\pi} v_L q_{max} \left| \sin\left(\frac{\pi}{2}\frac{q}{q_{max}}\right)\right| \tag{3.3}$$

As illustrated in Figure 3.18, for the 001 direction the v_L can be fairly well approximated by fitting the branch with a line and using the relation $v_L = \frac{\Delta E}{\hbar \cdot \Delta q}$. For the 010 direction, however, the branch is less pronounced, which is why q_{max} and $E(q_{max}) = \frac{2\hbar}{\pi} v_L q_{max}$ are approximated independently with individual error assignments.

The crystalline and noncrystalline models both yield a $v_L = 9240 \pm 70 \frac{m}{2}$, which translates into an elasticity modulus of $149 \pm 2\, GPa$ using the relationship $G = \rho \cdot v_L^2$ and $\rho = 1.676 \frac{g}{cm^3}$ which compares with the experimentally reported values of $v_{L,exp} = 11450 \pm 1290 \frac{m}{s}$ and $G = 220 \pm 50\, GPa$ [70]. Additionally, the crystalline model features a less pronounced phonon branch in the low energy region at the midpoint between the 002 and 004 peak, which is estimated here to $v_L = 3390 \pm 420 \frac{m}{s}$ and $G = 19 \pm 3\, GPa$. The branch along the 010 direction for crystalline cellulose is estimated to $v_L = 2900 \pm 500 \frac{m}{s}$ and $G = 14 \pm 3\, GPa$, which has been reported experimentally to lie at $v_L = 2973 \pm 85 \frac{m}{s}$ and $G = 14.8 \pm 0.8\, GPa$.

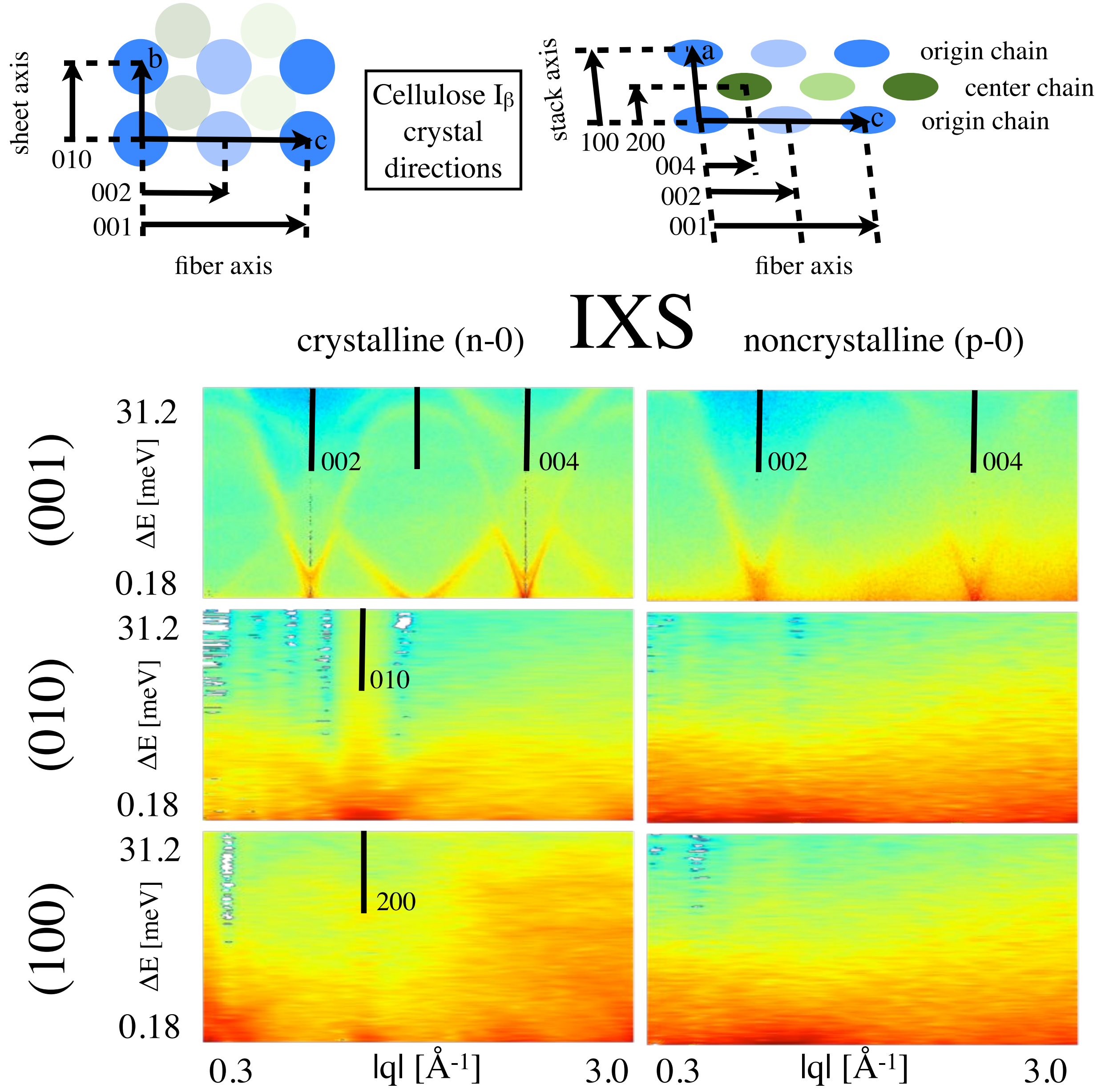
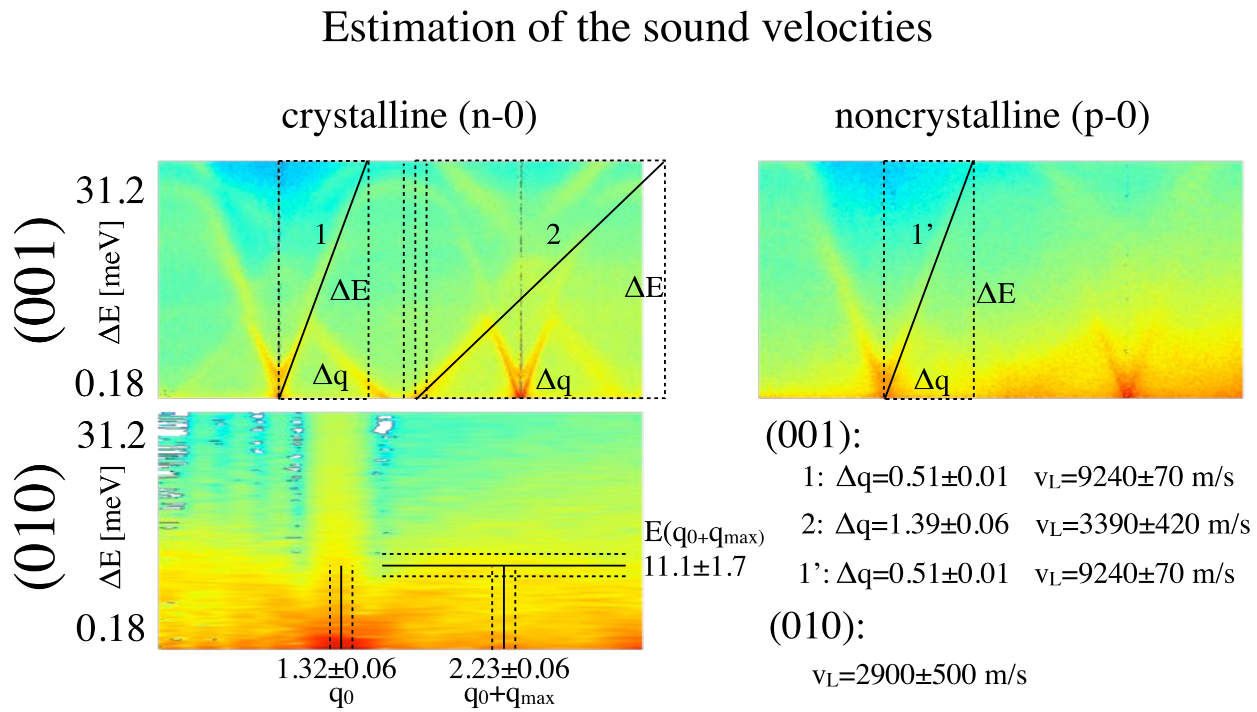


Figure 3.17.: IXS spectrum of crystalline and noncrystalline cellulose. All the significant Bragg peak locations are marked in the spectrum and illustrated in terms of cellulose crystal planes. The three chosen directions correspond to the directions of the unit cell vectors. The presence of strong interactions leads to phonon-like dispersion relations, which indicate the existence of under-damped lattice vibrations within the cellulose fiber.



(001):

1: $\Delta q=0.51\pm0.01$ $v_L=9240\pm70$ m/s

2: $\Delta q=1.39\pm0.06$ $v_L=3390\pm420$ m/s

1’: $\Delta q=0.51\pm0.01$ $v_L=9240\pm70$ m/s

(010):

$v_L=2900\pm500$ m/s

Figure 3.18.: Sound velocity estimation from dispersion in IXS spectrum.

3.2. Lignocellulose

The previous section provided extensive information on cellulose which is the main component of plant cells and the target substrate for biomass to biofuel conversion. However, in most terrestrial plants, cellulose is incorporated into a network of other biopolymers, which mainly comprise hemicellulose and lignin [58, 71]. To achieve efficient conversion of biomass towards simple sugars and the subsequent conversion into ethanol, the composite of cellulose, hemicellulose and lignin has to be modified to provide cellolytic enzymes increased accessibility to cellulose [62]. Different pretreatment strategies have been explored so far, each having a distinct effect on the molecular structure and arrangement of the various components [72]. For example, dilute acid pretreatment leads to the breakdown of the hemicellulose, a reduction of lignin content and enforces a redeposition of the fragments, while ammonia fiber expansion only leads to redeposition. Dilute acid pretreatment leads to an observed increase in cellulose crystallinity, while ammonia fiber expansion leads to a decrease [64].

Recalcitrance is strongly influenced by the physicochemical properties of lignocellulosic biomass [73]. However, the complexity of biomass has so far precluded detailed experimental characterization of the interaction of lignin with itself and cellulose at the molecular scale, and, in particular, the effect of cellulose crystallinity on lignin reprecipitation is not known.

Atomic-detail information on molecular processes in model biomass systems can be provided by MD simulation. Previous MD work have studied the structure and dynamics of cellulosic oligomers, whole cellulose fibrils [74, 75, 76, 77], and lignin [78]. Early MD studies of cellulose-lignin association examined the binding of 10- and 20-unit guiacyl oligomers to cellulose [79, 80], and these studies found that the adsorption of lignin onto different surfaces of a crystalline cellulose model yielded similar interaction energies. However, to study the aggregation processes of a statistically significant number of lignin molecules with a cellulose fiber the required time and length scales are considerable and have only recently been rendered attainable for all-atom MD simulation, facilitated by the establishment of peta-scale supercomputing facilities and the formulation of specialized parallelization strategies [56].

3.2.1. Modeling of Lignocellulose

Due to the heterogenous nature of lignin the construction of a realistic model for lignin reprecipitation onto cellulose requires a large set of individual lignin molecules. Lignin clusters found experimentally range in size from 5 nm up to 10 μm [81, 82], imposing a limit on the minimum size of the simulation box. Also, the degree of polymerization for cellulose is typically very large, ranging from 100 to 10000 [83]. The size of the system determines the time scale simulated, since lignin molecules must be able to translate within the simulation box, and the distance they travel is inherently diffusion limited. For a spherical particle with a radius of 2 nm, the self diffusion constant provided by the Stokes-Einstein equation is on the order of $10^{-6} cm^2 s^{-1}$, which results in an estimated translational diffusion of 17 nm in 500 ns. This determines the approximate minimum simulation time scale required to study lignin precipitation since lignin molecules must undergo multiple binding events with cellulose and other lignin molecules. Each lignin molecule modeled consists of 61 monomers, corresponding to a molecular weight of 13 kDa, i.e., within the experimentally determined range [84]. The monomer chemical composition and linkage were also obtained from experiment [85]. The cellulose fiber model consists of 36 chains with a chain length of 160 monomers and is depicted in Figure 3.19A. The atomic starting coordinates for the cellulose are based on the I-β crystal phase, derived from crystallographic studies [59], and the consensus model for the fiber [61]. The noncrystalline cellulose model was generated by simulating crystalline cellulose at a high temperature (650 K) for 1 ns, which is well above the melting point for the internal hydrogen bonding network. The crystallinity of the cellulose was assessed by computing the 1D and 2D Bragg scattering diagrams [19] and is shown in Figs. 3.19B and 3.19C. The fully crystalline model exhibits strong Bragg peaks typical for I-β, which are absent in the noncrystalline model.

Three models were simulated, each consisting of a cellulose fibril surrounded by 52 softwood lignin molecules in explicit water. The differences between the models lie in the initial random lignin placement and the crystallinity of the cellulose fiber and are visualized in Figure 3.20. The following nomenclature is used:

Model NC places the lignin molecules initially "near", i.e. 2-10 Å, from a crystalline fiber

Model FC places the lignin molecules initially "far", i.e. 4-20 Å, from a crystalline fiber

Model FN places the lignin molecules initially "far", i.e. 4-20 Å, from a noncrystalline fiber

3.2.2. Preferential Association of Lignin with Crystalline Cellulose

All lignocellulose simulation models showed the same overall behavior involving lignin molecules aggregating with each other and binding to the cellulose fiber. The precipitation of lignin onto cellulose and lignin self aggregation can be quantified by the time-evolution of the total number of atomic contacts, N, for each of the 3 simulation models. N exhibited strong linear correlation with the buried surface area (data not shown, but provided in the corresponding manuscript), and therefore these quantities can be interchanged without affecting the conclusions.

3.2.2.1. Cellulose-Lignin Aggregation

In all simulations the number of cellulose-lignin contacts, *N(CL)* rises sharply over the first 100 ns before converging by the end of the simulations (Fig. 3.21A), with a variation of less than 20% in the second half of the simulation. *N(CL)* is clearly different for each model throughout the whole simulation time. For the NC model the near placement of the lignin molecules leads to a larger *N(CL)* at all times of the simulation than in the "far" models, in which the lignin molecules were initially placed further away from the fiber. Interestingly, a significantly higher number of cellulose-lignin contacts is seen for FC than FN, even though the lignin molecules were initially placed in a similar manner around the fiber. Thus, Fig. 3.21A indicates that two factors affect the proportion of lignin reprecipitation onto cellulose: the average initial distance from the cellulose and the cellulose crystallinity. To obtain a thermodynamic understanding of the differences in lignin reprecipitation onto crystalline and noncrystalline cellulose, the cellulose-lignin and lignin-lignin interaction energy densities, ϵ, were calculated. ϵ is derived by dividing the interaction energy, E, by B, the buried surface area between cellulose-lignin or lignin-lignin. B and E exhibit strong linear correlation (data not shown). No significant dependence of the cellulose-lignin interaction energy on cellulose crystallinity is found (Table 3.4). This is consistent with earlier findings suggesting that the lignin monomer binding interaction energy is independent of the molecular structure of cellulose [80, 79]. As a next step, solvation was investigated as a possible origin of the difference between the FC and FN models. To characterize the solute-water interactions, separate simulations were performed in explicit water of individual lignin molecules and of a crystalline and a noncrystalline cellulose fiber. The solute-water interaction energy was then calculated as the interaction energy per unit solvent accessible surface area $\epsilon = E/S$.

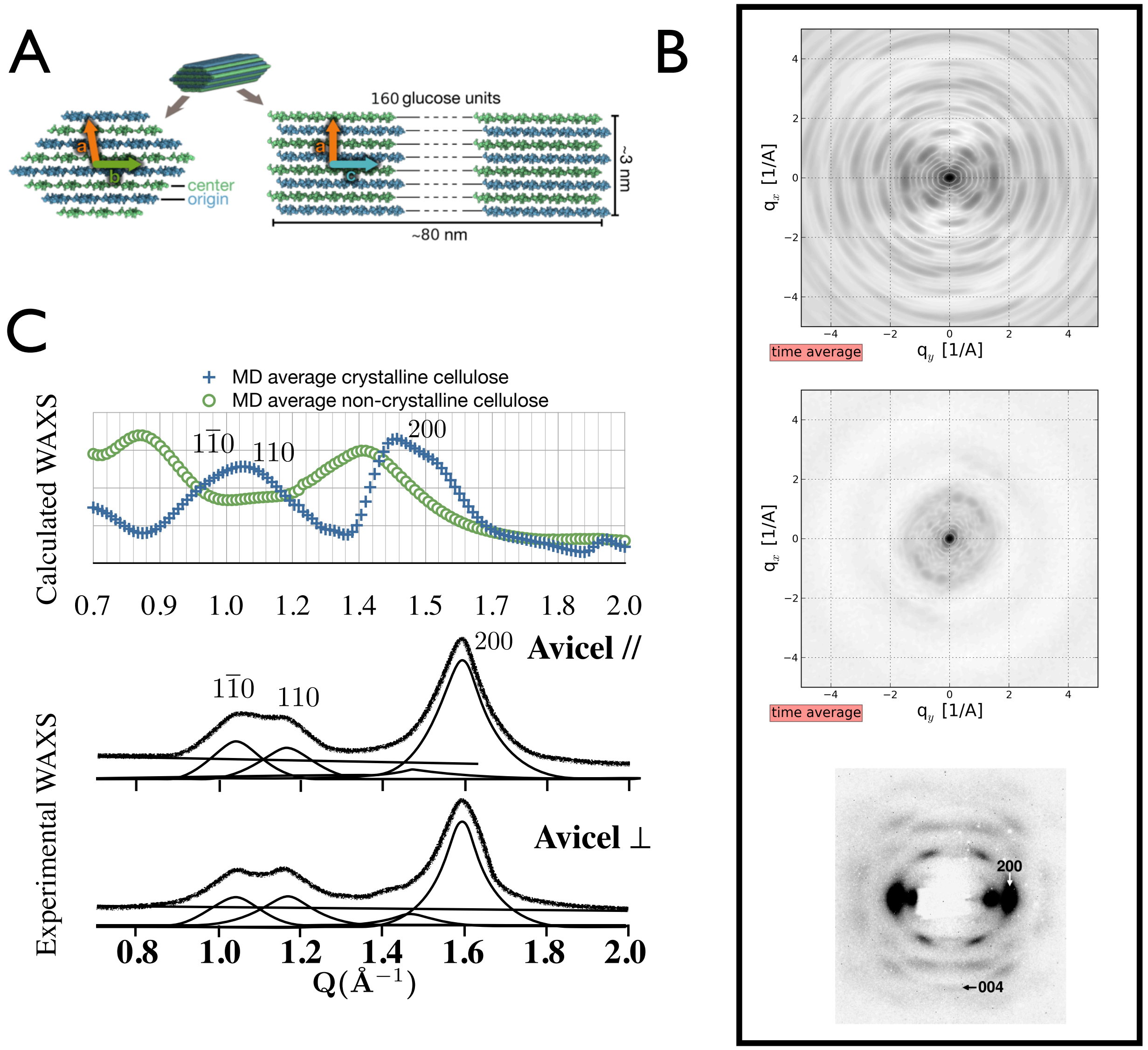


Figure 3.19.: A) Cross-sections of the $I\beta$ crystalline cellulose model. Left: perpendicular to the fiber axis, indicating the crystal surfaces in contact with the solvent. Right: along the fiber axis, indicating the fiber thickness and length. B) Top and Middle: 2D directional X-ray scattering patterns for the crystalline and noncrystalline cellulose models (based on the average of 1000 snapshots from a 20 *ns* MD simulation). Bottom: experimental X-ray scattering diagram from Ref. [70]. The Bragg peaks are a characteristic feature of the translational symmetry in crystalline cellulose perpendicular to the fiber axis, and are absent for noncrystalline cellulose. C) 1D spherically-averaged X-ray scattering profile for crystalline and noncrystalline cellulose simulations. The noncrystalline cellulose has residual peaks shifted away from the characteristic scattering peaks for crystalline cellulose. As a comparison the experimental WAXS is also shown for Avicel type cellulose from Ref. [86]. The calculated WAXS signal features peaks which are systematically shifted to lower q values when compared to the experimental one.

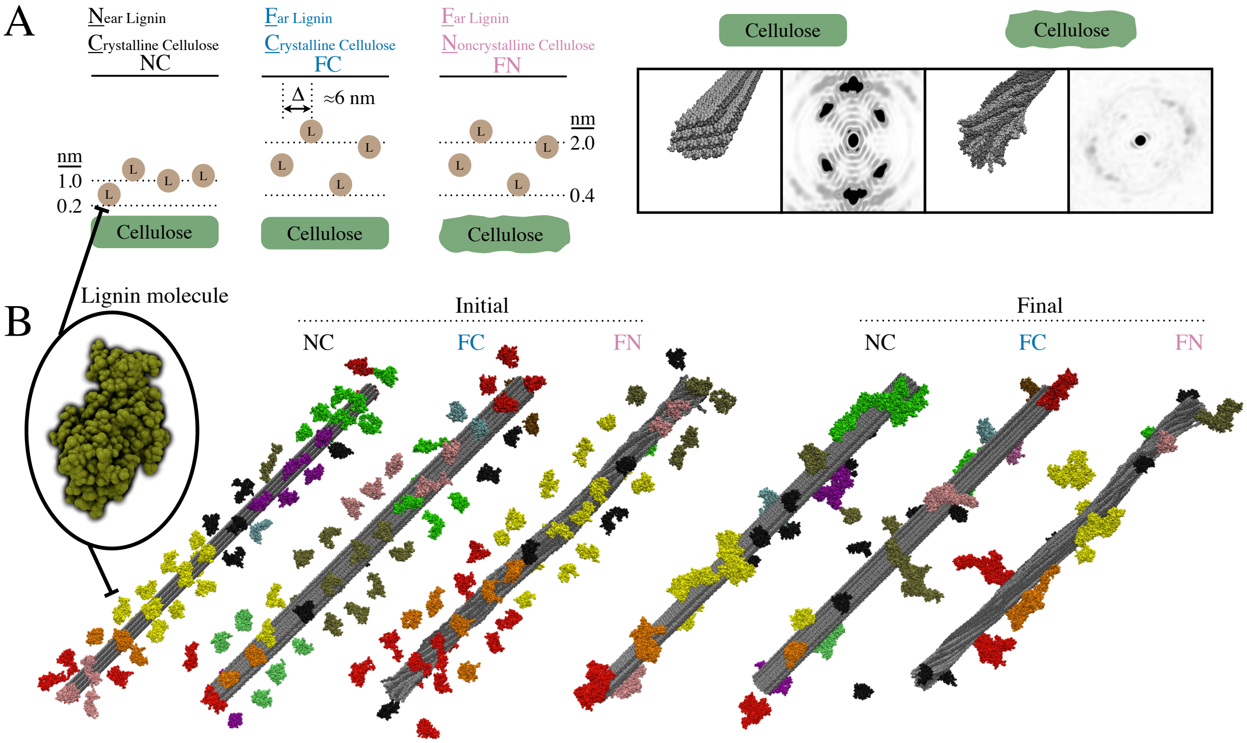


Figure 3.20.: A) Schematic of the initial configuration of the simulations. For each model 52 lignin molecules were placed around the central fiber at distances randomly chosen within the interval of 0.2 to 1.0 *nm* and 0.4 to 2.0 *nm* for the NC model and the FC and FN models, respectively. The directional X-ray scattering diagram along the fiber axis is shown for both the crystalline and noncrystalline cellulose, with the crystalline cellulose exhibiting strong Bragg peaks which are absent in the noncrystalline model. B) Visual representation of the initial and the final configurations of the systems. Lignin molecules belonging to the same cluster at the end of the simulation are colored identically. The lignin molecules participating in any given cluster tend to be in local proximity at the beginning of the simulation.

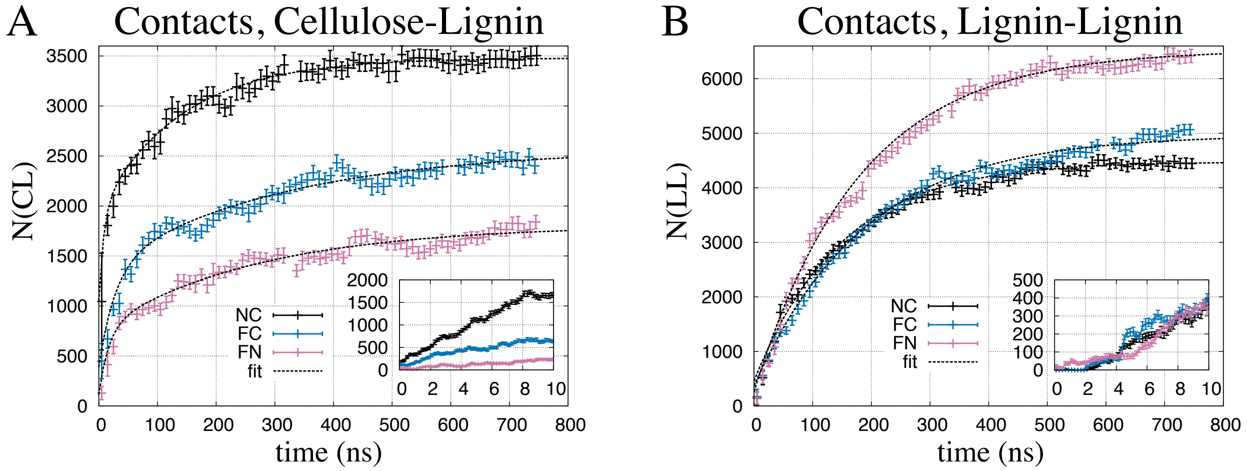


Figure 3.21.: A) Total number of cellulose-lignin atomic contacts, *N(CL)*. B) Total number of lignin-lignin atomic contacts, *N(LL)*. For both graphs the inset shows the first 10 ns of each simulation.

Lignocellulose interfacial interaction constants:

Interface	Model	Interaction Energy Densities		
		ϵ	ϵ_{Coul}	ϵ_{LJ}
Cellulose/Lignin	NC	-47.65	-20.43	-27.22
	FC	-51.80	-22.67	-29.13
	FN	-50.33	-24.10	-26.24
Lignin/Lignin	NC	-52.66	-22.19	-30.47
	FC	-52.95	-22.24	-30.72
	FN	-52.02	-21.54	-30.48

Table 3.4.: Lignocellulose interfacial interaction constants, ϵ, decomposed into coulombic, ϵ_{Coul}, and van der Waals contribution, ϵ_{LJ}. Energies are $kJ/mol/nm^2$. The assumed error is ±2.0 and ±0.67 for Cellulose/Lignin and Lignin/Lignin, which is derived from the difference of the values between the two simulations of the NC model (second NC simulation: $\epsilon(CL) = -50.46\,kJ/mol/nm^2$ and $\epsilon(LL) = -51.71\,kJ/mol/nm^2$). The true error of ϵ_{total} depends on the variations in the possible aggregation pathways for each model.

Solvent:biomass interaction constants:

Solute	Solvent Interaction Energy Densities		
	ϵ	ϵ_{Coul}	ϵ_{LJ}
Lignin	-74.1 ± 9.5	-63.0 ± 8.5	-11.2 ± 1.1
Crystalline Cellulose	-94.0 ± 1.7	-83.0 ± 1.8	-10.0 ± 0.2
Noncrystalline Cellulose	-107.3 ± 0.7	-102.0 ± 0.7	-5.0 ± 0.2

Table 3.5.: Solvent:biomass interaction constants, ϵ, decomposed into coulombic, ϵ_{Coul}, and van der Waals contribution, ϵ_{LJ}. Energies are $kJ/mol/nm^2$. Water-solute constants are based on the simulation of isolated molecules. The standard deviation for water-lignin reflects the spread across all of the 52 lignin molecules.

Table 3.5 shows that the enthalpic interaction per unit area between cellulose and water is significantly larger for the noncrystalline than for the crystalline form. The difference in interaction energy density was traced back to differences in the capacity of the cellulose to hydrogen bond with water, which increases from $3.2 \pm 0.1/nm^2$ for crystalline to $3.7 \pm 0.1/nm^2$ for noncrystalline cellulose. It was also found that, although the average number of hydrogen bonds made by cellulose (internal+solvent) per glucose molecule is very similar for the two models, with about 3.4 ± 0.1 for crystalline and 3.6 ± 0.1 for noncrystalline cellulose, the ratio of internal to solvent is markedly different, being 68% to 32% for the crystalline form compared to 47% to 53% for noncrystalline cellulose. The increased interaction energy between noncrystalline cellulose and water suggests that noncrystalline cellulose is effectively more hydrophilic than the crystalline form, and explains why the FN model exhibits less lignin precipitation on to cellulose than the FC model (Figure 3.21A). The lignin-water interaction energy densities are broadly distributed, ranging between -58 and $-95\, kJ/mol/nm^2$, and also were found to exhibit a strong correlation with the number of lignin-water hydrogen bonds but only a weak correlation with the total hydrophobic fraction of the surface area (Fig. 3.22). An example of two lignin molecules with the same solvent accessible surface area and chemical topology but a very different number of hydrogen bonds with water and thus lignin-water interaction energy density is also given in Fig. 3.22. In contrast to cellulose, the internal and solvent hydrogen bonds of lignin do not systematically compensate, and thus lignin molecules exhibit a varying degree of unsatisfied hydrogen bonding groups.

3.2.2.2. Lignin Aggregation

During the initial phase of lignin self aggregation in the simulations, lignin molecules bind to macromolecules in their local vicinity. As is apparent from Figure 3.21B, during the first 50 ns all models show a similar amount of lignin-lignin aggregation, which is a direct consequence of the models having similar initial distances between the lignin molecules. However, after 50 ns the FN model diverges from NC and FC, exhibiting a higher degree of lignin self aggregation. Hence, noncrystalline cellulose also promotes lignin self aggregation. To further quantify the self aggregation effect, the lignin aggregation process was decomposed into a discrete state model, in which each state represents a distinct arrangement of molecular contacts. Molecular contacts are defined here as two molecules having at least one atomic contact, and the state of a lignin molecule is then defined by the number of molecular contacts formed. The states were labeled L for isolated lignin, LX for a lignin molecule in contact with X-1 other lignin molecules and a leading C, if the lignin forms an additional contact with cellulose. The model and the time-dependent population for all states are provided in Fig. 3.23. At 50 ns, FN shows a striking increase in the L3 and L4 states, indicating the formation of larger lignin clusters without cellulose contact, which coincides with a strongly reduced population of the CL2 state, which corresponds to two lignin molecules bound to each other and to cellulose. Hence, the reduced affinity of noncrystalline cellulose for lignin during the initial 50 ns leads to a long-term increase in the lignin self aggregation.

3.2.2.3. Metastability and Stickiness

Since NC and FC were simulated with the same structural models for cellulose and lignin, the quantitative difference in *N(CL)* between NC and FC provided in Fig. 3.21A indicates that the simulations have reached different metastable states. One reason for this metastability is the "stickiness" of lignin for both cellulose and lignin, i.e. the persistent binding of lignin to cellulose or another lignin molecule. We define the stickiness, *S(t)*, as the conditional probability of finding a lignin molecule bound to another molecule at time *t*, given that the two molecules formed a contact

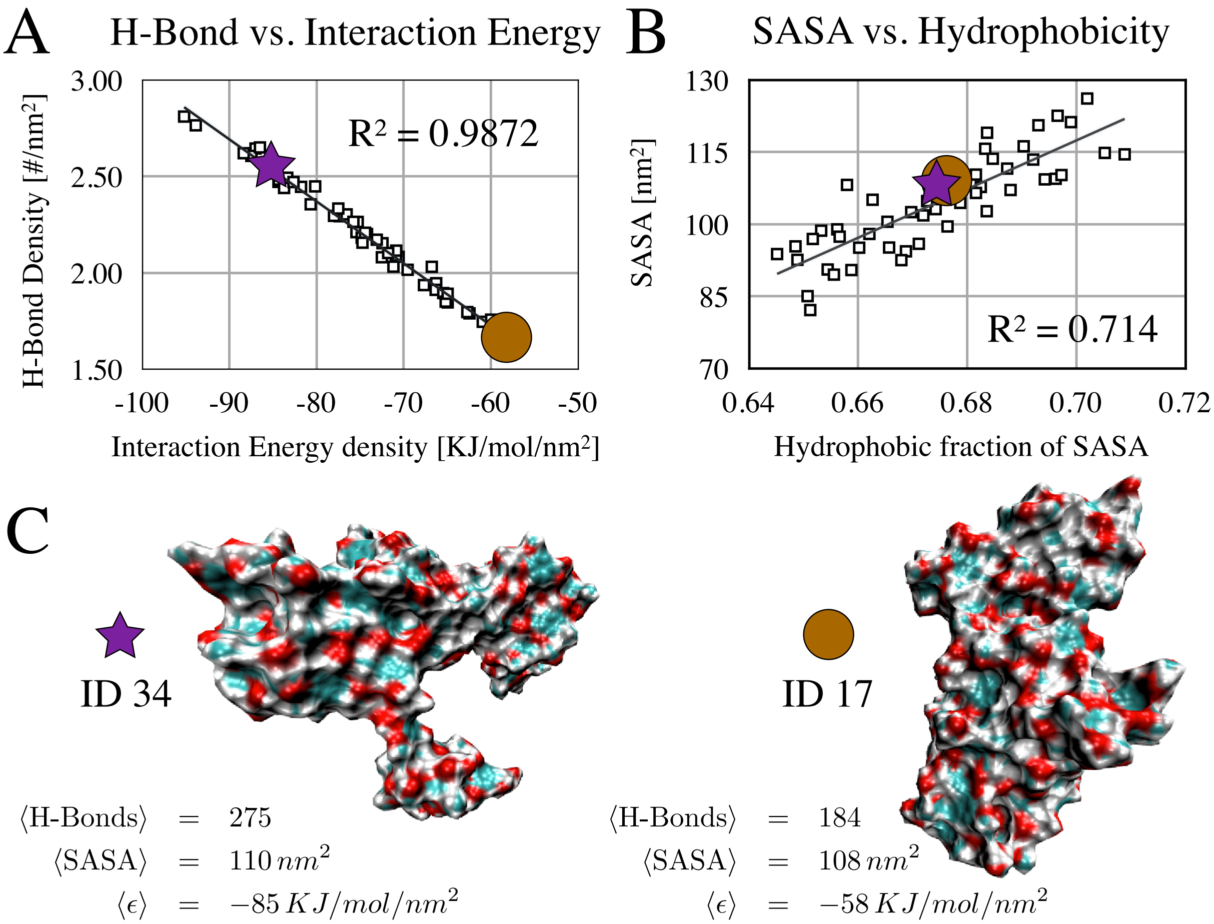


Figure 3.22.: A) The total *SASA* in relation to the fraction of hydrophobic for each lignin molecule. The linear regression yields a correlation coefficient of $R^2 = 0.714$. This suggests that the fraction of hydrophobic atoms at the surface is size dependent. B) The lignin-water hydrogen bond density, which is the number of hydrogen bonds divided by the SASA, in relation to the interaction energy density for each lignin molecule. The strong correlation with a coefficient of $R^2 = 0.9872$ suggests that the interaction energy density is governed by the hydrogen bonding capacity. C) The molecular configuration for two lignin molecules with very different interaction energy densities. Both lignin molecules have approximately the same SASA. However, ID 34 has a significantly larger hydrogen bonding capacity with water.

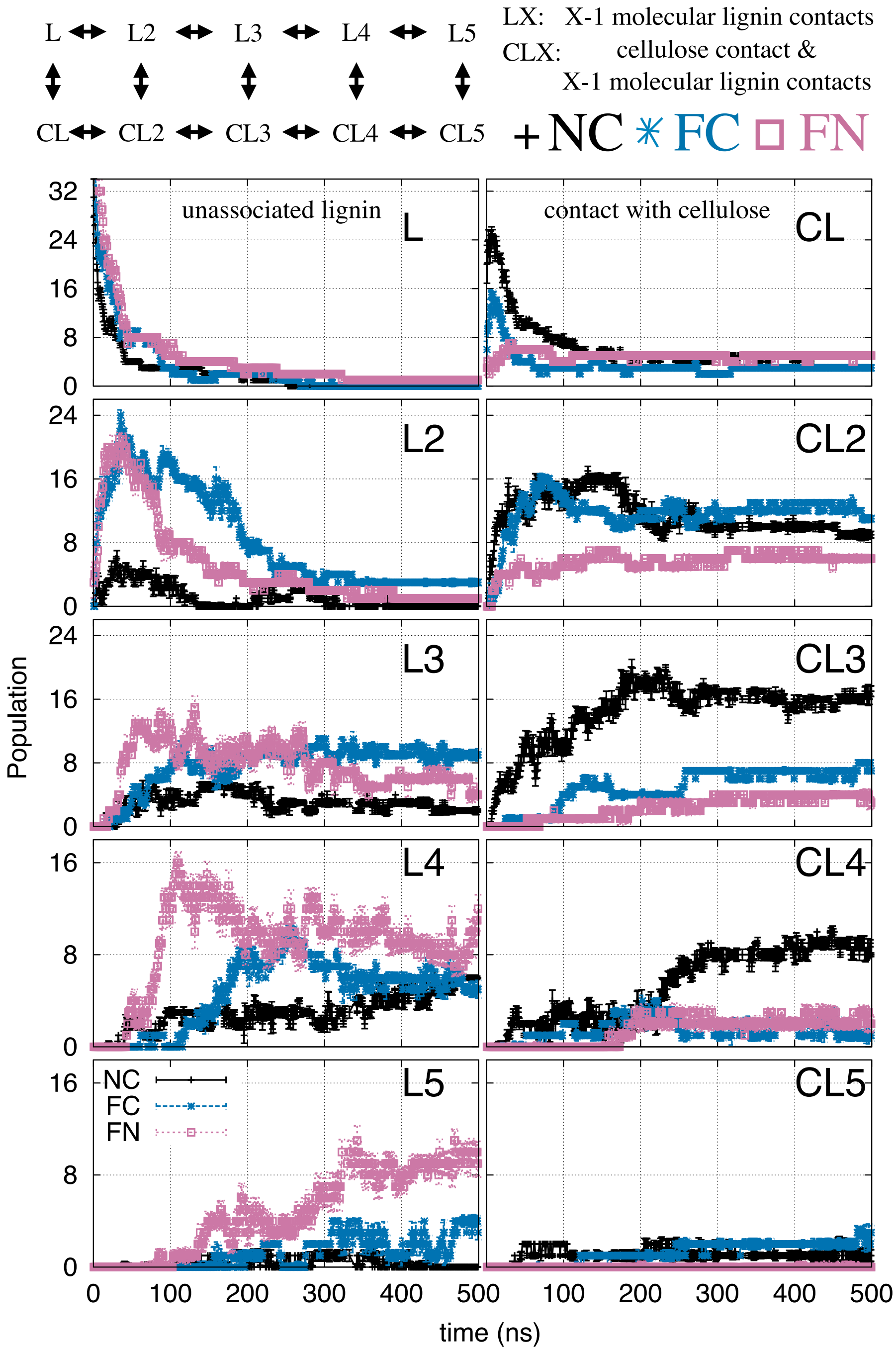


Figure 3.23.: Time dependence of the populations of the molecular lignin coordination number, shown for each model and state.

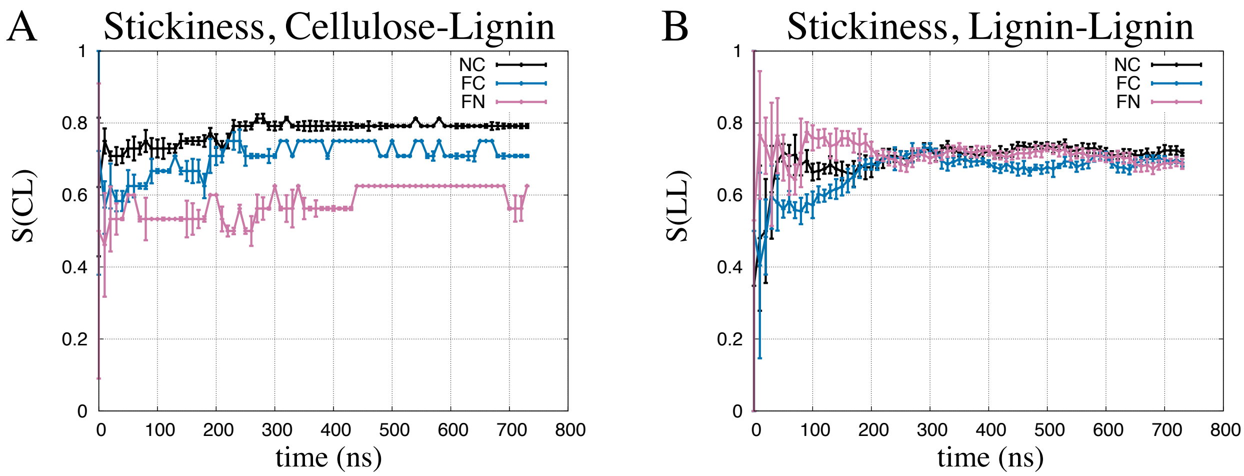


Figure 3.24.: A) Time dependence of the stickiness for lignin-lignin and B) cellulose-lignin molecular contacts. The cellulose-lignin stickiness shown only contains lignin molecules which enter the CL state first, i.e. bind to cellulose before to any other lignin, which emphasizes the intrinsic binding propensity of lignin with cellulose.

at time $\leq t$:

$$S_{LL}(t) = \langle\, P\,(\,N\,(L_i L_j, t) > 0 \mid N\,(L_i L_j, t_0) > 0, t_0 \leq t)\,\rangle_{ij} \tag{3.4}$$

$$S_{CL}(t) = \langle\, P\,(\,N\,(CL_i, t) > 0 \mid N\,(CL_i, t_0) > 0, t_0 \leq t)\,\rangle_{i} \tag{3.5}$$

Cellulose-lignin and lignin-lignin stickiness, $S_{CL}(t)$ and $S_{LL}(t)$ respectively, are shown in Figure 3.24. $S_{LL}(t)$ converges to 0.70 ± 0.05 for all models within the first 200 ns, which means that on average each lignin molecule retains 70% of the molecular contacts it initiates with other lignins. The association of a given lignin molecule with cellulose is influenced by the intrinsic propensity of the two molecules to bind in aqueous solution and also by extrinsic factors, such as the presence of other lignins that may block access to the cellulose. To probe the intrinsic factors contributing to stickiness of lignin on cellulose, $S_{CL}(t)$ was computed for the subset of lignin molecules which enter the CL state first, i.e. bind first to cellulose before binding to other lignins, which makes the derived $S_{CL}(t)$ less sensitive to competing binding of lignin to cellulose. As shown in Fig. 3.24A, $S_{CL}(t)$ converges to 0.80 ± 0.01, 0.75 ± 0.05, and 0.67 ± 0.01 for NC, FC, and FN, respectively. The difference in $S_{CL}(t)$ between the crystalline and noncrystalline cellulose models is consistent with the overall reduced lignin:noncrystalline cellulose association. Furthermore, the large values of $S_{CL}(t)$ are quantitatively indicative of the high degree of lignin stickiness for cellulose and explains why the initial conditions in the simulation have a significant effect on the morphology of the final metastable states.

3.2.2.4. Morphology

The impact of lignin stickiness and cellulose crystallinity on morphological features of the cellulose-lignin complexes is apparent in the molecular configurations visualized in Figure 3.20B, which contains 2D projections of the models at the start and end of each simulation[*]. The morphology of the cellulose-lignin aggregation was investigated by grouping individual lignin molecules either directly or indirectly connected through other lignin molecules into lignin clusters, and then

[*] Additional visualizations online: http:\\cmb.ornl.gov\material\lignocellulose-reprecipitation.

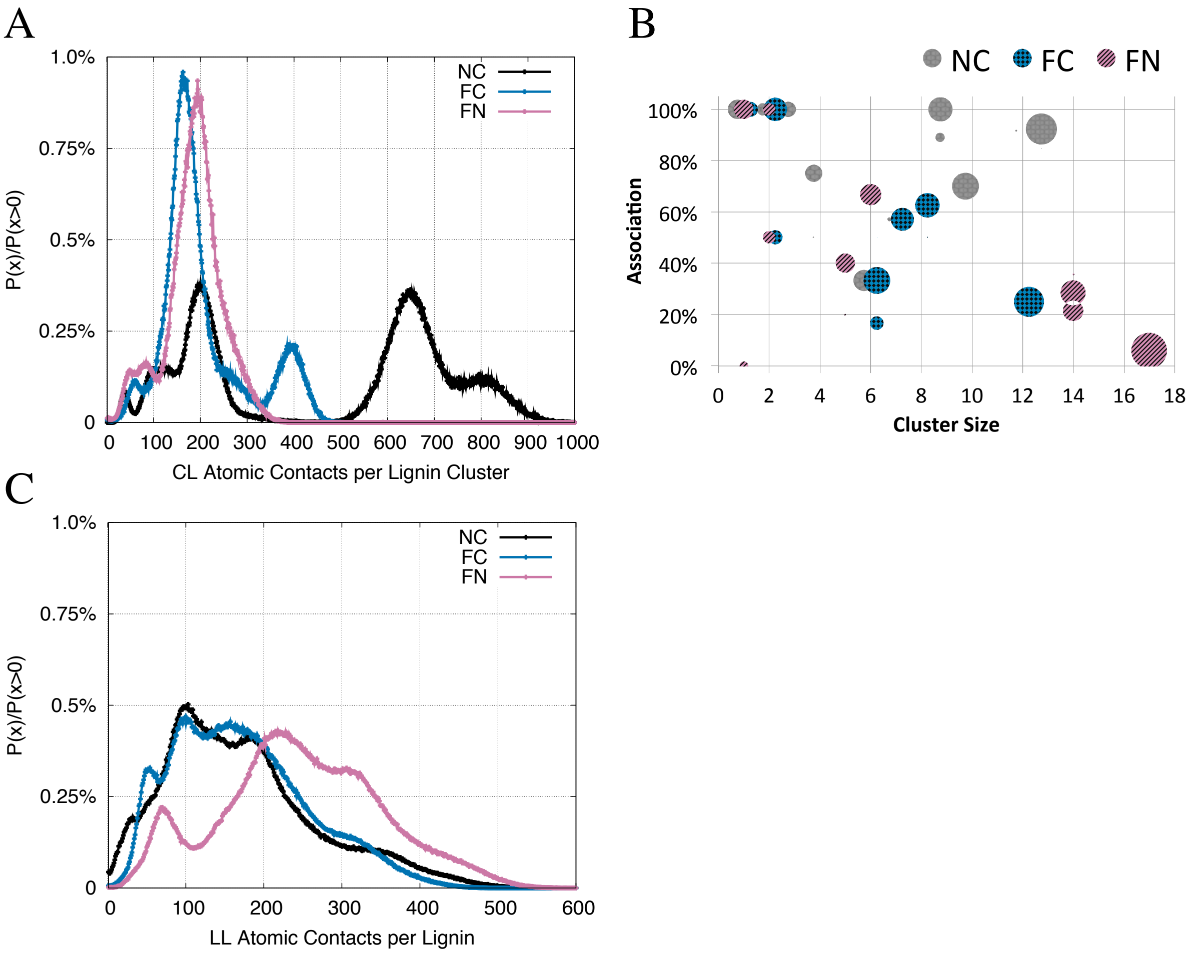


Figure 3.25.: A) Probability distributions of atomic contacts between cellulose and lignin based on cluster morphology, averaged over 50k frames (250 ns to 500 ns, minor structural rearrangements with a variation in total atomic contacts of less than 20%). B) Bubble Diagram of the Probability (Size of Bubbles) of lignin clusters of a given size (x axis) and their cellulose association (y axis), quantified by the fraction of lignin molecules in contact with cellulose. Cluster sizes of 1 mark lignin molecules in the L or CL state. FN has large clusters with low association, while NC results in medium sized clusters with high cellulose association. For visualization, the data for NC was shifted by 0.25 on the x axis to the lower values and FC was shifted 0.25 to higher values. C) Probability distributions of atomic contacts between lignin molecules with surrounding lignin, averaged over 50k frames (250 ns to 500 ns).

quantifying the degree of association of individual lignin clusters with cellulose. Figure 3.25A shows the probability of a random lignin molecule participating in a cluster forming $N(CL)$ contacts with cellulose. Due to lignin stickiness, the NC model, in which the lignin molecules are placed closer to the cellulose, contains lignin clusters which are highly associated with cellulose. The FC and FN models exhibit similar binding characteristics for clusters with $N(CL) \leq 300$. However, the maximum number of contacts for FN (280) is less than the maximum for FC (400). Similar information can be retrieved from Figure 3.25B, which shows the probability of a lignin cluster plotted against the cluster size and the degree of association with cellulose (the number of molecular lignin-cellulose contacts), and reveals, for example, that the FN model contains a large lignin cluster (17 molecules) with hardly any cellulose association (<10%). Differences in the morphology of lignin self aggregation can be seen from Figure 3.25C, which shows the number of atomic contacts made by each individual lignin molecule with its surrounding lignin molecules. While this number is very similar for NC and FC, for FN it is depleted in the range between 100 and 200 atomic contacts and significantly higher above 200, again consistent with the observation that the FN model exhibits larger lignin clusters with an increased number of molecular lignin-lignin contacts.

3.2.3. Modeling of a 24 Million Atom System

The investigation of the effect of cellulose crystallinity onto cellulose-lignin aggregation represents the first step in a series of investigations to probe and understand the complexity of biomass on the molecular level using simulation techniques. At the time of writing subsequent simulations have been performed, which will provide information on the effective interaction of the cellolytic enzymes Cel7A with crystalline and noncrystalline cellulose in a lignin environment. Here the converged lignocellulose structures act as building blocks for a cellulase-cellulose-lignin model, which comprises 24 million atoms in total using explicit water. Figure 3.26 provides a rendering of the 24 million atom system, using different color for cellulase, cellulose and lignin molecules. The model is comprised of 9 cellulose fibers, 9x52 lignin molecules and 54 Cel7A proteins.

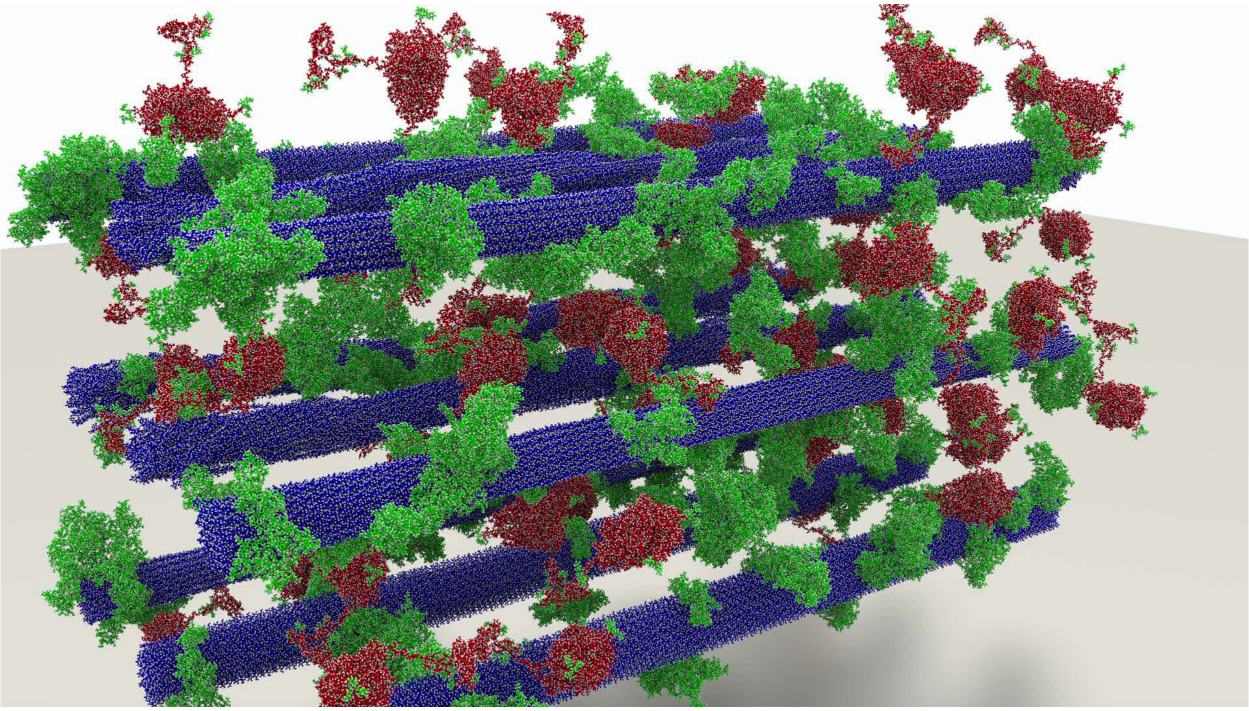


Figure 3.26.: Rendering of the 3x3 lignocellulose model derived from the 3 individual lignocellulose simulations, combined with the cellolytic enzyme Cel7A. The solvated system contains up to 24 million atoms.

3.3. Alanine Dipeptide

Achieving large time and length scales for molecular dynamics simulation using supercomputers is not the only way of pushing the envelope in understanding the dynamics of molecular systems. Another active field of research is the application of Markov Chain Modeling towards the analysis of the configurational dynamics of molecules: Markov Chain Models serve as a rigorous statistical tool to characterize and classify the molecular configurations and calculate the transition probabilities between them. Of particular interest is the application of Markov models in the analysis and interpretation of experimental data, e.g. for fluorescent microscopy [16], where it has been shown that the transition times derived from a Markov model analysis of a simulated alanine dipeptide, correspond well to the experimentally measured relaxation times.

Within the scope of this work, the application of Markov model analysis was extended towards the prediction and interpretation of dynamic neutron scattering spectra. It is shown that, with a properly chosen Markov model, the complex scattering profile of alanine dipeptide can be easily understood as the superposition of simultaneous relaxation processes, with distinct relaxation times and scattering intensities. The complete treatment of the theory and its application towards alanine dipeptide has been covered in a separate manuscript[†]. The necessary theoretical foundations are discussed in section 2.3.

3.3.1. Molecular Structure and Dynamics

The application of Markov State Modeling (MSM) towards the computation of neutron scattering is exemplified here using the simulated conformational dynamics of alanine dipeptide (N-acetyl-alanine-N'-methylamide) (see Fig. 3.27). This molecule has long served as a subject for methodological computational studies because the flexible backbone dihedral angles, Φ $(C - N - C_\alpha - C)$ and Ψ $(N - C_\alpha - C - N)$, adopt the conformations typical for the α helix and β strand motifs in proteins [87, 88, 89, 90, 91]. Furthermore, the alanine dipeptide contains one side-chain (χ) and two terminal methyl groups (N-ter and C-ter), which provide additional degrees of freedom relevant to spectroscopic techniques sensitive to atomic displacements, in particular dynamic neutron scattering. Previous work has indicated that the N-ter and C-ter methyl groups in the molecule have low rotational barriers, i.e., ≤ 0.1 kcal/mol [92], leading to rotations on the ps time scale at 300 K. In contrast, the side-chain methyl (χ) group exhibits an intrinsic intramolecular torsional barrier of about 3 kcal/mol [93] resulting in rotational jump diffusion on the sub ns time scale. This backbone, side-chain methyl and terminal methyl dynamics makes the system a good candidate for studying conformational dynamics with the aim of integrating spectroscopic experiments with MSM.

3.3.2. Modeling with Markov States

The derivation of a Markov state model (MSM) for the configurational dynamics of alanine dipeptide requires a number of steps, which are outlined in Fig. 3.28. These steps comprise state space reduction, discretization, implied time scale analysis and kinetic clustering. Each step builds upon the choices and results of the previous steps. Although the methodology for MSM construction is well established, the main points of the procedure are briefly summarized here and relevant information and construction parameters are provided. Further details are provided exhaustively in

[†] to be published

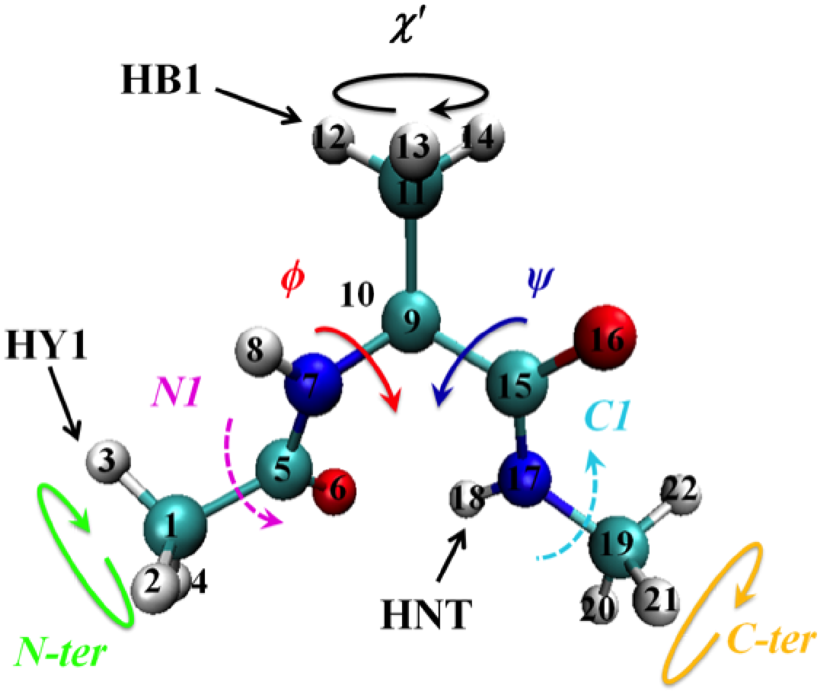


Figure 3.27.: Molecular structure of alanine dipeptide. All relevant dihedral and methyl group rotation angles are indicated.

the current literature [94, 95]. The software package EMMA[‡] was used for the discretization of the trajectory data, the implied time scale analysis and the quality assessment of the MSM.

3.3.2.1. State Space Reduction

The first step in the derivation of a Markov model is to find a number of coordinates to capture the anticipated transition processes. This commonly reduces the observed state space from 3N degrees of freedom to only a few. The maximum number of chosen degrees of freedom is limited in practice by the available sampling: more degrees lead to fewer sampling points per unique state and subsequently to larger errors in the estimated transition probabilities. The construction of a set of generalized coordinates is often guided by existing knowledge of the accessible configurations and by the ability to distinguish between them. The chosen coordinates are also referred to as state variables or reaction coordinates, due to their ability to distinguish different configurational states. An important feature of a set of state variables is that they together uniquely identify distinct configurations. In the case of alanine dipeptide, the internal dihedral rotations are a natural choice for distinguishing between different peptide configurations. In the present example, the set of rotational angles, Φ, Ψ, C1 (*CAY-CY-N-CA*), N1 (*CA-C-NT-CAT*), C-ter, N-ter, and χ methyl were selected as candidates for state variables (see Fig. 3.27), and their time series are illustrated in Fig. B.1. The values for Φ, Ψ and χ switch between distinguishable, long-lived (> 10 ps) plateaux, which means that their associated free energy profiles feature significant barriers with distinct minima. The values for the C1 and N1 peptide bonds fluctuate around $0 \pm 15°$ throughout the whole simulation of 1 μs owing to their strong rigidity, while, in contrast, the C- and N-ter methyl angles have no preferred orientation

[‡] EMMA: https://simtk.org/home/emma

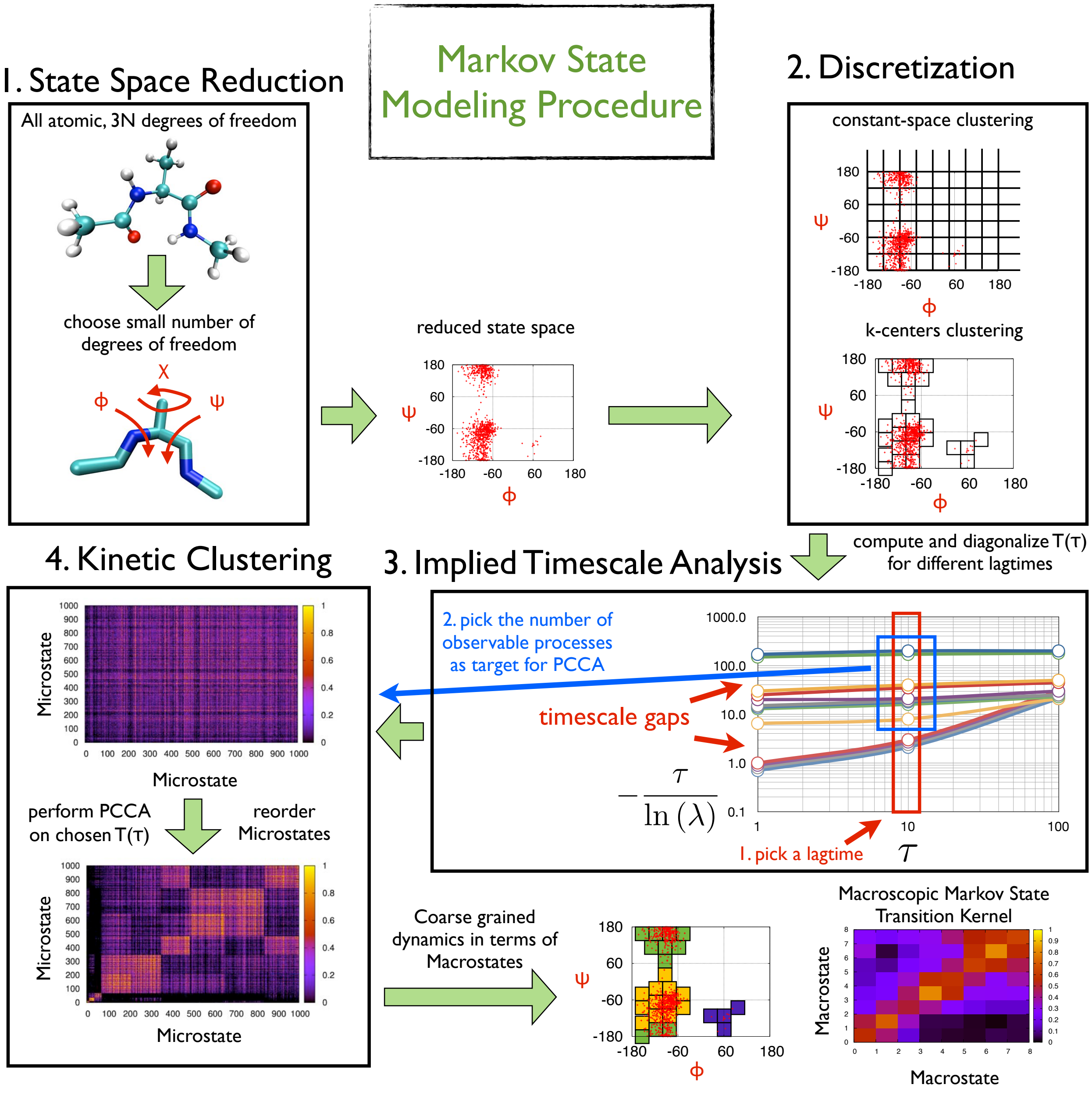

Markov State Modeling Procedure
1. State Space Reduction
All atomic, 3N degrees of freedom
choose small number of degrees of freedom
χ
ϕ
ψ
reduced state space
180
60
-60
-180
-180 -60 60 180
ψ
ϕ
2. Discretization
constant-space clustering
k-centers clustering
compute and diagonalize T(τ) for different lagtimes
3. Implied Timescale Analysis
2. pick the number of observable processes as target for PCCA
timescale gaps
1000.0
100.0
10.0
1.0
0.1
1
10
100
$-\frac{\tau}{\ln(\lambda)}$
1. pick a lagtime
τ
4. Kinetic Clustering
Microstate
Microstate
0 100 200 300 400 500 600 700 800 900 1000
0 0.2 0.4 0.6 0.8 1
perform PCCA on chosen T(τ)
reorder Microstates
Microstate
Microstate
Coarse grained dynamics in terms of Macrostates
Macroscopic Markov State Transition Kernel
Macrostate
Macrostate
0 1 2 3 4 5 6 7 8
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

Figure 3.28.: Markov modeling procedure. The modeling involves four major stages as indicated by the numbering.

and rotate freely, consistent with their low barriers ($\leq$ 0.1 kcal/mol) [92]. To further illustrate the individual time evolution of the selected angles, the corresponding NMR based angle-angle self-correlation functions $C_{NMR}(\tau) = \langle P_2 \left[\cos\left(\theta(t) - \theta(t+\tau)\right)\right]\rangle$ [96] were calculated and are plotted in Fig. B.2 P_2 is the second-order Legendre polynomial given by $P_2(x) = \frac{1}{2}\left[3x^2 - 1\right]$. The estimated relaxation time for χ is ~200 ps, for Φ and Ψ about ~50 ps, and for C1 and N1 and the C- and N-ter methyl angles ~0.1 ps and ~1 ps, respectively. Consequently, only Φ, Ψ and χ were selected as state variables for the subsequent Markov modeling.

3.3.2.2. Discretization

The three state variables, χ, Φ, and Ψ span a continuous and fully periodic three-dimensional configuration space, each point representing a distinguishable configuration of the alanine dipeptide. However, each sampling point occupies a unique location. To allow statistical analysis of the transitions within this state space, configurations which are geometrically close together must be clustered and assigned to the same microstate. Two common algorithms for this geometrical clustering are constant-space clustering and k-centers clustering [95, 97, 98, 50]. Constant-space clustering applies a regular grid to the complete state space and assigns distinct microstates to occupied grid elements, while the k-centers clustering algorithm traverses through the sampling data and generates new microstates for sampling points which do not fall within a certain geometrical distance of previous microstates. Here, the k-centers clustering algorithm was used to derive a set of microstates for the alanine dipeptide data using the software EMMA with a target of $k = 1000$ cluster centers and the Euclidian metric for distance calculation. It has been shown that the clustering results converge for large k (e.g., $k \geq 1000$), independent of the order by which the trajectory is traversed [95] (see Fig. B.3 for 1000 cluster centers in $\Phi - \Psi - \chi$ space). After discretization, each configuration of the alanine dipeptide was assigned to one of a finite set of distinct microstates. The dynamics of the alanine dipeptide can then be described by a simple time series of these microstates and serves as the basis for the calculation of transition probabilities from which the microstate transition kernel of the Markov model is derived. The elements of the microstate transition matrix, T_{ij}, can then be computed as

$$T_{ij} = \frac{P(s = j, t = \tau \ \wedge \ s = i, t = 0)}{P(s = i)} \tag{3.6}$$

where T_{ij} is the probability for finding the system in microstate j after a lagtime of τ, given that it was in microstate i at time t.

3.3.2.3. Time Scale Analysis

The choice of the best lagtime τ, with which to analyze the time series and for which the conditional probabilities in the Markov kernel should be defined, requires some consideration. Processes operating on time scales significantly faster than the lagtime cannot be accurately modeled, because they have already equilibrated at τ, precluding any detailed time-dependent information. On the other hand the lagtime must be sufficiently long that the described processes exhibit the Markov property, meaning that the path of the trajectory in configuration space must be free of hysteresis.

Hence, the choice of the lagtime is guided by an empirical approach in which the microstate Markov transition kernel is solved for a series of lagtimes and the scaled eigenvalues $t_i^* = -\frac{\tau}{\text{ln}|\lambda_i(\tau)|}$(implied times) [95, 94] are plotted against the lagtime itself. For processes exhibiting the Markov property the implied time, t_i^*, becomes independent of the lagtime, τ, which is indicated

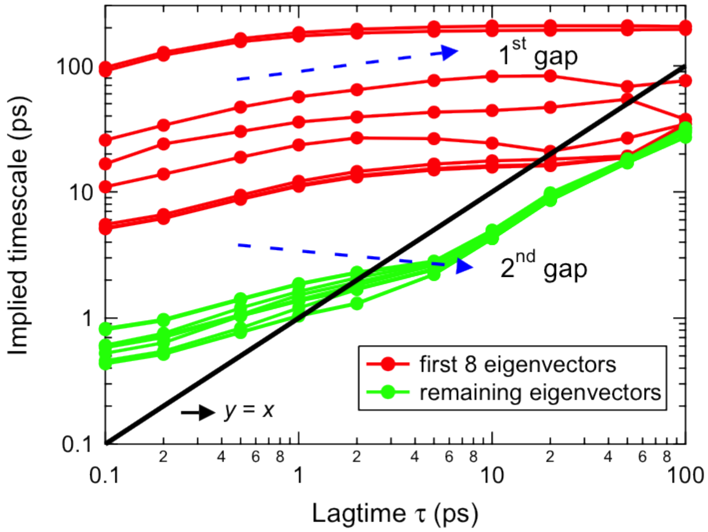


Figure 3.29.: Implied time scales of the processes associated with individual eigenvectors, depending on the lag time $n\tau$, computed as $\tau_k^*(n\tau) = \frac{n\tau}{\ln[\lambda_k(n\tau)]}$, where λ_k is the *k-th* eigenvalue. The implied time scales of the first 8 eigenvectors become flat at around 10 *ps*, which is below the lifetimes of the metastable states, indicating that the interstate transitions are Markovian.

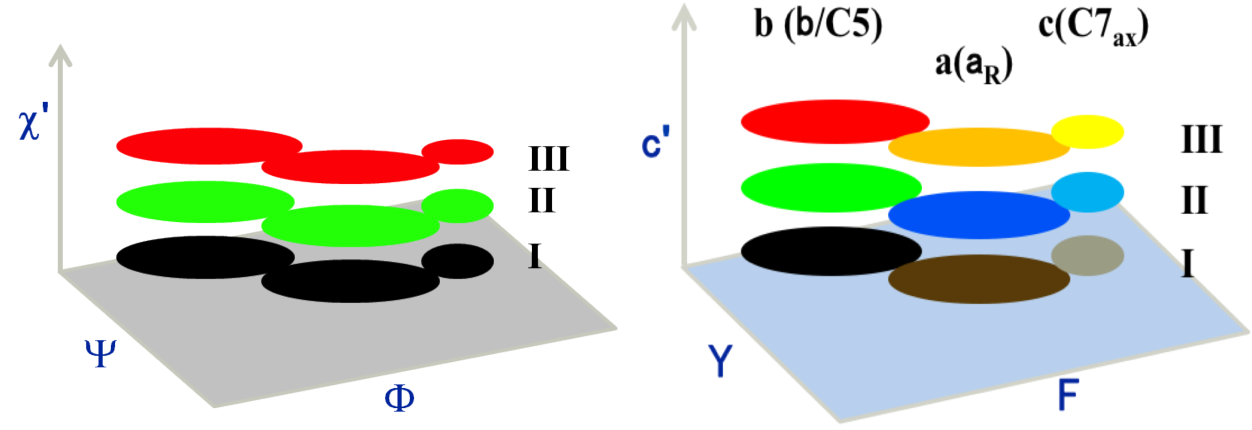
by reaching a plateau. The ideal lagtime is then given by the minimal value for which all relevant implied time scales have become independent of the value of the lagtime.

Here, the implied time scale analysis for the derived microstate transition kernel of alanine dipeptide was performed for the lagtimes (ps)=0.1, 0.2, 0.5, 1, 2, 5, 10, 20, 50, 100, and is shown in Fig. 3.29 for the 20 largest eigenvalues. The solid line in Fig. 3.29 marks the onset for the resolution limit of the Markov kernel which reflects the fact that fast processes cannot be reliably described by a kernel derived with a large lagtime (lower triangle).

Here, the implied time scale analysis graph suggests two possible reductions of the microstate towards a macrostate kernel, which are also visible as time scale gaps between the estimated relaxation times [95, 99]. The first reduction operates on a lagtime of 100 ps and takes advantage of the first time scale gap between the two slowest processes with any other, and the second reduction operates on a lagtime of 10 ps and reduces the number of distinguishable processes to 9.

3.3.2.4. Kinetic Clustering

The implied time scale analysis provides a rationale for determining the number of relaxation processes reliably described by a Markov kernel. Since the number of Markov states is inherently equal to the number of Markov processes (including the equilibrium distribution), this also provides the number of distinguishable Markov states. However, more generally, the microstate transition kernel contains explicit information about the kinetic connectivity between different groups of microstates expressed as conditional probabilities for transitions. This can be exploited by remodeling the microstate transition matrix so as to reassign the rows and columns such that those microstates close together in the matrix are also close kinetically. The result is a kinetic clustering of microstates into macrostates and is here performed using the improved Perron-Cluster-Cluster-Analysis (PCCA) method implemented in EMMA [100, 101]. This way two reduced macrostate Markov kernels were obtained: one 3 state/100 ps kernel and the other a 9 state/10 ps kernel.



(a)

(b)

Figure 3.30.: Cluster centers in metastable states I, II, and III in the configurational space of Φ, Ψ and χ from MSM using m = 3 (a) and m=9 (b)

The micro- to macrostate assignment was visually inspected using a 3-dimensional plot of the cluster location and assignment in the $\Phi - \Psi - \chi$ phase space. Fig. B.4 shows the assignment for the 3 state/100 ps kernel in which microstates with the same macrostate assignment have the same indicator, and Fig. 3.30a is a simplified schematic plot of the three macrostates: I ($\chi\epsilon\,[-120, 0]$), II ($\chi\epsilon\,[0, 120]$), and III ($\chi\epsilon\,[-180, -120] \cup [120, 180]$) in the $\Phi - \Psi - \chi$ space. These plots indicate that the 3 state/100 ps kernel describes transitions between states with differing χ, which means that it exclusively describes the methyl group rotational dynamics. The higher resolution of the 9 state/10 ps kernel is required to resolve transitions between point groups with different Φ and Ψ values, and these are shown in Fig. B.5a. The Markov states are labeled according to their position in χ (I, II, III) and their location within the $\Phi - \Psi$ plane (a, b, c). The projection of the distribution in $\Phi - \Psi - \chi$ phase space onto the $\Phi - \Psi$ plane provides the Ramachandran plot (Fig. B.5b), which leads to the identification of the Markov states with the Ramachandran states (a represents α_R, b represents $\beta/C5$, and c represents $C7_{ax}$). Fig. 3.30b is a simplified schematic plot of these 9 macrostates.

The derived macrostate kernel, is provided in Table 3.6. The corresponding eigenvalues and eigenvectors are plotted in Fig. 3.31, and the relaxation times and frequencies are listed in Table 3.7.

3.3.3. Incoherent and Coherent Scattering Analysis

The total incoherent intermediate scattering function, $F_{inc}(q, \tau)$, is shown in Fig. 3.32 for q=1 Å^{-1}, along with the single-atom incoherent intermediate scattering functions,$F_{inc}(q, \tau, \alpha)$ for α=HY1, HB1, and HNT, respectively. These three atoms were chosen as representatives of the hydrogen atoms on the terminal methyl groups, the side-chain methyl group, and the backbone, respectively.

$F_{inc}(q, \tau, \alpha)$ for HY1 decays within ~1 ps, that for HNT within ~100 ps, and that for HB1 within ~1 ns. The spread in relaxation times for the individual hydrogen atoms indicates that the associated structural relaxation processes do exist on significantly different time scales and affect individual atoms to a different degree.

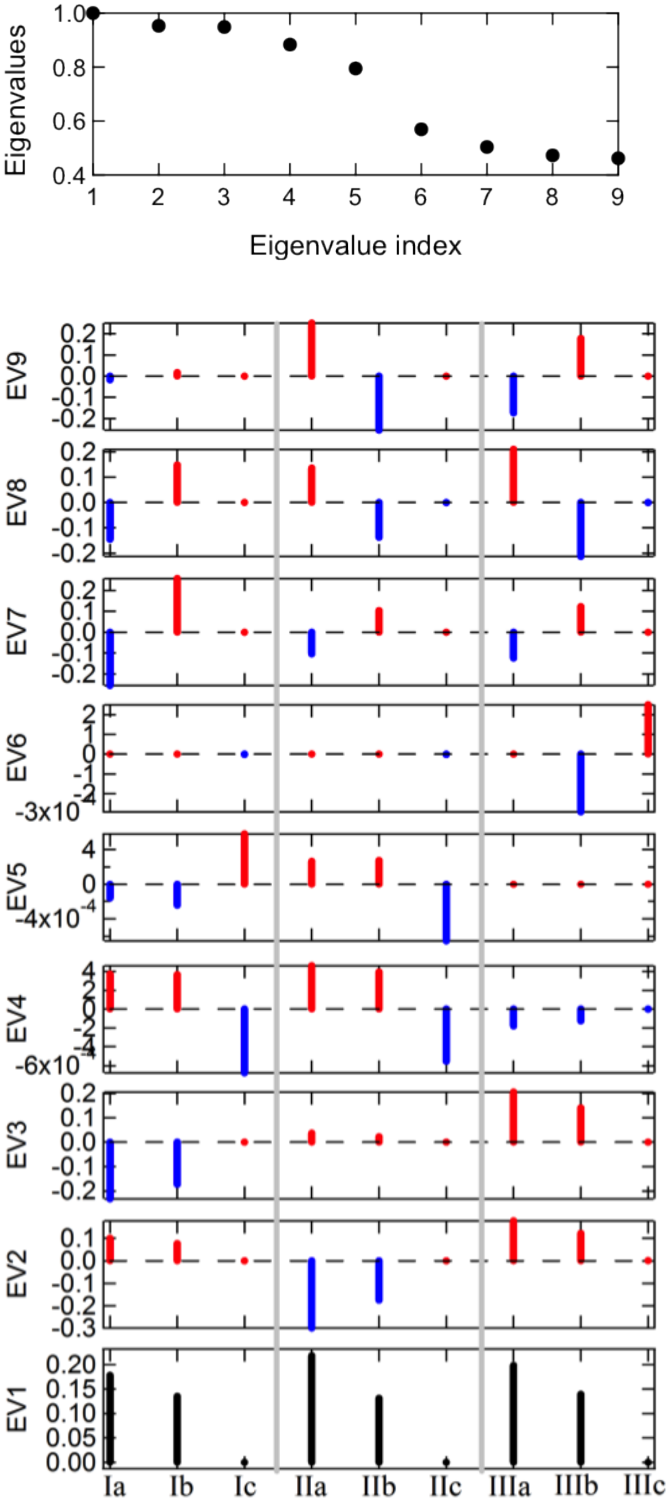


Figure 3.31.: Eigenvalues and Eigenvectors obtained from MSM using m=9.

9-state transition matrix $T^{\{\chi',\Phi,\Psi\}}$ for alanine dipeptide:

	$\to Ia$	$\to Ib$	$\to Ic$	$\to IIa$	$\to IIb$	$\to IIc$	$\to IIIa$	$\to IIIb$	$\to IIIc$
Ia	0.765	0.202	0.000	0.014	0.002	0.000	0.013	0.003	0.000
Ib	0.266	0.695	0.000	0.005	0.013	0.000	0.005	0.015	0.000
Ic	0.096	0.004	0.846	0.007	0.000	0.046	0.000	0.001	0.001
IIa	0.011	0.003	0.000	0.785	0.186	0.000	0.011	0.003	0.000
IIb	0.003	0.013	0.000	0.312	0.653	0.000	0.004	0.014	0.000
IIc	0.000	0.000	0.042	0.122	0.004	0.832	0.000	0.000	0.000
$IIIa$	0.012	0.004	0.000	0.012	0.003	0.000	0.767	0.202	0.000
$IIIb$	0.004	0.015	0.000	0.005	0.013	0.000	0.289	0.674	0.000
$IIIc$	0.000	0.000	0.002	0.001	0.001	0.002	0.304	0.029	0.662

Table 3.6.: Row stochastic transition matrix $T^{\{\chi',\Phi,\Psi\}}$ for m=9. Each matrix element describes the conditional probability for a transition between the originating state (designed by row) and the destination state (designed by column)

Eigen decomposition for $T^{\{\chi',\Phi,\Psi\}}$ with coherent and incoherent scattering amplitudes:

	Process / Eigenvector								
	1	2	3	4	5	6	7	8	9
Eigenvalue	1.0	0.95	0.94	0.88	0.79	0.66	0.50	0.47	0.46
τ_k (ps)	/	205.49	189.13	80.37	43.57	24.23	14.58	13.34	12.94
$A_{inc,k}$	0.71	3.4e-2	3.2e-2	1.4e-4	6.7e-5	1.4e-4	2.4e-2	5.5e-3	1.9e-3
$A_{coh,k}$	0.92	8.2e-6	2.6e-7	8.3e-5	7.9e-7	6.2e-5	3.5e-2	6.7e-3	5.4e-4

Table 3.7.: Eigenvalues, relaxation times/frequencies, and A_k for $m = 9$ and $q = 1$ Å^{-1}

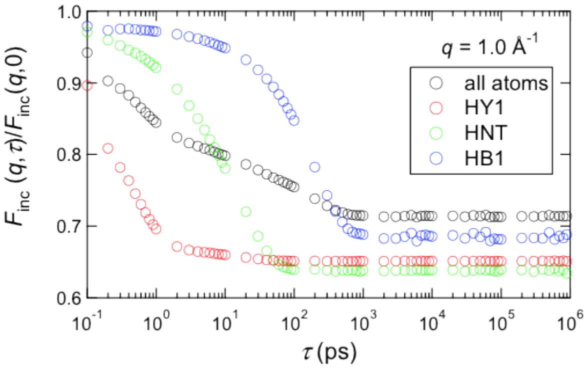


Figure 3.32.: $F_{inc}(q,\tau)$ for all atoms and selected hydrogens atoms at $q = 1$ Å^{-1} directly calculated from MD trajectories using Eq. 2.16.

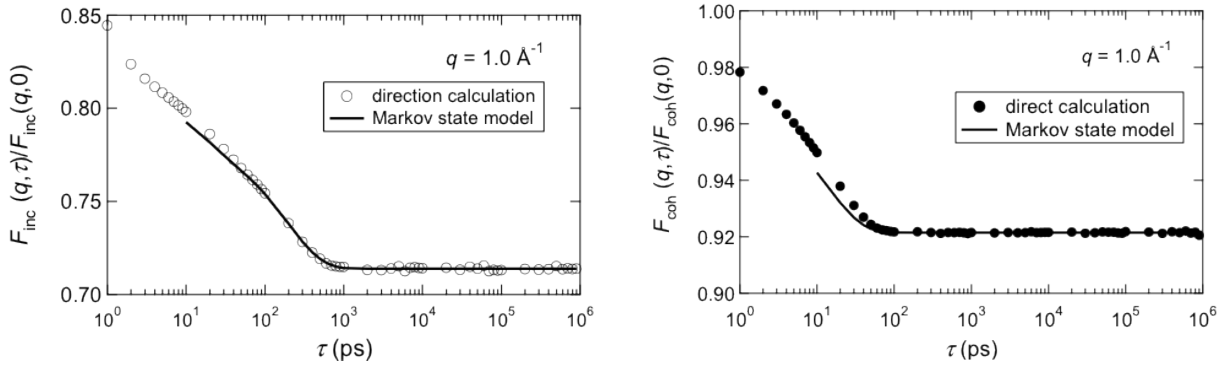


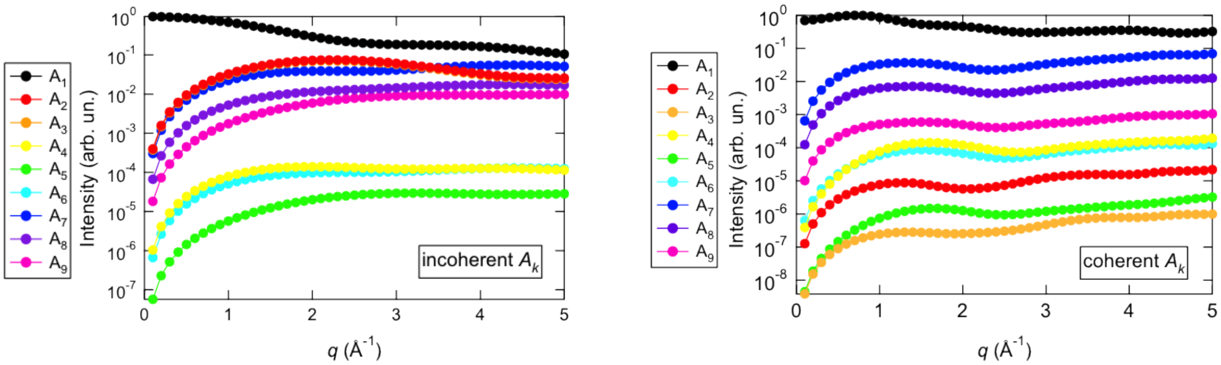
Figure 3.33.: Comparison of $F_{inc}(q,\tau)$, $F_{coh}(q,\tau)$ $q = 1$ Å^{-1} between the result from the Markov state model using Eq. 2.40 with coherent and incoherent amplitudes described by Eq. 2.41 and 2.42 and the result from the direct calculation from MD trajectories using Eq. 2.16 and 2.15.

$F_{inc}(q,\tau)$ for the whole molecule calculated from the 9-state kernel MSM is compared with the result from the direct calculation in Fig. 3.33. The good agreement of the two methods indicates that the MSM reproduces $F_{inc}(q,\tau)$ very accurately, with small differences found only on fast time scales as would be expected (<10 ps). The full function $F_{inc}(q,\tau)$ from the 9-state MSM consists of the 9 components according to Eq. 2.34. The first component, $A_{1,inc}(q)$, corresponding to $\lambda_1 = 1$, is the elastic structure factor and the other components, $\sum_{k=2}^{9} \exp\left(-\frac{\tau}{t_k}\right) \cdot A_{k,inc}(q)$, are associated with the relaxation processes in the MSM. The process-dependent scattering amplitudes, $A_{k,inc}(q)$, and the corresponding relaxation times, $\tau_{k,inc}$, are provided in Table 3.7.

The total incoherent intermediate scattering function, $F_{inc}(q,\tau)$, which is that which can be in principle determined in neutron scattering experiments, cannot be directly used to identify the set of structural relaxation processes that lead to the decay of $F_{inc}(q,\tau)$. Rather, in a typical experimental analysis, $F_{inc}(q,\tau)$ would be fitted with a set of simple exponential functions (assuming a small number of relaxation processes) or one single stretched exponential function (assuming a distribution of relaxation times [102]). This leads to an understanding of the system in terms of time scales, but not in terms of structural processes. In contrast, the MSM explicitly provides the decomposition into structural relaxation processes based on the molecular dynamics simulation and thus serves as an intermediary between the simulation and experiment.

$F_{coh}(q,\tau)$ calculated directly from the molecular dynamics trajectory using Eq. 2.15 and from the MSM using Eq. 2.40 together with Eq. 2.41 are plotted in Fig. 3.33. The agreement of the results illustrates that the MSM can also reliably reproduce the coherent scattering functions. The scattering amplitudes, $A_{k,coh}(q)$, are listed in Table 3.7 for $q = 1.0$ Å^{-1}. $F_{coh}(q,\tau)$ decays more rapidly than $F_{inc}(q,\tau,\alpha)$ the processes corresponding to $\lambda_2 = 0.96$ and $\lambda_3 = 0.94$ have small $A_{k,coh}(q)$ but relatively large $A_{k,inc}(q)$.

From inspection of Fig. 3.34 it follows that the dominant components for incoherent scattering are $A_{2,inc}(q)$ and $A_{3,inc}(q)$, while for coherent scattering the largest contribution arises from $A_{7,coh}(q)$. An analysis of the eigenvectors shows (see Figure 3.31) that the 2^{nd} and 3^{rd} process correspond to the methyl group rotation, while process $k = 7$ is associated with a backbone dihedral rotation. The relative lack of sensitivity of coherent scattering to the rotation of the methyl group is a result of the methyl group rotation involving a symmetry operation that does not change the value of the associated observable in Eq. 2.41 [103, 104, 105]. Conversely, structural changes that strongly affect



(a)

(b)

Figure 3.34.: Q dependence of scattering amplitudes $A_{k,inc}$ (a) and $A_{k,coh}$ (b) of each process.

the observables in Eq. 2.41 and Eq. 2.42 , are expected to manifest themselves in correspondingly large amplitudes $A_k(q)$.

3.4. High Performance Calculation of Scattering Profiles

Molecular dynamics simulation has traditionally been a high performance computing problem, which is why many research groups have spent significant effort on producing algorithms for molecular dynamics which run efficiently on massively parallel supercomputing hardware [106, 107, 54]. However, the analysis of the generated simulation data is still often performed with algorithms which cannot fully exploit the architecture of massively parallel computers. One reason for this is that in many cases the short-term incentive to refactor the analysis algorithms is not strong enough, which leads to a situation where a moderate amount of effort is spent on achieving a satisfactory scalability through the use of embarrassingly parallel schemes. E.g. the parallel computation of the time-averaged root-mean squared deviation (RMSD) can be achieved by splitting the trajectory into different time segments and assign each segment to a different processing unit.

However, some analysis algorithms are not embarrassingly parallelizable or are a mix between embarrassingly and delightfully parallel. Two of those cases, which are presented here, are the calculation of the incoherent and coherent dynamic scattering intensities from molecular dynamics simulation trajectories. While previous software solutions for calculating all the different scattering functions already exists [108, 49, 109, 110, 20], they have not been specifically designed for execution on a massively parallel computer. This work introduces algorithmic solutions for achieving good scalability on modern supercomputing architectures. The discussed data staging and calculation schemes were implemented in the software SASSENA [§] [19], which has been made available to the general public.

3.4.1. Data Staging Schemes

The calculation of the incoherent intermediate scattering function, $F_{inc}(q,\tau)$ requires the computation of the following quantity:

$$F_{inc}(q,\tau) = \sum_n \langle\langle a_n(\vec{q},t) \cdot a_n^*(\vec{q},t+\tau)\rangle_t\rangle_{\vec{q}} \tag{3.7}$$

where q is the scattering length, τ is the correlation time, $a_n(\vec{q},\tau)$ is the complex scattering amplitude for atom n and $\langle\langle\rangle_t\rangle_{\vec{q}}$ denotes the time and the orientational average. The coherent intermediate scattering function is given by

$$F_{coh}(q,\tau) = \langle\langle A(\vec{q},t) \cdot A^*(\vec{q},t+\tau)\rangle_t\rangle_{\vec{q}} \tag{3.8}$$

where $A(\vec{q},t) = \sum_n a_n(\vec{q},t)$ is the total scattering amplitude computed from the scattering amplitudes of all atoms within the system. The distinction between $a_{n,inc}(\vec{q},t)$ and $a_{n,coh}(\vec{q},t)$ is neglected for simplicity.

A significant difference between the calculation of the incoherent and coherent scattering is the position for the summation over atoms, $\sum_n$, which is the innermost part for coherent and the outermost part for incoherent scattering. Thus, the high-performance calculation of incoherent scattering requires the data to be organized into a sequence of blocks, each containing the coordinates of a single atom for all time instances (atom decomposition), while coherent scattering requires each block to contain the coordinates for all atoms for a particular time (frame decomposition). The relationship between data alignment and the algorithmic access pattern is illustrated in Figure 3.35. Patterns which place data elements as neighbors for access in a consecutive fashion benefit for two reasons: First the computing hardware can better avoid latencies, by loading several data

[§] http://www.sassena.org

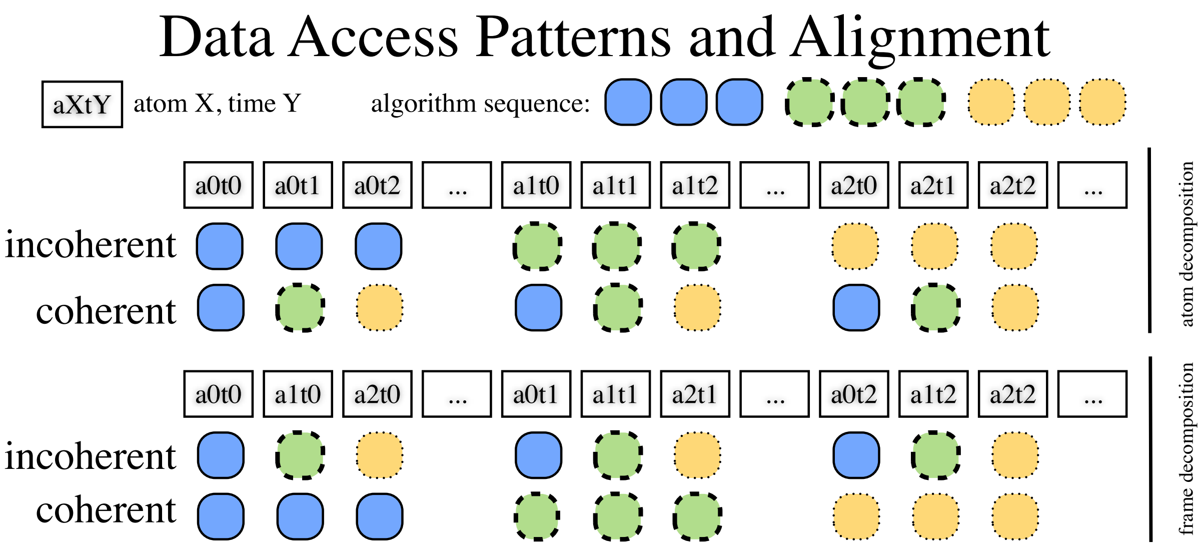


Figure 3.35.: Data alignment and access patterns. Consecutive access on neighboring data elements is the preferred scheme for high performance computing.

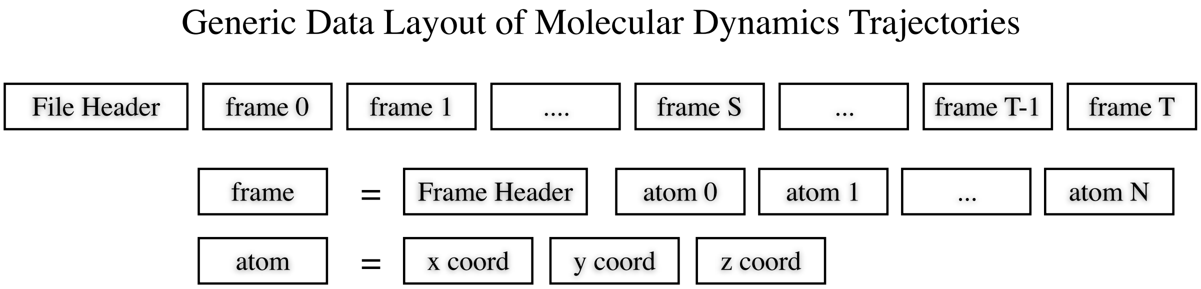


Figure 3.36.: Generic data layout of a molecular dynamics simulation file

elements at the same time into the processing unit. Second, which is even more important in a distributed memory environment, the alignment of the data into independent blocks leads to a clear task assignment, where each processing unit works on a consecutive chunk of data, with little or no overhead of moving data elements around.

Thus the first priority of any high performance algorithm targeted towards a massively parallel computing environment, is to place the data accordingly into the distributed memory. The generic data layout for a molecular dynamics trajectory file is provided in Figure 3.36. For coherent scattering the optimal data layout matches the data layout usually found in molecular dynamics simulation trajectories. Thus little effort is necessary to place the data accordingly into memory, which is illustrated in Figure 3.37. In contrast, incoherent scattering requires a transposed data layout, which requires a reshuffling of the data at the time the data are read from disk. This is achieved at almost no overhead cost by introducing a midstate during the reading of the trajectory data which prepares the data for a MPI AlltoAll call, after which the data is distributed among the processing units in a transposed form as illustrated in Figure 3.37.

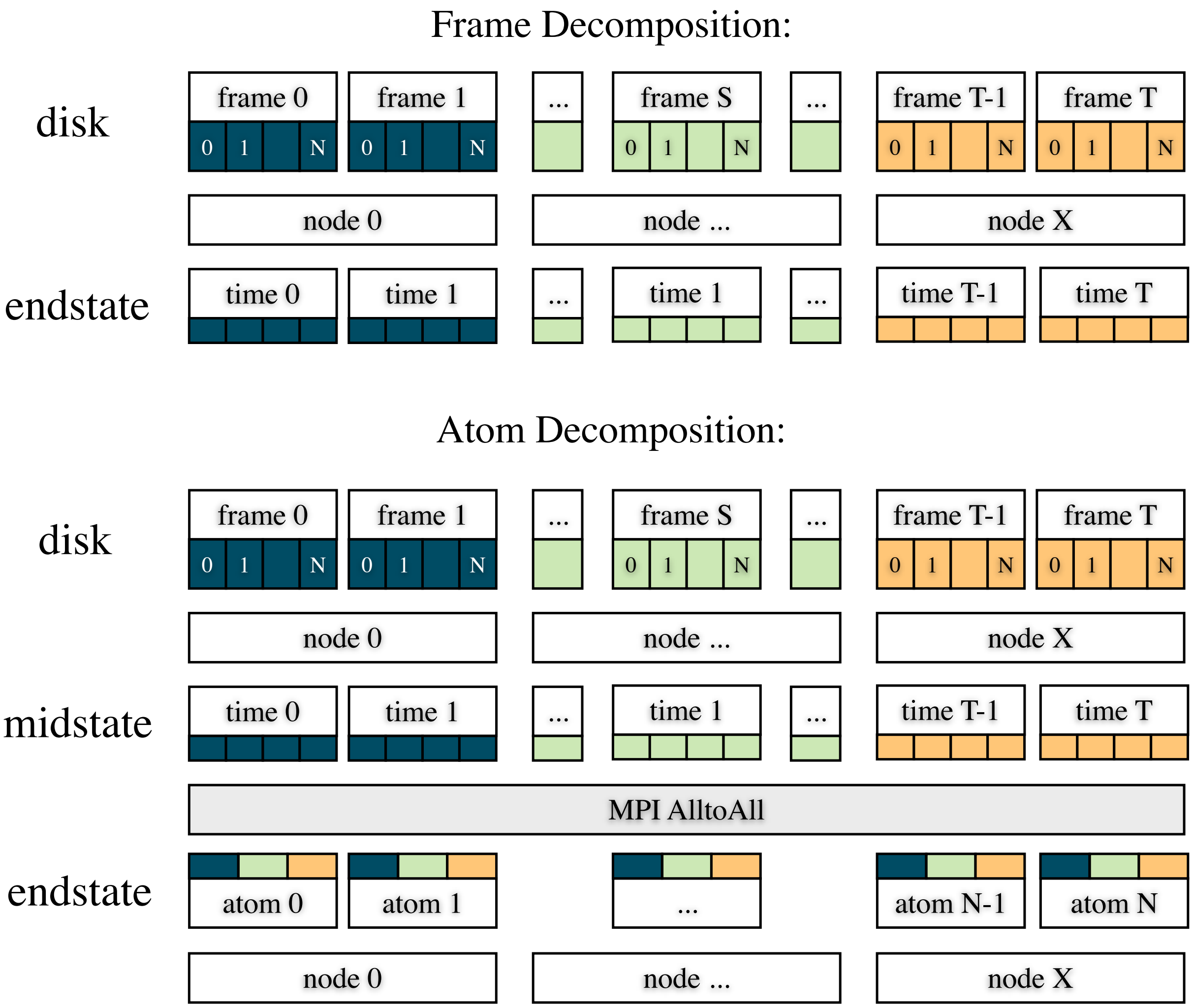


Figure 3.37.: Data staging schemes for coherent and incoherent scattering. In coherent scattering the required data layout matches the layout in the trajectory file. In incoherent scattering a transposition of the data is necessary, which is performed by introducing a midstate into the data staging, followed by a MPIAlltoAll call.

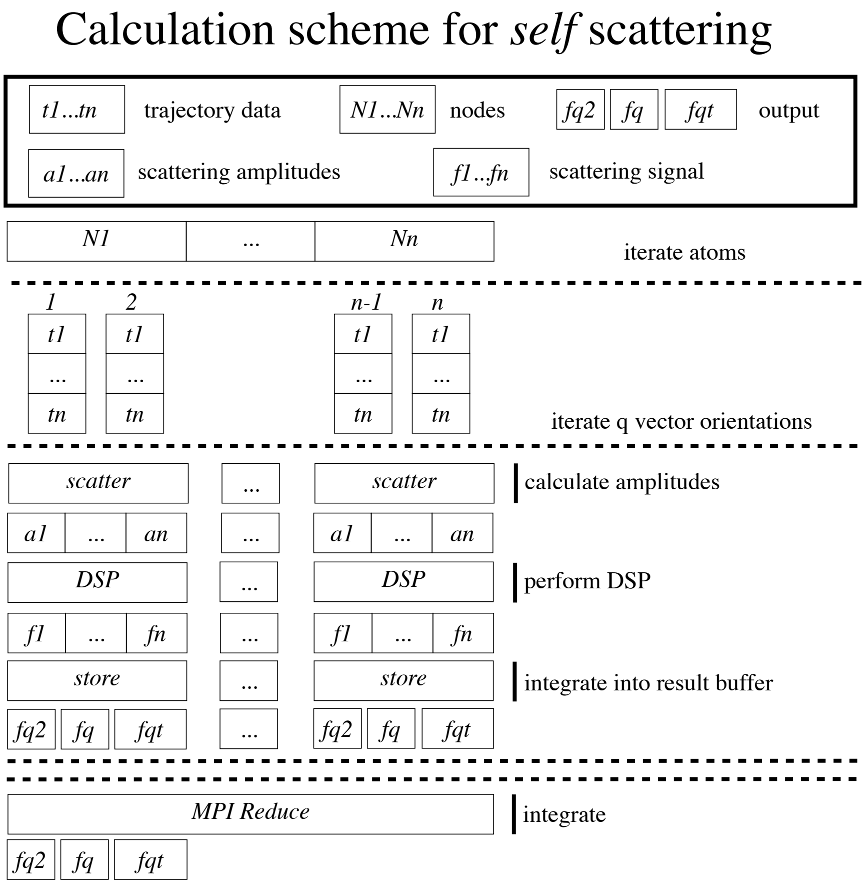


Figure 3.38.: Calculation scheme for incoherent (self) scattering.

3.4.2. Calculation Schemes

For incoherent scattering, also denoted as self scattering, the atom decomposition leads to a calculation scheme where each processing unit acts independently and no communication of intermediate results are necessary, as indicated in Figure 3.38. For coherent scattering, the frame decomposition achieves full data locality for the resulting calculation scheme. However, it requires a global communication for the exchange of intermediate results, which is implemented as MPIAlltoAll call, which is indicated in the corresponding Figure 3.39. The DSP step corresponds to the computation of the autocorrelation ($\langle a(t) \cdot a^*(t+\tau\rangle_t$ or $\langle A(t) \cdot A^*(t+\tau)\rangle_t$), and the final function is stored in a data array denoted by *fqt* (the remaining symbols are explained in Ref. [19]).

3.4.3. Partitioning Scheme

The calculation schemes for coherent (all) and incoherent (self) scattering impose a limit on the number of cores that can be used in parallel. For coherent (all) scattering this is equal to the number of frames, whereas for incoherent (self) scattering it is the number of atoms. However, most scattering calculations require the computation of the scattering signal for many independent scattering vectors $\vec{q}$. This allows an additional layer of parallelism to be added based on the number

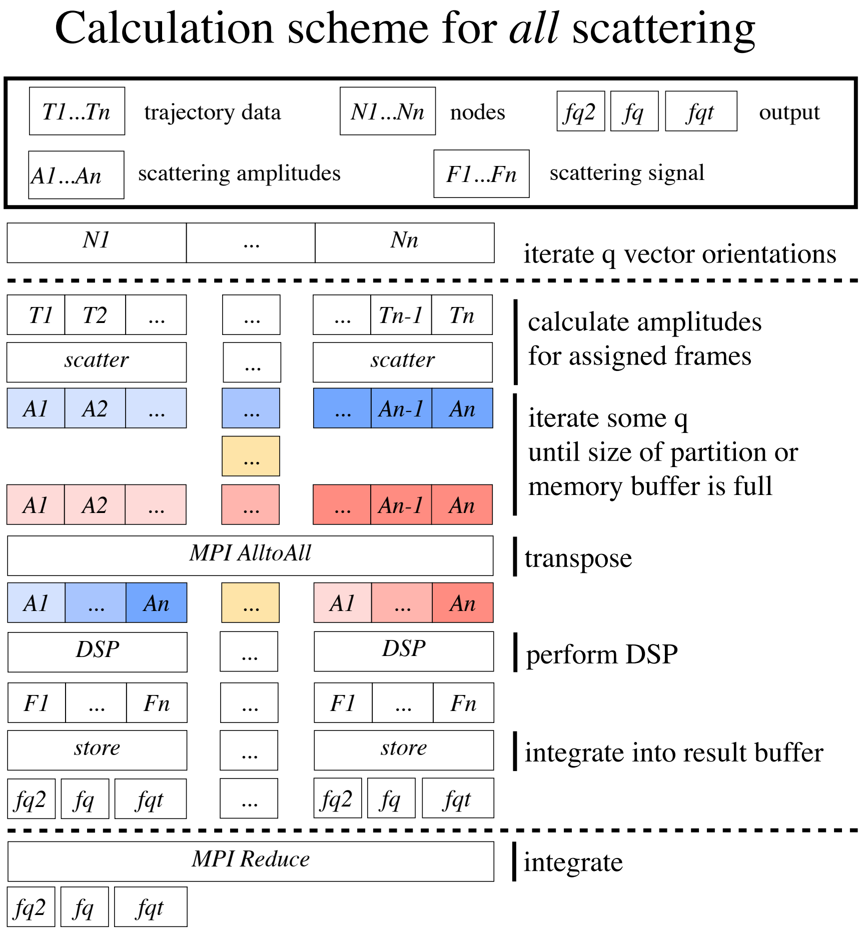
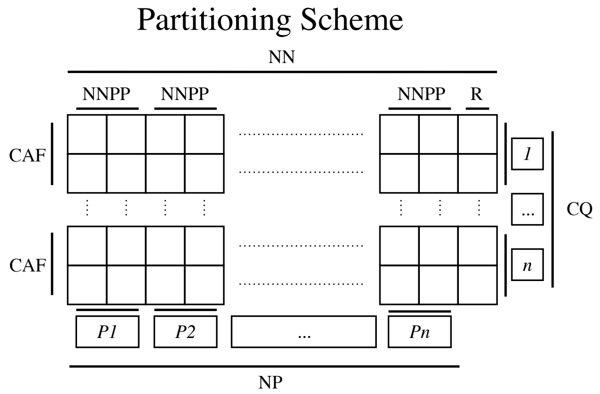


Figure 3.39.: Calculation scheme for coherent (all) scattering.



symbol	description
NN	all nodes
NP	number of partitions
$NNPP$	number of nodes per partition
NQ	number of independent q vectors
NAF	decomposition parameter (atoms or frames)
R	remaining nodes (not used)
CAF	cycles per q vector
CQ	q vector cycles

Figure 3.40.: Illustration of the underlying partitioning strategy (top) and definition of the symbols used (bottom). For a given partition size, $NNPP$, the utilization, U, of the available parallel bandwidth is pre-determined (see text).

of independent scattering vectors. This results in a 2-dimensional partitioning scheme with the partition size being a parameter of choice with the following restrictions: Memory and Utilization.

Memory is a limiting factor since each partition has a full copy of the trajectory data. For trajectory data which exceed the available memory per node, the minimum allowed partition size is determined by the minimum number of nodes capable of holding the trajectory data in memory.

The utilization, U is a performance characteristic of the chosen partition size and equals the fraction of time the cores are being used to compute scattering amplitudes. U can be precomputed:

$$U = \frac{NAF \cdot NQ}{NN \cdot CAF \cdot CQ} \tag{3.9}$$

$$\begin{aligned} CAF &= \begin{cases} NAF \div NNPP & 0 \equiv NAF \bmod NNPP \\ (NAF \div NNPP) + 1 & \text{else} \end{cases} \\ CQ &= \begin{cases} NQ \div NP & 0 \equiv NQ \bmod NP \\ (NQ \div NP) + 1 & \text{else} \end{cases} \end{aligned} \tag{3.10}$$

The partitioning scheme and the origin of equations 3.9 and 3.10 are illustrated in Figure 3.40 together with the list of symbols.

Since the utilization can be computed in advance, the software calculates U for each possible decomposition and selects that partition size $NNPP$ yielding the highest utilization. Sometimes a particular partition size is favorable even though it may not yield the best utilization U, as, for example, when the partition size should be equal to or a multiple of the number of cores per

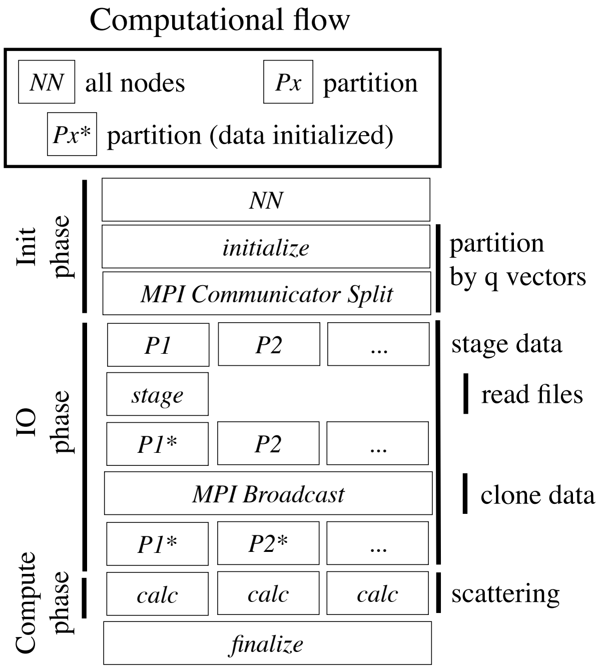


Figure 3.41.: Overall computational flow. After initialization, data is read from disk or the network file system and optimally distributed among the available computer nodes. During the compute phase all MPI communication is local to its partition.

computer node. In cases where the partition size is equal to the number of cores per computer node, communication within a partition is local and avoids network latencies. These non-trivial aspects of the partition scheme make it hard to automatically select the overall best partition size for a given problem, which is why the user usually has room for performance optimization, given enough knowledge about the hardware layout of the parallel computer. If the selected partitioning scheme results in more than one partition, the trajectory data are read by the first partition and subsequently cloned into the remaining partitions.

3.4.4. Software Performance and Scalability

The data staging and calculation schemes were implemented using the programming language C++ and the software was made available to the general public free of charge [19]. The I/O performance and the scalability of the software was characterized using a set of benchmark files. Each time the software is executed, it performs two main tasks: staging of the trajectory data and calculation of the scattering signal, which is illustrated in Figure 3.41. The performance characteristics of these two tasks depend on different aspects of the underlying hardware, which is why they are discussed separately. Data staging is dependent on factors such as the number of available file servers, the file protocol and the trajectory format, while scattering calculations depend mainly on the partitioning scheme, the available high performance network and, in particular, on the type of scattering function. The performance was assessed by software internal timers. The single node computational efficiency of the scattering routine (which computes the scattering amplitudes) was measured with CrayPAT and was 850 MFlops at double precision with default optimization settings using the GNU C++ compiler, which corresponds to about 8.2% of the single node peak performance.

Benchmarked computing platforms:

Feature	System		
	Moldyn	Jaguar	Hopper
File System	NFS/Gigabit	LUSTRE	LUSTRE
Interconnect	Infiniband	Cray Seastar2+	Cray Gemini
Cores/Node	12	12	24
CPU Type	AMD Opteron 2431	AMD Opteron 2435	AMD Opteron 8378
GB RAM/Node	16	16	32
Nodes	100	18688	6392

Table 3.8.: Systems used to run the performance benchmarks. Moldyn is a computational cluster owned by the Center for Molecular Biophysics at ORNL. Jaguar and Hopper are Cray supercomputers, administered by NCCS and NERSC, respectively.

Benchmark sets:

System ID	Atoms	Benchmark 8			Benchmark 48			Benchmark 240		
		Frames	XTC	DCD	Frames	XTC	DCD	Frames	XTC	DCD
SL	613	400k	831M	2.8G	2M	4.1G	14G	5M	11G	35G
MM	120k	2k	930M	2.8G	15k	6.9G	21G	50k	24G	68G
LS	3.7M	60	845M	2.6G	500	6.9G	22G	2.5k	35G	107G

Table 3.9.: Benchmark systems used characterized by their minimum job sizes. The symbols SL, MM and LS stand for small-large, medium-medium and large-small respectively. The benchmark files were tailored for minimum partition sizes of 8, 48 and 240.

3.4.4.1. IO

The IO performance is given by the time required to stage the trajectory data and was characterized by running a set of benchmarks on different computer systems, listed in Table 3.8. Moldyn is a computer cluster of 100 12-core compute nodes with an Infiniband network to perform MPI communication and possesses 2 file servers, one located on the head node and the other on a dedicated node, on which the benchmark data was placed. The file servers are connected to the compute nodes via Gigabit Ethernet. Jaguar is a Cray XT5 supercomputer at Oak Ridge National Laboratory and possesses 18688 12-core compute nodes, a LUSTRE file system and a CRAY SeaStar2+ network for high performance communication. Hopper is a Cray XE6 supercomputer hosted by NERSC and possesses 6392 24-core compute nodes, a LUSTRE file system and a CRAY Gemini network. Jaguar and Hopper are similar architectures common for supercomputers, while Moldyn is the intermediate cost solution of a high fidelity computational cluster.

Three benchmark sets were prepared, corresponding to partition sizes of 8, 48 and 240 cores, which also determines the number of instances used to read the trajectory in parallel. Each benchmark set contains three molecular systems with distinct characteristics (frames/atoms) and file formats (XTC/DCD). The benchmark sets are listed in Table 3.9.

The IO performance results are given in Figure 3.42. The performance was assessed as the time to finish the data staging phase and was determined as the minimum out of 10 trial runs for each case. The stage time was additionally renormalized by the file sizes to yield effective bandwidths in MByte/sec, shown in Figure 3.43.

Moldyn shows an approximately linear increase in data staging time with the partition size, stemming from the availability of only one file server and the bandwidth limitations of the Gigabit

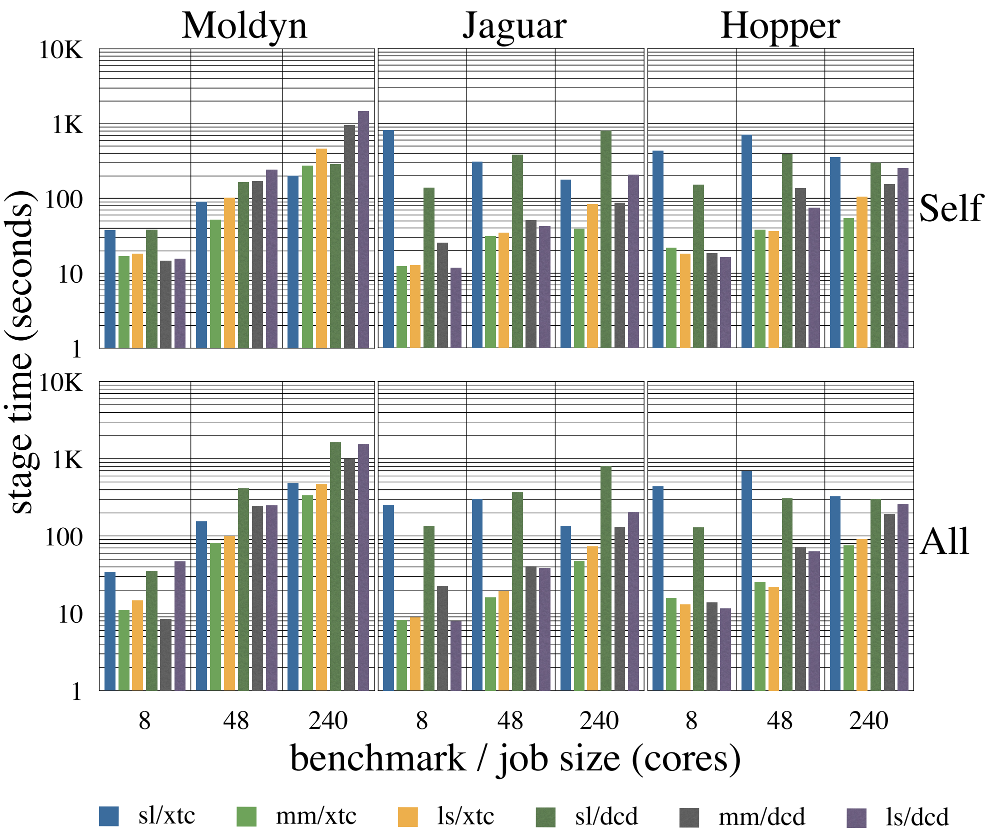


Figure 3.42.: IO performance results measured by time to completely load the trajectory data into memory for *all* and *self* scattering on Moldyn, Jaguar and Hopper for the three benchmark sets.

network. The time to stage data for *all* scattering is similar to that for *self* scattering. Jaguar shows significantly better scaling characteristics since more file servers and a better network are available. This causes the maximum effective bandwidth to approach the 1 GByte/sec mark. However, for the SL benchmark files Moldyn actually performs better than Jaguar, especially for small partition sizes. The SL system features a large number of small frames, thus generating much more file access requests than the other systems, and this seems to be a problem for the LUSTRE file system in the current case. Hopper exhibits IO characteristics similar to Jaguar, as expected since both supercomputers have very similar IO systems.

3.4.4.2. Scattering Calculations

The computational demand depends on the type of scattering calculation to be performed. *All* scattering requires global MPI communication during the calculation because the data are decomposed by frames and the DSP step requires the full time signal. Thus, for each orientation of the q vector, the signal must be aggregated on one node, which then performs post-processing operations and puts the result into local buffers. *Self* scattering does not require data exchange during the calculation since the trajectory data has been placed accordingly. However, *self* scattering requires a post-processing operation for each single atom, *i.e.*, the computation of an autocorrelation, which can become significant for long trajectories (FFTW scales with $O(T \cdot log(T))$).

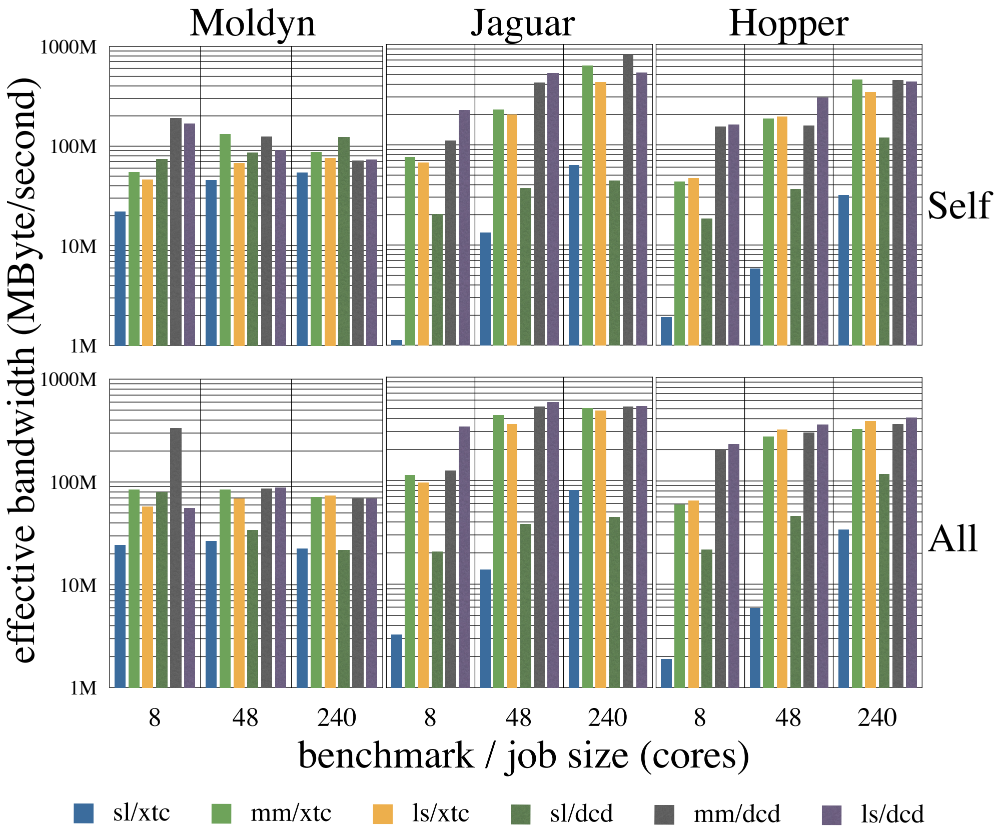


Figure 3.43.: IO performance results measured by the effective bandwidth achieved for *all* and *self* scattering on Moldyn, Jaguar and Hopper for the three benchmark sets.

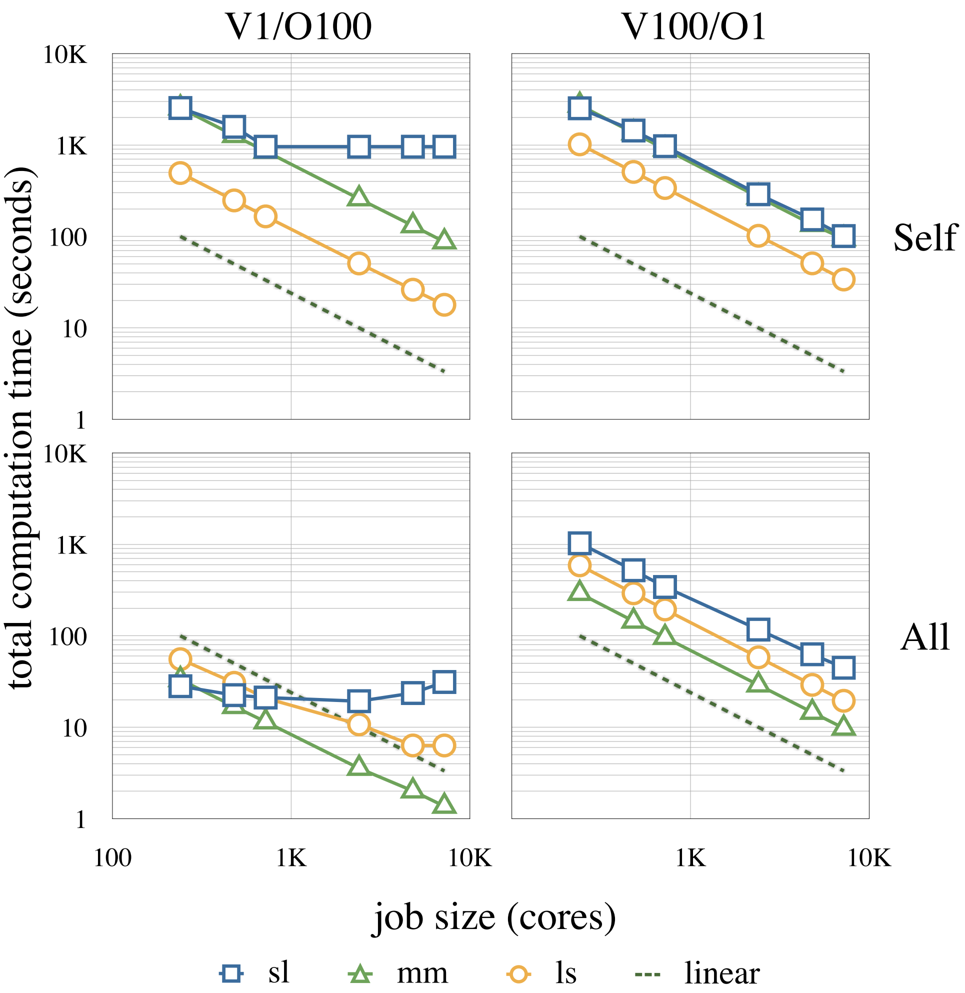


Figure 3.44.: Scaling results for Jaguar. The jobs sizes were 240, 480, 720, 2400, 4800 and 7200 cores. The partition size was chosen automatically to yield the best utilization. The scaling results for Moldyn and Hopper have similar characteristics (data not shown).

The performance characteristics were sampled with a set of 3 benchmark files and 4 variations of the scattering function, consisting of *all* vs. *self* scattering and a calculation for one scattering vector with 100 orientations (V1/O100) and 100 scattering vectors with one orientation each (V100/O1). The DSP setting was set to *autocorrelate*, which computes dynamic scattering functions and allowed the investigation of the impact of significantly different computational costs for DSP (complexity $O(T \cdot log(T))$ on the scalability for the three systems SL, MM and LS. Figure 3.44 shows the scaling results for Jaguar. Hopper and Moldyn have similar scaling characteristics (data not shown).

The heterogeneity in the type of function to compute makes it difficult to derive a single performance measure. However, from Figure 3.44, a few observations are apparent. Computing the signal for the SL system in case of V1/O100 does not scale well. For *self* scattering the reason is trivial since the partitioning is limited by the number of atoms. For *all* scattering the reason is more complex and has been investigated by decomposing the total calculation time into its components, shown in Figure 3.45.

The decomposition shows that the time to compute an autocorrelation becomes significantly larger than the time to compute the scattering amplitudes for larger partition sizes, and therefore there is

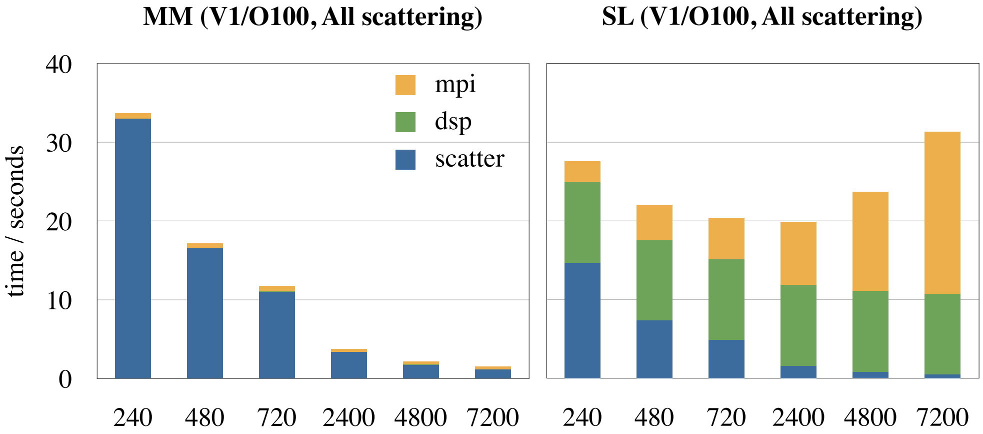


Figure 3.45.: Scattering calculation time for the SL and MM systems in the V1/O100 case decomposed into individual components of the scattering kernel. The time spent in the DSP phase and MPI communication is negligible for the MM and significant for the SL system.

no effective speed-up when increasing the number of cores. Instead the increase in the number of cores increases the MPI overhead in the global communication until the overhead becomes dominant.

The scalability of V1/O100 *self* scattering for the SL system can be increased by employing threads, which is illustrated in Figure 3.46. By placing a single MPI process on each 12-core compute node, the partition can span $12 \cdot 631$ cores instead of 631. The cores on each compute node are then utilized using threads.

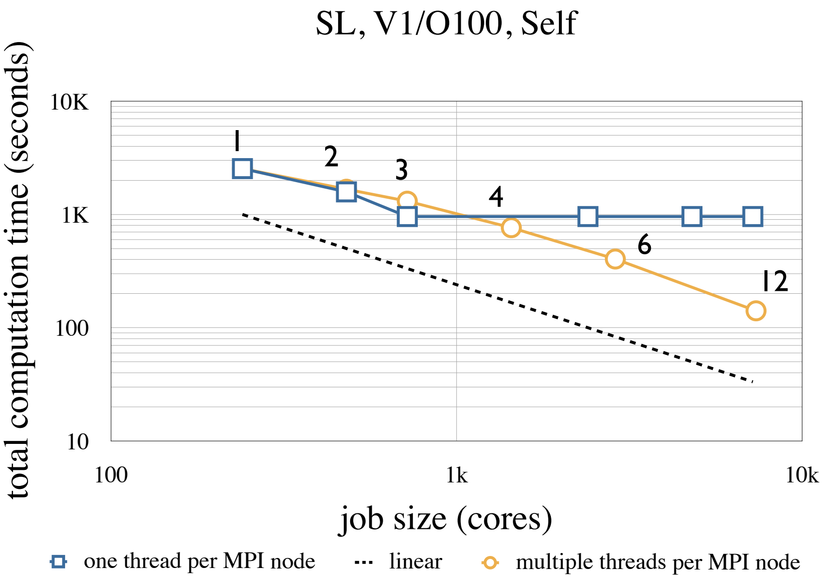


Figure 3.46.: Scalability result using multiple threads for V1/O100 *self* scattering of the SL system. For clarity the result without using multiple threads is also shown.

4. Conclusion

Physical experiments have long served as the cornerstone for scientific discovery and investigation. The growing complexity of the scientific data has inspired the development of a rich set of analytical and computational techniques to reduce the data into meaningful and digestible pieces of information. Starting out as yet another analytical technique [111, 112], in recent years, molecular dynamics has evolved from a simple computational tool for analyzing experiments towards a powerful framework of studying molecules and biomolecular systems in a level of detail unmatched by real physical experiments. Three factors have largely contributed to this development: The rapid progression of computational power, the sustained financial investment into computational infrastructure and tools, and the large scale support by the scientific community. However, despite all the progress of the molecular dynamics methodology, development efforts have never been stronger and they usually split into several camps: Increasing the raw computational performance of molecular dynamics code on the available computational platforms [55**?**], implementing intelligent algorithms to sample relevant configurations [23, 113], devising new analytical strategies to study complex systems [94, 98] and exploring the validity of new simulation protocols [56], to name a few. This work particularly benefited from the large scale parallelization work performed by the scientific code developer Roland Schulz.

As illustrated by the study of cellulose crystallinity and dynamics through the use of scattering theory in Section 3.1, the analysis of molecular dynamics simulation and atomistic models can leverage from the rich set of existing experimental techniques for studying complex matter, with the added benefit of having the full atomistic information available. This ultimately leads to a deeper understanding of the experimental method in terms of structure/dynamics vs. signal relationships. It also brings computational and experimental scientists closer together by developing a common analytical framework.

The study of lignocellulose, as discussed in Section 3.2, provides an example for a highly complex molecular system and process which cannot be easily studied in nature, due to experimental limitations. By leveraging the extensive computational resources made available through the Oak Ridge National Laboratory and the Department of Energy, this work provided the first clue in scientific history that an increase in cellulose crystallinity correlates with an increase in the amount of cellulose-lignin aggregation on the molecular level.

A proof that not all methodological advancements require extensive computational power, was provided in Section 3.3, which illustrated the utility of using Markov models for decomposing and interpreting dynamical neutron scattering functions. It shows how the molecular dynamics methodology and experimental scattering techniques can be combined in future to derive unequivocal descriptions of the conformational dynamics of molecules and their experimental observables.

The last section of this work, Section 3.4, presented the extensive work involved in creating a high performance software package for calculating the dynamical neutron scattering functions on modern supercomputer architectures. The recent shift in programming paradigms from serial towards parallel computing creates a performance gap which has to be closed by devising calculation schemes which work well on the available computing platforms. This work has shown that the full dynamic scattering function of molecules can be computed on massively parallel computing platform without making compromises in computational efficiency, leading to nearly perfect scaling.

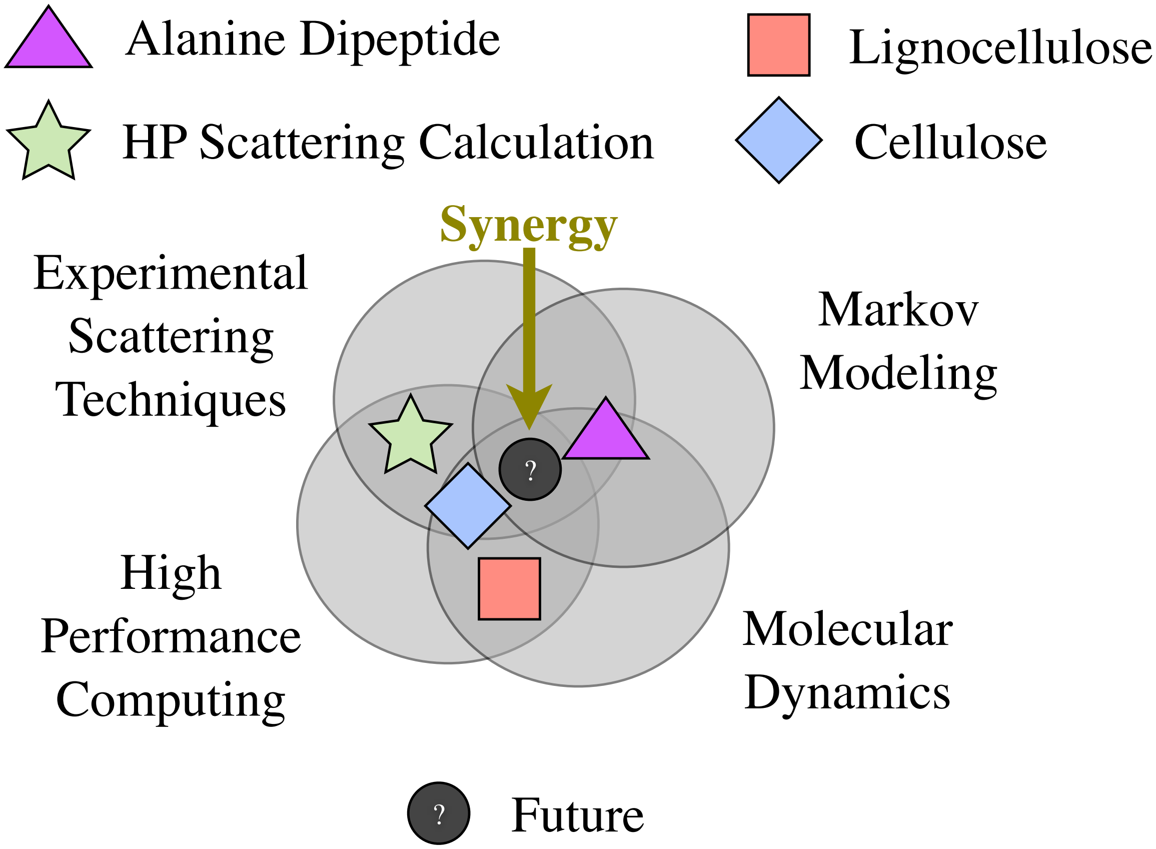


Figure 4.1.: The association of the various methods and approaches discussed here with the four scientific subdomains they cover: high performance computing, experimental scattering techniques, Markov modeling and molecular dynamics. The culmination point yields the direction for future developments.

The connection of the various methods and approaches discussed here with the various scientific subdomains is visually illustrated in Figure 4.1. Each of the projects operates on an intersection of at least two of the four domains: high performance computing, experimental scattering techniques, Markov modeling and molecular dynamics. The culmination point of all subdomains yields the logical progression of the methodology as a whole. It suggests that the future of molecular dynamics lies within combining experimental techniques (e.g. scattering theory) and stochastical tools (e.g. Markov modeling) to yield unmatched descriptions of molecular dynamics and its experimental observables. The level of complexity of the simulated systems and the level of analysis will determine the amount of computational resources required.

Bibliography

Bibliography

[1] Bryan S. Der and Brian Kuhlman. From computational design to a protein that binds. *Science*, 332(6031):801–802, May 2011. 1

[2] Sarel J. Fleishman, Timothy A. Whitehead, Damian C. Ekiert, Cyrille Dreyfus, Jacob E. Corn, Eva-Maria Strauch, Ian A. Wilson, and David Baker. Computational design of proteins targeting the conserved stem region of influenza hemagglutinin. *Science*, 332(6031):816–821, May 2011. 1

[3] Priscilla E. M. Purnick and Ron Weiss. The second wave of synthetic biology: from modules to systems. *Nature Reviews: Molecular Cell Biology*, 10(6):410–422, June 2009. 1

[4] Alyona Sukhanova, Klervi Even-Desrumeaux, Patrick Chames, Daniel Baty, Mikhail Artemyev, Vladimir Oleinikov, and Igor Nabiev. Engineering of ultra-small diagnostic nanoprobes through oriented conjugation of single-domain antibodies and quantum dots. *Nature: Protocol Exchange*, August 2012. 1

[5] Adarsh Sandhu. Biosensing: new probes offer much faster results. *Nature nanotechnology*, 2(12):746–748, December 2007. 1

[6] Horacio Cabral, Nobuhiro Nishiyama, and Kazunori Kataoka. Supramolecular nanodevices: from design validation to theranostic nanomedicine. *Accounts of Chemical Research*, 44(10):999–1008, October 2011. 1

[7] Enrico Mastrobattista, Marieke A. E. M. van der Aa, Wim E. Hennink, and Daan J. A. Crommelin. Artificial viruses: a nanotechnological approach to gene delivery. *Nature Reviews: Drug Discovery*, 5(2):115–121, February 2006. 1

[8] Kanjiro Miyata, Nobuhiro Nishiyama, and Kazunori Kataoka. Rational design of smart supramolecular assemblies for gene delivery: chemical challenges in the creation of artificial viruses. *Chemical Society Reviews*, 41(7):2562–2574, April 2012. 1

[9] Bruno Domon and Ruedi Aebersold. Mass spectrometry and protein analysis. *Science*, 312(5771):212–217, April 2006. 1

[10] Simon J. L. Billinge and Igor Levin. The problem with determining atomic structure at the nanoscale. *Science*, 316(5824):561–565, April 2007. 1

[11] Anthony Mittermaier and Lewis E. Kay. New tools provide new insights in NMR studies of protein dynamics. *Science*, 312(5771):224–228, April 2006. 1

[12] David J. Stephens and Victoria J. Allan. Light microscopy techniques for live cell imaging. *Science*, 300:82–86, April 2003. 1

[13] J. E. Tanner. Use of the Stimulated Echo in NMR Diffusion Studies. *Journal of Chemical Physics*, 52(5):2523–2526, 1970. 1

[14] Marc Bee. *Quasielastic Neutron Scattering: Principles and Applications in Solid State Chemistry, Biology and Material Science*. Adam Hilger, December 1988. 1, 4

[15] T Zemb and P. Lindner. *Neutrons, X-rays and Light: Scattering Methods Applied to Soft Condensed Matter.* Amsterdam ; Boston : Elsevier, 2002. 1, 4

[16] Frank Noé, Sören Doose, Isabella Daidone, Marc Löllmann, Markus Sauer, John D. Chodera, and Jeremy C. Smith. Dynamical fingerprints for probing individual relaxation processes in biomolecular dynamics with simulations and kinetic experiments. *PNAS*, 108(12):4822–4827, March 2011. 1, 7, 45

[17] Jack J. Dongarra and David W. Walker. The quest for petascale computing. *Computing in Science & Engineering*, 3(3):32–39, May 2001. 1

[18] Jack J. Dongarra and Aad J. van der Steen. High-performance computing systems: Status and outlook. *Acta Numerica*, 21(1):379–474, 2012. 1

[19] Benjamin Lindner and Jeremy C. Smith. Sassena — X-ray and neutron scattering calculated from molecular dynamics trajectories using massively parallel computers. *Computer Physics Communications*, 183(7):1491–1501, July 2012. 1, 14, 34, 55, 58, 61

[20] Konrad Hinsen, Eric Pellegrini, Sławomir Stachura, and Gerald R. Kneller. nMoldyn 3: using task farming for a parallel spectroscopy-oriented analysis of molecular dynamics simulations. *Journal of Computational Chemistry*, 33(25):2043–2048, September 2012. 1, 55

[21] Peter G. Bolhuis, David Chandler, Christoph Dellago, and Phillip L Geissler. Transition path sampling: throwing ropes over rough mountain passes, in the dark. *Annual Review of Physical Chemistry*, 53:291–318, October 2002. 2

[22] Yuji Sugita and Yuko Okamoto. Replica-exchange molecular dynamics method for protein folding. *Chemical Physics Letters*, 314(1-2), November 1999. 2

[23] Pu Liu and Gregory A. Voth. Smart resolution replica exchange: an efficient algorithm for exploring complex energy landscapes. *Journal of Chemical Physics*, 126(4):045106, January 2007. 2, 68

[24] Alexander D. MacKerell Jr. Empirical force fields for biological macromolecules: overview and issues. *Journal of Computational Chemistry*, 25(13):1584–1604, October 2004. 3

[25] Guillaume Lamoureux, Alexander D. MacKerell Jr., and Benoît Roux. A simple polarizable model of water based on classical drude oscillators. *Journal of Chemical Physics*, 119(10):5128, 2003. 3

[26] Arieh Warshel, Mitsunori Kato, and Andrei V. Pisliakov. Polarizable Force Fields: History, Test Cases, and Prospects. *Journal of Chemical Theory and Computation*, 3(6):2034–2045, November 2007. 3

[27] Karim Farah, Florian Müller-Plathe, and Michael C. Böhm. Classical reactive molecular dynamics implementations: state of the art. *ChemPhysChem*, 13(5):1127–1151, April 2012. 3

[28] Richard A. Friesner and Victor Guallar. Ab initio quantum chemical and mixed quantum mechanics/molecular mechanics (qm/mm) methods for studying enzymatic catalysis. *Annual Review of Physical Chemistry*, 56(1):389–427, May 2005. 3

[29] S. Mostaghim, M. Hoffmann, P. H. Konig, T Frauenheim, and J Teich. Molecular force field parametrization using multi-objective evolutionary algorithms. In *Proceedings of the 2004 Congress on Evolutionary Computation*, pages 212–219. IEEE, 2004. 3

[30] Yunling Liu, Lan Tao, Jianjun Lu, Shuo Xu, Qin Ma, and Qingling Duan. A novel force field parameter optimization method based on LSSVR for ECEPP. *FEBS Letters*, 585(6):888–892, March 2011. 3

[31] Jay W. Ponder and David A. Case. Force Fields for Protein Simulations. In *Advances in Protein Chemistry*, pages 27–85. Elsevier Ltd, 2003. 3

[32] Tom Darden, D. York, and Lee G. Pedersen. Particle mesh Ewald: An N log (N) method for Ewald sums in large systems. *Journal of Chemical Physics*, 98(12):10089–10092, June 1993. 3, 80

[33] David van der Spoel, Paul J van Maaren, and Herman J. C. Berendsen. A systematic study of water models for molecular simulation: Derivation of water models optimized for use with a reaction field. *Journal of Chemical Physics*, 108(24):10220–10230, 1998. 3

[34] Ulrich Essmann, Lalith Perera, Max L. Berkowitz, Tom Darden, Hsing Lee, and Lee G. Pedersen. A Smooth Particle Mesh Ewald Method. *Journal of Chemical Physics*, 103(19):8577–8593, 1995. 3, 80

[35] Wilfred F. van Gunsteren, Herman J. C. Berendsen, and Johan A C Rullmann. Inclusion of reaction fields in molecular dynamics. Application to liquid water. *Faraday Discussions of the Chemical Society*, 66(0):58–70, 1978. 3

[36] Martin Neumann. Dipole moment fluctuation formulas in computer simulations of polar systems. *Molecular Physics*, 50(4):841–858, November 1983. 3

[37] Martin Neumann. Dielectric relaxation in water. Computer simulations with the TIP4P potential. *Journal of Chemical Physics*, 85(3):1567, 1986. 3

[38] I. G. Tironi, R. Sperb, Paul E. Smith, and Wilfred F. van Gunsteren. A Generalized Reaction Field Method for Molecular-Dynamics Simulations. *Journal of Chemical Physics*, 102(13):5451–5459, 1995. 3, 80

[39] Maria M. Reif, Vincent Kräutler, Mika A. Kastenholz, Xavier Daura, and Philippe Hünenberger. Molecular dynamics simulations of a reversibly folding beta-heptapeptide in methanol: influence of the treatment of long-range electrostatic interactions. *Journal of Physical Chemistry B*, 113(10):3112–3128, March 2009. 3, 12

[40] L. Verlet. Computer "experiments" on classical fluids. *Physical Review*, 159(1):98–103, July 1967. 3

[41] S. Miyamoto. SETTLE: an analytical version of the SHAKE and RATTLE algorithm for rigid water models. *Journal of Computational Chemistry*, 13(8):952–962, 1992. 4, 80

[42] Jean-Paul Ryckaert, Giovanni Ciccotti, and Herman J. C. Berendsen. Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes. *Journal of Computational Physics*, 23(3):327–341, March 1977. 4, 80

[43] Berk Hess, Henk Bekker, and Herman J. C. Berendsen. LINCS: a linear constraint solver for molecular simulations. *Journal of Computational Chemistry*, 18(12):1463–1472, 1997. 4, 80

[44] K. Anton Feenstra, Berk Hess, and Herman J. C. Berendsen. Improving efficiency of large timescale molecular dynamics simulations of hydrogen-rich systems. *Journal of Computational Chemistry*, 20(8):786–798, May 1999. 4

[45] Wilfred F. van Gunsteren and Herman J. C. Berendsen. Computer Simulation of Molecular Dynamics: Methodology, Applications, and Perspectives in Chemistry. *Angewandte Chemie (International ed. in English)*, 29(9):992–1023, September 1990. 4

[46] G. L. Squires. *Introduction to the theory of Thermal Neutron Scattering*. Dover Publications : New York, 1978. 4

[47] J. M. Cowley, B. L. M. Peng, I. G. Ren, J. S. L. Dudarevc, and M. J. Whelanc. Parameterizations of electron atomic scattering factors. *International Tables for Crystallography*, C(Ch. 4.3):262, 2006. 4

[48] Varley F. Sears. Neutron scattering lengths and cross sections. *Neutron News*, 3:26–37, 1992. 5

[49] Franci Merzel and Jeremy C. Smith. SASSIM: a method for calculating small-angle X-ray and neutron scattering and the associated molecular envelope from explicit-atom models of solvated proteins. *Acta Crystallographica Section D Biological Crystallography*, 58:242–249, 2002. 5, 55

[50] Jan-Hendrik Prinz, Hao Wu, Marco Sarich, Bettina Keller, Martin Senne, Martin Held, John D. Chodera, Christof Schütte, and Frank Noé. Markov models of molecular kinetics: generation and validation. *Journal of Chemical Physics*, 134(17):174105, May 2011. 6, 7, 48

[51] Bettina Keller, Philippe Hünenberger, and Wilfred F. van Gunsteren. An Analysis of the Validity of Markov State Models for Emulating the Dynamics of Classical Molecular Systems and Ensembles. *Journal of Chemical Theory and Computation*, 7(4):1032–1044, April 2011. 7

[52] Alan R. Hoffman and J. F. Traub. *Supercomputers: directions in technology and applications*. The National Academic Press, January 1989. 10

[53] David W Walker Jack J Dongarra Jack J Dongarra David W Walker. MPI: A Standard Message Passing Interface. *Supercomputer*, 12:56–68, 1996. 10

[54] Steve Plimpton. Fast Parallel Algorithms for Short-Range Molecular Dynamics. *Journal of Computational Physics*, 117(1):1–19, March 1995. 12, 55

[55] Berk Hess, Carsten Kutzner, and David van der Spoel. Gromacs 4: Algorithms for highly efficient, load-balanced, and scalable molecular simulation. *Journal of Chemical Theory and Computation*, (4):435–447, 2008. 12, 68, 79

[56] Roland Schulz, Benjamin Lindner, Loukas Petridis, and Jeremy C. Smith. Scaling of Multimillion-Atom Biological Molecular Dynamics Simulation on a Petascale Supercomputer. *Journal of Chemical Theory and Computation*, 5(10):2798–2808, 2009. 12, 33, 68, 80

[57] S. Kimura, W. Laosinchai, T. Itoh, X. Cui, C. Linder, and R. Brown. Immunogold labeling of rosette terminal cellulose-synthesizing complexes in the vascular plant vigna angularis. *Plant Cell*, 11(11):2075–2086, November 1999. 14

[58] Neil G. Taylor. Cellulose biosynthesis and deposition in higher plants. *New Phytologist*, 178(2):239–252, 2008. 14, 15, 33

[59] Yoshiharu Nishiyama, Paul Langan, and Henri Chanzy. Crystal structure and hydrogen-bonding system in cellulose 1 beta from synchrotron X-ray and neutron fiber diffraction. *Journal of the American Chemical Society*, 124(31):9074–9082, 2002. 14, 16, 33, 79

[60] Daniel J. Cosgrove. Growth of the plant cell wall. *Nature Reviews: Molecular Cell Biology*, 6(11):850–861, November 2005. 14, 15

[61] She-You Ding and Michael E. Himmel. The maize primary cell wall microfibril: a new model derived from direct Visualization. *Journal of Agricultural and Food Chemistry*, (54):597–606, 2006. 15, 33, 79

[62] Michael E. Himmel, Shi-You Ding, D. K. Johnson, William S. Adney, Mark R. Nimlos, John W. Brady, and Thomas D. Foust. Biomass Recalcitrance: Engineering Plants and Enzymes for Biofuels Production. *Science*, 315(5813):804–807, February 2007. 14, 33

[63] Ajay Pal S. Sandhu, Gursharn S. Randhawa, and Kanwarpal S. Dhugga. Plant cell wall matrix polysaccharide biosynthesis. *Molecular Plant*, 2(5):840–850, September 2009. 14

[64] P. Alvira, E. Tomás-Pejó, M. Ballesteros, and M. J. Negro. Pretreatment technologies for an efficient bioethanol production process based on enzymatic hydrolysis: A review. *Bioresource Technology*, 101(13):4851–4861, July 2010. 16, 33

[65] Miron Abramson, Oded Shoseyov, and Ziv Shani. Plant cell wall reconstruction toward improved lignocellulosic production and processability. *Plant Science*, 178(2):61–72, 2010. 16

[66] Anders Thygesen, Jette Oddershede, Hans Lilholt, Anne Belinda Thomsen, and Kenny Stahl. On the determination of crystallinity and cellulose content in plant fibres. *Cellulose*, (12):563–576, 2005. 16, 19

[67] U.P. Agarwal, R.S. Reiner, and S.A. Ralph. Cellulose I crystallinity determination using FT–Raman spectroscopy: univariate and multivariate methods. *Cellulose*, 17(4):721–733, 2010. 16, 19

[68] Yoshiharu Nishiyama, Junji Sugiyama, Henri Chanzy, and Paul Langan. Crystal structure and hydrogen bonding system in cellulose 1(alpha), from synchrotron X-ray and neutron fiber diffraction. *Journal of the American Chemical Society*, 125(47):14300–14306, 2003. 16

[69] Shawn D. Mansfield, Caitriona Mooney, and John N. Saddler. Substrate and Enzyme Characteristics that Limit Cellulose Hydrolysis. *Biotechnology Progress*, 15(5):804–816, October 1999. 19

[70] Imke Diddens, Bridget Murphy, Michael Krisch, and Martin Müller. Anisotropic Elastic Properties of Cellulose Measured Using Inelastic X-ray Scattering. *Macromolecules*, 41:9755–9759, November 2008. 28, 30, 35

[71] Ewa J. Mellerowicz and Björn Sundberg. Wood cell walls: biosynthesis, developmental dynamics and their implications for wood properties. *Current Opinion in Plant Biology*, 11(3):293–300, June 2008. 33

[72] Henning Jørgensen, Jan Bach Kristensen, and Claus Felby. Enzymatic conversion of lignocellulose into fermentable sugars: challenges and opportunities. *Biofuels, Bioproducts and Biorefining*, 1(2):119–134, 2007. 33

[73] Tongye Shen, Paul Langan, Alfred D. French, Glenn P. Johnson, and S. Gnanakaran. Conformational flexibility of soluble cellulose oligomers: chain length and temperature dependence. *Journal of the American Chemical Society*, 131(41):14786–14794, October 2009. 33

[74] Oliver Biermann, Erich Hädicke, Sebastian Koltzenburg, and Florian Müller-Plathe. Hydrophilicity and Lipophilicity of Cellulose Crystal Surfaces. *Angewandte Chemie (International ed. in English)*, 40(20):3822–3825, October 2001. 33

[75] Karim Mazeau and Laurent Heux. Molecular Dynamics Simulations of Bulk Native Crystalline and Amorphous Structures of Cellulose. *Journal of Physical Chemistry B*, 107(10):2394–2403, March 2003. 33

[76] James F. Matthews, Cathy E. Skopec, Philip E. Mason, Pierfrancesco Zuccato, Robert W. Torget, Junji Sugiyama, Michael E. Himmel, and John W. Brady. Computer simulation studies of microcrystalline cellulose Ibeta. *Carbohydrate Research*, 341(1):138–152, January 2006. 33

[77] Toshifumi Yui and Sachio Hayashi. Molecular Dynamics Simulations of Solvated Crystal Models of Cellulose I alpha and III I. *Biomacromolecules*, 8(3):817–824, March 2007. 33

[78] Loukas Petridis, Roland Schulz, and Jeremy C. Smith. Simulation analysis of the temperature dependence of lignin structure and dynamics. *Journal of the American Chemical Society*, 133(50):20277–20287, December 2011. 33

[79] Stéphane Besombes and Karim Mazeau. The cellulose/lignin assembly assessed by molecular modeling. Part 2: seeking for evidence of organization of lignin molecules at the interface with cellulose. *Plant Physiology And Biochemistry*, 43(3):277–286, 2005. 33, 34

[80] Stéphane Besombes and Karim Mazeau. The cellulose/lignin assembly assessed by molecular modeling. Part 1: adsorption of a threo guaiacyl beta-O-4 dimer onto a I beta cellulose whisker. *Plant Physiology And Biochemistry*, 43(3):299–308, 2005. 33, 34

[81] Michael J. Selig, Sridhar Viamajala, Stephen R. Decker, Melvin P. Tucker, Michael E. Himmel, and Todd B. Vinzant. Deposition of lignin droplets produced during dilute acid pretreatment of maize stems retards enzymatic hydrolysis of cellulose. *Biotechnology Progress*, 23(6):1333–1339, 2007. 33

[82] Bryon S. Donohoe, Stephen R. Decker, Melvin P. Tucker, Michael E. Himmel, and Todd B. Vinzant. Visualizing lignin coalescence and migration through maize cell walls following thermochemical pretreatment. *Biotechnology and Bioengineering*, 101(5):913–925, December 2008. 33

[83] Rajeev Kumar, Gaurav Mago, Venkatesh Balan, and Charles E. Wyman. Physical and chemical characterizations of corn stover and poplar solids resulting from leading pretreatment technologies. *Bioresource Technology*, 100(17):3948–3962, September 2009. 33

[84] Gösta Brunow, Ilkka Kilpeläinen, Catherine Lapierre, Knut Lundquist, Liisa Kaarina Simola, and Juha Lemmetyinen. The chemical structure of extracellular lignin released by cultures of Picea abies. *Phytochemistry*, 32(4):845–850, March 1993. 33

[85] Yunqiao Pu, Dongcheng Zhang, Preet M. Singh, and Arthur J. Ragauskas. The new forestry biofuels sector. *Biofuels, Bioproducts and Biorefining*, 2(1):58–73, 2008. 33, 79

[86] Yoshiharu Nishiyama, Glenn P. Johnson, Alfred D. French, V. Trevor Forsyth, and Paul Langan. Neutron Crystallography, Molecular Dynamics, and Quantum Mechanics Studies of the Nature of Hydrogen Bonding in Cellulose I-beta. *Biomacromolecules*, 9(11):3133–3140, 2008. 35

[87] G. N. Ramachandran, C. Ramakrishnan, and V. Sasisekharan. Stereochemistry of polypeptide chain configurations. *Journal of Molecular Biology*, 7:95–99, July 1963. 45

[88] Hao Hu, Marcus Elstner, and Jan Hermans. Comparison of a QM/MM force field and molecular mechanics force fields in simulations of alanine and glycine "dipeptides" (Ace-Ala-Nme and Ace-Gly-Nme) in water in relation to the problem of modeling the unfolded peptide backbone in solution. *Proteins*, 50(3):451–463, February 2003. 45

[89] Jan Hermans. The amino acid dipeptide: small but still influential after 50 years. *PNAS*, 108(8):3095–3096, February 2011. 45

[90] M.-P. Gaigeot. Unravelling the conformational dynamics of the aqueous alanine dipeptide with first-principle molecular dynamics. *Journal of Physical Chemistry B*, 113(30):10059–10062, July 2009. 45

[91] Paul E. Smith. The alanine dipeptide free energy surface in solution. *Journal of Chemical Physics*, 111(12):5568–5579, 1999. 45

[92] Takeshi Kojima, Eiichi Yano, Kuniaki Nakagawa, and Shozo Tsunekawa. Microwave spectrum of acetamide in the ground torsional state. *Journal of Molecular Spectroscopy*, 112(2):494–495, August 1985. 45, 48

[93] Jerome Baudry and Jeremy C. Smith. Can Proteins and Crystals Self-Catalyze Methyl Rotations? *Journal of Physical Chemistry B*, 109(43):20572–20578, November 2005. 45

[94] Frank Noé, Illia Horenko, Christof Schütte, and Jeremy C. Smith. Hierarchical analysis of conformational dynamics in biomolecules: transition networks of metastable states. *Journal of Chemical Physics*, 126(15):155102, April 2007. 46, 48, 68

[95] Gregory R. Bowman, Xuhui Huang, and Vijay S. Pande. Using generalized ensemble simulations and Markov state models to identify conformational states. *Methods (San Diego, Calif.)*, 49(2):197–201, October 2009. 46, 48, 49

[96] M. Krishnan, V. Kurkal-Siebert, and Jeremy C. Smith. Methyl group dynamics and the onset of anharmonicity in myoglobin. *Journal of Physical Chemistry B*, 112(17):5522–5533, May 2008. 48

[97] Sanjoy Dasgupta and Philip M. Long. Performance guarantees for hierarchical clustering. *Journal of Computer and System Sciences*, 70(4):555–569, June 2005. 48

[98] Gregory R. Bowman, Kyle A. Beauchamp, George Boxer, and Vijay S. Pande. Progress and challenges in the automated construction of Markov state models for full protein systems. *Journal of Chemical Physics*, 131(12):124101, September 2009. 48, 68

[99] Frank Noé and Stefan Fischer. Transition networks for modeling the kinetics of conformational change in macromolecules. *Current Opinion in Structural Biology*, 18(2):154–162, April 2008. 49

[100] Marcus Weber. Improved Perron cluster analysis. In *ZIB Report 2003*. Konrad-Zuse-Zentrum für Informationstechnik Berlin, 2003. 49

[101] Peter Deuflhard and Marcus Weber. Robust Perron cluster analysis in conformation dynamics. *Linear Algebra and its Applications*, 398:161–184, March 2005. 49

[102] J. H. Roh, J. E. Curtis, S. Azzam, V. N. Novikov, I. Peral, Z Chowdhuri, R B Gregory, and A P Sokolov. Influence of Hydration on the Dynamics of Lysozyme. *Biophysical Journal*, 91(7):2573–2588, October 2006. 53

[103] Gerrit Coddens. Coherent quasielastic neutron scattering: A theorem about total neutron scattering functions for rotational jump diffusion of molecules on a lattice. *Physical Review B*, 63(6):064105, January 2001. 53

[104] Gerrit Coddens. Coherent quasielastic neutron scattering and correlations between rotational jumps of molecules on a periodic lattice. *The European Physical Journal B*, 31(4):533–543, February 2003. 53

[105] Irene Calvo-Almazán, Tilo Seydel, and Peter Fouquet. Questions arising for future surface diffusion studies using scattering techniques–the case of benzene diffusion on graphite basal plane surfaces. *Journal of Physics: Condensed Matter*, 22(30):304014, August 2010. 53

[106] Herman J. C. Berendsen, David van der Spoel, and Rudi van Drunen. GROMACS: A message-passing parallel molecular dynamics implementation. *Computer Physics Communications*, 91(1-3):43–56, September 1995. 55

[107] Mark T. Nelson, William Humphrey, Attila Gursoy, Andrew Dalke, Laxmikant V. Kalé, Robert D. Skeel, and Klaus Schulten. NAMD: a parallel, object-oriented molecular dynamics program. *The International Journal of Supercomputer Applications and High Performance Computing*, 10(4), December 1996. 55

[108] Alexandru M. Micu and Jeremy C. Smith. SERENA: a program for calculating X-ray diffuse scattering intensities from molecular dynamics trajectories. *Computer Physics Communications*, 91(1-3):331–338, September 1995. 55

[109] Gerald R. Kneller, Volker Keiner, Meinhard Kneller, and Matthias Schiller. nMOLDYN: A program package for a neutron scattering oriented analysis of Molecular Dynamics simulations. *Computer Physics Communications*, 91(1-3):191–214, September 1995. 55

[110] T. Róg, K. Murzyn, Konrad Hinsen, and Gerald R. Kneller. nMoldyn: a program package for a neutron scattering oriented analysis of molecular dynamics simulations. *Journal of Computational Chemistry*, 24(5):657–667, April 2003. 55

[111] B. J. Alder and T. E. Wainwright. Studies in Molecular Dynamics. I. General Method. *Journal of Chemical Physics*, 31(2):459–466, 1959. 68

[112] Aneesur Rahman. Molecular Dynamics Study of Liquid Water. *Journal of Chemical Physics*, 55(7):3336–3359, 1971. 68

[113] Jaegil Kim, John E. Straub, and Tom Keyes. Replica exchange statistical temperature molecular dynamics algorithm. *Journal of Physical Chemistry B*, 116(29):8646–8653, July 2012. 68

[114] William L. Jorgensen, Jayaraman Chandrasekhar, Jeffry D. Madura, Roger W. Impey, and Michael L. Klein. Comparison of simple potential functions for simulating liquid water. *Journal of Chemical Physics*, 79(2):926–935, 1983. 79, 80

[115] Michelle Kuttel, John W. Brady, and Kevin J. Naidoo. Carbohydrate Solution Simulations: Producing a Force Field with Experimentally Consistent Primary Alcohol Rotational Frequencies and Populations . *Journal of Computational Chemistry*, 23(13):1236–1243, 2002. 79

[116] Olgun Guvench, Elizabeth Hatcher, Richard M. Venable, Richard W. Pastor, and Alexander D. MacKerell Jr. CHARMM Additive All-Atom Force Field for Glycosidic Linkages between Hexopyranoses. *Journal of Chemical Theory and Computation*, 5(9):2353–2370, September 2009. 79

[117] Loukas Petridis and Jeremy C. Smith. A molecular mechanics force field for lignin. *Journal of Computational Chemistry*, 30(3):457–467, February 2009. 79

[118] David van der Spoel, Erik Lindahl, Berk Hess, Carsten Kutzner, Aldert R. van Buuren, Emile Apol, Pieter J. Meulenhoff, D. Peter Tieleman, Alfons L. T. M. Sijbers, K. Anton Feenstra, Rudi van Drunen, and Herman J. C. Berendsen. *Gromacs User Manual, Version 4.0*, 2010. 80

[119] W. G. Hoover. Canonical dynamics: Equilibrium phase-space distributions. *Physical Review A*, 31(3):1695–1697, March 1985. 80

[120] Herman J. C. Berendsen and J Postma. Molecular dynamics with coupling to an external bath. *Journal of Chemical Physics*, 81(8):3684, 1984. 80

[121] M. Parrinello and Aneesur Rahman. Polymorphic Transitions in Single-Crystals - a New Molecular-Dynamics Method. *Journal of Applied Physics*, 52(12):7182–7190, 1981. 80

[122] Alexander D. MacKerell Jr., D. Bashford, M. Bellott, R. L. Dunbrack Jr, J. D. Evanseck, M. J. Field, Stefan Fischer, J. Gao, H. Guo, S. Ha, D. Joseph-McCarthy, L. Kuchnir, K. Kuczera, F. T. K. Lau, C. Mattos, S Michnick, T. Ngo, D. T. Nguyen, B. Prodhom, W. E. Reiher III, Benoît Roux, M. Schlenkrich, Jeremy C. Smith, R. Stote, J. Straub, M. Watanabe, J. Wiórkiewicz-Kuczera, D. Yin, and Martin Karplus. All-atom empirical potential for molecular modeling and dynamics studies of proteins. *Journal of Physical Chemistry B*, 102(18):3586–3616, 1998. 80

Appendices

A. Simulation and Calculation Parameters

A.1. Cellulose

The library of cellulose models was constructed by starting from a fully I-β crystalline model based on the crystallinity parameters provided in Ref. [59] and a consensus model for the fiber provided in Ref. [61]. The noncrystalline regions in cellulose were attained by aligning the fiber axis along the z direction and applying strong harmonic positional constrains to the x and y coordinates of a selected segment within the cellulose fibers and simulate the fiber at a high temperature (650 K). These simulations were performed with the GROMACS MD suite [55] using the TIP3P water model [114] and the CHARMM Carbohydrate Solution Force Field (CSFF) [115] for cellulose. Since this work was completed the carbohydrate force field has been revised [116]. However, the results concerning the crystallinity vs. noncrystallinity of cellulose are general enough, to allow for inaccuracies in the cellulose force field description.

The calculation of the WAXS pattern of cellulose was performed as an isotropic orientational average over 10 times 100,000 vectors. The intense orientational averaging is necessary for the WAXS to converge, which takes signifantly longer for crystalline samples. All scattering calculations were performed using the software SASSENA.

The INS calculations were based on simulations performed at room temperatue (300 K) for solvated models of fully crystalline (n-0) and noncrystalline (p-0) cellulose and averaged over 100 random orientations. Simulations were performed for 25 ns, with a time step of 2 fs and using LINCS for hydrogen bond constrains. The coordinates were saved at intervals of 5 ps. The short time INS analysis and IXS calculation was based on simulations performed for 100 ps and a 1 fs timestep, and the coordinates were saved at each time step. The simulation protocol of the cellulose fiber was otherwise identical to the one described for Lignocellulose in Section A.2.

A.2. Lignocellulose

All models were hydrated with a 1 nm shell of explicit water, resulting in simulation sizes of 3.31, 3.43 and 3.80 million atoms for the NC, FC and FN models, respectively. All models contain 1.56 MDa of biomass. The size of the cellulose fiber and the amount of lignin used in the simulation matches the experimentally-observed ratio of molecular weights in softwood (cellulose:lignin = 1:0.52) [85]. The simulations were performed with the GROMACS MD suite [55] using the TIP3P water model [114] and the CHARMM Carbohydrate Solution Force Field (CSFF) [115] for cellulose and the CHARMM Lignin Force Field [117]. Since this work was completed the carbohydrate force field has been revised [116]. However, the changes made, which affect principally details of crystal structure geometries, are unlikely to impact any of the properties examined is the present work, which concern principally solvation and interfacial energies. Each of the models was simulated for 500 ns at 300 K using the NPT ensemble at 1 atm, with the NC model simulated twice with different initial starting velocities to examine the dependence of the aggregation on random fluctuations in the thermostat and barostat, giving a total of *2* μs simulation time. No significant difference was found between the two NC simulations (see Supporting Information of the corresponding manuscript) and hence only data for one NC model are shown. The trajectories were saved for analysis every 5 ps.

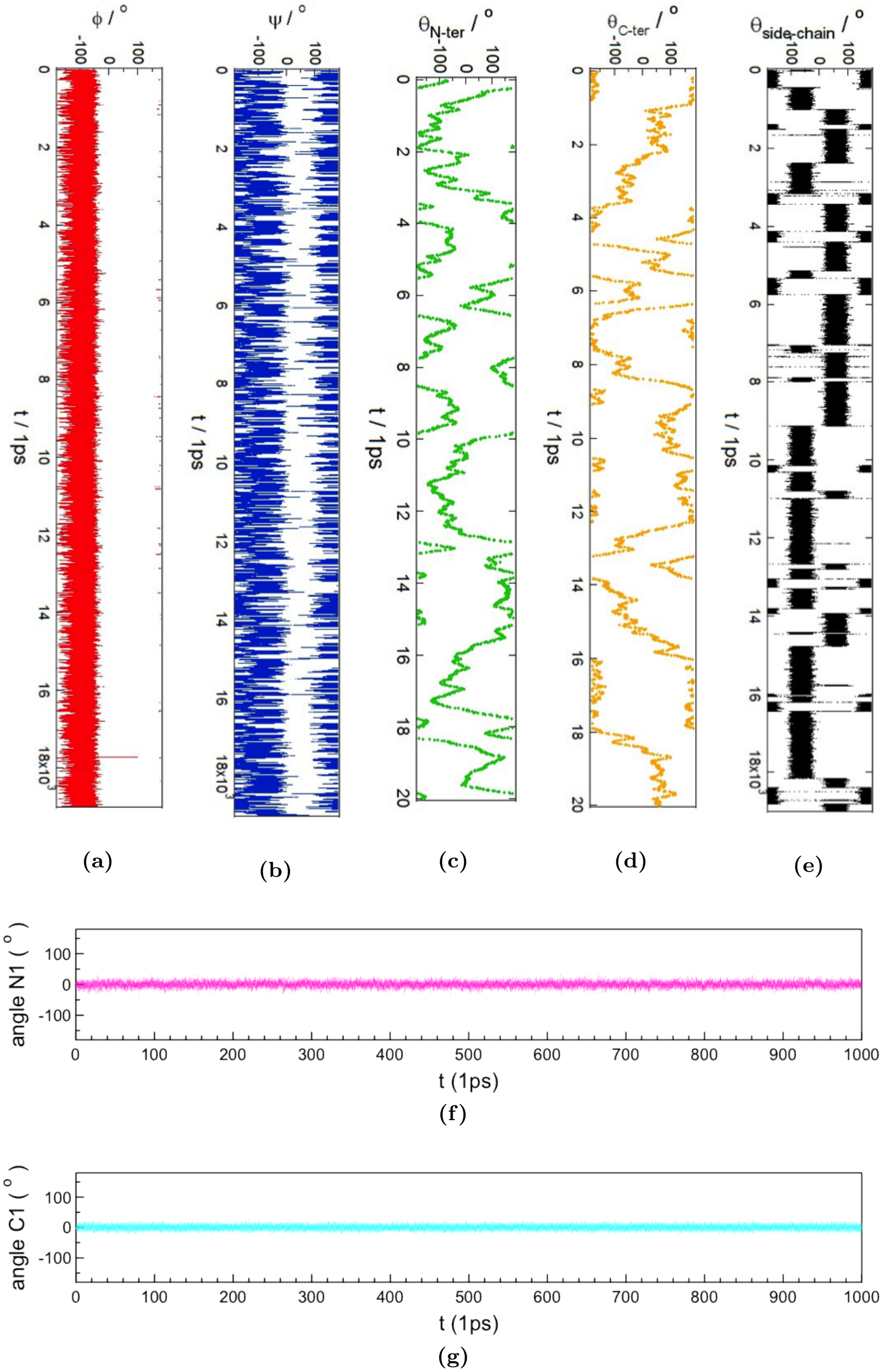
The system was simulated with periodic boundary conditions. The size of the simulation box is the result of the geometric dimensions of the cellulose fiber, the placement criteria for the lignin molecules and the additional solvent shell to allow for sufficient solvation. The mass specific hydration (biomass:water) is 8.2:91.8, 7.5:92.5 and 7.3:92.7 for the NC, FC and FN models, respectively. The non-bonded electrostatic interactions were calculated using the Reaction Field Zero (RFZ) method [38, 118] with a 1.2 nm force and 1.5 nm neighbor-list cut-off. It has been shown that RFZ is of accuracy similar to the commonly-used Particle Mesh Ewald method [34] for biomass systems while allowing significantly better parallel computational efficiency above 10000 cores [56]. Van der Waals interactions were reduced to zero between 0.8 nm and 1.2 nm using a "switch" function [118]. Bonds containing hydrogen were constrained using LINCS [43]. Water internal dynamics was constrained using the SETTLE routine [41]. All systems were simulated in the NPT ensemble. The systems were first simulated for 1 ns to reach equilibrated values for temperature and pressure. Initial equilibration for 1 ns up to 300 K in steps of 30 K and semi-anisotropic pressure at 1 atm with a simulation time step of 1 fs was followed by a production run at *300 K* and isotropic pressure coupling and a simulation time step of 2 fs. During equilibration, the temperature and pressure were controlled with the Nose-Hoover (t =1 ps) **(author?)** [119] and Berendsen algorithms **(author?)** [120] (t =4 ps), respectively, while for production, the temperature and pressure were controlled using the Berendsen thermostat (t=0.1 ps) and the Parrinello-Rahman barostat *(*t =4 ps) **(author?)** [121]. The values for t above denote the characteristic relaxation constants for the respective thermostat and barostat. Neighbor searching was performed every 10 time steps. The simulations were carried out on the Jaguar XT5 Petaflop Supercomputer at the Oak Ridge National Laboratory, using 40000 cores at a peak performance of 27 ns/day.

A.3. Alanine Dipeptide

Equilibrium MD simulations were performed using NAMD [**?**] with the CHARMM22 all-atom force field [122] for the alanine dipeptide and TIP3P for the explicitly modeled water molecules [114]. The alanine dipeptide was placed inside a cubic box with a distance of 7 Å^{-1} to the closest edge, and was solvated by water molecules. Periodic boundary conditions were used and electrostatic interactions were calculated using the Particle Mesh Ewald (PME) method [32] with a grid spacing of 1 Å^{-1}. Short range electrostatic and van der Waals interactions were switched to zero between 10 Å^{-1} and 12 Å^{-1}. Neighbor lists were updated every 10 steps and the non-bonded interactions were calculated every second step. Internal water motion was constrained using the SHAKE algorithm [42]. The simulated system was first energy minimized for 10000 steps using the conjugate gradient algorithm, followed by 1 ns MD equilibration and 1μs of production. The integration time step of the MD simulations was 1 fs and the recorded trajectory time step was 100 fs. During the equilibration the temperature was gradually increased from 0 to 300 K at a rate of 10 K/ps. The temperature was kept at 300 K using the Langevin thermostat with a 5 ps time constant coupled to the heavy atoms. The pressure was maintained at *1 atm* using the Nosé-Hoover Langevin piston barostat with a period of 100 fs, a decay time of 50 fs and a temperature of 300 K.

B. Supporting Data

B.1. Alanine Dipeptide



(a)

(b)

(c)

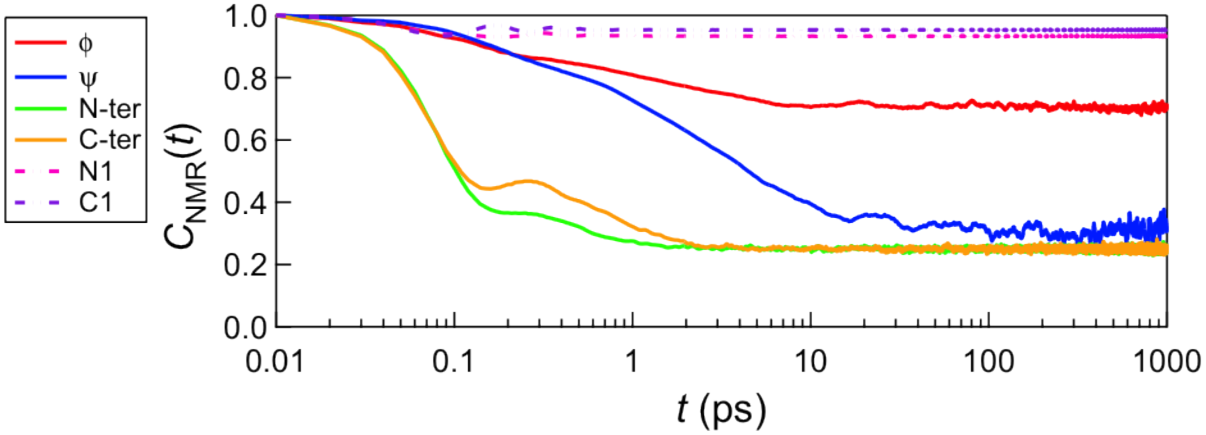
(d)

(e)

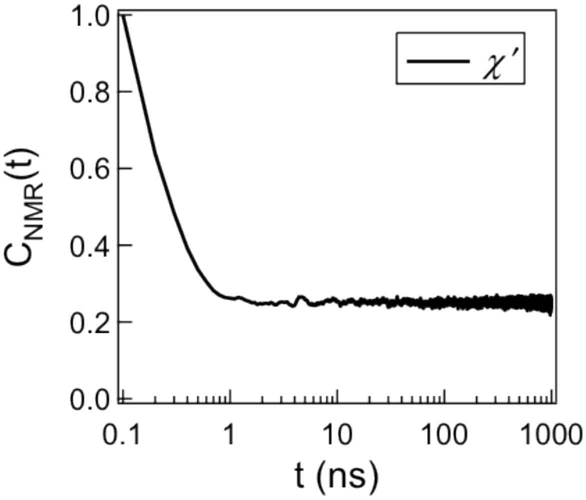
(f)

(g)

Figure B.1.: Time-dependence of seven dihedral angles (Φ, Ψ, $C1$, $N1$, $C-ter$, $N-ter$, and χ methyl angles)



(a)



(b)

Figure B.2.: The self-correlation functions of the seven dihedral angles from the $1\,\mu s$ MD trajectory.

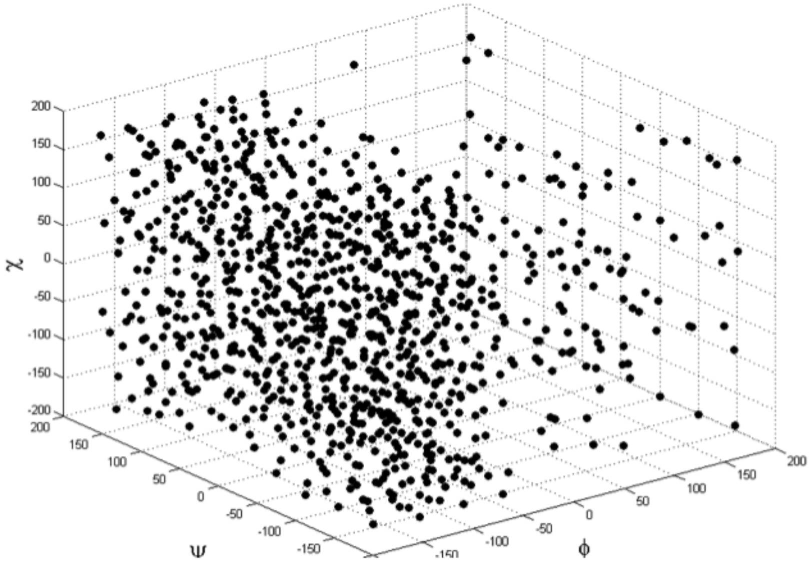
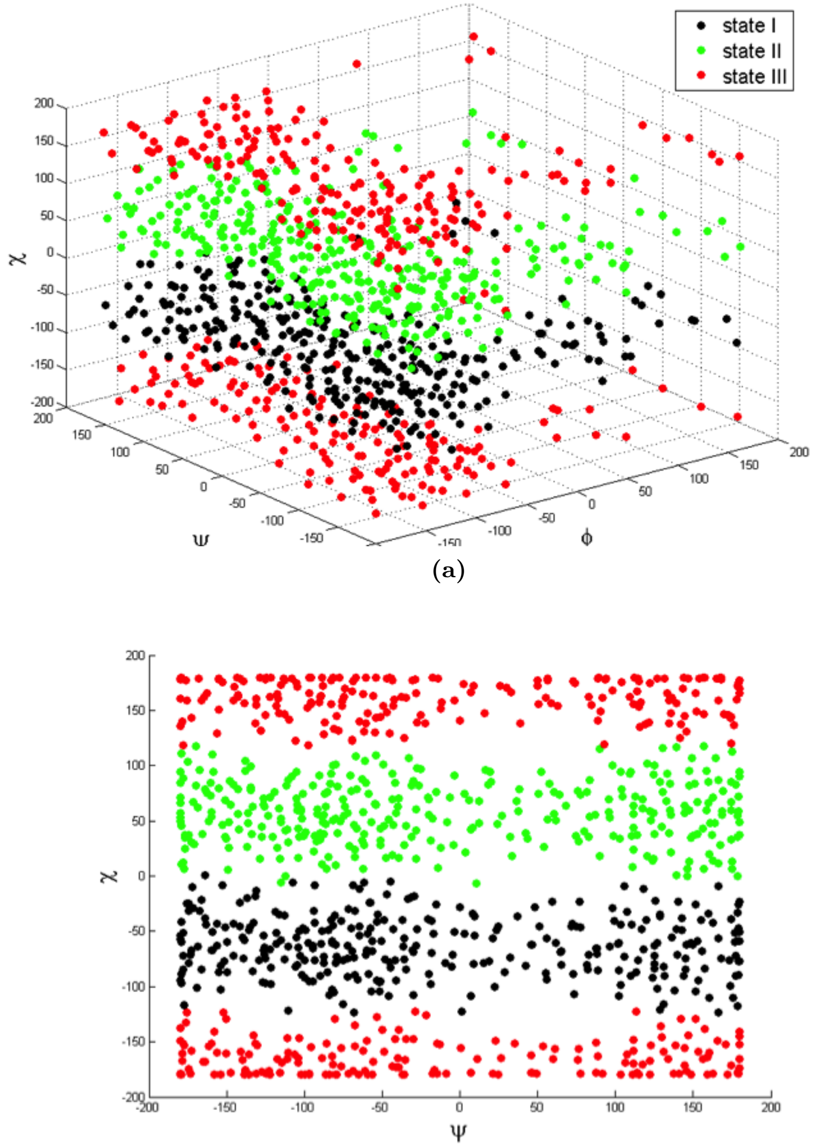


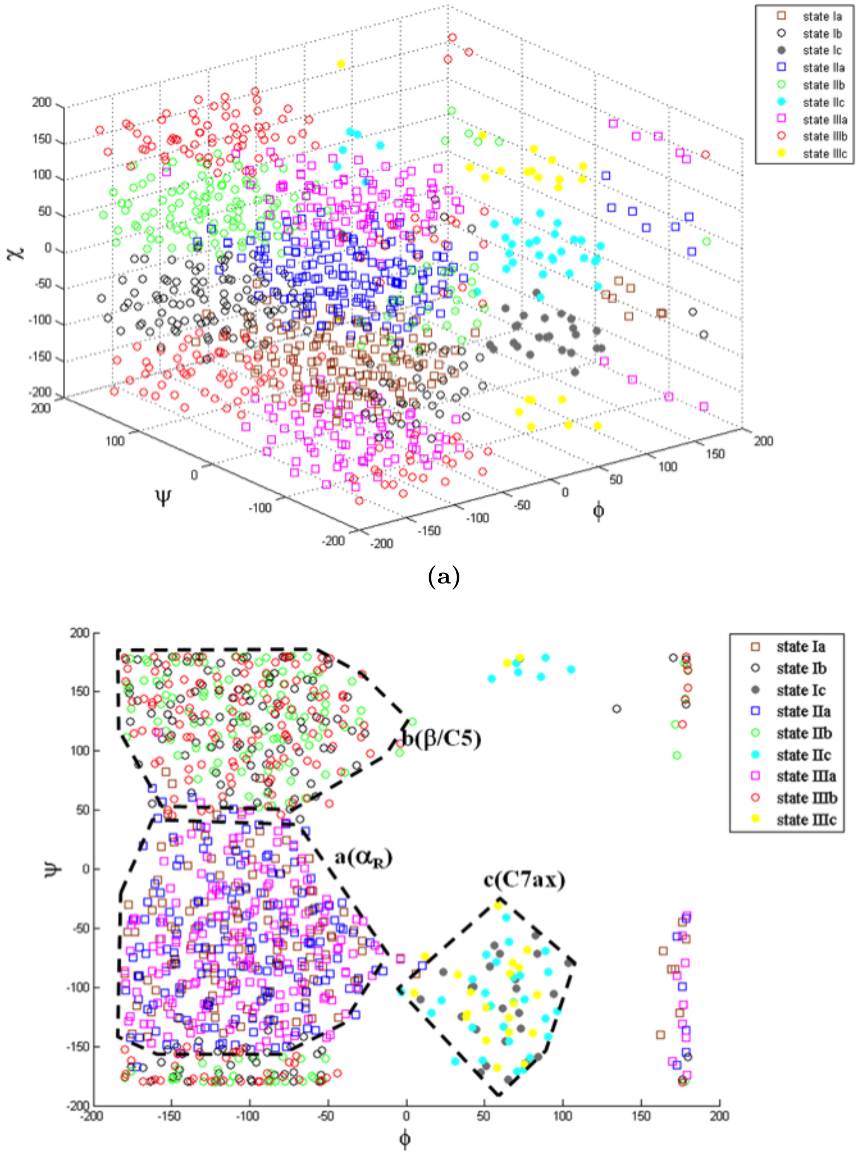
Figure B.3.: 1000 cluster centers in the configurational space of Φ, Ψ and χ using k-means clustering method.



(a)

(b)

Figure B.4.: and III in the configurational space of Φ, Ψ and χ from MSM using $m = 3$.



(a)

(b)

Figure B.5.: Cluster centers in 9 metastable states in the configurational space of Φ, Ψ and χ from MSM using $m = 9$.

Vita

Benjamin Lindner was born in Frankfurt, Germany, January 1979, and is the son of Rudolf Lindner and Eva Nusspickel. His early life was determined by his success as a sportsman in wrestling: He won the German national championship in his age and weight class in 1992, 1993, 1994, 1997 and 1998, and attended the European and World championship in 1998. He made a transition from sports to science during 1998, when he engaged in a youth science project in physics with Marcus Benna and Dominik Jednoralski, which resulted in the winning of the Bavarian youth science contest for physics in 1999 and the attendance of the national contest in the same year. After serving the community for one year by contributing to the welfare service Malteser, he pursued a Master's degree at the University of Wuerzburg in Nanostructure Technology from 2000 to 2006. During that time he participated in a student exchange program with the University of British Columbia, Vancouver, from 2002 to 2003, where he learned about soft matter science and the computational analysis of the molecular interactions within proteins. After he graduated from the University of Wuerzburg, Germany in 2006 with a diploma in Nanostructure Technology, he joined the Genome, Science & Technology program at the University of Tennessee and Oak Ridge National Laboratory in 2007 to obtain his doctoral degree in Life Sciences in 2012. During that time he acquired a diverse set of skills, stretching from the application and analysis of molecular dynamics simulation towards experimental data to high performance computational science, which is underlined by his certification by the Interdisciplinary Graduate Minor in Computational Science program.