

## nMoldyn - Interfacing spectroscopic experiments, molecular dynamics simulations and models for time correlation functions

V. Calandrini<sup>1,2,6</sup>, E. Pellegrini<sup>3</sup>, P. Calligari<sup>4</sup>, K. Hinsén<sup>1,2</sup> and G.R. Kneller<sup>1,2,5</sup>

<sup>1</sup> Centre de Biophysique Moléculaire, CNRS, Rue Charles Sadron, 45071 Orléans, France

<sup>2</sup> Synchrotron Soleil, L'Orme de Merisiers, 91192 Gif-sur-Yvette, France

<sup>3</sup> Institut Laue-Langevin, BP. 156, 6, rue Jules Horowitz, 38042 Grenoble Cedex 9, France

<sup>4</sup> Département de Chimie, associé au CNRS, École Normale Supérieure, 24 rue Lhomond, 75231 Paris Cedex 05, France

<sup>5</sup> Université d'Orléans, Château de la Source, Av. du Parc Floral, 45067 Orléans, France

<sup>6</sup> Centre de Physique Théorique, CNRS, Campus de Luminy, 13288 Marseille, France

**Abstract.** This article gives an introduction into the program *nMoldyn*, which has been originally conceived to support the interpretation of neutron scattering experiments on complex molecular systems by the calculation of appropriate time correlation functions from classical and quantum molecular dynamics simulations of corresponding model systems. Later the functionality has been extended to include more advanced time series analyses of molecular dynamics trajectories, in particular the calculation of memory functions, which play an essential role in the theory of time correlation functions. Here we present a synoptic view of the range of applications of the latest version of *nMoldyn*, which includes new modules for a simulation-based interpretation of data from nuclear magnetic resonance spectroscopy, far infrared spectroscopy and for protein secondary structure analysis.

### 1. INTRODUCTION

With the advent of computers, their usefulness for studying condensed matter systems has been immediately understood, as they permit to simulate a more or less representative part of a real system on an atomistic level in fully controlled conditions. Since the fundamental goal of statistical physics is to understand macroscopic phenomena on a microscopic, atomistic basis, computer simulations have been used from the very beginning for theoretical studies. We mention here the invention of the Monte Carlo (MC) technique by Metropolis *et al.* [1] to study the equation of state of model liquids and the pioneering molecular dynamics (MD) study of liquid argon by Rahman, which extended the scope of computer simulations to time dependent phenomena [2]. With the technical and methodological progress over the last decades, increasingly complex systems could be treated, including also electronic degrees of freedom [3]. In this way computer simulations became also interesting for experimentalists, as the systems under consideration could be represented more and more realistically.

For most spectroscopic experiments, the link between computer simulations and experimental data is established by a specific time correlation function, which is either measured directly, or indirectly in form of a Fourier spectrum. Classical molecular dynamics simulations are particularly useful to simulate spectra which are obtained from scattering of thermal neutrons. The reason is that neutrons are scattered by the atomic nuclei in the sample, if the surrounding electrons are not spin-polarized, as they are in magnetic systems [4]. Since the atomic nuclei are the objects in classical molecular

dynamics simulations (the “particles”) and since the wavelength and the energy of thermal neutrons coincide, respectively, with typical atomic distances and energies in condensed matter systems, the comparison between experiment and simulation is very direct. If simulated and measured data match well, the simulated trajectories can be analyzed in detail and give information which is not accessible in experiments. In this way hypotheses and theoretical models can be tested and the information content in the experimental data can be fully appreciated.

In this spirit, the program package *nMoldyn* has been developed since the 1990’s, focusing first on a neutron scattering oriented analysis of the dynamics of complex molecular systems (molecular liquids and solids, biomolecules) in terms of rigid body motions [5]. To enable an efficient time series analysis of the MD trajectories, the latter were treated as two-dimensional arrays, enabling direct access to the time series of single atoms. In this way Fast Fourier Transform (FFT) techniques could be used for the calculation of certain types of correlation functions, allowing for speedups of several orders of magnitude compared to conventional techniques. Typical applications can be found in [6–9]. A completely revised version of *nMoldyn* was published in 2003 [10], extending the functionality essentially to autoregressive (AR) modeling of MD trajectories, including the calculation of memory functions [11], which are the pivot element in the theory of the generalized Langevin equation [12]. Applications in this context were the study of fractional Brownian dynamics in proteins on the nanosecond time scale [13] and the refinement of a memory-function based model for liquid water [14]. The advantage of AR modeling for a computation of Fourier spectra without broadening through spectral windows has been exploited in recent studies of lysozyme under pressure [15, 16]. From a technical point of view, the code was completely rewritten, passing from Fortran 77 to object-oriented programming in Python, using as a basis the Molecular Modeling Toolkit (MMTK) [17]. The 2003 version (*nMoldyn2*) shares in particular the data structure for trajectories with MMTK and has a graphical user interface.

Over the last years, an effort has been made to interface *nMoldyn* with NMR relaxation spectroscopy, and recently also with Terahertz spectroscopy, for the study of protein dynamics and to include in this context also analyses of the dynamics of secondary structure elements with the ScrewFit algorithm [18]. In the following we give a short overview of the latest version of *nMoldyn*, emphasizing in particular the new functionalities.

## 2. TIME CORRELATION FUNCTIONS

### 2.1 Definition

Most of the functionalities of *nMoldyn* imply the computation of *time correlation functions* (TCF) of specific dynamical variables. They are the pivot elements in both theoretical and experimental studies of systems close to thermal equilibrium. In the following we recall some basic properties of TCFs.

TCFs are special forms of thermal averages, in which the quantity to be averaged is the product of two dynamical variables, each taken at a different time. In statistical mechanics the thermal average of a dynamical variable  $A$  is defined as an *ensemble average*,

$$\langle A(t) \rangle = \int_{\Gamma} d^{6N} \Gamma f(\Gamma) A(\Gamma; t), \quad (2.1)$$

where  $\Gamma$  is the  $6N$  dimensional “position vector”  $\Gamma = \{\mathbf{R}_1, \dots, \mathbf{R}_N, \mathbf{P}_1, \dots, \mathbf{P}_N\}$  in the phase space spanned by all coordinates and momenta of the  $N$  particles (atoms) in the system, and  $f(\Gamma)$  is the equilibrium probability density which is determined by the Hamiltonian of the system. Correspondingly a TCF is defined as

$$c_{AB}(t_1, t_0) \doteq \langle A(t_1)B(t_0) \rangle = \int_{\Gamma} d\Gamma f(\Gamma) A(\Gamma; t_1) B(\Gamma; t_0). \quad (2.2)$$

Since molecular dynamics simulations generate phase space trajectories of fixed length  $T$  by a stepwise integration of the equations of motion, ensemble averages are computed through time averages, assuming that the ergodic hypothesis holds [19]

$$\langle A(t_0) \rangle = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} d\tau A(\tau + t_0), \quad (2.3)$$

$$\langle A(t_1)B(t_0) \rangle = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} d\tau A(\tau + t_1)B(\tau + t_0). \quad (2.4)$$

In thermal equilibrium, thermodynamic averages and TCFs are invariant with respect to a shift of the time arguments (stationarity), which means that  $\langle A(t) \rangle$  does not depend on  $t$  and that  $\langle A(t_1)B(t_0) \rangle$  depends only on the time-difference  $t_1 - t_0 = t$ , i.e.  $\langle A(t_1)B(t_0) \rangle = \langle A(t_1 - t_0)B(0) \rangle \equiv \langle A(t)B(0) \rangle$ . In the following we will use the notation

$$c_{AB}(t) \doteq \langle A(t)B(0) \rangle \quad (2.5)$$

and it follows that from the stationarity of equilibrium TCFs that

$$c_{AB}(t) = c_{BA}(-t). \quad (2.6)$$

In the framework of classical statistical physics, autocorrelation functions (ACFs) are thus symmetric in time.

## 2.2 Wiener-Khinchin theorem

There exists an important relation between the Fourier spectrum of a signal  $A$  and the Fourier transform of its autocorrelation function. Defining the Fourier transform of  $f$  and its inverse, respectively, through

$$\tilde{f}(\omega) = \int_{-\infty}^{+\infty} dt f(t) \exp(-i\omega t) \equiv \mathcal{F}\{f(t), t, \omega\}, \quad (2.7)$$

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} d\omega \tilde{f}(\omega) \exp(i\omega t) \equiv \mathcal{F}^{-1}\{\tilde{f}(\omega), \omega, t\}, \quad (2.8)$$

and the correlation integral

$$(f \circ g)(t) = \int_{-\infty}^{+\infty} d\tau f(t + \tau)g^*(\tau), \quad (2.9)$$

it is straightforward to show that

$$\mathcal{F}\{(f \circ g)(t), t, \omega\} = \tilde{f}(\omega)\tilde{g}^*(\omega). \quad (2.10)$$

Defining a signal  $a_T(t)$  of length  $T$ , such that

$$a_T(t) = \begin{cases} a(t) & \text{if } |t| < T/2, \\ 0 & \text{otherwise,} \end{cases}$$

it follows that  $c_{aa^*}(t) = \lim_{T \rightarrow \infty} (a_T \circ a_T)(t)/T$ , and with (4.18)

$$\tilde{c}_{aa^*}(\omega) = \lim_{T \rightarrow \infty} \frac{1}{T} |\tilde{a}_T(\omega)|^2. \quad (2.11)$$

This is the famous Wiener-Khinchin theorem, relating the Fourier transformed ACF of a signal to its power spectrum [19, 20].

### 2.3 Mean square displacements

The mean square displacement (MSD) of a diffusing particle, defined as

$$W(t) \equiv \langle (\mathbf{x}(t) - \mathbf{x}(0))^2 \rangle = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} d\tau (\mathbf{x}(t + \tau) - \mathbf{x}(\tau))^2, \quad (2.12)$$

where  $\mathbf{x}$  is its position, plays a fundamental role in the description of diffusion processes. It has a close relation to the velocity autocorrelation function of the tagged particle,

$$W(t) = 2 \int_0^t dt' (t - t') \langle \mathbf{v}(t') \cdot \mathbf{v}(0) \rangle. \quad (2.13)$$

For spatially unconstrained, “normal” diffusion, the MSD follows asymptotically Einstein’s law [21],

$$W(t) \sim 2n_d D t, \quad t \rightarrow \infty, \quad (2.14)$$

where  $D$  is the diffusion coefficient and  $n_d$  the dimension of  $\mathbf{x}$ . The diffusion constant can be expressed in form of the Kubo-relation [22]

$$D = \frac{1}{n_d} \int_0^\infty dt \langle \mathbf{v}(t) \cdot \mathbf{v}(0) \rangle. \quad (2.15)$$

If the motion of the diffusing particle is spatially constrained, the MSD tends for long times to a plateau value,

$$\lim_{t \rightarrow \infty} W(t) = 2(\langle \mathbf{x}^2 \rangle - \langle \mathbf{x} \rangle^2). \quad (2.16)$$

### 2.4 Generalized Langevin equation

Memory functions play a key role in the theory of the generalized Langevin equation (GLE). To introduce the concept, it is useful to consider the GLE of a tagged particle in a liquid or a many particle system in general. For simplicity we consider motion in a particular direction and  $v$  is the velocity of the particle in that direction. The equation of motion of the tagged particle can then be cast into the form [12],

$$\dot{v} = - \int_0^t d\tau \zeta(t - \tau) v(\tau) + f^+(t), \quad (2.17)$$

where  $\zeta(\cdot)$  is the *memory function*. The GLE (2.17) resembles the Langevin equation [23], with  $f^+(\cdot)$  being the “random force” and the convolution integral being the “friction force”. The GLE is, however, an *exact* equation of motion, and the memory function as well as  $f^+(\cdot)$  can be expressed in terms of the microscopic forces acting between the tagged particle and the rest of the system. They represent, in fact, the remaining system. For details we refer to the excellent monograph by Zwanzig [12]. The important point is that  $f^+(\cdot)$  is constructed such that

$$\langle v(0) f^+(t) \rangle = 0, \quad t > 0, \quad (2.18)$$

in analogy to the corresponding relation in the normal Langevin equation, where  $f^+(t)$  is white noise. Due to relation (2.18), one can derive a closed equation for the VACF,

$$\frac{dc_{vv}(t)}{dt} = - \int_0^t d\tau \zeta(t - \tau) c_{vv}(\tau), \quad (2.19)$$

which has the starting point for the development of various models for the time evolution of the VACF [22, 24]. The reason is that rather effective approximations to the time correlation functions of interest can be obtained by relatively simple models for the memory function. In the simplest case, where  $\zeta(t) = \gamma \delta(t)$ , Eq. (2.19) becomes a normal differential equation, whose solution is an exponential,

$c_{vv}(t) = c_{vv}(0) \exp(-\gamma t)$ , which is well known as the VACF of a particle whose dynamics follows a Langevin equation. The latter describes thus “memory-less” motion, representing more precisely a time scale separation between the dynamics of the tagged particle and its environment.

The derivation of the expressions for the memory function and  $f^+(t)$  does not depend on the dynamical variable under consideration, and one can derive a generalized Langevin equation for any dynamical variable,  $A$

$$\dot{A} = - \int_0^t d\tau \xi_{AA}(t - \tau)v(\tau) + f_A^+(t), \quad (2.20)$$

and the ACF of  $A$  verifies

$$\frac{dc_{AA}(t)}{dt} = - \int_0^t d\tau \xi_{AA}(t - \tau)c_{AA}(\tau). \quad (2.21)$$

For a complete description on the memory function formalism the reader is referred to standard textbooks (see for example [22]).

### 3. CALCULATIONS WITH NMOLDYN

In this section we give an overview of the quantities which can be calculated by  $n$ Moldyn, with some examples of applications. Most analyses involve averages over atoms, which can be attributed specific weights. For this purpose a new and powerful atom selection scheme has been implemented.

#### 3.1 Velocity autocorrelation function and density of states

$n$ Moldyn allows to calculate the atom-averaged velocity autocorrelation function,

$$c_{vv}(t) = \sum_{\alpha=1}^N w_{\alpha} c_{vv,\alpha}(t), \quad \text{with} \quad \sum_{\alpha} w_{\alpha} = 1, \quad (3.1)$$

where the atomic VACF can be calculated with two options:

1. Isotropic average:  $c_{vv,\alpha}(t) \doteq \frac{1}{3} \langle \mathbf{v}_{\alpha}(0) \cdot \mathbf{v}_{\alpha}(t) \rangle$ , where  $\mathbf{v}_{\alpha}$  is the velocity of atom  $\alpha$ .
2. Along a given direction:  $c_{vv;\alpha}(t) \doteq \langle v_{\alpha,n}(0) v_{\alpha,n}(t) \rangle$ , where  $v_{\alpha,n}(t) = \mathbf{n} \cdot \mathbf{v}_{\alpha}(t)$  and  $\mathbf{n}$  is an arbitrary unit vector.

Furthermore,  $n$ Moldyn calculates the so-called density of states (DOS), which is defined as the Fourier cosine transform of the VACF,

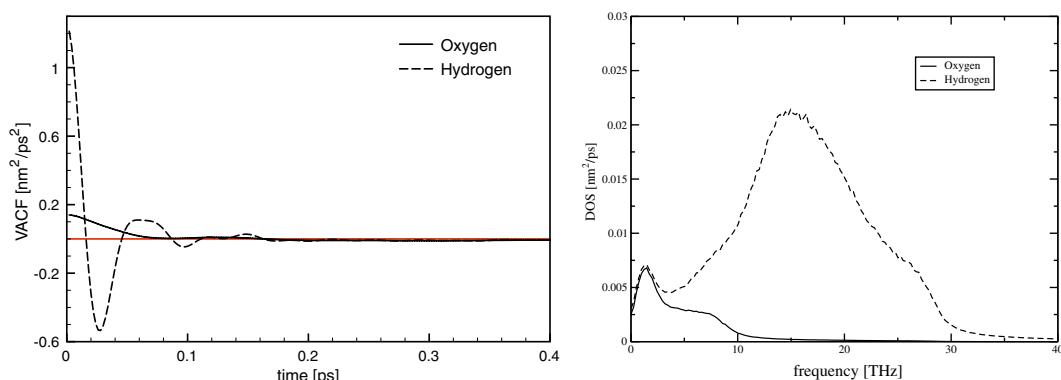
$$g(\omega) = \int_0^{\infty} dt \cos(\omega t) c_{vv}(t). \quad (3.2)$$

From the definition of the DOS and the Kubo relation (2.15) it follows that

$$g(0) = D. \quad (3.3)$$

**Algorithmic considerations:** The VACF can be either calculated with the Fast Fourier Transform (FFT) based Algorithm described in Section 4.1 or by autoregressive modeling of the underlying velocity time series, which is described in Section 4.5. Correspondingly, the DOS is calculated by a smoothed discrete Fourier transform or by the so-called maximum entropy method, which are described in Sections 4.3 and 4.4, respectively. A comparison between these computational approaches can be found in references [11] and [10] for liquid argon and pure water respectively.

**Application:** Figure 1 shows the VACF and the DOS of hydrogen and oxygen atoms in water. In the DOS plot one can recognize two peaks centered at about 2 and 8 THz, appearing in both the hydrogen and the oxygen spectra, and a more pronounced peak at about 15 THz, appearing only in the hydrogen



**Figure 1.** *Left:* VACF of hydrogen and oxygen atoms in water. *Right:* DOS of hydrogen and oxygen atoms in water. Here the contribution of each atom has been weighted by the incoherent scattering length. The input trajectory has been computed using the simulation program DLPOLY (version2) [25]. The simulated system consists of 256 water molecules in a cubic box of edge length  $1.9552\text{ nm}$ . The simulations have been performed in the thermodynamic  $NpT$ -ensemble at a temperature of  $T = 300\text{ K}$  and a pressure of  $1\text{ bar}$ , using the SPC/E potential [26] and Ewald summation for long-range electrostatic interactions. The integration time step was  $1\text{ fs}$ . The trajectory length is  $100\text{ ps}$  with a sampling interval of  $10\text{ fs}$ .

spectrum. Since the two peaks at lower frequency are identical in both hydrogen and oxygen spectra, they are usually associated to the diffusion motion of the whole water molecule. On the contrary, the peak at higher frequency is usually interpreted in terms of hindered motions around the center of mass of the water molecule, which is almost coincident with the oxygen atom. This behavior is reflected in the VACFs. The latter are characterized by a damped oscillatory decay, more pronounced for the hydrogen than for the oxygen, which can be related to the confinement of the tagged particle in the cage formed by its neighbors. Negative values of the the VACF reflect likely changes in the particle velocity due to “rattling” motions of a tagged particle in the cage of nearest neighbors.

### 3.2 Mean-square displacement

In  $n$ Moldyn the mean square displacement is an atom-averaged quantity,

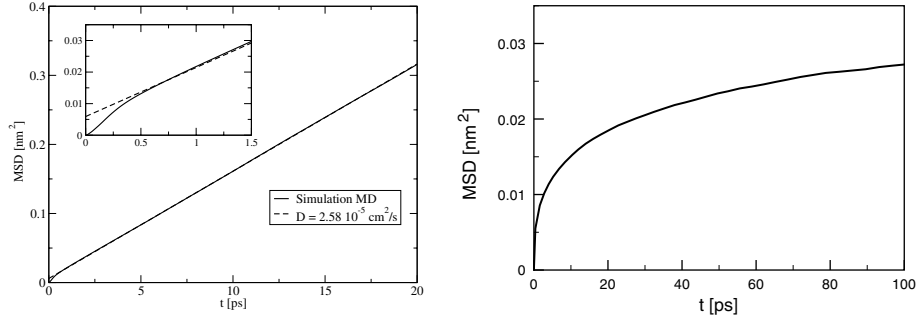
$$W(t) = \sum_{\alpha=1}^N w_{\alpha} W_{\alpha}(t), \quad \text{with} \quad \sum_{\alpha} w_{\alpha} = 1, \quad (3.4)$$

where  $W_{\alpha}(t)$  is the MSD of atom  $\alpha$  which can be calculated according to the following options:

1. Total isotropic MSD:  $W_{\alpha}(t) \doteq \langle (\mathbf{x}_{\alpha}(t) - \mathbf{x}_{\alpha}(0))^2 \rangle$ .
2. Along a given direction  $\mathbf{n}$ :  $W_{\alpha}(t) \doteq \langle (x_{\alpha,n}(t) - x_{\alpha,n}(0))^2 \rangle$ , where  $x_{\alpha,n}(t) = \mathbf{n} \cdot \mathbf{x}_{\alpha}(t)$  and  $\mathbf{n}$  is an arbitrary unit vector.

**Algorithmic considerations:** The MSD is calculated with the Fast Fourier Transform (FFT) based Algorithm described in Section 4.2.

**Application:** In the left panel of figure 2 we show as an example the MSD of pure water. For time lags larger than  $0.5\text{ ps}$ , the MSD is well described by a straight line as expected for a Brownian particle diffusing in the absence of external forces. Using equation (2.14) one can calculate the diffusion coefficient, whose value is reported in the legend of the left panel of figure 2. The atomic contributions have been weighted by  $w_{\alpha} \propto b_{\alpha,inc}^2$ , where  $\{b_{\alpha,inc}\}$  are the incoherent scattering lengths of the atoms. In this type of average the contributions from the hydrogen atoms dominate. The diffusion coefficient can



**Figure 2.** *Left:* MSD of pure water. Here the contribution of each atom has been weighted by the incoherent scattering length. The input trajectory is the one used for the analysis shown in Figure 1. *Right:* MSD of lysozyme in water. Molecular dynamics simulations have been performed for a system consisting of one lysozyme molecule surrounded by 3403 water molecules, using the program package MMTK [17] with the AMBER94 force field. [27] The system was enclosed in a rectangular box of an average size of  $6.15 \times 4.10 \times 4.61 \text{ nm}^3$ , applying periodic boundary conditions. The lysozyme structure was taken from the Brookhaven protein data bank [28] (code 193L[29]) and hydrogen atoms were added to the structure according to standard criteria concerning the chemical bond structure of amino acids. To mimic realistic thermodynamic conditions the system was simulated in the thermodynamics  $NpT$ -ensemble, employing the extended system method [30, 31] with an integration time step of  $0.001 \text{ ps}$ . Global translations and rotations of the lysozyme molecule have been subtracted by superposing its configuration for each time frame with the corresponding initial configuration [32]. This option is now available within the new version of *nMoldyn*.

also be extracted from the DOS shown in the right panel of Fig. 1, using relation (3.3) and noting that the long time diffusion constants of hydrogen and oxygen in a water molecule are the same.

In the case of particles performing confined motions in space the MSD tends to plateau value in the long time limit (see Eq. (2.16)). An example is provided by atomic motions in a protein once the global motion has been removed. To illustrate this fact we show, in the right panel of figure 2, the MSD averaged over all atoms in a lysozyme molecule, using again the weights  $w_\alpha \propto b_{\alpha,inc}^2$ .

### 3.3 Neutron intermediate scattering functions and dynamic structure factors

#### 3.3.1 Intermediate scattering functions

In the following we consider correlation functions of the spatially Fourier-transformed particle density. With

$$\rho_\alpha(\mathbf{x}, t) = \delta(\mathbf{x} - \mathbf{x}_\alpha(t)) \quad (3.5)$$

being the probability density to find a single “point-like” particle (atom)  $\alpha$  at time  $t$  at position  $\mathbf{x}$ , ( $\delta(\cdot)$  denotes the Dirac delta distribution) and

$$\tilde{\rho}_\alpha(\mathbf{q}, t) = \int d^3x \exp(-i\mathbf{q} \cdot \mathbf{x}) \rho_\alpha(\mathbf{x}, t) = \exp(-i\mathbf{q} \cdot \mathbf{x}_\alpha(t)), \quad (3.6)$$

its spatial Fourier transform, we define

$$\tilde{\rho}(\mathbf{q}, t) = \sum_\alpha w_{\alpha,coh} \tilde{\rho}_\alpha(\mathbf{q}, t). \quad (3.7)$$

In terms of the above definitions the coherent and incoherent *intermediate scattering functions* [4, 33, 34], respectively, are defined as the (weighted) autocorrelation functions

$$\mathcal{F}_{coh}(\mathbf{q}, t) = \langle \tilde{\rho}(\mathbf{q}, t) \tilde{\rho}^*(\mathbf{q}, 0) \rangle, \quad (3.8)$$

$$\mathcal{F}_{\text{inc}}(\mathbf{q}, t) = \sum_{\alpha} w_{\alpha, \text{inc}} \langle \tilde{\rho}_{\alpha}(\mathbf{q}, t) \tilde{\rho}_{\alpha}^*(\mathbf{q}, 0) \rangle. \quad (3.9)$$

One sees that  $\mathcal{F}_{\text{coh}}(\mathbf{q}, t)$  gives information about collective motions, whereas  $\mathcal{F}_{\text{inc}}(\mathbf{q}, t)$  describes single particle correlations. The weights are normalized, such that

$$\sum_{\alpha} w_{\alpha, \text{coh}}^2 = 1, \quad (3.10)$$

$$\sum_{\alpha} w_{\alpha, \text{inc}} = 1. \quad (3.11)$$

The standard choice for comparison to experimental data is  $w_{\alpha, \text{coh}} \propto b_{\alpha, \text{coh}}$  and  $w_{\alpha, \text{inc}} \propto b_{\alpha, \text{inc}}^2$ , where  $b_{\alpha, \text{coh}}$  and  $b_{\alpha, \text{inc}}$  are, respectively, the coherent and incoherent scattering length of atom  $\alpha$ . With the normalizations (3.10) and (3.11) it follows that

$$\lim_{|\mathbf{q}| \rightarrow \infty} \mathcal{F}_{\text{coh}}(\mathbf{q}, 0) = 1, \quad (3.12)$$

$$\mathcal{F}_{\text{inc}}(\mathbf{q}, 0) = 1, \quad (3.13)$$

where

$$\mathcal{F}_{\text{coh}}(\mathbf{q}, 0) \equiv \mathcal{S}(\mathbf{q}) \quad (3.14)$$

is the *static structure factor*.

### 3.3.2 Gaussian approximation of the incoherent intermediate scattering function

The mean-square displacement can be related to the incoherent intermediate scattering function via the cumulant expansion [35, 36]

$$\mathcal{F}_{\text{inc}}(\mathbf{q}, t) = \sum_{\alpha} w_{\alpha, \text{inc}} \exp[-q^2 \rho_{\alpha, 1}(t) + q^4 \rho_{\alpha, 2}(t) \mp \dots]. \quad (3.15)$$

Here  $q = |\mathbf{q}|$ , and the cumulants  $\rho_{\alpha, k}(t)$  are defined as

$$\rho_{\alpha, 1}(t) = \frac{1}{2!} \langle d_{\alpha}^2(t; \mathbf{n}_q) \rangle, \quad (3.16)$$

$$\rho_{\alpha, 2}(t) = \frac{1}{4!} \{ \langle d_{\alpha}^4(t; \mathbf{n}_q) \rangle - 3 \langle d_{\alpha}^2(t; \mathbf{n}_q) \rangle^2 \}, \quad (3.17)$$

⋮

where  $d_{\alpha}(t; \mathbf{n}_q)$  is defined as

$$d_{\alpha}(t; \mathbf{n}) \doteq \mathbf{n}_q \cdot (\mathbf{x}_{\alpha}(t) - \mathbf{x}_{\alpha}(0)), \quad (3.18)$$

and  $\mathbf{n}_q$  is the projection of the displacement  $\mathbf{x}_{\alpha}(t) - \mathbf{x}_{\alpha}(0)$  onto the unit vector in the direction of  $\mathbf{q}$ . In the Gaussian approximation the above expansion is truncated after the  $q^2$ -term. For certain model systems like the ideal gas, the harmonic oscillator, and a particle undergoing Einstein diffusion, this is exact. For these systems the incoherent intermediate scattering function is completely determined by the mean-square displacement. A comparison between the full calculation of the incoherent scattering function and its Gaussian approximation is shown in reference [10] for pure water.



### 3.3.3 Dynamic structure factor

Closely related to the intermediate scattering functions is the *dynamic structure factor*,  $\mathcal{S}(\mathbf{q}, \omega)$ , which is the quantity directly measured by neutron scattering experiments. Defining its coherent and incoherent component through

$$\mathcal{S}_{\text{coh/inc}}(\mathbf{q}, \omega) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} dt \exp(-i\omega t) \mathcal{F}_{\text{coh/inc}}(\mathbf{q}, t), \quad (3.19)$$

the total differential neutron scattering cross section is given by

$$\frac{d^2\sigma}{d\Omega d\omega} = \frac{|\mathbf{k}|}{|\mathbf{k}_0|} \left\{ \frac{\sigma_{\text{tot,coh}}}{4\pi} \mathcal{S}_{\text{coh}}(\mathbf{q}, \omega) + \frac{\sigma_{\text{tot,inc}}}{4\pi} \mathcal{S}_{\text{inc}}(\mathbf{q}, \omega) \right\} \quad (3.20)$$

Here  $d\Omega$  denotes a differential solid angle element,  $\hbar\mathbf{k}_0$  is the momentum of the incident neutrons,  $\hbar\mathbf{k}$  the momentum of the scattered neutrons,  $\hbar\mathbf{q} = \hbar(\mathbf{k}_0 - \mathbf{k})$  the momentum transfer from the neutron to the sample, and  $\hbar\omega = E_0 - E$  the corresponding energy transfer. Moreover  $\sigma_{\text{tot,coh}} = 4\pi \sum_{\alpha} b_{\alpha,\text{coh}}^2$  denotes the total coherent scattering cross section and  $\sigma_{\text{tot,inc}} = 4\pi \sum_{\alpha} b_{\alpha,\text{inc}}^2$  the incoherent counterpart. On account of (3.14) the static structure factor may be written as

$$\mathcal{S}(\mathbf{q}) = \int_{-\infty}^{+\infty} d\omega \mathcal{S}_{\text{coh}}(\mathbf{q}, \omega). \quad (3.21)$$

### 3.3.4 Symmetry relations and detailed balance

The dynamic structure factor being a measurable quantity, it must be real. This condition is fulfilled if and only if (the indices ‘‘coh’’ and ‘‘inc’’ are omitted)

$$\mathcal{F}^*(\mathbf{q}, t) = \mathcal{F}(\mathbf{q}, -t). \quad (3.22)$$

This relation is indeed verified, which follows from (2.6), setting  $A(t) = \tilde{\rho}(\mathbf{q}, t)$ ,  $B(0) = \tilde{\rho}^*(\mathbf{q}, 0)$  and using the stationarity of equilibrium time correlation functions. By definition, the intermediate scattering functions calculated from MD simulations are classical correlation functions and must be considered as approximations for the quantum correlation functions describing the full neutron scattering law [37]. The quantum intermediate scattering functions fulfill also (3.22), ensuring thus the reality of the dynamic structure factor. The following symmetry relation,

$$\mathcal{F}(\mathbf{q}, t) = \mathcal{F}(-\mathbf{q}, -t - i\beta\hbar), \quad (3.23)$$

differs, however, from the corresponding relation of their classical counterparts, which are obtained by the formal limit  $\hbar \rightarrow 0$ . From (3.23) one obtains the *detailed balance relation* for the dynamic structure factor

$$\mathcal{S}(\mathbf{q}, \omega) = \mathcal{S}(-\mathbf{q}, -\omega) \exp(-\beta\hbar\omega). \quad (3.24)$$

Here  $\beta = (k_B T)^{-1}$ , where  $k_B$  is the Boltzmann constant and  $\hbar$  the Planck constant divided by  $2\pi$ . Since  $\hbar\omega$  is here the energy transfer to the scattering system, relation (3.24) expresses that energetically higher levels in the scattering system are lower weighted than energetically lower levels, the weights being given by the Boltzmann factor. From a physical point of view the description of neutron scattering by a scattering law derived from classical correlation functions corresponds to neglecting momentum transfer from the neutron to the sample, even when the scattering system can be described by the laws of classical mechanics [38].

### 3.3.5 Partial terms

In the analysis of neutron scattering spectra it is sometimes desirable to consider groups of atoms with common properties (species) in the analysis of MD simulations. A typical application is the decomposition of the coherent intermediate scattering functions or dynamic structure factors into “self” and “distinct” contributions from such groups, which are denoted  $\mathcal{M}_I$  ( $I = 1, \dots, N_{\text{species}}$ ) in the following. For all atoms  $\alpha \in \mathcal{M}_I$ , we then have the same weight. Defining

$$\rho_I(\mathbf{q}, t) = \frac{1}{\sqrt{n_I}} \sum_{\alpha \in \mathcal{M}_I} \tilde{\rho}_\alpha(\mathbf{q}, t), \quad (3.25)$$

where  $n_I$  is the number of atoms in  $\mathcal{M}_I$ , with  $\sum_{I=1}^{N_{\text{species}}} n_I = N_{\text{atoms}}$ , and

$$w_{I,\text{coh}} = \sqrt{n_I} w_{\alpha,\text{coh}}, \quad \text{for } \alpha \in \mathcal{M}_I, \quad (3.26)$$

we have

$$\mathcal{F}_{\text{coh}}(\mathbf{q}, t) = \sum_{I,J=1}^{N_{\text{species}}} w_{I,\text{coh}} w_{J,\text{coh}} \mathcal{F}_{\text{coh},IJ}(\mathbf{q}, t), \quad (3.27)$$

where the *partial intermediate scattering functions* are given by

$$\mathcal{F}_{\text{coh},IJ}(\mathbf{q}, t) = \langle \tilde{\rho}_I(\mathbf{q}, t) \tilde{\rho}_J^*(\mathbf{q}, 0) \rangle. \quad (3.28)$$

Analogously one may define the partial incoherent intermediate scattering functions

$$\mathcal{F}_{\text{inc},I}(\mathbf{q}, t) = \frac{1}{n_I} \sum_{\alpha \in \mathcal{M}_I} \langle \tilde{\rho}_\alpha(\mathbf{q}, t) \tilde{\rho}_\alpha^*(\mathbf{q}, 0) \rangle, \quad (3.29)$$

and the weights

$$w_{I,\text{inc}} = n_I w_{\alpha,\text{inc}}, \quad \text{for } \alpha \in \mathcal{M}_I, \quad (3.30)$$

such that the total incoherent scattering takes the form

$$\mathcal{F}_{\text{inc}}(\mathbf{q}, t) = \sum_{I=1}^{N_{\text{species}}} w_{I,\text{inc}} \mathcal{F}_{\text{inc},I}(\mathbf{q}, t). \quad (3.31)$$

It follows from (3.26) and (3.30) that the species weights are normalized as their atomic counterparts, i.e.  $\sum_{I=1}^{N_{\text{species}}} w_{I,\text{coh}}^2 = 1$  and  $\sum_{I=1}^{N_{\text{species}}} w_{I,\text{inc}} = 1$ .

The partial intermediate scattering functions introduced above fulfill the symmetry relations

$$\mathcal{F}_{\text{coh},IJ}^*(\mathbf{q}, t) = \mathcal{F}_{\text{coh},JI}(\mathbf{q}, -t), \quad (3.32)$$

$$\mathcal{F}_{\text{inc},I}^*(\mathbf{q}, t) = \mathcal{F}_{\text{inc},I}(\mathbf{q}, -t), \quad (3.33)$$

which show that the Fourier transforms of the coherent cross terms are not necessarily real and thus cannot be interpreted as physically meaningful partial dynamic structure factors. Setting

$$\mathcal{F}_{\text{coh},IJ}^{\text{sym}}(\mathbf{q}, t) = \frac{1}{2} \{ \mathcal{F}_{\text{coh},IJ}(\mathbf{q}, t) + \mathcal{F}_{\text{coh},JI}(\mathbf{q}, t) \} \quad (3.34)$$

the latter maybe defined as

$$\mathcal{S}_{\text{coh},IJ}(\mathbf{q}, \omega) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} dt \exp(-i\omega t) \mathcal{F}_{\text{coh},IJ}^{\text{sym}}(\mathbf{q}, t), \quad (3.35)$$

$$\mathcal{S}_{inc,I}(\mathbf{q}, \omega) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} dt \exp(-i\omega t) \mathcal{F}_{inc,I}(\mathbf{q}, t). \quad (3.36)$$

**Algorithmic considerations:** The intermediate scattering functions are either calculated by the FFT-based method described in Section 4.1, or by autoregressive modeling of the underlying time series, which is described in Section 4.5. The associated dynamics structure factors are, respectively, obtained from a smoothed discrete Fourier transform (see Section 4.3) or by maximum entropy estimation (see Section 4.4). All quantities are normally isotropically averaged over a user-defined number of  $\mathbf{q}$ -vectors, which are grouped in “shells” of resolution  $\Delta q$ . In this way one can account for non-crystalline samples, like liquids, solutions and powders, on which the majority of combined neutron scattering and MD simulation experiments are performed.

### 3.4 Neutron elastic incoherent structure factor

$n$ Moldyn allows the computation of the *elastic incoherent structure factor* (EISF), which is defined as the limit

$$EISF(\mathbf{q}) = \lim_{t \rightarrow \infty} \mathcal{F}_{inc}(\mathbf{q}, t) \quad (3.37)$$

of the incoherent intermediate scattering function. The EISF owes its name to the fact that it describes the elastic part of the incoherent neutron scattering spectrum. Writing

$$\mathcal{F}_{inc}(\mathbf{q}, t) = EISF(\mathbf{q}) + \mathcal{F}'_{inc}(\mathbf{q}, t) \quad (3.38)$$

it follows that

$$\mathcal{S}_{inc}(\mathbf{q}, \omega) = EISF(\mathbf{q})\delta(\omega) + \mathcal{S}'_{inc}(\mathbf{q}, \omega), \quad (3.39)$$

where  $\mathcal{S}'_{inc}(\mathbf{q}, \omega)$  describes quasi-elastic and inelastic scattering. One may also define an elastic coherent structure factor, but this quantity is not considered here, since incoherent scattering dominates from hydrogen atoms in most neutron scattering studies of soft matter systems that are typically accompanied by MD simulations. In practice expression (3.37) is not useful for the calculation of EISFs from MD simulations. For this purpose one can rely on the relation

$$\langle \exp(-i\mathbf{q} \cdot \mathbf{x}_\alpha(0)) \exp(i\mathbf{q} \cdot \mathbf{x}_\alpha(t)) \rangle \stackrel{t \rightarrow \infty}{=} \langle |\exp(i\mathbf{q} \cdot \mathbf{x}_\alpha)|^2 \rangle, \quad (3.40)$$

which reduces its calculation to the evaluation of thermal averages,

$$EISF(\mathbf{q}) = \sum_{\alpha} w_{\alpha,inc} \langle |\exp(i\mathbf{q} \cdot \mathbf{x}_\alpha)|^2 \rangle. \quad (3.41)$$

In terms of contributions from different species, the EISF may be written as

$$EISF(\mathbf{q}) = \sum_{I=1}^{N_{species}} w_{I,inc} EISF_I(\mathbf{q}), \quad (3.42)$$

with

$$EISF_I(\mathbf{q}) = \frac{1}{n_I} \sum_{\alpha \in \mathcal{M}_I} \langle |\exp(i\mathbf{q} \cdot \mathbf{x}_\alpha)|^2 \rangle. \quad (3.43)$$

being the partial contribution from the different species. Strictly speaking, the EISF is non-zero only for systems in which the atomic motions are confined in space, but in practice spurious elastic contributions to a neutron scattering spectrum arise from the limited resolution of spectrometers, or from an insufficient trajectory lengths in case of calculation from MD simulations.

A typical application for the calculation of EISFs concerns elastic incoherent neutron scattering from hydrated powders of proteins [39], and in this case one usually applies the Gaussian approximation. The latter follows from the cumulant expansion (3.15) in the limit  $t \rightarrow \infty$ , keeping only the leading term. Defining

$$\mathbf{u}_{\alpha,q} = \mathbf{n}_q \cdot (\mathbf{x}_\alpha - \langle \mathbf{x}_\alpha \rangle), \quad (3.44)$$

where  $\mathbf{n}_q$  is the unit vector in the direction of  $\mathbf{q}$ , the Gaussian approximation of the EISF reads

$$EISF(\mathbf{q}) \approx \sum_{\alpha} w_{\alpha,\text{inc}} \exp(-q^2 \langle \mathbf{u}_{\alpha,q}^2 \rangle). \quad (3.45)$$

This relation becomes correct in the limit  $q \rightarrow 0$ . If one neglects motional heterogeneity for different atoms, one obtains the cruder approximation

$$EISF(\mathbf{q}) \approx \exp(-q^2 \langle \mathbf{u}_q^2 \rangle), \quad (3.46)$$

which is often used in practice. Here  $\mathbf{u}_q$  is the displacement of a “representative atom” in the sample.

**Algorithmic considerations:** As described above, EISFs are computed as static time averages, and, in analogy to the intermediate scattering functions, they are normally isotropically averaged over a user-defined number of  $\mathbf{q}$ -vectors, which are grouped in shells of resolution  $\Delta q$ .

**Application:** A recent study [40] explores the origin of non-Gaussian behavior of the EISF of proteins, using lysozyme as an example. As Fig. 3 shows, a simple Gaussian model (dashed lines) is a very poor fit of the EISF calculated from a Molecular Dynamics trajectory, even for small  $q$ . It should be noted that the Gaussian model was fitted to the EISF data over the entire  $q$  range; a fit to small- $q$  data would lead to a better agreement in this region. A much better fit (drawn-out line) can be obtained using a model based on a distribution of many Gaussians:

$$EISF(q) = \int_0^{\infty} d\mu w(\mu) \exp(-\mu q^2). \quad (3.47)$$

The integration variable  $\mu$  represents the mean-square atomic fluctuation and the distribution  $w(\mu)$  represents the heterogeneity of the fluctuations. The simple Gaussian model motivated by the “representative atom” view corresponds to

$$w(\mu) = \delta(\mu - \langle \mathbf{u}_q^2 \rangle), \quad (3.48)$$

where  $\delta(\cdot)$  denotes the Dirac distribution. The model that was used for the fit shown in Fig. 3 is given by

$$w(\mu) = \frac{1}{\sigma^2} p\left(\frac{\mu}{\sigma^2}\right). \quad (3.49)$$

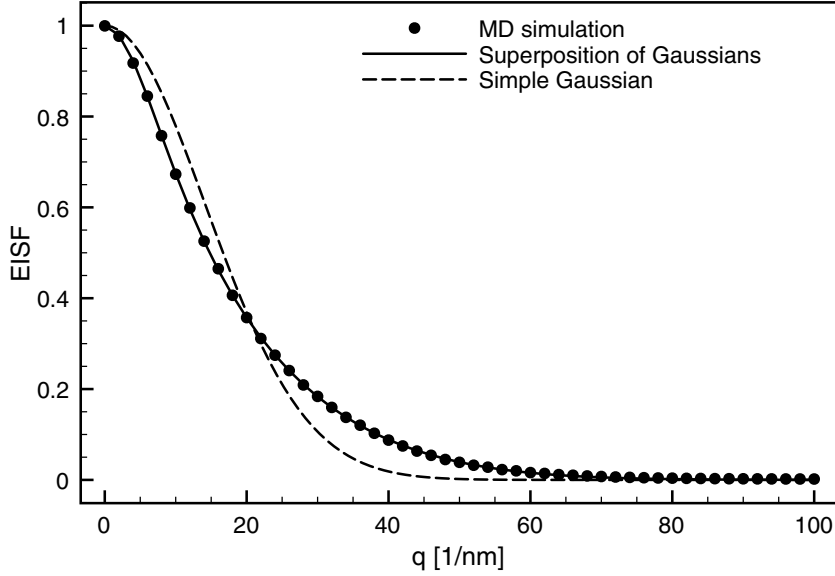
where  $p(\lambda)$  is given by the shifted Gamma distribution [41],

$$p(\lambda; \alpha, \beta) = p_0(\lambda - \alpha; \beta), \quad \alpha > 0, \quad (3.50)$$

the standard Gamma distribution being

$$p_0(\lambda; \beta) = \theta(\lambda) \frac{(\beta\lambda)^{\beta}}{\lambda\Gamma(\beta)} \exp(-\beta\lambda), \quad 0 < \beta < \infty, \quad (3.51)$$

where  $\theta(\cdot)$  is the Heaviside unit step function. The model defined by Eq. (3.49) has in total three parameters. The parameter  $\sigma^2$  describes the scale of the atomic fluctuations in the protein. The parameter  $\beta$  describes how fast the distribution decays with increasing  $\mu$ . Finally, the shift parameter  $\alpha$  must be introduced because atomic fluctuations in proteins have a clearly non-zero minimal value that corresponds to the fastest vibrational modes in the molecule.



**Figure 3.** Simulated EISF of lysozyme (points) fitted with a single Gaussian (dashed line) and with a superposition of multiple Gaussians (drawn-out line). The simulation has been performed for a single lysozyme molecule [29] in the presence of 3403 water molecules. The molecular dynamics simulation over a time of 1ns was performed with the MMTK simulation package [42], using the Amber94 force field [27], periodic boundary conditions and a velocity-Verlet integrator with a time step of 1 *f.s.* A thermodynamic  $NpT$ -ensemble [30, 31], with  $T = 300K$  and  $p = 1\text{atm}$  was used. Prior to the EISF calculation, the global translational and rotational motions of the lysozyme molecule have been removed by *nMoldyn*. The data have been taken from Ref. [40].

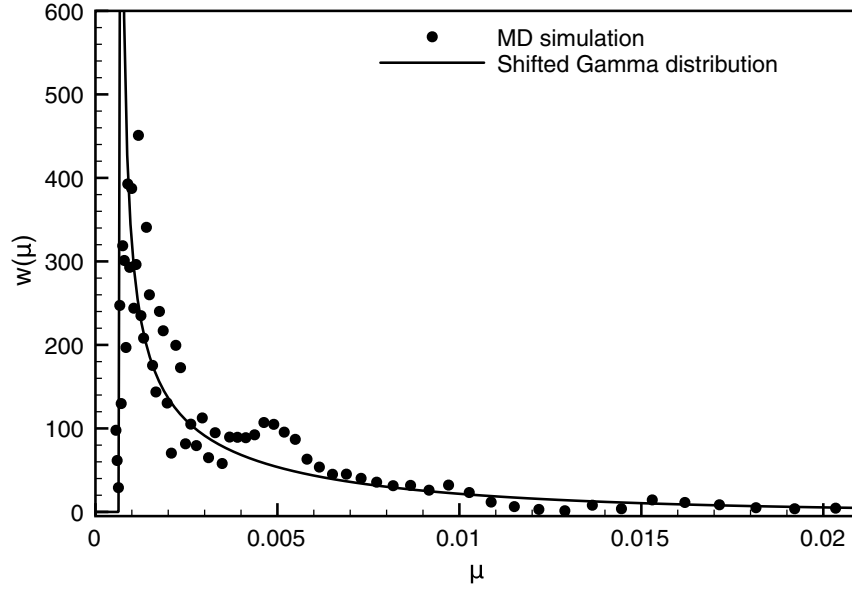
Since  $w(\mu)$  is the distribution of the atomic fluctuation values over the atoms of the protein, it can be computed directly from the Molecular Dynamics trajectory in the form of a histogram. Fig. 4 shows this distribution in comparison to the shifted Gamma distribution that was fitted to the EISF. The agreement is quite remarkable and supports the conclusion that the deviation of the EISF from a simple Gaussian is essentially due to the heterogeneity of atomic fluctuations in a protein. The “representative atom” does not exist.

### 3.5 Isotropic static structure factors

The static structure factor introduced in (3.14) is the central object in structural studies of liquids and amorphous solids, and in many applications a decomposition into cross and self contributions from different groups of atoms is of interest. Using the decomposition (3.27) of the coherent intermediate scattering function we may write

$$\mathcal{S}(\mathbf{q}) = \sum_{I,J=1}^{N_{\text{species}}} w_{I,\text{coh}} w_{J,\text{coh}} \mathcal{S}_{IJ}(\mathbf{q}), \quad (3.52)$$

where  $\mathcal{S}_{IJ}(\mathbf{q}) = \mathcal{F}_{\text{coh},IJ}(\mathbf{q}, 0)$ . In the static case the symmetrization (3.34) leads to  $\mathcal{S}_{IJ}^*(\mathbf{q}) = \mathcal{S}_{JI}(\mathbf{q})$  and may be neglected if the partial static structure factor is invariant under the reflection  $\mathbf{q} \rightarrow -\mathbf{q}$ . This is in particular the case in all studies of liquids and amorphous solids, where the static structure factor is invariant under rotations of the coordinate system and depends thus on  $|\mathbf{q}|$ , rather than on the vector  $\mathbf{q}$ . This situation is considered in *nMoldyn* and can be exploited to calculate partial static structure factors



**Figure 4.** The distribution of atomic fluctuations relevant to the Molecular Dynamics trajectory from which the EISF shown in figure 3 was computed (points), and the shifted Gamma distribution that was fitted to the EISF (line). The atomic fluctuations were computed as time-averages over the trajectory and their distribution was obtained as a histogram over the values for the different atoms.

by performing the isotropic average over all directions of  $\mathbf{q}$  analytically. One then obtains

$$S_{IJ}(q) = \delta_{IJ} + \frac{1}{n_I} \sum'_{\alpha \in \mathcal{M}_I, \beta \in \mathcal{M}_J} \left\langle \frac{\sin(qr_{\alpha\beta})}{qr_{\alpha\beta}} \right\rangle \quad (3.53)$$

where  $q = |\mathbf{q}|$  and  $r_{\alpha\beta} = |\mathbf{x}_\alpha - \mathbf{x}_\beta|$ . The prime indicates that terms  $\alpha = \beta$  for  $I = J$  are to be excluded.

**Algorithmic considerations:** In  $n$ Moldyn expression (3.53) is evaluated for a user-defined grid of  $q$ -values, averaging over the configurations in the input MD trajectory. The  $q$ -dependent contributions for each configuration are computed by summing over all pairs of atoms in  $\mathcal{M}_I$  and  $\mathcal{M}_J$ .

### 3.6 Pair correlation functions

The partial static structure factors have a direct relation to the corresponding pair correlation functions, which are also referred to as radial distribution functions [4, 43, 44]. Interpreting the average in Expression (3.28) as an ensemble average, one arrives at

$$S_{IJ}(\mathbf{q}) = \delta_{IJ} + \sqrt{\rho_I \rho_J} \int d^3r \exp(i\mathbf{q} \cdot \mathbf{r}) g_{IJ}(\mathbf{r}), \quad (3.54)$$

where  $\rho_I = n_I/V$  and  $\rho_J = n_J/V$  are the densities of species  $I$  and  $J$ , respectively, and  $g_{IJ}(\mathbf{r})$  is the pair correlation function. Multiplied by  $\rho_J$ , it gives the conditional probability density to find an atom of species  $J$  at position  $\mathbf{r}$  if an atom of species  $I$  is at the origin. For isotropic systems, relation (3.54) can be cast into the form

$$S_{IJ}(q) = \delta_{IJ} + \frac{\sqrt{4\pi\rho_I\rho_J}}{q} \int_0^\infty dr \sin(qr) r g_{IJ}(r), \quad (3.55)$$

where  $q \equiv |\mathbf{q}|$  and  $r \equiv |\mathbf{r}|$ . Here

$$g_{IJ}(r) = \frac{n_{IJ}(r)}{\rho_J 4\pi r^2 dr} \quad (3.56)$$

can be interpreted as the average number of atoms of species  $J$  in a shell of width  $dr$  at distance  $r$  from an atom of species  $I$ , divided by the volume  $4\pi r^2 dr$  of that shell and the average density  $\rho_J$ .

**Algorithmic considerations:** In *nMoldyn* the partial pair correlation functions are computed via corresponding histograms for  $n_{IJ}(r)$ , averaging over the configurations in the MD trajectory.

### 3.7 Memory functions

The numerical study of memory functions by Molecular Dynamics simulations has for a long time been hindered by the lack of a suitable numerical algorithm for their calculation. Such an algorithm has been published in [11] and it has been implemented in *nMoldyn* to calculate according to the general scheme (2.21) the memory functions corresponding to the following time correlation functions,

- Velocity autocorrelation function.
- Incoherent intermediate scattering function.
- Coherent intermediate scattering function.

Some examples are given in the following.

**Algorithmic considerations:** The algorithm is described in Section 4.6, referring to Sections 4.4 and 4.5. It is important to note that memory functions associated with atom-weighted self-correlation functions are computed as averages over the corresponding individual memory functions. Consider for example the VACF (see definition (3.1)). Here a memory function is first computed for each atom, according to the definition

$$\frac{dc_{vv,\alpha}(t)}{dt} = - \int_0^t d\tau \zeta_\alpha(t - \tau) c_{vv,\alpha}(\tau), \quad (3.57)$$

and *nMoldyn* computes the final result as average over the individual memory functions,

$$\zeta(t) = \sum_\alpha w_\alpha \zeta_\alpha(t - \tau). \quad (3.58)$$

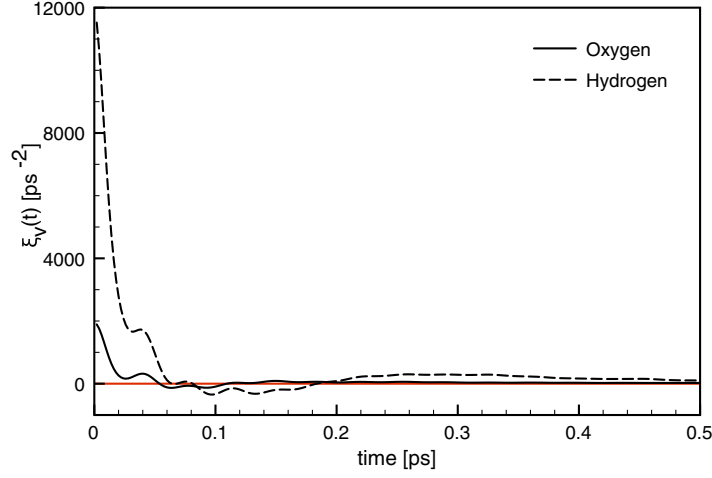
The same applies to the calculation of the memory function associated with the incoherent intermediate scattering function.

**Application:** In Figure 5 we show the memory functions relevant to the VACF of hydrogen and oxygen atoms in water [14, 45]. In both curves, the initial sharp decay reflects the effects of individual interactions between the tagged particle and its neighbors. For larger times the memory function of the oxygen atom stays close to zero, thus indicating the absence of appreciable memory effects. The hydrogen memory function shows, in contrast, another component for  $t > 0.2$  ps which displays a slower decay than the initial one. This second component can be interpreted as a manifestation of the collective dynamics of the hydrogen bond network.

From the initial values of the memory functions one obtains the square of the so-called Einstein frequency,

$$\zeta(0) = \frac{\langle \dot{v}^2 \rangle}{\langle v^2 \rangle} \equiv \Omega^2, \quad (3.59)$$

for the hydrogen and oxygen atoms, respectively, which directly relates to the mean square force and can be interpreted as a ‘‘rattling frequency’’ in the cage of nearest neighbors. For hydrogen and oxygen we find  $11881 \text{ ps}^{-2}$  and  $1941 \text{ ps}^{-2}$ , respectively.



**Figure 5.** Memory functions relevant to the VACFs of hydrogen and oxygen atoms in water. The data have been taken from Refs. [14, 45]. The input trajectory is the one described in Figure 1.

Memory functions can also be used to estimate diffusion coefficients. From a Laplace transform of equation (2.19) one obtains

$$\hat{c}_{vv}(s) = \frac{\langle v^2 \rangle}{s + \hat{\xi}(s)}, \quad (3.60)$$

where  $\hat{\xi}(s)$  and  $\hat{c}_{vv}(s)$  are, respectively, the Laplace transform of the normalized VACF and the corresponding memory function. Using that  $\hat{c}_{vv}(0) = \int_0^\infty dt c_{vv}(t) = D$ , Eq. (3.60) yields

$$D = \frac{\langle v^2 \rangle}{\gamma_v}, \quad (3.61)$$

where  $\gamma_v$  is the friction constant

$$\gamma_v := \hat{\xi}(0) = \int_0^\infty dt \xi(t). \quad (3.62)$$

For the hydrogen atom this yields  $D_H = 3.8 \cdot 10^{-5} \text{ cm}^2/\text{s}$ , and for the oxygen atoms  $D_O = 2.8 \cdot 10^{-5} \text{ cm}^2/\text{s}$ , which is higher than the estimation from the slope of the MSD ( $D = 2.5 \cdot 10^{-5} \text{ cm}^2/\text{s}$ ) and the experimental value ( $D = 2.23 \cdot 10^{-5} \text{ cm}^2/\text{s}$  [46]).

### 3.8 Time correlation functions relevant to NMR spectroscopy

*n*Moldyn allows the calculation of time autocorrelation functions relevant to nuclear magnetic resonance (NMR) spectroscopy

$$C_{ii}(t) = \langle P_2(\boldsymbol{\mu}_i(t) \cdot \boldsymbol{\mu}_i(0)) \rangle. \quad (3.63)$$

Here  $P_2(\cdot)$  is the second order Legendre polynomial and  $\boldsymbol{\mu}_i(t)$  is the unit vector pointing along the instantaneous direction of the vector relating the pair of interacting nuclear spins considered for analysis. For example in the case of the relaxation measurement of amide backbone  $^{15}\text{N}$  nuclei, which is routinely used for the investigation of protein backbone dynamics, relaxation primarily occurs through fluctuations of the  $^{15}\text{N}$ - $^1\text{H}$  dipole-dipole interactions with the directly bonded amide proton and of the  $^{15}\text{N}$  chemical shift anisotropy tensor, which is commonly assumed to be axially symmetric with its axis parallel to the NH bond. For a general reference, see Ref. [47]. In this case  $\boldsymbol{\mu}_i(t)$  is the unit vector



pointing along the direction of the  $^{15}\text{N}^1\text{H}$  bond of residue  $i$ . The new atom selection engine of *nMoldyn* allows the user to select the vector of interest.

To compute the time correlation functions from the simulated trajectory, *nMoldyn* uses the representation of  $P_2(\boldsymbol{\mu}_i(t) \cdot \boldsymbol{\mu}_i(0))$  in terms of spherical harmonic functions,

$$C_{ii}(t) = \frac{4\pi}{5} \sum_{m=-2}^2 (-1)^m \langle Y_2^{-m}(\boldsymbol{\Omega}_i(0)) Y_2^m(\boldsymbol{\Omega}_i(t)) \rangle, \quad (3.64)$$

which allows for an efficient calculation by the FFT-based algorithm described in Section 4.1. Here,  $\boldsymbol{\Omega}_i$  denotes the polar angles defining the instantaneous orientation of the  $i$ th vector in the given reference frame, and  $Y_2^m(\cdot)$  are the second rank spherical harmonics [41].

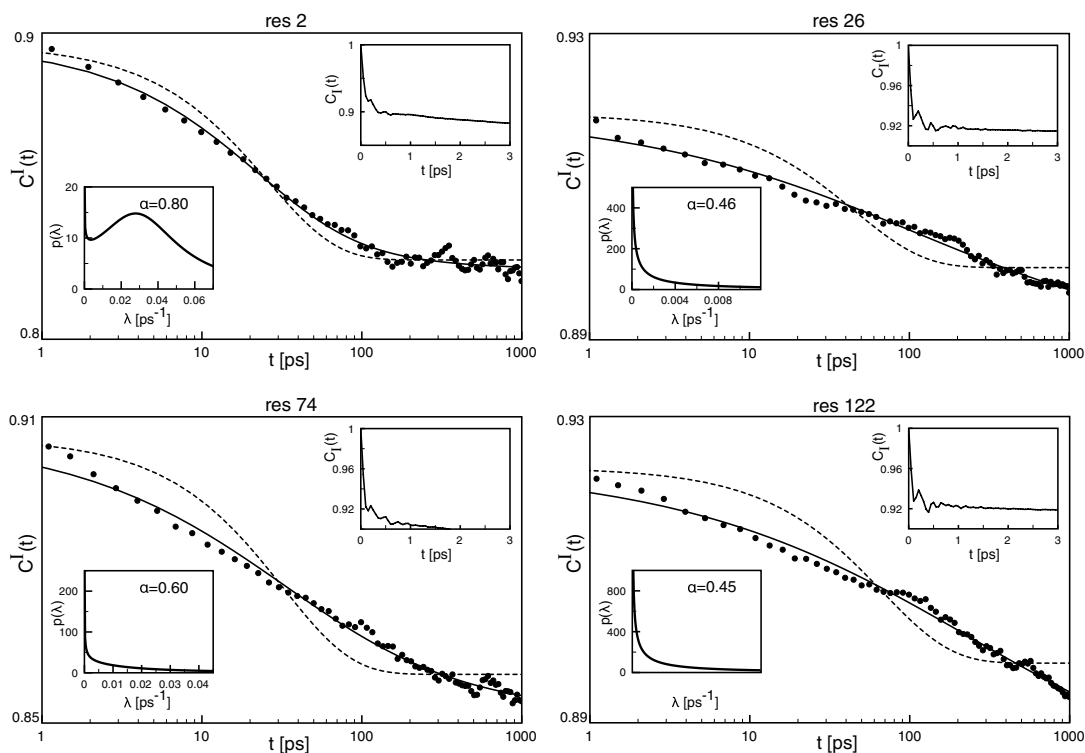
The program provides also the asymptotic value of the time correlation functions  $C_{ii}(\infty) = S_i^2$  – known as generalized order parameter [48] – which indicates the degree of spatial restriction of the motions of the vector in the given reference frame (for isotropic tumbling it is zero). The asymptotic values is computed through

$$S_i^2 = \lim_{t \rightarrow \infty} C_{ii}^I(t) = \frac{4\pi}{5} \sum_{m=-2}^2 | \langle Y_2^m(\boldsymbol{\Omega}_i) \rangle |^2. \quad (3.65)$$

**Algorithmic considerations:** The NMR time correlation functions are computed according to the Fast Fourier Transform (FFT) based Algorithm described in Section 4.1.

**Application:** One of the main reasons for computing the microscopic time correlation functions relevant to the NMR relaxation spectroscopy by MD simulations is that experimental approaches provide their spectra at a very limited number of frequencies, corresponding to the Larmor frequencies of the nuclei involved in the spin relaxation. This fact makes it difficult to infer the underlying dynamics by a numerical reconstruction of the corresponding spectra. Here MD simulations provide very detailed information about the time correlation functions, which allows discriminating among different models of microscopic dynamics.

We cite as an example a recent publication dealing with the computation of the internal time correlation functions relevant to the relaxation of the amide backbone  $^{15}\text{N}$  nuclei of lysozyme [49]. Some typical curves are displayed in Fig. 6. The computed time correlation functions have been used to test a new model for the reorientation of the  $^{15}\text{N}^1\text{H}$  bond, which is based on fractional Brownian Dynamics (fBD) and describes internal protein dynamics more realistically than the traditional model-free (MF) approach by Lipari and Szabo [48]. In the MF approach the assumption is made that the internal reorientational correlation functions decay exponentially, which is a natural assumption in absence of detailed information on the rotational diffusion of NH vectors. However a direct calculation of the time correlation function from MD simulations shows systematic deviations from an exponential form, which can be well described in terms of a superposition of exponential functions, with a broad spectrum of decay rates, which is typical for non-markovian stochastic models, such as fractional Brownian dynamics [50]. Some typical fits of the time correlation functions to the MF and fBD models are shown in figure 6. In each panel the top-right inset shows a zoom on the first few picoseconds of the time correlation function characterized by the well-known initial fast decay associated to librations, which is usually neglected in the analysis of NMR relaxation data since these fast motions cannot be experimentally resolved. Besides the initial fast decay the time correlation functions show an asymptotic power law behavior and the bottom-left inset shows the underlying distribution of relaxation rates. The latter is calculated in the framework of the fBD approach. For more details concerning the models we refer to Ref. [49]. Some technical information concerning the input trajectory are given in the figure caption.



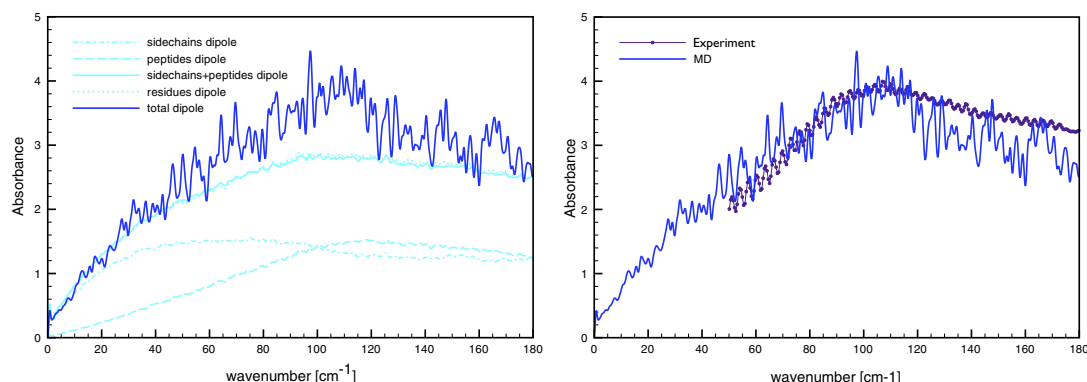
**Figure 6.** Internal correlation functions  $C_{ii}^I(t)$  for selected residues (Val 2, Gly 26, Asn 74, Ala122) in Lysozyme computed by  $n$ Moldyn from MD simulation (dots) together with the fits to the fBD (solid line) and MF (short dashes) models. Both fits are performed in the time window from 1 ps to 1000 ps. In each panel the top-right inset shows a zoom on the first few picoseconds of the time correlation function and the bottom-left inset shows the distribution of relaxation rates. The MD trajectory used for the calculation  $n$ Moldyn has been computed by the program package NAMD [51]. The simulated system consists of one hen egg white lysozyme molecule [29] and 6775 water molecules. The simulations have been performed in the  $NpT$  ensemble using the the AMBER99SB [52] and SPC/E [26] force fields for the protein and the water molecules, respectively. To mimic a macroscopic system, periodic boundary conditions have been applied and electrostatic interactions have been computed using the particle mesh Ewald method (PME) [53], with a cut-off of 12 Å. The integration time step was set to 1 fs and atomic configurations were recorded with a sampling time step of 50 fs over 10 ns in total. In order to extract a trajectory describing only the internal dynamics of the simulated lysozyme molecule, global translations and rotations of the protein molecule have been filtered out by performing for each sampling time step a rigid body fit of its actual conformation to its initial conformation in the trajectory [32, 54].

### 3.9 IR-THz spectroscopy

One of the most recent functionalities implemented in  $n$ Moldyn concerns the computation of IR spectra in the THz frequency region. The collective modes probed in this frequency region can be described in the framework of classical physics and are, a priori, suitable for comparison with MD simulation data. The program calculates the time correlation functions describing the fluctuations of the total dipole moment  $\mathbf{M}(t) = \sum_{\alpha} q_{\alpha} \mathbf{R}_{\alpha}(t) - \langle \sum_{\alpha} q_{\alpha} \mathbf{R}_{\alpha} \rangle$  and its frequency spectrum

$$I(\omega) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} dt \exp(-i\omega t) \langle \mathbf{M}(0) \mathbf{M}(t) \rangle \quad (3.66)$$

The above computation is performed over the whole system and, for complex systems that consist of several components, on the different components as well. In this way their individual contribution to the



**Figure 7.** *Left:* IR spectra of Lysozyme calculated from MD simulations. *Total dipole* refers to the spectrum of the fluctuations of the total dipole moment of the protein, while *side chains*, *peptides*, and *residues dipole* refer to the spectra of the fluctuations of the dipole moment associated to side chains, peptides, and residues, respectively. The *side chains+peptides dipole* results from the summation of the spectra *side chains dipole* and *peptide dipole*. *Right:* Comparison between measured and calculated spectra. The experimental data have been collected by the spectrometer AILES at Synchrotron Soleil. Figure from ref. [56]

total spectrum as well as the contribution from the cross terms can be evaluated. For example, in the case of a protein, the program recognizes the side chains, the peptide groups and the amino acids, as substructures.

Knowing  $I(\omega)$ , the absorption cross section is computed according to [55]

$$\alpha(\omega) = \frac{4\pi^2\omega(1 - \exp(-\beta\hbar\omega))}{3\hbar cn} I(\omega) \quad (3.67)$$

which is related to the measured absorbance spectrum via a constant factor depending on the number density of absorbers and on the path length of the light. In the above equation the refractive index is assumed to be constant over the analyzed frequency range.

**Algorithmic considerations:** The time correlation functions of the total dipole moment are computed according to the Fast Fourier Transform (FFT) based Algorithm described in Section 4.1.

**Application:** In the left panel of figure 7 we show as an example the IR spectra of hydrated lysozyme in the terahertz frequency region [56]. We calculate the fluctuations of the permanent dipole moment of the whole protein and of different components of the protein (side chains, peptides groups, amino acids), to evaluate their contributions to the total spectrum. The spectra labeled “side-chains” result from the fluctuation of the permanent dipole moments associated to the side-chains (no cross correlations between different side-chains are considered) the spectra labeled “peptide” result from the fluctuation of the permanent dipole moments associated to the peptide groups (no cross correlation between different peptide groups are considered) and the spectra labeled “residues” result from the fluctuation of the permanent dipole moments associated to the different residues (no cross correlations between different residues are considered, while the cross correlations between side-chains and peptides group are taken into account). This analysis shows that the main contribution to the absorption band at  $110 \text{ cm}^{-1}$  arises from the fluctuations of the peptide groups and from the cross correlations between different residues. Moreover, the MD simulation suggests that the absorption spectrum in the frequency range  $1\text{--}25 \text{ cm}^{-1}$  is mainly dominated by side-chain fluctuations, without cross correlation contributions. In the right panel of figure 7 we compare the spectrum computed by *nMoldyn* with a measured spectrum. The comparison shows that MD simulations are able to qualitatively reproduce the experimental data.

### 3.10 Protein secondary structure analysis

The latest version of *nMoldyn* also contains a module for the analysis of the time-varying secondary structure of proteins in MD simulations.

#### 3.10.1 Protein secondary structure

The description of protein structures is commonly hierarchized into primary, secondary, tertiary and quaternary structure, following the characterization of atomic conformations in three-dimensional space in a progressively larger scale. While the primary structure only gives information about the sequence of amino acids which composes proteins, the secondary structure concerns the configuration of local segments of proteins. But the latter does not take into account the overall atomic positions in three-dimensional space, which are instead considered to be the tertiary structure.

Motifs in protein secondary structure are traditionally described in terms of two torsional angles,  $\phi$  and  $\psi$ , per amino acid, which define for each  $C_\alpha$ -atom the rotation of the left and right peptide plane about the  $N-C_\alpha$  and  $C_\alpha-C$  bond, respectively [57]. Alternatively they can be defined by the patterns of hydrogen bonds between backbone amide and carboxyl groups. The knowledge of secondary structure is an essential tool to assess local flexibilities in proteins and to interpret the structural changes induced, for example, by ligand binding or by different environmental conditions (pressure, temperature or molecular crowding, for instance).

The ScrewFit algorithm [18] implemented in *nMoldyn* provides an efficient description of structural motifs in terms helicoidal motions, relating consecutive peptide planes (see left panel of Fig. 8). It provides, in particular, a description of non-ideal protein secondary structures and of their structural changes due to external stress (pressure, chemical agents etc.). The method is based on Chasles' theorem, which states that any rigid body displacement can be described by a screw motion. The parameters are

1. The orientational (angular) distance between consecutive peptide planes
2. The radius of the local screw motion
3. The straightness of consecutive screw axes

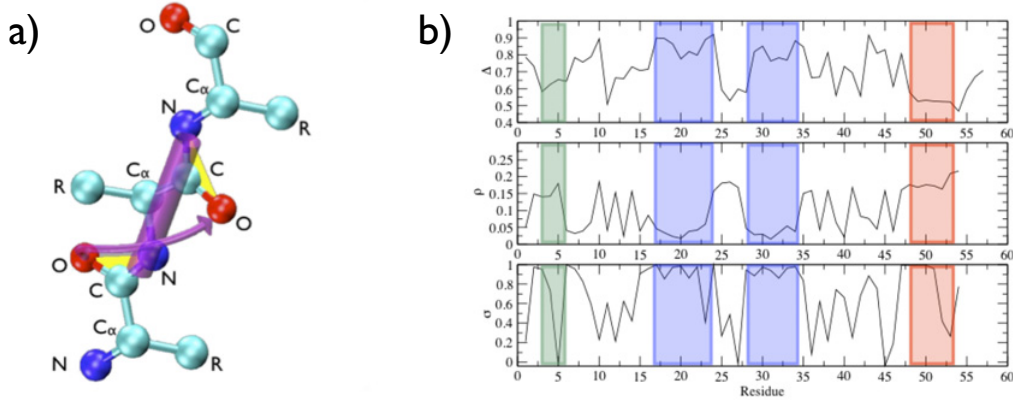
Panel a) of Figure 8 gives an illustration of the helix (screw motion) parameters. By definition, the  $C$ -atoms of the peptide planes are on the surface of the cylinder defining the envelope of the screw motion. Details about the algorithms are described in Section 4.7 and an application to protein ligand binding, a typical problem relevant to structural biology, can be found in Ref [58].

**Application:** In the following we consider an application of the ScrewFit algorithm to a small protein, bovine pancreatic trypsin inhibitor. Panel b) of Fig. 8 gives a sketch of a typical ScrewFit profile of a protein (in this figure, bovine pancreatic trypsin inhibitor). By applying ScrewFit to different protein configurations obtained from an MD simulation, one can easily recognize the stability of structural motifs and the precise points where their changes occur, which is more difficult to achieve by the standard analysis of positional differences. The reason is that the latter might indicate important structural differences in a large region, although the corresponding position differences are induced by one single localized change in the winding of the protein backbone.

## 4. ALGORITHMS

### 4.1 Fast Correlation Algorithm

In practice the time averages are calculated over a finite time window, which corresponds to the trajectory length and the dynamical variables are sampled at fixed time intervals,  $A = A(k \cdot \Delta t)$  and  $B = B(k \cdot \Delta t)$  with  $k = 0, \dots, (N_t - 1)$ . The discrete representation of Eqs. (2.3) and (2.4) yields,



**Figure 8.** Panel a) A tri-peptide with two peptide bonds in the extended conformation. The yellow triangles formed by the atoms  $\{O, C, N\}$  of the peptide planes define the local helix structure of the polypeptide. The green spheres, labeled with “R”, indicate dummy atoms replacing the side-chains. The radius of the cylinder shown in the figure defines the radius  $\rho$  of the screw motion relating the two consecutive peptide planes. Panel b) A typical ScrewFit profile for a protein structure. In this figure the three parameters refers to the BPTI (PDB code: 4PTI). Vertical stripes superimposed on ScrewFit plots represent motifs detected by the latter method ( $\beta$ -strands: blue;  $\alpha$ -helices: red; 3–10-helices: green).

respectively,

$$\langle A \rangle = \frac{1}{N_t} \sum_{k=0}^{N_t-1} A(k \cdot \Delta t), \quad (4.1)$$

and

$$c_{AB}(m \cdot \Delta t) = \begin{cases} \frac{1}{N_t - m} \sum_{k=0}^{N_t-m-1} B^*(k \cdot \Delta t) A((k+m) \cdot \Delta t), & m = 0 \dots N_t - 1, \\ \frac{1}{N_t - |m|} \sum_{k=|m|}^{N_t-1} B^*(k \cdot \Delta t) A((k-|m|) \cdot \Delta t) & m = -(N_t - 1) \dots -1. \end{cases} \quad (4.2)$$

In the above equations  $N_t$  is the number of sampled values of the dynamical variables,  $\Delta t$  is the sampling time step  $(N_t - 1) \cdot \Delta t$  is the trajectory length and  $c_{AB}(m \cdot \Delta t)$  is the TCF relevant to  $A$  and  $B$ . The prefactors in front of the sums ensure the proper normalization. According to (4.2),  $c_{AB}(m \cdot \Delta t)$  has  $2N_t - 1$  data points.

In practice the direct calculation of the time correlation functions by Eq. (4.2) is prohibitive because of the computational complexity (measured by the number of multiplications),  $\propto N_t^2$ . To increase the computational efficiency, all TCFs in *nMoldyn* are calculated by the Fast Correlation Algorithm (FCA), which relies on the Fast Fourier Transform (FFT) technique [59]. Compared to the direct computation, FCA allows the reduction of the computational complexity (measured by the number of multiplications), from  $\propto N_t^2$  to  $\propto N_t \log_2(N_t)$ . To exploit the correlation theorem for discrete periodic functions we define an extended periodic version of the the time series  $A(k \cdot \Delta t)$  and  $B(k \cdot \Delta t)$  introduced in Section 2.1. In the following,  $\Delta t$  is dropped for simplicity.

$$a(k) = \begin{cases} A(k) & k = 0 \dots N_t - 1 \\ 0 & k = N_t \dots 2N_t - 1 \end{cases}, \quad (4.3)$$

$$b(k) = \begin{cases} B(k) & k = 0 \dots N_t - 1 \\ 0 & k = N_t \dots 2N_t - 1 \end{cases}, \quad (4.4)$$

and

$$a(k) = a(k + m \cdot 2N_t), \quad b(k) = b(k + m \cdot 2N_t), \quad m = 0, \pm 1, \pm 2, \dots \quad (4.5)$$

The discrete, cyclic correlation of  $a(k)$  and  $b(k)$  is defined as

$$S_{ab}(m) = \sum_{k=0}^{2N_t-1} a^*(k)b(k+m). \quad (4.6)$$

It is easy to see that

$$c_{AB}(m) = \frac{1}{N_t - |m|} S_{ab}(m), \quad -(N_t - 1) \leq m \leq N_t - 1. \quad (4.7)$$

Using the correlation theorem for discrete periodic functions [59],  $S_{ab}(m)$  can be written as

$$S_{ab}(m) = \frac{1}{2N_t} \sum_{n=0}^{2N_t-1} \exp\left[2\pi i \left(\frac{mn}{2N_t}\right)\right] \tilde{a}^*\left(\frac{n}{2N_t}\right) \tilde{b}\left(\frac{n}{2N_t}\right) \quad (4.8)$$

where  $\tilde{a}\left(\frac{n}{2N_t}\right)$  and  $\tilde{b}\left(\frac{n}{2N_t}\right)$  are the discrete Fourier transforms of  $a(k)$  and  $b(k)$ , respectively:

$$\tilde{a}\left(\frac{n}{2N_t}\right) = \sum_{k=0}^{2N_t-1} \exp\left[-2\pi i \left(\frac{nk}{2N_t}\right)\right] a(k), \quad (4.9)$$

$$\tilde{b}\left(\frac{n}{2N_t}\right) = \sum_{k=0}^{2N_t-1} \exp\left[-2\pi i \left(\frac{nk}{2N_t}\right)\right] b(k). \quad (4.10)$$

The FCA algorithm and the direct scheme (4.2) give, apart from round-off errors, *identical results* and we note here that the FCA is, in fact, based on the discrete version of the Wiener-Khinchin theorem (2.11), performing an appropriate “zero padding” of the discrete signals to be correlated.

#### 4.2 Efficient Computation of Mean-square displacements

In the discrete case, the mean-square displacement of a particle is given by

$$\Delta^2(m) = \frac{1}{N_t - m} \sum_{k=0}^{N_t-m-1} [\mathbf{r}(k+m) - \mathbf{r}(k)]^2, \quad m = 0 \dots N_t - 1, \quad (4.11)$$

where  $\mathbf{r}(k) \equiv \mathbf{r}(k\Delta t)$  is the particle trajectory. We now define the auxiliary function

$$S(m) \doteq \sum_{k=0}^{N_t-m-1} [\mathbf{r}(k+m) - \mathbf{r}(k)]^2, \quad m = 0 \dots N_t - 1, \quad (4.12)$$

which is split as follows:

$$S(m) = S_{AA+BB}(m) - 2S_{AB}(m), \quad (4.13)$$

$$S_{AA+BB}(m) = \sum_{k=0}^{N_t-m-1} [\mathbf{r}^2(k+m) + \mathbf{r}^2(k)], \quad (4.14)$$

$$S_{AB}(m) = \sum_{k=0}^{N_t-m-1} \mathbf{r}(k) \cdot \mathbf{r}(k+m). \quad (4.15)$$

The function  $S_{AB}(m)$  can be computed using the FCA method described in Section 4.1. For  $S_{AA+BB}(m)$  the following recursion relation holds:

$$S_{AA+BB}(m) = S_{AA+BB}(m-1) - \mathbf{r}^2(m-1) - \mathbf{r}^2(N_t - m), \quad (4.16)$$

$$S_{AA+BB}(0) = 2 \sum_{k=0}^{N_t-1} \mathbf{r}^2(k). \quad (4.17)$$

This allows one to construct the following efficient scheme for the computation of mean-square displacements:

1. Compute  $DSQ(k) = \mathbf{r}^2(k)$ ,  $k = 0 \dots N_t - 1$ ;  $DSQ(-1) = DSQ(N_t) = 0$ .
2. Compute  $SUMSQ = 2 \cdot \sum_{k=0}^{N_t-1} DSQ(k)$ .
3. Compute  $S_{AB}(m)$  using the FFT method.
4. Compute the mean-square displacement  $MSD(m)$  in the following loop:

$$\begin{aligned} SUMSQ &\leftarrow SUMSQ - DSQ(m-1) - DSQ(N_t - m) \\ MSD(m) &\leftarrow (SUMSQ - 2 \cdot S_{AB}(m)) / (N_t - m) \\ m &\text{ running from } 0 \text{ to } N_t - 1 \end{aligned}$$

It should be noted that the efficiency of this algorithm is the same as for the FCA computation of time correlation functions since the number of operations in step (1), (2), and (4) grows linearly with  $N_t$ .

### 4.3 Direct computation of frequency spectra

In contrast to the estimation of TCFs, the estimation of their Fourier spectra by straight-forward application of discrete Fourier transforms is not improved by adding data points for longer time intervals. On the contrary, such additional points tend to add noise to the spectra. A classical method to obtain smooth Fourier spectra is to use convolutions with frequency windows [60]. Using the convolution theorem of the Fourier transform in frequency space,

$$\mathcal{F}\{f(t)g(t), t, \omega\} = \frac{1}{2\pi} \int_{-\infty}^{+\infty} d\omega' \tilde{f}(\omega - \omega') \tilde{g}(\omega'), \quad (4.18)$$

one sees that  $\tilde{f}(\omega)$  acts as a smoothing window for  $\tilde{g}(\omega)$  and vice versa. Defining a window function with  $w(0) = 1$ , one obtains a smoothed estimation for the Fourier spectrum of a correlation function  $c(t)$  by computing the discrete Fourier transform

$$\tilde{c}_{\text{smoothed}}\left(\frac{n}{2N_t}\right) = \Delta t \sum_{m=-(N_t-1)}^{N_t-1} \exp\left[-2\pi i \left(\frac{nm}{2N_t}\right)\right] w(m)c(m), \quad (4.19)$$

where  $c(m) \equiv c(m\Delta t)$ ,  $m = -(N_t - 1), \dots, N_t - 1$ . The choice of the frequency window can be optimized using various criteria [61]. In *nMoldyn* a Gaussian window is used,

$$w(t) = \exp\left(-\frac{1}{2} \left[\frac{t}{\sigma_t}\right]^2\right), \quad (4.20)$$

where  $\sigma_t > 0$  is the width of the Gaussian. The latter is defined relatively to the trajectory length  $T$ ,

$$\sigma_T = \alpha T, \quad (4.21)$$

where  $\alpha > 0$  is a dimensionless parameter. We note here that

$$\tilde{w}(\omega) = \sqrt{2\pi}\sigma_t \exp\left(-\frac{1}{2}\sigma_t^2\omega^2\right), \quad (4.22)$$

which shows that the the width in (angular) frequency space is simply given by

$$\sigma_\omega = \frac{1}{\sigma_t}. \quad (4.23)$$

This well-known relation facilitates the estimation of the frequency resolution of the window function.

#### 4.4 Maximum entropy estimation of power spectra

The estimation of frequency spectra according to the method described in Section 4.3 suffers from some arbitrariness concerning the choice of the frequency window. For autocorrelation functions, which are most important in spectroscopic experiments, there exists an alternative method, known as the Maximum Entropy Method (MEM) [20, 62]. The idea is to estimate the spectrum of an ACF such that its Fourier spectrum is as “white as possible”, using the information of the ACF for a fixed number of points as constraints. For this purpose the (discrete) signal,  $A(n) \equiv A(n\Delta t)$  ( $n = 0, \dots, N_t - 1$ ), to be correlated is fitted by an autoregressive (AR) model

$$A(t) = \sum_{n=1}^P a_n^{(P)} A(t - n\Delta t) + \epsilon_P(t). \quad (4.24)$$

Here  $P \ll N_t$  is the order of the process and  $\epsilon_P(t)$  is *white noise* with zero mean and amplitude  $\sigma_P$ . A system of linear equations for the coefficients  $\{a_n^{(P)}\}$  can be obtained correlating Eq. (4.24) for  $t = j\Delta t$  with  $A(t')$  for  $t' = k\Delta t$  and using the stationarity of the ACF  $c_{AA}(t)$ . This yields the so-called Yule-Walker equations

$$\sum_{k=1}^P c_{AA}(|j-k|\Delta t) a_k = c_{AA}(j\Delta t), \quad j = 1, \dots, P. \quad (4.25)$$

As input for  $c_{AA}(t)$  one can use the estimation (4.2). Since the matrix of coefficients of the system of equations (4.25) has Toeplitz form, particularly efficient algorithms can be established. In *nMoldyn* the Burg algorithm [62, 63] is used. Once the coefficients  $\{a_n^{(P)}\}$  have been found,  $\sigma_P$  can be computed via

$$\sigma_P^2 = c_{AA}(0) - \sum_{n=1}^P a_n^{(P)} c_{AA}(n\Delta t). \quad (4.26)$$

The coefficients  $\{a_n^{(P)}\}$  and  $\sigma_P$  being determined, the Fourier transform of  $c_{AA}(t)$  can then be obtained through Z-transform techniques. Introducing the two-sided z-transform of a discrete function  $f(n)$  and its inverse through

$$\hat{F}(z) = \sum_{n=-\infty}^{+\infty} f(n) z^{-n} \equiv \mathcal{Z}\{f(n), n, z\}, \quad (4.27)$$

$$f(n) = \frac{1}{2\pi i} \oint_C dz z^{n-1} \hat{F}(z) \equiv \mathcal{Z}^{-1}\{\hat{F}(z), z, n\}, \quad (4.28)$$

where the contour  $C$  lies in the region of convergence of  $\hat{F}(z)$ , it follows from (4.24) that

$$\hat{A}(z) = \frac{\hat{\epsilon}_P(z)}{1 - \sum_{k=1}^P a_k^{(P)} z^{-k}}. \quad (4.29)$$

As for the Fourier transform, there exists a correlation theorem for the Z-transform. Defining

$$(f \circ g)(n) = \sum_{k=-\infty}^{\infty} f(n+k) g^*(k) \quad (4.30)$$



it follows that

$$\mathcal{Z}\{(f \circ g)(n), n, z\} = F(z)G^*(1/z^*) \quad (4.31)$$

if the individual Z-transforms exist. For the finite signal

$$f_N(n) = \begin{cases} f(n) & \text{if } n = 0, 1, \dots, N-1, \\ 0 & \text{otherwise,} \end{cases} \quad (4.32)$$

one obtains

$$c_{ff}(n) = \lim_{N \rightarrow \infty} (f_N \circ f_N)(n)/(2N+1) \quad (4.33)$$

and

$$\hat{c}_{ff}(z) = \lim_{N \rightarrow \infty} \frac{1}{2N+1} f_N(z)f_N^*(1/z^*). \quad (4.34)$$

This is the Wiener-Khinchin theorem for the Z-transform, from which one can infer an estimation of the power spectrum of  $A(n)$ , noting that with (4.29)

$$\hat{c}_{AA}^{(AR)}(z) = \frac{\hat{c}_{\epsilon\epsilon}(z)}{\left(1 - \sum_{k=1}^P a_k^{(P)} z^{-k}\right) \left(1 - \sum_{l=1}^P a_l^{(P)*} z^l\right)}.$$

Since  $\epsilon_P(n)$  is white noise, its autocorrelation function is given by  $c_{\epsilon\epsilon}(n) = \sigma_P^2 \delta_{n,0}$ , and therefore  $\hat{c}_{\epsilon\epsilon}(z) = \sigma_P^2$ . This leads to the so-called all-pole form of  $\hat{c}_{AA}(z)$ ,

$$\hat{c}_{AA}^{(AR)}(z) = \frac{\sigma_P^2}{\left(1 - \sum_{k=1}^P a_k^{(P)} z^{-k}\right) \left(1 - \sum_{l=1}^P a_l^{(P)*} z^l\right)}. \quad (4.35)$$

From the approximation

$$\tilde{c}_{AA}^{(AR)}(\omega) \approx \Delta t \sum_{n=-\infty}^{+\infty} c_{AA}(n) \exp(-in\omega\Delta t) = \Delta t \hat{c}_{AA}(e^{i\omega\Delta t}) \quad (4.36)$$

one obtains finally an approximation for the Fourier spectrum of the ACF of  $a(t)$ , which is based on the AR model [61, 64],

$$\tilde{c}_{aa}^{(AR)}(\omega) = \frac{\Delta t \sigma_P^2}{\left|1 - \sum_{k=1}^P a_k^{(P)} e^{-ik\omega\Delta t}\right|^2}. \quad (4.37)$$

In contrast to the estimation of Fourier spectra through smoothed discrete Fourier transforms described in Section 4.3, no window functions are applied here and the frequency scale is moreover continuous. Its evaluation for small values of  $\omega$  must be considered as an estimation on the basis of the MEM approach, which amounts effectively to extrapolating  $c_{AA}(t)$  beyond the trajectory length. The calculation of ACFs according to the AR model is described in Section 4.5.

#### 4.5 Autoregressive (AR) model for autocorrelation functions

The estimation for the ACF of  $A$  corresponding to the power spectrum (4.37) can be obtained by inverse Z-transform of (4.35),

$$c_{AA}^{(AR)}(n) = \frac{1}{2\pi i} \oint_C dz z^{n-1} \hat{c}_{AA}^{(AR)}(z), \quad (4.38)$$

where  $n$  can be any integer number, in particular  $|n| > N_t$ , where  $N_t$  is the number of data points in the discrete signal  $A(n) = \{A(0), \dots, A(N_t - 1)\}$  available from an MD simulation. The inverse

Z-transform (4.38) can be performed by transforming  $\hat{c}_{AA}^{(AR)}(z)$  into a rational function [11], which is achieved by multiplying the numerator and the denominator by  $-z^P/a_p^{(P)}$ .

$$\hat{c}_{AA}^{(AR)}(z) = \frac{1}{a_p^{(P)}} \frac{-z^P \sigma_p^2}{\underbrace{\left( z^P - \sum_{k=1}^P a_k^{(P)} z^{P-k} \right)}_{p(z)} \underbrace{\left( \sum_{k=1}^P \left[ \frac{a_k^{(P)*}}{a_p^{(P)}} \right] z^k - \frac{1}{a_p^{(P)}} \right)}_{q(z)}}. \quad (4.39)$$

Now the contour integral (4.38) can be evaluated by applying the residue theorem, choosing a contour in the annulus of convergence of  $\hat{c}_{AA}^{(AR)}(z)$ , which is given by

$$|z_k|_{max} < |z| < \frac{1}{|z_k|_{max}}. \quad (4.40)$$

This condition follows from the zeros of the polynomial  $q(z)$  in (4.35) which are given by  $1/z_k$  if  $z_k$  ( $k = 1, \dots, P$ ) are the zeros of the characteristic polynomial

$$p(z) = z^P - \sum_{k=1}^P a_k^{(P)} z^{P-k}. \quad (4.41)$$

Clearly, the domain of convergence of (4.35) is non-empty only if all zeros of  $p(z)$  lie within the unit circle,

$$|z_k| < 1, \quad k = 1, \dots, P. \quad (4.42)$$

This condition is fulfilled if the Burg algorithm [20, 62] is chosen for the calculation of the coefficients  $\{a_k^{(P)}\}$ . In this case the integration contour in (4.38) can be chosen to be the unit circle and one obtains

$$c_{AA}^{(AR)}(n) = \sum_{j=1}^P \beta_j z_j^{|n|}, \quad (4.43)$$

where the coefficients  $\{\beta_j\}$  have the form

$$\beta_j = \frac{1}{a_p^{(P)}} \frac{-z_j^{P-1} \sigma_p^2}{\prod_{k=1, k \neq j}^P (z_j - z_k) \prod_{l=1}^P (z_j - z_l^{-1})}, \quad (4.44)$$

assuming that the characteristic polynomial  $p(z)$  has only simple roots. In the light of expression (4.43) relation (4.42) appears effectively as a *stability criterion*.

#### 4.6 Discrete memory function within the AR model

An estimation of the memory function associated with a given numerical ACF can be obtained from the discretized version of the memory function equation (2.21). Defining for convenience the normalized ACF of the dynamical variable  $A$  under consideration through

$$\psi(t) = \frac{C_{AA}(t)}{C_{AA}(0)}, \quad (4.45)$$

we then have

$$\frac{\psi(n+1) - \psi(n)}{\Delta t} = - \sum_{k=0}^{n-1} \Delta t \xi(n-k) \psi(k). \quad (4.46)$$

This equation is now subjected to a *one-sided* Z-transform, which is defined as

$$\hat{f}_{>}(z) = \sum_{n=0}^{\infty} f(n)z^{-n}. \quad (4.47)$$

This leads to

$$\hat{\xi}_{>}(z) = \frac{1}{\Delta t^2} \left( \frac{z}{\Psi_{>}(z)} + 1 - z \right), \quad (4.48)$$

using that  $\psi(0) = 1$ . As shown in [11], the AR model allows to express  $\Psi_{>}(z)$  as

$$\Psi_{>}^{(AR)}(z) = \sum_{n=0}^{\infty} \psi^{(AR)}(n)z^{-n} = \sum_{j=1}^P \beta_j \frac{z}{z - z_j}, \quad (4.49)$$

where the coefficients  $\beta_j$  are given by Eq. (4.44). Inserting (4.49) into (4.48) yields  $\hat{\xi}_{>}^{(AR)}(z)$ , the z-transform of the discrete memory function within the AR model,

$$\hat{\xi}_{>}^{(AR)}(z) = \frac{1}{\Delta t^2} \left( \frac{z}{\Psi_{>}^{(AR)}(z)} + 1 - z \right). \quad (4.50)$$

In *nMoldyn* the time series  $\zeta^{(AR)}(n)$  is obtained from  $\hat{\xi}_{>}^{(AR)}(z)$  by polynomial division. Writing (4.50) in the form  $\hat{\xi}_{>}^{(AR)}(z) = c_0 + c_1 z^{-1} + c_2 z^{-2} + \dots$  one can identify  $c_0 \equiv \zeta(0)$ ,  $c_1 \equiv \zeta(1)$ ,  $\dots$

A numerical estimation of the generalized friction coefficient is obtained from

$$\zeta_0^{(AR)} = \sum_{n=0}^{\infty} \Delta t \zeta(n) = \Delta t \hat{\xi}_{>}^{(AR)}(1). \quad (4.51)$$

which corresponds to the integral over the memory function. Using (4.50) we obtain thus within the AR model

$$\zeta_0^{(AR)} = \frac{1}{\Delta t} \frac{1}{\sum_{j=1}^P \beta_j \frac{1}{1-z_j}}, \quad (4.52)$$

where  $\{z_j\}$  are again the zeros of the characteristic polynomial (4.41).

#### 4.7 ScrewFit algorithm

The ScrewFit parameters for the secondary structure analysis of proteins are obtained through a quaternion-based superposition fit of the protein peptide planes (formed by the atoms O, C, N in the protein backbone). The superposition fit of molecular structures is described by the target function

$$m(q) = \sum_{\alpha=1}^N w_{\alpha} (\mathbf{D} \cdot \mathbf{x}_{\alpha} - \mathbf{x}'_{\alpha})^2, \quad (4.53)$$

where  $\{x_{\alpha}\}$  and  $\{x'_{\alpha}\}$  are the coordinate sets giving the atomic positions of two successive planes relative to the respective rotation centers. In ScrewFit, the rotation centers are chosen to be the C atoms. The weights  $w_{\alpha}$  are a positive numbers that satisfy  $\sum_{\alpha} w_{\alpha} = 1$ , and  $\mathbf{D}$  is the orthogonal rotation matrix expressed in terms of the parameters of a normalized quaternion  $q = \{q_0, q_1, q_2, q_3\}$ :

$$\mathbf{D}(q) = \begin{pmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(-q_0q_3 + q_1q_2) & 2(q_0q_2 + q_1q_3) \\ 2(q_0q_3 + q_1q_2) & q_0^2 + q_2^2 - q_1^2 - q_3^2 & 2(-q_0q_1 + q_2q_3) \\ 2(-q_0q_2 + q_1q_3) & 2(q_0q_1 + q_2q_3) & q_0^2 + q_3^2 - q_1^2 - q_2^2 \end{pmatrix} \quad (4.54)$$

The function (4.53) is to be minimized with respect to these four parameters, which must obey the normalization condition  $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$ . As demonstrated in previous works [32, 54], the

constrained minimization problem can be mapped to an eigenvector problem for a positive semi-definite matrix of the type  $\mathbf{M} \equiv \mathbf{M}(\{\mathbf{x}_\alpha, \mathbf{x}'_\alpha\})$ ,

$$\mathbf{M} \cdot \mathbf{q} = \lambda \mathbf{q}, \quad (4.55)$$

where  $\lambda$  is the superposition error. One then obtains four eigenvectors  $\mathbf{q}_j$  and four associated eigenvalues  $\lambda_j = m(\mathbf{q}_j)$ , the smallest of which is associated with the quaternion yielding the optimal superposition. The description of a rotation in terms of quaternion parameters does not only lead to an efficient and elegant solution of the superposition problem, but also allows the rotation axis to be easily determined. Defining  $\mathbf{n}$  to be a unit vector along the rotation axis and  $\phi$  the rotation angle, we have  $\mathbf{q} \equiv (q_0, \mathbf{q}_v) = (\cos(\phi/2), \sin(\phi/2)\mathbf{n})$ . The corresponding rotation axis  $\mathbf{n}$  is also the direction of the screw motion, according to Chasles' theorem [18].

The ScrewFit Analysis (SFA) implemented in *nMoldyn* provides three ScrewFit parameters to characterize protein secondary structures and their evolution in time during a molecular dynamics simulation:

1. The *orientational distance* is defined as

$$\Delta = \sqrt{\frac{\sum_{\alpha=1}^3 (\mathbf{x}_\alpha - \mathbf{x}'_\alpha)^2}{\lambda_{max}}}. \quad (4.56)$$

where the quaternion associated with the largest eigenvalue,  $\lambda_{max}$ , describes the “worst” superposition and thus the eigenvalue  $\lambda_{max}$  defines the maximal Euclidean distance between the two peptide planes. By definition  $0 \leq \Delta \leq 1$ .

2. The *radius of the screw motion* defined by the Chasles' theorem and relating two consecutive peptide planes,

$$\rho_i = \frac{|\mathbf{t}_\perp|}{2} \sqrt{1 + \cot^2(\phi/2)}. \quad (4.57)$$

where, defining  $\mathbf{t}$  to be the vector relating the C-atoms of consecutive peptide planes,  $\mathbf{t}_\perp$  is its component perpendicular to the unit vector  $\mathbf{n}$  which defines the rotation axis. The radius  $\rho$  is a measure of the local curling of the backbone conformation. In a flat backbone conformation, such as an extended  $\beta$ -strand,  $\rho$  is close to zero; when the local backbone conformation is curled, as in the case of  $\alpha$ -helices and  $\beta$ -turns,  $\rho$  increases.

3. The *straightness* parameter  $\sigma$ . For residue  $i$  the latter is defined as

$$\sigma_i = \mu_i^T \cdot \mu_{i+1}, \quad (4.58)$$

where

$$\mu_i = \frac{\mathbf{R}_{i+1}^\perp - \mathbf{R}_i^\perp}{|\mathbf{R}_{i+1}^\perp - \mathbf{R}_i^\perp|}, \quad (4.59)$$

and  $\mathbf{R}_i^\perp$  is the point on the helix axis, which is closest to the C-atom of peptide plane  $i$ . The straightness gives information about the curvature of a secondary structure element.

#### 4.8 File formats

*nMoldyn* expects trajectory files to follow the conventions of the Molecular Modelling Toolkit. This means that all trajectory data is stored in a netCDF file, which also contains the definition of the system. The advantage of using this format is that the files are platform-independent, compact, and self-contained, which reduces the risk of user errors. Moreover, it is possible to read trajectory data atom by atom efficiently, i.e. without reading in the whole file each time. In the latest version of *nMoldyn*, several format converters have been implemented to allow the treatment of trajectories produced by simulation programs other than MMTK, such as Amber, CHARMM/X-PLOR, DL POLY, MaterialsStudio, NAMD

and VASP. The output files produced by *nMoldyn* are in netCDF format, which allows more compact data storage especially for the functions of two variables. *nMoldyn* also provides a plotting facility for these functions, with both 3D and 2D representations. Furthermore, the variables contained in a netCDF file (frequency, time, momentum transfer, time correlation function, frequency spectrum, . . . ) can be extracted and exported to ASCII file.

## 5. CONCLUSIONS

*nMoldyn* is a flexible and user-friendly analysis programs for simulation data. Among the new functionalities introduced in the latest version we emphasize the computation of radial distribution function, static structure factor, partial terms of all scattering-related properties, time correlation functions relevant to NMR relaxation spectroscopy, and secondary structure analysis of proteins. An effort has been made to offer also a new and much more powerful atom selection engine. The implementation of analyses devoted to the IR-THz spectroscopy is in progress. *nMoldyn* is now also parallelized and the corresponding details will be published elsewhere.

## References

- [1] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J. Chem. Phys.*, 21:1087–1092, 1953.
- [2] A. Rahman. Correlations in the motion of liquid argon. *Phys. Rev.*, 136(2A):405–411, 1964.
- [3] R. Car and M. Parrinello. Unified approach for Molecular Dynamics and density-functional theory. *Phys. Rev. Lett.*, 55(22):2471–2474, 1985.
- [4] S.W. Lovesey. *Theory of Neutron Scattering from Condensed Matter*, volume I. Clarendon Press, Oxford, 1984.
- [5] G.R. Kneller, V. Keiner, M. Kneller, and M. Schiller. Nmoldyn, a program package for the calculation and analysis of neutron scattering spectra from MD simulations. *Comp. Phys. Comm.*, 91:191–214, 1995. Full description in report ILL95KN02T, Institut Laue-Langevin, 156 X, F-38042 Grenoble Cedex, France.
- [6] G.R. Kneller, J.C. Smith, S. Cusack, and W. Doster. Methyl group dynamics in the crystalline alanine dipeptide: A molecular dynamics and incoherent neutron scattering analysis. *J. Chem. Phys.*, 97(12):8864–8879, 1992.
- [7] A.J. Dianoux, G.R. Kneller, J.L. Sauvajol, and J.C. Smith. The polarized density of states of crystalline polyacetylene: Molecular dynamics Analysis and comparison with neutron scattering results. *J. Chem. Phys.*, 99(7):5586–5596, 1993.
- [8] A.J. Dianoux, G.R. Kneller, J.L. Sauvajol, and J.C. Smith. Dynamics of sodium-doped polyacetylene. *J. Chem. Phys.*, 101(1):634–644, 1994.
- [9] G.R. Kneller and J.C. Smith. Liquid-like side-chain motions in myoglobin. *J. Mol. Biol.*, 242:181–185, 1994.
- [10] T. Rog, K. Murzyn, K. Hinsén, and G.R. Kneller. nMoldyn : A Program Package for a Neutron Scattering Oriented Analysis of Molecular Dynamics Simulations. *J. Comp. Chem.*, 24(5):657–667, 2003.
- [11] G.R. Kneller and K. Hinsén. Computing memory functions from Molecular Dynamics simulations. *J. Chem. Phys.*, 115(24):11097–11105, 2001.
- [12] R. Zwanzig. *Nonequilibrium statistical mechanics*. Oxford University Press, 2001.
- [13] G.R. Kneller and K. Hinsén. Fractional brownian dynamics in proteins. *J. Chem. Phys.*, 121(20):10278–10283, 2004.
- [14] V. Calandrini, G. Sutmann, A. Deriu, and G.R. Kneller. The role of effective atomic masses in memory function-based models for liquids : A simulation study of liquid water. *J. Chem. Phys.*, 125:236102, 2006.

- [15] V. Calandrini, V. Hamon, K. Hinsens, P. Calligaris, M.-C. Bellissent-Funel, and G.R. Kneller. Relaxation dynamics of lysozyme in solution under pressure: Combining molecular dynamics simulations and quasielastic neutron scattering. *Chem. Phys.*, 345:289–297, 2008.
- [16] V. Calandrini and G.R. Kneller. Influence of pressure on the fractional relaxation dynamics in proteins: A simulation study. *J. Chem. Phys.*, 128(6):065102, 2008.
- [17] K. Hinsens. The molecular modeling toolkit: A new approach to molecular simulations. *J. Comp. Chem.*, 21:79–85, 2000.
- [18] G.R. Kneller and P. Calligaris. Efficient characterisation of protein secondary structure in terms of screw motions. *Acta Crystallographica D*, 62:302–311, 2006.
- [19] L.E. Reichl. *A Modern Course in Statistical Physics*. Arnold Publishers, 1980.
- [20] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw Hill, New York, 3rd edition, 1991.
- [21] A. Einstein. ‘Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen. *Ann. Phys.*, 322(8), 1905.
- [22] J.P. Boon and S. Yip. *Molecular Hydrodynamics*. McGraw Hill, New York, 1980.
- [23] P. Langevin. Sur la théorie du mouvement brownien. *C. Rendus Acad. Sci. Paris*, 146:530–533, 1908.
- [24] J.-P. Hansen and I.R. McDonald. *Theory of simple liquids*. Academic Press, 2nd edition, 1986.
- [25] W. Smith, C.W. Yong, and P.M. Rodger. Dlpoly: Application to molecular simulation. *Molecular Simulation*, 28(1):385–471, 2002.
- [26] H.J.C. Berendsen, J.R. Grigera, and T.P. Straatsma. The missing term in effective pair potentials. *J. Phys. Chem.*, 91:6269 – 6271, 1987.
- [27] W.D. Cornell, P. Cieplak, C.I. Bayly, I.R. Gould, K.M. Merz Jr, D.M. Ferguson, D.C. Spellmeyer, T. Fox, J.W. Caldwell, and P.A. Kollman. A second generation force field for the simulation of proteins and nucleic acids. *J. Am. Chem. Soc.*, 117:5179, 1995.
- [28] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28:235–242, 2000.
- [29] M.C. Vaney, S. Maignan, M. RiesKautt, and A. Ducruix. High-resolution structure (1.33 angstrom) of a HEW lysozyme tetragonal crystal grown in the APCR apparatus. Data and structural comparison with a crystal grown under microgravity from spacehab-01 mission. *Acta Cryst. D Biol. Cryst.*, 52:505–517, 1996.
- [30] H.C. Andersen. Molecular dynamics at constant pressure and/or constant temperature. *J. Chem. Phys.*, 72(4):2384–2393, 1980.
- [31] S. Nosé. A unified formulation of the constant temperature molecular dynamics methods. *J. Chem. Phys.*, 81(1):511–519, 1984.
- [32] G.R. Kneller. Superposition of molecular structures using quaternions. *Mol. Sim.*, 7:113–119, 1991.
- [33] L. van Hove. Correlations in space and time and born approximation in systems of interacting particles. *Phys. Rev.*, 93(1):249–262, 1954.
- [34] M. Bée. *Quasielastic Neutron Scattering: Principles and Applications in Solid State Chemistry, Biology and Materials Science*. Adam Hilger, Bristol, 1988.
- [35] A. Rahman, K.S. Singwi, and A. Sjolander. Theory of slow neutron scattering by liquids. I. *Phys. Rev.*, 126(3):986–996, 1961.
- [36] J.P. Boon and S. Yip. *Molecular Hydrodynamics*. McGraw Hill, 1980.
- [37] G.C. Wick. The scattering of neutrons by systems containing light nuclei. *Phys. Rev.*, 94(5):1228–1242, Jun 1954.
- [38] G.R. Kneller. Neutron scattering from classical systems: Stationary phase approximation of the scattering law. *Mol. Phys.*, 83(1):63–87, 1994.

- [39] G. Zaccai. How soft is a protein? A protein dynamics force constant measured by neutron scattering. *Science*, 288:1604–1607, 2000.
- [40] G.R. Kneller and K. Hinsen. Quantitative model for the heterogeneity of atomic position fluctuations in proteins: A simulation study. *J. Chem. Phys.*, 131(4):045104, 2009.
- [41] M. Abramowitz and I.A. Stegun. *Handbook of Mathematical Functions*. Dover Publications, New York, 1972.
- [42] K. Hinsen and G.R. Kneller. Projection methods for the analysis of complex motions in macromolecules. *Mol. Sim.*, 23:275–292, 2000.
- [43] D.A. McQuarrie. *Statistical Mechanics*. Harper’s Chemistry Series. Harper Collins Publishers, 1976.
- [44] D. Chandler. *Introduction to Modern Statistical Mechanics*. Oxford University Press, New York, Oxford, 1987.
- [45] Vania Calandrini, Godehard Sutmann, Anotnio Deriu, and Gerald R. Kneller. Rigid molecule approximation in memory function-based models for molecular liquids: Application to liquid water. *Z. Phys. Chem.*, 223:957–978, 2009.
- [46] K.T. Gillen, D.C. Douglas, and M.J.R. Hoch. Self-diffusion in liquid water to  $-31^{\circ}$  C. *Journal of Chemical Physics*, 57:5117, 1972.
- [47] D. M. Korzhnev, M. Billeter, A. S. Arseniev, and V. Y. Orekhov. NMR studies of Brownian tumbling and internal motions in proteins. *Progress in Nuclear Magnetic Resonance Spectroscopy*, 38(3):197–266, April 2001.
- [48] G. Lipari and A. Szabo. Model-free approach to the interpretation of nuclear magnetic resonance relaxation in macromolecules. 1. theory and range of validity. *J. Am. Chem. Soc.*, 104:4546–4559, 1982.
- [49] Vania Calandrini, Daniel Abergel, and Gerald R. Kneller. Fractional protein dynamics seen by NMR spectroscopy: Relating molecular dynamics simulation and experiment. *J. Chem. Phys.*, 133:145101, 2010.
- [50] R. Metzler and J. Klafter. The random walk’s guide to anomalous diffusion: A fractional dynamics approach. *Phys. Rep.*, 339:1–77, 2000.
- [51] James C. Phillips, Rosemary Braun, Wei Wang, James Gumbart, Emad Tajkhorshid, Elizabeth Villa, Christophe Chipot, Robert D. Skeel, Laxmikant Kalé, and Klaus Schulten. Scalable molecular dynamics with namd. *J Comput Chem.*, 26(16):1781–1802, 2005.
- [52] Viktor Hornak, Robert Abel, Asim Okur, Bentley Strockbine, Adrian Roitberg, and Carlos Simmerling. Comparison of multiple amber force fields and development of improved protein backbone parameters. *Proteins*, 65(3):712–25, Nov 2006.
- [53] Tom Darden, Darrin York, and Lee Pedersen. Particle mesh ewald: An  $n \cdot \log(n)$  method for ewald sums in large systems. *J. Chem. Phys.*, 98(12):10089, 1993.
- [54] G.R. Kneller. Comment on “Using quaternions to calculate RMSD” [J. Comp. Chem. 25, 1849 (2004)]. *J. Comp. Chem.*, 26(15):1660–1662, 2005.
- [55] D.A. McQuarrie. *Statistical Mechanics*. Harper’s Chemistry Series. Harper Collins Publishers, 1976. CBM.
- [56] V. Calandrini. Lysozyme dynamics in the thz frequency region: effect of the solvent and thermal activation of functional motions. Experimental Report 20090288, Synchrotron Soleil, 2010.
- [57] L. Stryer. *Biochemistry*. W.H. Freeman and Company, 1988.
- [58] P.A. Calligari, G.R. Kneller, A. Giansanti, Ascenzi P., A. Porrello, and A. 1 Bocedi. Group-1 and group-2 neuraminidases: Screw motion analysis reveals fine structural changes induced by binding with oseltamivir. *Biophysical Chemistry*, 141(1):117–123, 2009.
- [59] E.O. Brigham. *The Fast Fourier Transform*. Prentice Hall, 1974.
- [60] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw Hill, 3rd edition, 1991. CBM.

- [61] A. Papoulis. *Signal Analysis*. McGraw Hill, New York, 1984.
- [62] J. Burg. *Maximum entropy spectral analysis*. PhD thesis, Stanford University, Stanford (CA), USA, 5 1975.
- [63] J. Makhoul. Stable and efficient lattice methods for linear prediction. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-25(5):423–428, 1977.
- [64] S. Haykin. *Adaptive Filter Theory*. Prentice Hall, 1996.