DISSERTATION

WATER DYNAMICS IN CONFINED SYSTEMS AND AQUEOUS MIXTURES EXPLORED BY NEUTRON SCATTERING AND SIMULATIONS

Submitted by

Michael Robert Harpham

Department of Chemistry

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Spring 2007

COLORADO STATE UNIVERSITY

28 February 2007

WE HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER OUR SUPERVISION BY MICHAEL R. HARPHAM ENTITLED "WATER DYNAMICS IN CONFINED SYSTEMS AND AQUEOUS MIXTURES EXPLORED BY NEUTRON SCATTERING AND SIMULATIONS" BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.

Committee on Graduate Work

B. G. Barisas

David A. Krueger

Advisor

Co-Advisor

Department Chair and Committee Member

ABSTRACT OF DISSERTATION

WATER DYNAMICS IN CONFINED SYSTEMS AND AQUEOUS MIXTURES EXPLORED BY NEUTRON SCATTERING AND SIMULATIONS

We investigate binary mixtures of water and DMSO through MD simulations. We evaluate the self-intermediate scattering functions (ISF's), $F_S(\mathbf{Q},t)$, which are related by a Fourier transform to the incoherent structure factors, $S(\mathbf{Q},\omega)$, measured in quasielastic neutron scattering (QNS) experiments. We find that translational and rotational motion is slowest at 0.5 mole fraction DMSO. We use our simulation results to also assess the range of validity of the dynamical models used in the analysis of QNS data. We find that the jump-translational diffusion model, used to obtain information about translational diffusion from QNS spectra, represents water translational dynamics quite well. Our MD data suggests that the decoupling approximation, used to extract information on rotational relaxation from QNS spectra, becomes less accurate as the mixture composition approaches $X_{DMSO} = 0.5$. We find that rotational relaxation deviates quite strongly from the isotropic rotational diffusion model, and we explore this issue further by investigating the behavior of orientational time correlation functions for different unit vectors.

We explore the mobility and structure of water molecules in Aerosol OT (bis(2-ethylhexyl) sulfosuccinate, AOT) reverse micelles (RM's) of varying water content and surfactant counterion through simulations and experiments. Experimental QNS results

indicate that the water translational and rotational dynamics become more hindered with decreasing water content, and less hindered with increasing counterion size. MD simulations were performed using the Faeder/Ladanyi model (*J. Phys. Chem. B*, **2000**, *104*, 1033) of the RM interior to evaluate the ISF's for water hydrogen atoms. Our simulated QNS spectra match experimental data quite well. Our simulations show that the decay of $F_S^{CM}(\mathbf{Q},t)$ is nonexponential; we attribute this to confinement effects and decreased water mobility for molecules close to the interface. Rotational relaxation is anisotropic and also exhibits nonexponential decay. Differences in interfacial mobility of water molecules are strongly correlated with structural features, especially ion-water coordination, and the extent of disruption of the water hydrogen bond network by the counterions. Our QNS investigation of surfactant effects on water dynamics within quaternary reverse micelles showed that water motion within these systems is significantly hindered relative to that of bulk water or even water in ternary reverse micelles.

Michael Robert Harpham
Chemistry Department
Colorado State University
Fort Collins, CO 80523
Spring 2007

Acknowledgements

How to begin? If the person reading this knows me at all, they know my feelings on an acknowledgement section of a dissertation. I have recently cracked a number of jokes about the sentimental feelings expressed by others (people I know, not strangers) in their Acknowledgements, and I fully expect that others will do the same to me. So, I have been dreading writing this. In fact, this is the last part I am writing before I print this out.

I guess I'll start by thanking my mother and father. They have been a great help in keeping me motivated. And for the last two years, they have endured my deflections when they asked "So, when are you going to graduate?"

In addition to the research knowledge I gained, working with Profs. Branka Ladanyi and Nancy Levinger taught me a number of things - how to argue my point (in a scientific way), present and write effectively, and how to think critically about my research and the research of others. Also, Prof. Bridget Gourley is not one of my advisors, but she has acted as another co-advisor in the last few years, and has had an influence in my decision about the next step of my career.

I would also like to thank Elliot Bernstein and Grzegorz Szamel for discussions about research or whatever random topic about which I stopped in to talk to them. (You see there how I avoided ending with a preposition? That is because of Nancy's bad joke about prepositions.) I can also say that I did not mind at all when Grzegorz would tell me to solve a thermo or stat. mech. problem on the board. Whether I initially knew the answer or not, I would learn something each time I went up there.

I have had a desk in Levinger group office since the beginning, so I've spent most of my time interacting with the people there. I would like to thank (and blame), in the order that I met them, Benjamin Langdon, Elizabeth Corbeil, George Bullock, Wendy Schatzberg, Kristopher Lange, Richard Cole, Myles Sedgwick, and Jennifer O'patchen, for making it difficult to leave here. I will always remember working with and hanging out with them.

I should also thank the people at Panino's. I have gone there once a week for the last three years, and I don't know how many problems I've solved over lunch there. While I am thanking them, I should thank the Sweet Sinsations for supplying the caffeine necessary to make me productive after lunch at Panino's.

I should probably wrap this up before I start to ramble. I have been here for awhile. When I am comfortable with people and a place, I find it hard to move on, even when I know I should move on. As I mentioned before, my parents would ask about when I thought I would graduate, and if they nagged enough, I would respond to their question with "When I feel like it! All right!?!" Their follow-up would sometimes be "When is that?" Finally, I think I have an answer, and I feel good about it.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Chemicals and Systems	
Sodium bis(2-ethylhexyl) sulfosuccinate	NaAOT, Aerosol OT
Cetyl trimethyl ammonium bromide	CTAB
Sodium dodecyl sulfate	SDS
Dimethyl sulfoxide	DMSO
Reverse Micelle	RM
Techniques, Functions, and Potentials	
Quasielastic Neutron Scattering (Technique)	QNS
Dynamic Light Scattering	DLS
Molecular Dynamics	MD
Monte Carlo	MC
Lennard-Jones	LJ
Pair Density Function	PDF
Radial Density Profile	RDP
Intermediate Scattering Function	ISF
Time Correlation Function	TCF
Locations/Divisions	
Argonne National Laboratory	ANL
Intense Pulsed Neutron Source	IPNS
National Institute of Standards and Technology	NIST
NIST Center for Neutron Research	NCNR
Spallation Neutron Source	SNS
Instruments	
Disk Chopper Spectrometer	DCS
Quasielastic Neutron Scattering (Spectrometer)	QENS

Chapter 1

Introduction

1.I. Systems

1.I.A. Reverse Micelles

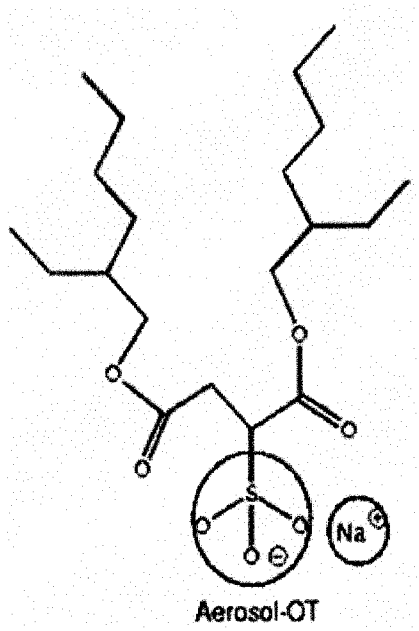
In the appropriate concentration range, mixtures of polar, nonpolar, and amphiphilic molecules can form self-assembled structures called reverse micelles (RMs), in which the polar phase is sequestered from the nonpolar phase by the amphiphilic surfactant. The most commonly-studied surfactant for RMs is sodium bis-(2-ethylhexyl) sulfosuccinate (Aerosol-OT, NaAOT, Fig. 1a), which forms monodisperse, spherical, and stable RMs over a large concentration range. Single-tailed surfactants such as cetyl-trimethylammonium bromide (CTAB, Fig. 1b) and sodium dodecyl sulfate (SDS, Fig. 1c) require a cosurfactant, usually an aliphatic alcohol, to form stable microemulsions. The purpose of the cosurfactant is to intercalate between the surfactant molecules, and minimize the repulsive forces between the surfactant headgroups.

Reverse micelles are often discussed in terms of w_0, which is defined as:

$$w_0 = [\text{polar solvent}]/[\text{surfactant}] \tag{1.1}$$

In the case of spherical RMs, the radius of the RM varies linearly with w_0. Although there are a number of polar solvents which can form RMs, aqueous RMs will be discussed in this work.

a)



b)

c)

Figure 1.1. Structure of common surfactants NaAOT (a), CTAB (b), and SDS (c).

Reverse micelles are used in a range of applications, from templates for semiconductor and nanoparticle synthesis, [1-5] to restricted "nanobeakers" for chemical reactions,[4] and to models for biological membranes.[6] Reverse micelles have been investigated by a number of techniques and methods, including dynamic light scattering, NMR, fluorescence spectroscopy, neutron scattering, viscometry, X-ray scattering, and computer simulation. These studies have shown that the properties of the polar solvent within the RM can differ greatly from the properties of bulk liquid, both structurally and dynamically, due to the geometric size constraints of the RM as well as the intermolecular interactions between the polar solvent and the micellar interface.[7-11]

Surfactant and counterion charge is thought to have a significant effect on the dynamics and structure of water near the RM interface. Specifically, interfacial water is thought to have reduced mobility and reduced hydrogen bonding when compared to core water or bulk water. These effects are generally attributed to the structuring of water molecules around the charged particles at the RM interface.

1.I.B. Binary Mixtures

Two-component mixtures receive a considerable amount of attention from both experimentalists and theoreticians. Even though there are only two species involved, the interaction between these constituents is complicated, especially in aqueous mixtures involving hydrogen bonding liquids such as acetonitrile, [12-17] methanol, [11,13,18-20] or dimethyl sulfoxide.[14,21-42] Aqueous binary mixtures have been explored through a variety of techniques, including steady-state spectroscopy, X-ray scattering, ultrafast spectroscopy, neutron scattering experiments, and molecular dynamics simulations.

These techniques provide insights into the structure and dynamics of water and co-solvent within aqueous binary mixtures.

Binary mixtures are studied for a number of reasons. From a pure science perspective, the structural and dynamical features that arise from the interactions between the species involved are quite interesting. The combination of hydrophobic and hydrophilic effects present in the aqueous binary mixtures results in a variety of interspecies interactions, which greatly affects the dynamics of each component. Practically, binary mixtures can be used as "tunable solvents", where various properties such as viscosity, dielectric constant, and density can be varied by changing the composition of the mixture.

It is well known that the dynamics of water are typically slower in aqueous mixtures than in neat liquid. There are numerous possible explanations for this, such as increased structuring of water molecules around the co-solvent molecules and disruption of the water hydrogen bond network. The relationship between a slowing down of dynamics as a consequence of increased water structure has numerous explanations, such as the so-called "iceberg hypothesis" of Frank and Evans.[43] The combination of hydrophobic and hydrophilic effects may cause self-aggregation of water and the co-solvent, and the degree of this self-aggregation may vary by the composition of the mixture studied.

1.II. Goals

The overall goal of this research is to investigate the dynamics of water in binary mixtures and in confined systems. In both cases, we expect to see a disruption of the water hydrogen bond network. However, the cause of the disruption as well as the degree of disruption may vary by system. We expect these systems will influence dynamical and structural properties of water in different ways. Water within ionic reverse micelles will experience confinement effects as well as ion-dipole interactions with the charged surfactant and counterions, while water mixtures will be influenced by the hydrophobic and hydrophilic interactions present in the binary mixture.

We focus our binary mixtures study on mixtures of water with dimethyl sulfoxide (DMSO, $(CH_3)_2SO$). DMSO, an aprotic dipolar solvent, mixes readily with water in all proportions, and has uses in biological systems. Specifically, DMSO has the ability to readily penetrate cell membranes without compromising their structural integrity, which makes DMSO a good drug-delivery system.[22] Because water forms stronger hydrogen bonds to DMSO than to water,[44] we expect that the structural and dynamical properties of the mixture will be strongly influenced by the water-DMSO hydrogen bonding network.

Our investigation of water in confined systems is focused on ionic RM systems. We expect to see ion-dipole interactions between surfactants and counterions with the water within the reverse micelles. This effect should be the greatest at the interface, where the electric field gradient is the strongest and where we expect to see a reduction in the number of hydrogen bonds. We explore two ionic RM systems, ternary reverse micelles in which we use the negative-charged surfactant AOT, and quaternary reverse micelles with the anionic surfactant SDS or cationic surfactant CTAB. We use pentanol

as a cosurfactant for these quaternary RM studies. Our ternary RM studies focus on two effects - the effect of RM size and the effect of surfactant counterion on water dynamics within the RM. Our quaternary RM studies investigate the effects of surfactant headgroup and counterion charge on water dynamics within the RM.

We investigate these systems through a combination of quasielastic neutron scattering (QNS) experiments and molecular dynamics (MD) simulations because each of these techniques relies on a set of assumptions. The analysis of experimental QNS data relies on several assumptions whose validity cannot be determined solely on the basis of the experimental data. Experimental QNS analyses are limited in the timescale of the motion they probe; the resolution function of the instrument determines the fastest and slowest motion that can be measured. MD simulations can be used to evaluate the validity of these experimental assumptions, as well as determine properties that may not be observable in the QNS experiment. In the case of MD simulation, assumptions are made in the functional form of the potentials used, as well as in the rigidity of the molecules simulated.

1.III. Techniques

1.III.A. Neutron Scattering

Neutrons are elementary particles with zero charge. When a neutron passes near a nucleus, there are two possibilities: the neutron is absorbed, or the neutron is scattered. In the case of absorption, a compound nucleus in an excited state is formed. This compound nucleus then decays into the ground state, typically through γ emission. The excited state may also decay through the emission of charged particles such as tritons and α particles,

or through fission. In the case of neutron scattering, the direction and energy of the neutron is changed. The energy of a thermal neutron is too small to cause an internal excitation of a nucleus or an electronic shell. However, nuclear motion can be felt by the scattered neutron, and give rise to inelastic neutron scattering.

The kinetic energy of a neutron can simply be related to its velocity v by:

$$E = \frac{1}{2}mv^2 \tag{1.2}$$

where m is the mass of the neutron, 1.066×10^{-24} kg. One can consider a neutron as a plane wave, with a wavevector $\mathbf{k}$ given by:

$$\mathbf{k} = \frac{2\pi m}{h}\mathbf{v} \tag{1.3}$$

and λ, the wavelength of the neutron, is given as:

$$\lambda = \frac{2\pi}{|\mathbf{k}|} = \frac{h}{m|\mathbf{v}|} \tag{1.4}$$

Using equations (1.3) and (1.4), we can rewrite the kinetic energy as:

$$E = \frac{h^2k^2}{8\pi^2 m} = \frac{h^2}{2m\lambda^2} \tag{1.5}$$

and relate the kinetic energy of a neutron to its wavevector and wavelength.

A neutron scattering experiment measures two basic quantities, the energy transfer, $\hbar\omega$ or ΔE, and the scattering vector $\mathbf{Q}$, corresponding to the wavevector transfer. For an initial state with energy E_0 and wavevector $\mathbf{k}_0$ and final energy E and wavevector $\mathbf{k}$, we can define the energy transfer as:

$$\hbar\omega = E - E_0 = \frac{h^2}{2m}\left(k^2 - k_0^2\right) \tag{1.6}$$

and the scattering vector as:

$$\mathbf{Q} = \mathbf{k} - \mathbf{k}_0 \tag{1.7}$$

The relationship between **k**, $\mathbf{k}_0$, **Q**, and scattering angle θ is depicted in the so-called "Scattering Triangle" in Figure 1.2. A scattering event which has zero energy transfer is called elastic scatter, while a nonzero energy transfer is considered inelastic scatter. Quasielastic scatter is considered a small energy transfer, where "small" is relative to the energy of the neutron.

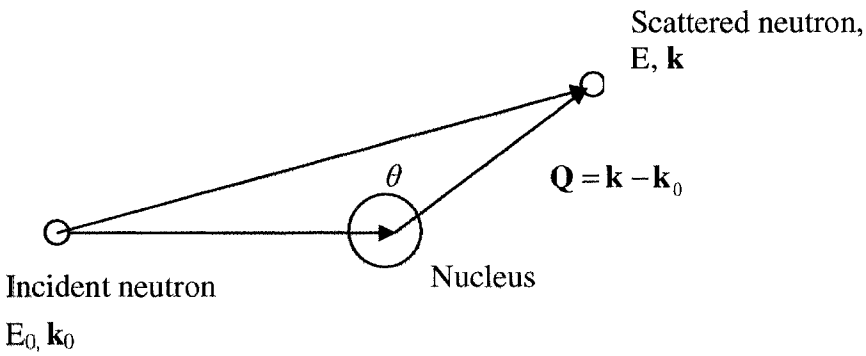


Figure 1.2. Depiction of a neutron scattering event

Assuming a current of I_0 neutrons per second per square centimeter incident on the sample, we can determine from the number of scattering and absorption events occurring each second, I_s and I_a, a scattering and an absorption cross section, σ_s and σ_a, respectively, by the relation:

$$I_s = I_0 \sigma_s \tag{1.8}$$

$$I_a = I_0 \sigma_a \tag{1.9}$$

Both cross-sections have units of surface, typically expressed in terms of barns. (1 barn = 10^{-24} cm^{-2}) At low energies, σ_a is a smooth function of neutron energy, and is proportional to the neutron wavelength.

The differential cross section, $\frac{\partial\sigma}{\partial\Omega}$, in one direction Ω gives the probability that a neutron leaves the sample in a solid angle $\partial\Omega$ about the direction Ω. The double-differential cross section

$$\frac{\partial^2\sigma}{\partial\Omega\partial E}=\frac{1}{\hbar}\frac{\partial^2\sigma}{\partial\Omega\partial\omega} \tag{1.10}$$

gives the probability that a neutron with initial energy E_0 leaves in a solid angle $\partial\Omega$ about the direction Ω with an energy exchange between $\hbar\omega=E-E_0$ and $\hbar(\omega+d\omega)=(E-E_0)+dE$. It is clear that

$$\sigma=\int d\Omega\frac{\partial\sigma}{\partial\Omega}=\int d\omega\int d\Omega\frac{\partial^2\sigma}{\partial\Omega\partial\omega} \tag{1.11}$$

Neutrons interact with a nucleus by short-range magnetic and nuclear forces, and this interaction has a very short range compared to the neutron wavelength and nuclear radius of the atom.[45] From this, it can be shown that the neutron nucleus scattering is isotropic and is characterized by a the scattering length, b, which is independent of the neutron energy. The scattering length b can be complex, where the real portion is either positive or negative according to the attractive or repulsive nature of the interaction. The imaginary portion of the scattering length represents absorption.

In an experiment, the sample is composed of i atomic species, each of them being a mixture of several isotopes possessing a nuclear spin. The scattering length b_i vary by element and by isotope of the element, because the nature of the neutron-nucleus

interaction depends on the nature of the nucleus as well as the nuclear spin state. The average $\langle b_i \rangle$ over all isotopes and spin states is defined as the coherent scattering length. The incoherent scattering length is defined as the root mean square deviation of b_i from $\langle b_i \rangle$.

$$b_i^{coh} = \langle b_i \rangle \tag{1.12}$$

$$b_i^{inc} = \left[\langle b_i^2 \rangle - \langle b_i \rangle^2 \right]^{1/2} \tag{1.13}$$

The so-called bound cross section is given as:

$$\sigma_{coh,i} = 4\pi \langle b_i \rangle^2 \tag{1.14}$$

for the coherent cross section, and

$$\sigma_{inc,i} = 4\pi \left[\langle b_i^2 \rangle - \langle b_i \rangle^2 \right] \tag{1.15}$$

for the incoherent cross section. The total bound cross section is then a sum of Eqs. (1.14) and (1.15). An important atom in neutron scattering is the hydrogen atom, which has an incoherent cross section of 79.9 barns, and a coherent cross section of 1.8 barns.

The double differential cross section for a sample of N nuclei is given as:

$$\frac{\partial^2 \sigma}{\partial \Omega \partial \omega} = \frac{k}{k_0} \frac{1}{2\pi} \sum_i \sum_j \int_{-\infty}^{\infty} \frac{1}{N} \left\langle b_i b_j \exp\{i\mathbf{Q}\cdot\mathbf{r}_i(t)\} \times \exp\{i\mathbf{Q}\cdot\mathbf{r}_j(0)\} \right\rangle \exp(-i\omega t)\, dt \tag{1.16}$$

for nucleus i at position $\mathbf{r}_i$ at time t and nucleus j at position $\mathbf{r}_j$ at time t=0. As a general rule, we assume there is no coupling between the scattering length of a nucleus and its position. [45] When $i \neq j$, the scatter is coherent, whereas for $i = j$, the scattering is incoherent.

Quasielastic neutron scattering (QNS) is an incoherent neutron scattering method that measures the dynamic structure factor $S_{inc}(\mathbf{Q},\omega)$ [45] as a function of momentum transfer $\mathbf{Q}$ and frequency ω. For a nucleus of a given type, denoted by α, $S_{inc}^{\alpha}(\mathbf{Q},\omega)$ is the frequency Fourier transform of the self-intermediate scattering function (ISF), $F_s^{\alpha}(\mathbf{Q},t)$

$$S_{inc}^{\alpha}(\mathbf{Q},\omega)=\frac{1}{2\pi}\int_{-\infty}^{\infty}F_S^{\alpha}(\mathbf{Q},t)e^{i\omega t}dt \tag{1.17}$$

$$F_s^{\alpha}(\mathbf{Q},t)=\frac{1}{N_{\alpha}}\sum_{j=1}^{N_{\alpha}}\left\langle\exp\left\{i\mathbf{Q}\cdot\left[\mathbf{r}_{j,\alpha}(0)-\mathbf{r}_{j,\alpha}(t)\right]\right\}\right\rangle \tag{1.18}$$

where N_{α} is the number of atoms of a given type α and $\mathbf{r}_{j,\alpha}(t)$ is the position of the jth atom of a given type α.

To interpret the QNS results, one typically assumes that translation, vibration, and rotation are decoupled.[45] Furthermore, since vibrational motion occurs on a very short timescale, it is assumed to be energy-independent.[45] Within these assumptions, the incoherent structure factor is rewritten as:

$$S_{inc}^{\alpha}(Q,\omega)\simeq V(Q)\cdot T(Q,\omega)\otimes R(Q,\omega) \tag{1.19}$$

where V, T, and R are vibrational, translational, and rotational terms, and $\otimes$ denotes a convolution in ω. Vibrational motion is typically expressed as a Debye-Waller factor,[45] whose functional form is given as:

$$V^{\alpha}(Q)=e^{-\langle\mu^2\rangle Q^2/3} \tag{1.20}$$

where $\langle\mu^2\rangle$ is the mean-squared vibrational displacement of a given atom.

There are a number of models that are used in the analysis of translational motion. This work will briefly discuss the models used in soft-matter systems – continuous

diffusion and jump-translational diffusion.[46] These models assume an isotropic system, in which there is no preferred direction of translational motion.

The simplest of the translational diffusion models is the continuous diffusion model, which assumes that a molecule undergoes Brownian motion, where particles move under the influence of the forces that arise from collisions between them. Between collisions, particles move along straight lines, and after a collision, a particle moves in a random direction that is independent of the previous one. The condition of "memory loss" assumes a small distance, relative to any relevant distance between two positions at two well-separated times, between the positions between collision N and collision N+1. Within the continuous diffusion model, the translational terms is expressed as:

$$T(Q,\omega) = \frac{D_T Q^2}{\omega^2 + \left(D_T Q^2\right)^2} \tag{1.21}$$

where D_T is the translational diffusion constant.

The jump-translational diffusion model[46] has been demonstrated[47] to be very useful for describing water dynamics. The jump-translational diffusion model is based on the hypothesis that an atom remains at a given location for a time interval τ_{jump}, vibrating about a center of equilibrium, and building a thermal cloud. After this time, the atom rapidly moves to another site, within a negligible amount of time. The length of the jump vector between these two sites, $|L|$, is assumed to be larger than the dimensions of the thermal cloud. Within the jump translation model,

$$T(Q,\omega) = \frac{\Gamma(Q)}{\omega^2 + \left[\Gamma(Q)\right]^2} \tag{1.22}$$

where the Lorentzian linewidth $\Gamma(Q)$ is given as:

$$\Gamma(Q) = \frac{D_{T,jump} Q^2}{1 + D_{T,jump} Q^2 \tau_{jump}} \tag{1.23}$$

and the translational diffusion constant, $D_{T,jump}$, is given as:

$$D_{T,jump} = \frac{\langle L \rangle^2}{6\tau_{jump}} \tag{1.24}$$

In the case of infinitesimally small L, the jump-translational diffusion model reduces to the continuous diffusion model.

To account for rotational diffusion, one typically assumes an isotropic rotational diffusion model.[48] In the isotropic rotational diffusion model, rotations are assumed to occur through small-angle, random, reorientations. Over a time-average, no preferred orientation exists. The functional form of the rotational contribution to the incoherent structure factor is given as:

$$R(Q,\omega) = A_0(Q)\delta(\omega) + \frac{1}{\pi}\sum_{l=1}^{\infty} A_l(Q)\left[\frac{\tau_l^{diff}}{1+\omega^2\left(\tau_l^{diff}\right)^2}\right] \tag{1.25}$$

where τ_l^{diff} is the rotational period, and

$$A_l(Q) = (2l+1)\, j_l^2(Qr) \tag{1.26}$$

and j_l is the l^{th} spherical Bessel function, r is the radius of gyration, which is the O-H bond length in the case of a water molecule. The rotational diffusion constant is related to the rotational relaxation rate τ_l^{diff} by:

$$\left[l(l+1)\tau_l^{diff}\right]^{-1} = D_R \tag{1.27}$$

The relatively simplistic isotropic rotational model is usually a starting point for the analysis of systems, even those that are most likely not isotropic.

Another model which has been applied to analyze rotational motion is the Ivanov model,[49] which can be thought of as the rotational equivalent of the jump translational diffusion model. In the Ivanov model, a molecule is assumed to be rotationally frozen for a characteristic time between jumps, τ_0, after which the molecule instantaneously rotates an angle θ. The time τ_0 is related to the forward rate constant of the hydrogen bond switching.[50] This angle jump behavior is visualized in Figure 1.3, with each frame being a snapshot of a switching event. The green sticks designate hydrogen bonds within the water trimer involved in this switching event, while blue sticks represent hydrogen bonds to molecules outside of this trimer. In frame A, oxygen atom O^a is five-coordinate because of collective fluctuations of the hydrogen bonding network, while O^b is three-coordinate. This situation of overcoordination of O^a and undercoordination of O^b leads to O^a motion away from O^*, and O^b motion towards O^*, to produce frame B, where O^b and O^a are equidistant from H^*, and the hydrogen bond can switch. At this point, O^b and O^a have the same coordination number. In frame C, the hydrogen bond has flipped; H^* is hydrogen bonded to O^b, leading to an overcoordination of O^b and an undercoordination of O^a, which is the reverse of the initial situation.

Within the Ivanov model, the rotational period τ_l^{jump} is given as:

$$\tau_l^{jump} = \frac{\tau_0}{\left[1 - \frac{1}{2l+1}\frac{\sin\left[(l+1/2)\theta\right]}{\sin(\theta/2)}\right]} \tag{1.28}$$

where τ_0 is a characteristic time. When the angular jump θ is very small, the rotational period τ_l^{jump} formulated in Eq. (1.28) reduces to:

$$\tau_l^{jump} = \frac{1}{l(l+1)D_R} \equiv \tau_l^{diff}, \tag{1.29}$$

which is identical to the rotational period described by the isotropic rotation model in Eq. (1.27).

An extended jump rotational diffusion model, which builds on the Ivanov model, has recently been proposed by Laage and Hynes.[51] In the extended jump model, an oxygen-oxygen reorientation time, τ_l^{OO}, is introduced. This τ_l^{OO} corresponds to the slow diffusional process that occurs between the hydrogen bond switching events depicted in Figure 1.3.

The functional form of the extended jump model is given as:

$$\frac{1}{\tau_l} = \frac{1}{\tau_l^{jump}} + \frac{1}{\tau_l^{OO}} \tag{1.30}$$

where τ_l^{jump} is the same as in the Ivanov model, defined in Eq. (1.28). In a recent article, Laage and Hynes show that their extended jump model has applications in the analysis of bulk water data. [51] They find that τ_l^{OO} is diffusive in character, while the τ_l^{jump} is not.

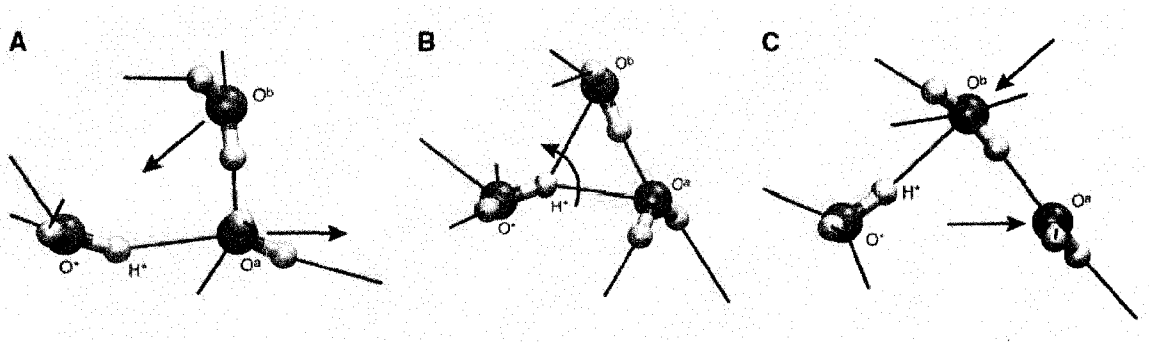


Figure 1.3. Jump-rotational behavior of water molecules.[51] Each frame corresponds to a snapshot of a hydrogen bond switching event. Arrows indicate translational or rotational motion.

1.III.B. Molecular Dynamics simulations

Molecular dynamics (MD) simulations provide the means of obtaining dynamical and structural information from molecular trajectory data. By comparing the experimental results with those of MD simulations, we can increase our level of understanding of the systems we study, and we can assess whether assumptions made in the analysis of experimental and simulation data are valid.

As experimental techniques use approximations and assumptions in their analyses, so do MD simulations in the generation of simulation data. Most common is the classical treatment of electrons, which requires that a molecule-specific, and possibly experiment- and system-specific, potential be developed to reproduce behavior observed either in experiment or in quantum chemistry calculation.

In a classical MD simulation, intermolecular interactions are usually expressed in a Lennard-Jones (LJ) 6-12 plus Coulomb form. An interaction between site α and β on two molecules separated by a distance r is then expressed as:

$$u_{\alpha,\beta}(r) = 4\left(\varepsilon_\alpha \varepsilon_\beta\right)^{1/2}\left[\left(\frac{\sigma_\alpha+\sigma_\beta}{2r}\right)^{12} - \left(\frac{\sigma_\alpha+\sigma_\beta}{2r}\right)^{6}\right] + \frac{q_\alpha q_\beta}{4\pi\varepsilon_0 r} \qquad (1.31)$$

where σ_α and ε_α are Lennard Jones diameter and well-depth, respectively, and q_α is the (partial) charge on site α.

Intermolecular potentials are usually developed by varying the LJ parameters, partial charges, and positions of the atoms and partial charge sites to reproduce experimental and/or quantum chemical data. As such, they can sometimes produce mixed results when used to calculate something other than what they were intended to do, and may also sacrifice structural accuracy for dynamical or electrostatic accuracy. Similarly,

two potentials that can reproduce their respective pure solvent data might not work in a binary mixture.

MD simulations are usually performed on the femtosecond to nanosecond timescale, so it is frequently assumed that intramolecular bonds are rigid because vibration takes place on a significantly shorter timescale than the MD timestep. Because of this, molecules in a MD simulation are typically frozen in a geometry determined by a quantum chemistry calculation, experimental result, or potential parameterization.

Although the potential parameters for the molecules in the simulation are important, so is the algorithm for determining how those molecules move. MD simulations simulate molecular dynamics, as the name would imply, which requires that each molecule in the system is moved at a velocity commensurate with the forces applied to it by other molecules. It is standard to divide molecular motion into two components – translation of the center of mass and rotation about the center of mass. Translational motion is simply determined from the vector sum of all forces acting on the center of mass. We discuss propagation of translational motion in terms of the Verlet,[52,53] velocity Verlet,[54] and leapfrog algorithms.[55] The rotational motion is given by the torque about the center of mass. We use two methods to enforce the bond rigidity in our simulations – quaternions and bond constraints.

The Verlet algorithm[52,53] is one of the most common algorithms used to propagate translational motion within molecular dynamics simulations. This algorithm involves two third-order Taylor expansions for the positions $\mathbf{r}(t)$, one forward and one backward in time, given as:

$$\mathbf{r}(t+\Delta t)=\mathbf{r}(t)+\mathbf{v}(t)\Delta t+\frac{1}{2}\mathbf{a}(t)\Delta t^2+\frac{1}{6}\mathbf{b}(t)\Delta t^3+... \tag{1.32}$$

$$\mathbf{r}(t-\Delta t)=\mathbf{r}(t)-\mathbf{v}(t)\Delta t+\frac{1}{2}\mathbf{a}(t)\Delta t^2-\frac{1}{6}\mathbf{b}(t)\Delta t^3+... \tag{1.33}$$

where $\mathbf{v}$, $\mathbf{a}$, and $\mathbf{b}$ are the velocity, acceleration, and the third time derivative of $\mathbf{r}(t)$, respectively. Adding the above equations produces:

$$\mathbf{r}(t+\Delta t)=2\mathbf{r}(t)-\mathbf{r}(t-\Delta t)+\mathbf{a}(t)\Delta t^2+O(\Delta t^4) \tag{1.34}$$

We are integrating Newton's equations, so $\mathbf{a}(t)$ is the force divided by the mass, and the force is a function of the positions $\mathbf{r}(t)$. We can rewrite $\mathbf{a}(t)$ as:

$$\mathbf{a}(t)=-(1/m)\nabla V(\mathbf{r}(t)) \tag{1.35}$$

When the system is evolved by Δt, errors produced by this algorithm are on the order $O(\Delta t^4)$. One problem with the Verlet algorithm is that velocities are not directly generated. Knowledge of the velocities is not necessary for this algorithm, but is necessary to compute the kinetic energy of the system. The velocity could be computed from the positions by the relation:

$$\mathbf{v}(t)=\frac{\mathbf{r}(t+\Delta t)-\mathbf{r}(t-\Delta t)}{2\Delta t}, \tag{1.36}$$

however, the errors will be on the order of Δt^2 rather than Δt^4. To overcome this difficulty, several variants of the Verlet algorithm have been developed.[54,55]

The leap-frog algorithm[55] attempts to overcome the errors associated with the determination of velocity by using a three-point central difference formula with time half-steps. The stored quantities are the current positions $\mathbf{r}(t)$ and accelerations $\mathbf{a}(t)$, as well as mid-step velocities $\mathbf{v}(t-\Delta t/2)$. Within this method, we define the position at time $t+\Delta t$ as:

$$\mathbf{r}(t+\Delta t)=\mathbf{r}(t)+\mathbf{v}(t+\Delta t/2)\Delta t \tag{1.37}$$

with the mid-step velocity $\mathbf{v}(t+\Delta t/2)$ given by:

$$\mathbf{v}(t+\Delta t/2)=\mathbf{v}(t-\Delta t/2)+\mathbf{a}(t)\Delta t \tag{1.38}$$

Equation (1.38) is applied first, and the velocities $\mathbf{v}(t-\Delta t/2)$ "leap" over the coordinates to produce the next mid-step velocities $\mathbf{v}(t+\Delta t/2)$. Current velocities can then be calculated by:

$$\mathbf{v}(t)=\frac{1}{2}\left[\mathbf{v}(t-\Delta t/2)+\mathbf{v}(t+\Delta t/2)\right] \tag{1.39}$$

Following this velocity leap, we propel the positions ahead of the velocities with Eq. (1.37). Since the leapfrog algorithm explicitly uses (mid-step) velocities in the calculations, we are allowed to perform velocity rescaling, as in constant temperature or constant energy simulations.

Although the leapfrog algorithm is an improvement over the Verlet algorithm, still does not handle velocities in a completely convenient way. The velocity Verlet algorithm[54] stores positions, accelerations, and velocities all at the same time *t*, rather than at mid-steps. First, the velocity Verlet algorithm advances position by:

$$\mathbf{r}(t+\Delta t)=\mathbf{r}(t)+\mathbf{v}(t)\Delta t+\frac{1}{2}\mathbf{a}(t)\Delta t^2 \tag{1.40}$$

The velocity is then calculated at a half-step $\Delta t/2$ to produce:

$$\mathbf{v}(t+\Delta t/2)=\mathbf{v}(t)+\frac{1}{2}\mathbf{a}(t)\Delta t \tag{1.41}$$

The forces and accelerations are then calculated at time $t+\Delta t$, so the velocity at a full timestep can then be calculated by:

$$\mathbf{v}(t+\Delta t)=\mathbf{v}(t+\Delta t/2)+\frac{1}{2}\mathbf{a}(t+\Delta t)\Delta t \tag{1.42}$$

The velocity Verlet algorithm is generally considered to be superior to the leapfrog and Verlet algorithms based on its simplicity, numerical stability, and explicit use of current velocities $\mathbf{v}(t)$.

Turning now to rotational motion, we assume intramolecular bond rigidity in our simulations of binary mixtures and use quaternions[56,57] to maintain this. Briefly, quaternions are a set of four coordinates that are used to identify the location and orientation of a given molecule. These coordinates are normalized and related to the Euler angles that are used to determine the orientation of a molecule relative to the laboratory axis by:

$$\begin{aligned}
q_{0,i} &= \sin(\theta_i/2)\sin\left[(\phi_i-\psi_i)/2\right] \\
q_{1,i} &= \sin(\theta_i/2)\cos\left[(\phi_i-\psi_i)/2\right] \\
q_{2,i} &= \sin(\theta_i/2)\sin\left[(\phi_i+\psi_i)/2\right] \\
q_{3,i} &= \sin(\theta_i/2)\cos\left[(\phi_i+\psi_i)/2\right] \\
q_{0,i}^2+q_{1,i}^2+q_{2,i}^2+q_{3,i}^2 &= 1
\end{aligned} \tag{1.43}$$

The rotation matrix $\mathbf{R}_i$, which relates unit vectors $\mathbf{e}^b$ and $\mathbf{e}^s$ for the body-fixed and space-fixed frames, is given by:

$$\mathbf{R}_i=\begin{pmatrix}
q_{0,i}^2+q_{1,i}^2-q_{2,i}^2-q_{3,i}^2 & 2\left(q_{1,i}q_{2,i}+q_{0,i}q_{3,i}\right) & 2\left(q_{1,i}q_{3,i}-q_{0,i}q_{2,i}\right) \\
2\left(q_{1,i}q_{2,i}-q_{0,i}q_{3,i}\right) & q_{0,i}^2-q_{1,i}^2+q_{2,i}^2-q_{3,i}^2 & 2\left(q_{2,i}q_{3,i}+q_{0,i}q_{1,i}\right) \\
2\left(q_{1,i}q_{3,i}+q_{0,i}q_{2,i}\right) & 2\left(q_{2,i}q_{3,i}-q_{0,i}q_{1,i}\right) & q_{0,i}^2-q_{1,i}^2-q_{2,i}^2+q_{3,i}^2
\end{pmatrix} \tag{1.44}$$

Therefore, the quaternions for each molecule satisfy the equation:

$$\begin{pmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} -q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{pmatrix} \begin{pmatrix} 0 \\ \omega_x^b \\ \omega_y^b \\ \omega_z^b \end{pmatrix} \tag{1.45}$$

where $\omega_x^b, \omega_x^b, \omega_x^b$ are components of the angular velocity defined relative to the body-fixed axis of the molecule in question. The side term of this Eq. (1.45) is designated is the first time derivative of the quaternion matrix $\mathbf{Q}(t)$. We use a leapfrog algorithm for integrating the rotational equation of motion,[56,57] which is based on storing angular momenta $\mathbf{I}^S(t)$, quaternions $\mathbf{Q}(t)$, and torques $\boldsymbol{\tau}^S(t)$ from positions and orientations at time *t*. First, we calculate the angular momenta from the previous half-step by:

$$\mathbf{I}^S(t) = \mathbf{I}^S\left(t - \frac{1}{2}\Delta t\right) + \frac{1}{2}\Delta t \boldsymbol{\tau}^S(t) \tag{1.46}$$

These quantities allow us to determine the body-fixed angular velocity at time *t*, which in turn allows calculation of the first time derivative of the quaternion matrix $\mathbf{Q}(t)$ through Eq. (1.45). A guess at $\mathbf{Q}\left(t + \frac{1}{2}\Delta t\right)$ is made by:

$$\mathbf{Q}\left(t + \frac{1}{2}\Delta t\right) = \mathbf{Q}(t) + \frac{1}{2}\Delta t \dot{\mathbf{Q}}(t) \tag{1.47}$$

Equations (1.46) and (1.47) are auxiliary equations; their purpose is to obtain an estimate of $\mathbf{Q}\left(t + \frac{1}{2}\Delta t\right)$. Knowing this, we can implement the main equations:

$$\mathbf{I}^S\left(t + \frac{1}{2}\Delta t\right) = \mathbf{I}^S\left(t - \frac{1}{2}\Delta t\right) + \Delta t \boldsymbol{\tau}^S(t) \tag{1.48}$$

and

$$\mathbf{Q}(t + \Delta t) = \mathbf{Q}(t) + \Delta t \dot{\mathbf{Q}}\left(t + \frac{1}{2}\Delta t\right) \tag{1.49}$$

where Eq. (1.48) is obtained after determining $\dot{\mathbf{Q}}\left(t+\frac{1}{2}\Delta t\right)$ from Eqs. (1.46) and (1.47). After the main equations are implemented, the torques and forces may then be evaluated and the entire process repeated for the next timestep.

For our simulations of water in RMs, we use the approach of constraint dynamics.[54,58] This approach fixes the bond lengths by employing a set of undetermined multipliers that represent the forces along the bonds. The constraint dynamics approach is to solve the equations of motion for one timestep without constraints, then determine the magnitudes of these multipliers and correct the atomic positions. Bond angles can be fixed in a similar manner by introducing fictitious bond length constraints.

To advance any of these algorithms, one must know the forces that act on the particles in the system. In simulations of many-molecule systems, it is computationally efficient to impose a cutoff radius (R_{cut}) for LJ interactions, as these are relatively short-range, and the calculation of forces accounts for most of the time involved in performing a simulation. Care must be taken in determining the cutoff radius; too small an R_{cut} makes the computation meaningless, as LJ interactions that should contribute to the dynamics are being excluded from the calculation, while too large an R_{cut} includes interactions that do not contribute to the dynamics of the system, and is thus computationally inefficient. The reverse micelle simulations presented in Chapter 3 of this work do not employ any interaction cutoffs, as these simulations involve small, individual RMs with characteristic interfaces. In a system with periodic boundaries, such as that employed in our water-DMSO simulations, we can express the electrostatic interactions in a Ewald sum. If we designate our simulation box as having coordinates $(0,0,0)$, and repeat cells as

(n_x, n_y, n_z), we can group the repeat cells in approximately spherical shells around the origin cell according to $n = (n_x^2 + n_x^2 + n_x^2)^{\frac{1}{2}}$. We then have the case of a continuum with a relative permittivity ε_S, where the subscript denotes the spherical character. In the case of a "good" conductor, $\varepsilon_S \to \infty$, whereas in a vacuum, we have $\varepsilon_S \to 1$. Within this, the Coulombic interaction between sites z is given as:

$$V^{zz}(\varepsilon_S = \infty) = V^{zz}(\varepsilon_S = 1) - \frac{2\pi}{3L}\left|\sum_i z_i \mathbf{r}_i\right|^2 \tag{1.50}$$

where L is the length of the simulation box side, z_i and $\mathbf{r}_i$ are the charge and position of the ith charge site, respectively. During a simulation, the (central) simulation box is a unit cell in a neutral lattice. According to the Ewald method, each point charge is surrounded by a distribution of charges which spreads out radially from and is equal in charge and opposite in magnitude from that point charge. This distribution is assumed to be Gaussian in nature, given by:

$$\rho_i^z(r) = \frac{z_i \kappa^3 \exp(-\kappa^2 r^2)}{\pi^{3/2}} \tag{1.51}$$

where an arbitrary parameter κ determines the width of the distribution. The final potential energy for a vacuum is given by:

$$V^{zz}(\varepsilon_s = 1) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\sum_{|\mathbf{n}|=0}^{\infty}{}' z_i z_j \frac{erfc(\kappa|\mathbf{r}_{ij}+\mathbf{n}|)}{|\mathbf{r}_{ij}+\mathbf{n}|} + \left(\frac{1}{\pi L^3}\right)\sum_{\mathbf{k}\neq 0} z_i z_j \left(\frac{4\pi^2}{k^2}\right)\exp\left(-\frac{k^2}{4\kappa^2}\right)\cos(\boldsymbol{\kappa}\cdot\mathbf{r}_{ij})\right)$$
$$-\left(\frac{\kappa}{\pi^{\frac{1}{2}}}\right)\sum_{i=1}^{N} z_i^2 + \left(\frac{2\pi}{3L^2}\right)\left|\sum_{i=1}^{N} z_i \mathbf{r}_i^2\right|^2 \tag{1.52}$$

where *erfc*(x) is given by

$$erfc(\mathrm{x}) = \left(\frac{2}{\pi^{\frac{1}{2}}}\right)\int_{x}^{\infty}\exp\left(-t^2\right)dt \tag{1.53}$$

If κ is determined to be large enough, the first term only contributes when **n**=0, which corresponds to the (0, 0, 0) origin cell. The second term in Eq. (1.52) is a sum over reciprocal vectors **k**, where

$$\mathbf{k} = \frac{2\pi\mathbf{n}}{L^2}. \tag{1.54}$$

κ is set to a value that leads to a truncation at $\mathbf{n} = 0$ but still yields a reasonable number of k vectors. In practice, this value of κ corresponds to $5/L$ and yields 10^2 wavevectors for the k-space sum. Our simulations of water-DMSO mixtures assume a conducting boundary condition, $\varepsilon_S \to \infty$, so by substituting Eq. (1.52) into Eq. (1.50), we obtain:

$$V^{zz}\left(\varepsilon_s = \infty\right) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\sum_{|\mathbf{n}|=0}^{\infty}{}' z_i z_j \frac{erfc\left(\kappa\left|\mathbf{r}_{ij}+\mathbf{n}\right|\right)}{\left|\mathbf{r}_{ij}+\mathbf{n}\right|} + \left(\frac{1}{\pi L^3}\right)\sum_{\mathbf{k}\neq 0} z_i z_j \left(\frac{4\pi^2}{k^2}\right)\exp\left(-\frac{k^2}{4\kappa^2}\right)\cos\left(\boldsymbol{\kappa}\bullet\mathbf{r}_{ij}\right)\right) - \left(\frac{\kappa}{\pi^{\frac{1}{2}}}\right)\sum_{i=1}^{N} z_i^2 \tag{1.55}$$

MD simulations used in this dissertation are performed at constant energy. The temperature is maintained by using a Berendsen thermostat[59] with a long time constant. This thermostat attempts to maintain constant temperature by correcting deviations of

temperature T from a set-point T_0 at intervals by rescaling the velocities of molecules within the system by a scaling factor λ, given by:

$$\lambda = \left[1 + \frac{\Delta t}{\tau_T}\left(\frac{T}{T_0} - 1\right)\right]^{\frac{1}{2}} \tag{1.56}$$

where τ_T is the "rise time" or time constant of the thermostat. This time constant τ_T describes the strength of the coupling between the system and a hypothetical heat bath. Larger τ_T means weaker coupling, and a longer time to achieve a given T_0 after an instantaneous temperature change from a previous T_0.

Most of the MD simulations discussed in this work focus on the evaluation of single-particle time-correlation functions (TCF's). Determination of the error within a TCF produced by a simulation is slightly less straightforward than it is in an experiment. Given a time-dependent property, $A(t)$, we can denote the run average of the TCF, $C_{AA}^{run}(t)$, by:

$$C_{AA}^{run}(t) = \langle A(t)A(0)\rangle_{run} = \frac{1}{t_{run}}\int_0^{t_{run}} dt' A(t')A(t'+1). \tag{1.57}$$

The error in $C(t)$ is given by:

$$\begin{aligned} \delta C(t) &= C_{AA}^{run}(t) - C_{AA}(t) \\ &= \langle A(t)A(0)\rangle_{run} - \langle A(t)A(0)\rangle \\ &= \frac{1}{(t_{run}-t)}\int_0^{t_{run}} dt' \left(A(t')A(t'+t) - \langle A(t')A(t'+t)\rangle\right) \end{aligned} \quad , \tag{1.58}$$

where $\langle ... \rangle$ is the ensemble average. Since we are dealing with finite run length, the error in $C(t)$ increases with t. The variance of the mean is given by:

$$\sigma^2\left(\left\langle A(t)A\right\rangle_{run}\right)=\frac{1}{\left(t_{run}-t\right)^2}\int_0^{t_{run}}\int_0^{t_{run}}dt'dt''\left(\left\langle A(t')A(t'+t)A(t'')A(t''+t)\right\rangle-\left\langle A(t)A(0)\right\rangle^2\right) \quad (1.59)$$

Assuming Gaussian statistics, Eq. (1.59) reduces to:

$$\sigma^2\left(\left\langle A(t)A\right\rangle_{run}\right)\approx 2t_A\left[C_{AA}(0)\right]^2/\left(t_{run}-t\right) \quad (1.60)$$

where t_A is a correlation time defined by

$$t_A=2\int_0^{\infty}dt\left\langle\delta A(t)\delta A\right\rangle/\left\langle\delta A^2\right\rangle^2 \quad (1.61)$$

For a TCF that decays exponentially,

$$t_A=\lim_{t\to\infty}t\frac{\sigma^2\left(\left\langle A\right\rangle_t\right)}{\sigma^2(A)}. \quad (1.62)$$

The standard error in the (normalized) TCF is then given by

$$\sigma\left(\left\langle A(t)A\right\rangle_{run}\right)/\left\langle A^2\right\rangle\approx\left(t_A/\left(t_{run}-t\right)\right)^{1/2} \quad (1.63)$$

From this, the error at $t=0$ is zero, but one would need to average 10^5 time steps to obtain a relative precision ~ 1% for $C_{AA}(t)$ if the correlation time t_A is 10^1 time steps. When looking at single-particle time correlation functions, such as the bond and dipole vector time correlation functions and intermediate scattering functions discussed in Chapters 2 and 3, we have N single-particle correlations, and

$$C_{AA}(t)=\frac{1}{N}\sum_{i=1}^{N}\left\langle A_i(t)A_i(0)\right\rangle \quad (1.64)$$

If we assume that the functions are statistically independent, in other words, a single-particle TCF for a given particle is not correlated with that of another, the error at long times is then $\approx\left(2t_A/Nt_{run}\right)^{1/2}$, reflecting an increase in precision as the system size

increases. Therefore, single-particle TCF's are generally found to be less noisy than collective TCF's.[58]

1.IV. Instrumentation

We performed QNS experiments with two instruments – QENS and DCS. The QENS instrument is located in the Intense Pulsed Neutron Source (IPNS) division of Argonne National Laboratory (ANL), in Argonne, Illinois. DCS, an abbreviation for disk-chopper spectrometer, is located in the NIST Center for Neutron Research (NCNR) in Gaithersburg, Maryland.

The DCS is a direct geometry spectrometer (Fig 1.4), in which a set of choppers is used to generate a monochromatic beam of incident neutrons. By adjusting the relative timing of the choppers, we can select a given wavelength of neutrons from a wavelength range of 2 to 10 Å, as well as an energy resolution between 10000 to 5 μeV . For a given neutron wavelength, the instrument can operate in low, medium, or high energy-resolution mode by adjusting the timing of the choppers. DCS is a direct geometry instrument, in which the energy gain/loss of a neutron upon a scattering event is determined by knowing the initial energy and the total time of flight of that neutron.

The QENS instrument uses the inverse geometry or crystal analyzer geometry, given in Fig. 1.5, to detect scattered neutrons. In this configuration, a so-called "white-beam" of incident neutrons scatters off a sample. Scattered neutrons strike an array of crystal analyzers, which will then reflect neutrons of a specific final energy to the

detector bank. By knowing the final energy of the neutrons, as well as the angle of scatter and the total time of flight, one can determine the initial energy and therefore the change in energy of the scattered neutron. The QENS instrument has a Q-range of $0.3 \leq Q \leq 2.8 \text{Å}^{-1}$, and an energy resolution of ~90 μeV. This instrument has a relatively low energy resolution, which makes it most useful for measuring fast processes such as rotational diffusion. However, at higher Q, both translational and rotational motions will contribute to the observed signal.

Because QNS spectra contain convolutions of translational and rotational processes, it is often desirable to fit low-Q data, where the rotational contribution to the scatter is negligible, to a translational model, and then use the derived translational parameters for higher-Q data. DCS can operate at a higher resolution and lower Q than QENS, and is capable of resolving slow translational motions within a momentum transfer regime where rotational contributions are minimal. QENS operates at a low energy resolution and high-Q, and so is less useful for measuring translational motions, but can more effectively measure rotational motion than can DCS.

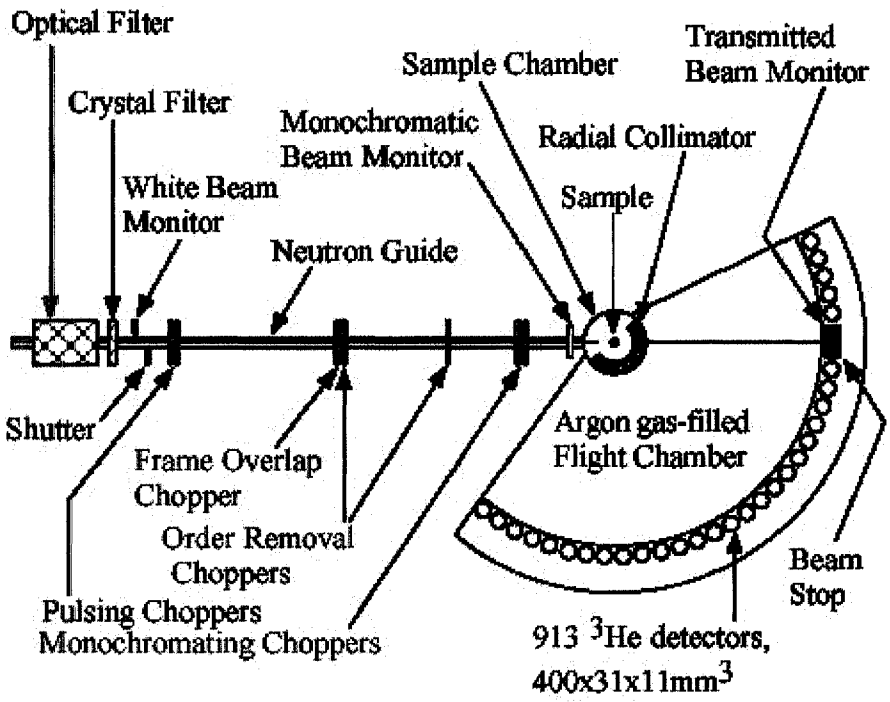


Figure 1.4. Depiction of the disk chopper spectrometer (DCS). [60]

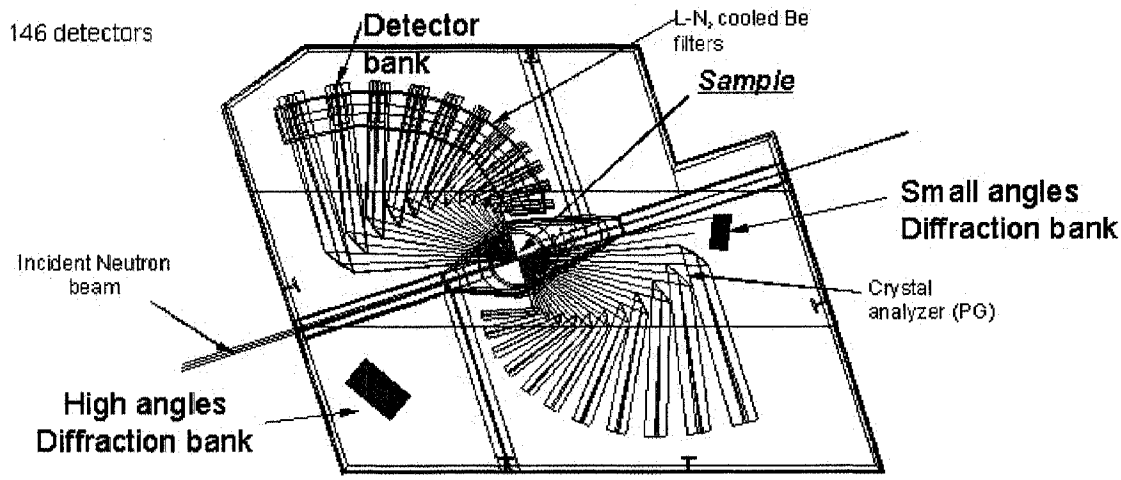


Figure 1.5. Depiction of the QENS spectrometer.[61]

References and Notes

(1) Calandra, P.; Longo, A.; Liveri, V. T. *J. Phys. Chem. B* **2003**, *107*, 25.

(2) Cason, J. P.; Miller, M. E.; Thompson, J. B.; Roberts, C. B. *J. Phys. Chem. B* **2001**, *105*, 2297.

(3) Pavel, F. M.; Mackay, R. A. *Langmuir* **2000**, *16*, 8568.

(4) Pileni, M. P. *J. Phys. Chem.* **1993**, *97*, 6961.

(5) Pileni, M. P. *Langmuir* **1997**, *13*, 3266.

(6) *Reverse Micelles: Biological and Technological Relevance of Amphiphilic Structures in Apolar Media*; Luisi, P. L.; Straub, B. E., Eds.; Plenum Press: New York, 1984.

(7) Amararene, A.; Gindre, M.; LeHuerou, J. Y.; Nicot, C.; Urbach, W.; Waks, M. *J. Phys. Chem. B* **1997**, *101*, 10751.

(8) Boicelli, C. A.; Giomini, M.; Giuliani, A. M. *Appl. Spec.* **1984**, *38*, 537.

(9) Brubach, J. B.; Mermet, A.; Filabozzi, A.; Gerschel, A.; Lairez, D.; Krafft, M. P.; Roy, P. *J. Phys. Chem. B* **2001**, *105*, 430.

(10) D'Angelo, M.; Fioretto, D.; Onori, G.; Palmieri, L.; Santucci, A. *Phys. Rev. E* **1996**, *54*, 993.

(11) Venables, D. S.; Huang, K.; Schmuttenmaer, C. A. *J. Phys. Chem. B* **2001**, *105*, 9132.

(12) Loughnane, B. J.; Farrer, R. A.; Scodinu, A.; Fourkas, J. T. *J. Chem. Phys.* **1999**, *111*, 5116.

(13) Venables, D. S.; Schmuttenmaer, C. A. *J. Chem. Phys.* **2000**, *113*, 11222.

(14) Aminabhavi, T. M.; Gopalakrishna, B. *J. Chem. Eng. Data* **1995**, *40*, 856.

(15) Catalan, J.; Diaz, C.; Garcia-Blanco, F. *Organic & Biomolecular Chemistry* **2003**, *1*, 575.

(16) Gaeva, E. B.; Pimienta, V.; Metelitsa, A. V.; Voloshin, N. A.; Minkin, V. I.; Micheau, J. C. *J. Phys. Org. Chem.* **2005**, *18*, 315.

(17) Bako, I.; Megyes, T.; Palinkas, G. *Chem. Phys.* **2005**, *316*, 235.

(18) Ladanyi, B. M.; Skaf, M. S. *J. Phys. Chem.* **1996**, *100*, 1368.

(19) Skaf, M. S.; Fonseca, T.; Ladanyi, B. M. *J. Chem. Phys.* **1993**, *98*, 8929.

(20) Puxty, G.; Maeder, M.; Radack, K. P.; Gemperline, P. J. *Applied Spectroscopy* **2005**, *59*, 329.

(21) Crupi, V.; Majolino, D.; Migliardo, P.; Venuti, V. *J. Phys. Chem. A* **2000**, *104*, 11000.

(22) *Dimethyl Sulfoxide*; Marcel Dekker: New York, 1971.

(23) Soper, A. K.; Luzar, A. *J. Chem. Phys.* **1992**, *97*, 1320.

(24) Soper, A. K.; Luzar, A. *J. Phys. Chem.* **1996**, *100*, 1357.

(25) Vechi, S. M.; Skaf, M. S. *J. Chem. Phys.* **2005**, *123*.

(26) Benjamin, I. *J. Chem. Phys.* **1999**, *110*, 8070.

(27) Cabral, J. T.; Luzar, A.; Teixeira, J.; Bellissent-Funel, M. C. *J. Chem. Phys.* **2000**, *113*, 8736.

(28) Cabral, J. T.; Luzar, A.; Teixeira, J.; Bellissent-Funel, M. C. *Phys. B* **2000**, *276*, 508.

(29) Chalaris, M.; Samios, J. *J. Mol. Liq.* **2002**, *98-9*, 399.
(30) Gordalla, B. C.; Zeidler, M. D. *Mol. Phys.* **1986**, *59*, 817.
(31) Gordalla, B. C.; Zeidler, M. D. *Mol. Phys.* **1991**, *74*, 975.
(32) Kirchner, B.; Searles, D. J.; Dyson, A. J.; Vogt, P. S.; Huber, H. *J. Am. Chem. Soc.* **2000**, *122*, 5379.
(33) Laria, D.; Skaf, M. S. *J. Chem. Phys.* **1999**, *111*, 300.
(34) Lei, Y.; Li, H. R.; Han, S. J. *Chem. Phys. Lett.* **2003**, *380*, 542.
(35) Luzar, A. *J. Mol. Liq.* **1990**, *46*, 221.
(36) Luzar, A.; Chandler, D. *J. Chem. Phys.* **1993**, *98*, 8160.
(37) Luzar, A.; Soper, A. K.; Chandler, D. *J. Chem. Phys.* **1993**, *99*, 6836.
(38) Kobara, H.; Wakisaka, A.; Takeuchi, K.; Ibusuki, T. **2003**, *107*, 11827.
(39) Anastopoulos, A. G.; Latsinoglou, T. *J. Sol. Chem.* **2005**, *34*, 1341.
(40) Hernandez-Perni, G.; Leuenberger, H. *Eur. J. Pharm. Biopharm.* **2005**, *61*, 201.
(41) Bordallo, H. N.; Herwig, K. W.; Luther, B. M.; Levinger, N. E. *J. Chem. Phys.* **2004**, *121*, 12457.
(42) Martins, L. R.; Tamashiro, A.; Laria, D.; Skaf, M. S. *J. Chem. Phys.* **2003**, *118*, 5955.
(43) Frank, H. S.; Evans, M. W. *J. Chem. Phys.* **1945**, *13*, 507.
(44) Borin, I. A.; Skaf, M. S. *J. Chem. Phys.* **1999**, *110*, 6412.
(45) Bee, M. *Quasielastic Neutron Scattering*; Hilger: Bristol, 1988.
(46) Egelstaff, P. A. *An Introduction to the Liquid State*; Academic: London, 1967.
(47) Di Cola, D.; A., D.; Sampoli, M.; Torcini, A. *J. Chem. Phys.* **1996**, *104*, 4223.
(48) Sears, V. F. *Can. J. Phys.* **1967**, *45*, 237.
(49) Ivanov, E. N. *Soviet Physics JETP* **1964**, *18*, 1041.
(50) Luzar, A.; Chandler, D. *Nature* **1996**, *379*, 55.
(51) Laage, D.; Hynes, J. T. *Science* **2006**, *311*, 832.
(52) Schiff, D.; Verlet, L. *Phys. Rev.* **1967**, *160*, 208.
(53) Verlet, L. *Phys. Rev.* **1967**, *159*, 98.
(54) Ciccotti, G.; Ryckaert, J.-P. *Comput. Phys. Rep.* **1986**, *4*, 345.
(55) Hockney, R. W. *Methods Comput. Phys.* **1970**, *9*, 136.
(56) Evans, D. J. *Mol. Phys.* **1977**, *34*, 317.
(57) Evans, D. J.; Murad, S. *Mol. Phys.* **1977**, *34*, 327.
(58) Allen, M. P.; Tildesley, D. J. *Computer Simulations of Liquids*.Clarendon Press: Oxford,U.K., 1987; Vol. 1987.
(59) Berendsen, H. J. C.; Postma, J. P. M.; van Gunsteren, W. F.; DiNola, A.; Haak, J. R. *J. Chem. Phys.* **1984**, *81*, 3684.
(60) http://www.ncnr.nist.gov/instruments/dcs/PosterNov2002.pdf; NIST Center for Neutron Research, 2002.
(61) deSouza, N. http://www.pns.anl.gov/instruments/qens/, 2006.

Chapter 2

Dynamics of Water in Binary Mixtures with DMSO

2.I. Introduction

Mixtures of dimethyl sulfoxide (DMSO) and water are of considerable importance as chemical reaction media, and in biomedical applications,[1-4] which exploit the cryoprotection properties of these solutions.[3,4] Given that interspecies hydrogen (H) bonds are formed between water hydrogen atoms and the oxygen atoms of DMSO, these mixtures are strongly associating and exhibit nonideal behavior, especially in their transport properties.[3-11] Maximum deviations from ideality are reported to occur between 0.3 to 0.4 mole fraction DMSO (X_{DMSO}). For example, the mixture has a higher viscosity than the pure components, with the peak occurring around X_{DMSO}=0.33.[5,6]

These strong deviations from ideality have been hypothesized to arise from the formation of 2:1 water to DMSO complexes.[8,11,12] A neutron diffraction study of water-DMSO mixtures [13,14] found that water-water H-bonding in mixtures was reduced relative to bulk water and that a large proportion of the water hydrogen atoms within the mixture were associated with the lone pairs on the DMSO oxygen. These data also suggest the presence of a high concentration of 1:2 DMSO-water complexes at the composition

$X_{DMSO} = 0.33$.[14] The neutron diffraction data show that there exists local tetrahedral order of water molecules at this mole fraction.

A number of molecular dynamics (MD) simulations aimed at elucidating several aspects of structure and dynamics of water-DMSO mixtures have been performed.[15-24] Simulations performed by Luzar and Chandler[16] show that the water-DMSO H-bonds live longer than water-water H-bonds, that DMSO is usually a double H-bond acceptor that forms 2:1 aggregates of DMSO and water with almost tetrahedral symmetry. An MD simulation study by Borin and Skaf[19] explored the local structures and H-bond distributions over the full composition range of water-DMSO mixtures. Based on their analysis of site-site pair distribution functions, they find that the water-DMSO hydrogen bonding is stronger than water-water hydrogen bonding. Furthermore, by adding DMSO to water, an H-bond accepting water molecule is replaced by a DMSO molecule, which preserves the local tetrahedral order. They found that the 1:2 complexes of DMSO and water predominate over 2:1 complexes for water-rich mixtures ($X_{DMSO} < 0.5$), and the opposite is true for water-poor mixtures ($X_{DMSO} > 0.5$). They explored the self-diffusion coefficients and single-particle rotational relaxation times of water and DMSO as functions of composition, and found that the water dynamics are slowest at the $X_{DMSO} = 0.5$ composition.

Single-molecule dynamics of water and DMSO molecules in the mixtures have been studied by several spectroscopic methods, including Raman[9] and NMR[10,11,25] spectroscopy. A method that provides information about translational and rotational mobility of molecules in the liquid phase is quasielastic neutron scattering (QNS). QNS experiments measure the incoherent structure factor, $S(\mathbf{Q},\omega)$, as a function of

momentum transfer **Q** and energy transfer ω. Since the ^1H isotope of hydrogen has a much larger incoherent cross section than D, selective deuteration can be used to obtain information on the dynamics on each of the mixture components. QNS has recently been applied to water-DMSO mixtures, first as a function of temperature at the eutectic composition[12,26] (2:1 water-DMSO mole ratio) and then as a function of composition at room temperature.[27] Specifically, in the latter study, Bordallo et al. reported[27] results from QNS experiments on water/DMSO mixtures over a range of X_{DMSO} from 0.1 to 0.8. In both studies, information of the rotational and translational dynamics was extracted from experimental data using the jump-translational diffusion and isotropic rotational diffusion models, which are well-established in their application to analysis of QNS data from liquid samples.[28]

In the case of *d*6-DMSO/H_2O mixtures, Bordallo et al.'s experiments indicated that the translational diffusion constant of water reached a minimum of 0.3×10^{-5} cm^2/s at $X_{DMSO} = 0.35$. This is about one order of magnitude smaller than the diffusion coefficient of bulk water[29] The rotational diffusion coefficient did not exhibit as much variation within this range, which seems to conflict with the MD simulation data[19] as well as NMR data.[10] Estimates of the rotational diffusion coefficient of water obtained by NMR[10] show a similar trend to the QNS data, but a larger variation with X_{DMSO}. That is, both techniques indicate that $X_{DMSO} = 0.33$ shows the slowest rotational motion, however, the rotational relaxation period varies over a larger range in the NMR experiments than it does in the QNS experiments. The rotational relaxation period of water in the $X_{DMSO} = 0.33$, when measured by NMR[10], is approximately 8 ps, compared to 2 ps measured by QNS.[27]

In the work reported here, we perform a series of MD simulations on water-DMSO mixtures to gain information about structural and dynamical properties at a variety of compositions, with the aim of better understanding the QNS experiments. We focus on the systems in which DMSO is deuterated and investigate the single-molecule dynamics of water. The primary quantity that we evaluate is the self-intermediate scattering function, which is related to the incoherent structure factor measured in QNS experiments. Specifically, $S(\mathbf{Q},\omega)$ is the frequency Fourier transform of the self-intermediate scattering function, $F_S(\mathbf{Q},t)$.

$$S(\mathbf{Q},\omega)=\frac{1}{2\pi}\int_{-\infty}^{\infty}F_S(\mathbf{Q},t)e^{i\omega t}dt. \tag{2.1}$$

where

$$F_S(\mathbf{Q},t)=\frac{1}{N}\sum_{j=1}^{N}\left\langle \exp\left\{i\mathbf{Q}\cdot\left[\mathbf{r}_j(0)-\mathbf{r}_j(t)\right]\right\}\right\rangle. \tag{2.2}$$

N is the number of water hydrogen atoms and $\mathbf{r}_j(t)$ is the position of the j^{th} hydrogen atom. MD simulation provides a means of obtaining $F_S(\mathbf{Q},t)$ from molecular trajectory data. By evaluating the contributions to these self-intermediate scattering functions from translational and rotational dynamics, we can determine how these dynamics contribute to QNS and assess the range of validity of the dynamical models used in the analysis of experimental QNS data.

In Section 2.II of this chapter, we present the model we use in our MD simulations, specify the time-correlation functions evaluated, and introduce the models that have been used in the interpretation of QNS data from water-DMSO mixtures. Our results and discussion are presented in Section 2.III. The MD results from $F_S(\mathbf{Q},t)$ are

discussed in terms of rotational and translational relaxations of water molecules within the mixture, and comparisons between our data and the experimental QNS data obtained by Bordallo et al. are presented.[27] The chapter is concluded in Section 2.IV.

2.II. Molecular Dynamics Simulations

2.II.A. Interaction Model

We have considered water/*d6*-DMSO mixtures of X_{DMSO} = 0.124, 0.25, 0.33, 0.50, 0.666, and 0.75 at ambient temperatures. For water, we use the SPC/E model,[30] and for DMSO, the four-site P2 potential developed Luzar and Chandler.[16] Both SPC/E and P2 models are rigid and nonpolarizable and in DMSO the methyl group is a single site.

Intermolecular interactions are described in terms of a Lennard-Jones plus Coulomb form, with the interactions between sites α and β on different molecules given as:

$$u_{\alpha\beta}(r) = 4\left(\varepsilon_\alpha \varepsilon_\beta\right)^{1/2}\left[\left(\frac{\sigma_\alpha + \sigma_\beta}{2r}\right)^{12} - \left(\frac{\sigma_\alpha + \sigma_\beta}{2r}\right)^{6}\right] + \frac{q_\alpha q_\beta}{4\pi\varepsilon_0 r} \qquad (2.3)$$

Standard Lorentz-Berthelot combining rules are used for sites of different types. Lennard-Jones parameters and charges are summarized in Table 2.1. Figure 2.1 displays the bond angles and bond lengths for the P2 DMSO (2.1a) and SPC/E water (2.1b) molecules.

The simulations were performed on constant number, volume, energy (NVE) ensembles of 500 molecules in a cubic box with periodic boundaries at an average temperature of 298 K. Box lengths were chosen to match the experimental densities[7] at room temperature and pressure. Compositions and molar volumes are listed in Table 2.2.

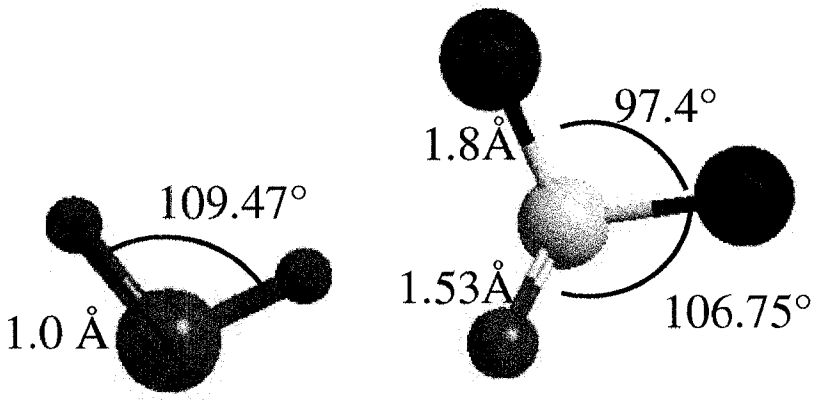


Figure 2.1. SPC/E water (left) and P2-DMSO (right) models with structural parameters superimposed. Oxygen, carbon, hydrogen, and methyl groups are given in red, yellow, light blue, and dark blue, respectively.

Table 2.1: Lennard Jones and Coulomb potential parameters

		ε/K	σ/Å^{-1}	q/e
DMSO	Oxygen	35.988	2.8	-0.459
	Sulfur	119.96	3.4	0.139
	Methyl	147.93	3.8	0.16
Water	Oxygen	78.177	3.166	-0.8476
	Hydrogen	------	------	0.4238

Lennard-Jones interactions were cut off at half the box length, while Ewald sums with conducting boundary conditions[31] were used for Coulomb forces. Integration of equations of motion was accomplished using the velocity-leapfrog algorithm for translations[32] and the quaternion algorithm for rotations[33,34]. Production trajectories were 0.5 to 1.5 ns, based on water content and expected mobility, with longer trajectories for lower water content and lower mobility compositions. Before the production trajectories, mixtures were equilibrated starting from a face-centered cubic structure, with molecules of different species randomly placed at lattice sites, with random orientations, at ~15% higher molar volume, decreasing in steps to the experimental molar volume[5] over a period of 100 ps, followed by ~100 ps of further simulation. In some cases, such as X_{DMSO} = 0.50, additional heating and cooling cycles were necessary to successfully equilibrate the mixture.

Table 2.2: Composition and Molar Volumes of Water-DMSO mixtures

X_{DMSO}	n_{Water}	n_{DMSO}	Molar Volume (10^{-5} m^3/mol)
0.124	438	62	2.52
0.25	375	125	3.12
0.33	335	165	3.5
0.5	250	250	4.37
0.66	165	335	5.37
0.75	125	375	5.81

2.II.B. Evaluation of the QNS Intermediate Scattering Function

The intermediate scattering function $F_S(\mathbf{Q},t)$ can be obtained from MD simulation through Eq. (2.2). We focus our calculations on the dynamics of water molecules and the Q-range of the experiment on H_2O/d6-DMSO mixtures performed by Bordallo et al.[27] The timescale of a QNS experiment is determined by the resolution of the instrument. Because experimentally accessible timescales are typically on the order of tens of picoseconds, we calculate time correlations over similar time intervals. The QNS experiments of Bordallo et al. were performed on the QENS instrument at Argonne National Laboratory, which has an instrument resolution of 90 μeV, corresponding to a temporal window of ~50 ps, and a Q-range from $0.38\,\text{Å}^{-1} < Q < 2.63\,\text{Å}^{-1}$. In our simulations, we use a Q range from $0.32\,\text{Å}^{-1} < Q < 2.93\,\text{Å}^{-1}$.

In our simulations we use a rigid model for water. We can thus predict the contributions to QNS from rotational and translational, but not from vibrational, motions of water molecules. This is not a serious drawback, given that vibrations in water occur at high frequencies that exceed the limits of detection in QNS.[28] As a result, they contribute to the water QNS spectrum $S(\mathbf{Q},\omega)$ only through a frequency-independent multiplicative Debye-Waller factor.[28]

In our simulations, the location of a hydrogen atom in a water molecule can be expressed as

$$\mathbf{r} = \mathbf{r}_{CM} + \mathbf{d}, \tag{2.4}$$

where $\mathbf{d}$ is the distance of the hydrogen atom from the molecular center-of-mass, $\mathbf{r}_{CM}$. In the rigid SPC/E water model, the molecular geometry is fixed and therefore $\mathbf{d}$ changes

only through molecular rotation. The translational component of the water motion is tracked through $\mathbf{r}_{CM}$.

The total intermediate scattering function (ISF) for a hydrogen atom in a water molecule is then given by

$$F_S(\mathbf{Q},t) = \left\langle \exp\left\{i\mathbf{Q}\cdot[\mathbf{r}_{CM}(0) - \mathbf{r}_{CM}(t)]\right\} \exp\left\{i\mathbf{Q}\cdot[\mathbf{d}(0) - \mathbf{d}(t)]\right\} \right\rangle. \tag{2.5}$$

In real systems, molecular rotational and translational motions are coupled. However, QNS experiments are frequently analyzed by neglecting rotation-translation coupling and Eq. (2.5) would be factored into separate contributions from rotation and translation. To test this approximation for the water-DMSO mixtures and to make contact with experiments of Bordallo et al.[27], we compare the MD results for $F_S(\mathbf{Q},t)$ in the presence and in the absence of coupling. Assuming independence of rotational and translational dynamics in Eq. (2.5) results in the product approximation:

$$F_S(\mathbf{Q},t) \cong F_P(\mathbf{Q},t) = F_S^{CM}(\mathbf{Q},t) F_S^R(\mathbf{Q},t), \tag{2.6}$$

where the translational and rotational ISFs are given by

$$F_S^{CM}(\mathbf{Q},t) = \left\langle \exp\left\{i\mathbf{Q}\cdot[\mathbf{r}_{CM}(0) - \mathbf{r}_{CM}(t)]\right\} \right\rangle \tag{2.7}$$

and

$$F_S^R(\mathbf{Q},t) = \left\langle \exp\left\{i\mathbf{Q}\cdot\left[\mathbf{d}(0) - \mathbf{d}(t)\right]\right\} \right\rangle. \tag{2.8}$$

While the rotational ISF, $F_S^R(\mathbf{Q},t)$, can be evaluated from MD trajectory data directly using Eq.(2.8), one can also express it in terms of the length d of the vector $\mathbf{d}$ and its orientation, $\hat{\mathbf{d}} = \mathbf{d}/d$, to compare with models for rotational relaxation. This is done by using the Rayleigh expansion of the exponential[35]

$$F_S^R(\mathbf{Q},t) = \left[j_0(Qd)\right]^2 + \sum_{l=1}^{\infty}(2l+1)\left[j_l(Qd)\right]^2 C_l(t), \tag{2.9}$$

where j_l is the spherical Bessel function of order l and $C_l(t)$ is the time correlation for the Legendre polynomial P_l of order l of the change in direction of unit vector $\hat{\mathbf{d}}$:

$$C_l(t) = \left\langle P_l\left[\hat{\mathbf{d}}(0)\cdot\hat{\mathbf{d}}(t)\right]\right\rangle. \tag{2.10}$$

To extract information on rotational and translational dynamics from QNS data, it is convenient to use dynamical models that relate the results to familiar parameters such as the rotational and translational diffusion coefficients. Given that molecular motion in liquids is diffusive on intermediate and long time scales, a dynamical model appropriate for diffusive relaxation is usually applied to liquid-state QNS data. For bulk liquid water, a combination of jump diffusion model for translation and isotropic diffusion model for rotation fits experimental data reasonably well.[28] In the present case, Bordallo et al.[27] used this dynamical model in their data analysis. According to this model $F_S^{CM}(Q,t)$ and $C_l(t)$ decay exponentially, specifically

$$F_S^{CM}(Q,t) \cong e^{-\Gamma(Q)t}, \tag{2.11}$$

and

$$C_l(t) \cong \exp\left[-l(l+1)D_R t\right] \equiv \exp\left[-t/\tau_l\right], \tag{2.12}$$

where D_R is the rotational diffusion constant and τ_l is the rotational relaxation time.

The decay constant of the translational ISF in the jump-diffusion model[36] is given by

$$\Gamma(Q) = \frac{D_T Q^2}{1 + D_T Q^2 \tau_0} \tag{2.13}$$

where D_T is the translational diffusion constant and τ_0 is the time between jumps. These two quantities are related through the jump length L by:

$$D_T = \frac{\langle L^2 \rangle}{6\tau_0} \tag{2.14}$$

By comparing the ISFs from MD to the functional forms assumed in the translational jump diffusion-isotropic rotational diffusion model, we can determine to what extent nondiffusive processes influence the dynamics of water in its mixtures with DMSO.

Within the models for translational and rotational relaxation discussed above, $S(\mathbf{Q},\omega)$ is a sum of Lorentzians:

$$S(\mathbf{Q},\omega)e^{Q^2\langle u^2\rangle/3} = [j_0(Qd)]^2 \frac{1}{\pi}\frac{\Gamma(Q)}{\left[\Gamma(Q)\right]^2+\omega^2} + \sum_{l=1}^{\infty}(2l+1)\frac{[j_l(Qd)]^2}{\pi}\frac{\Gamma(Q)+l(l+1)D_R}{\left[\Gamma(Q)+l(l+1)D_R\right]^2+\omega^2} \tag{2.15}$$

Model parameters are obtained by fitting the experimental results to this functional form. As indicated in Eq.(2.15), the QNS spectrum also depends on molecular vibrations, which give rise to the Debye-Waller factor, $\exp\left(-\langle u^2\rangle Q^2/3\right)$. This factor, which depends on the mean squared vibrational displacements, $\langle u^2\rangle$, modulates the intensity of $S(\mathbf{Q},\omega)$, but does not affect its frequency-dependence.

Eq. (2.15) shows that the low-Q spectrum is dominated by the translational contribution, given that $\left[j_l(Qd)\right]^2 \propto (Qd)^{2l}$ at small Qd. At higher Q, the D_R-dependent terms become important and increase the width and intensity of the spectral wings. Because the rotations contribute less to signals at lower Q than at higher Q, the assumption that translation and rotation are decoupled should show less effect at low Q

than at high Q. Although Eq. (2.15) includes an infinite sum of Lorentzians, in practice no more than four terms in the sum over l are needed to obtain convergence in the accessible range of Q values.

2.III. Results and Discussion

2.III.A. Computer Simulation Results

We explore the dynamics of water in these mixtures primarily through evaluation of the self-intermediate scattering functions, $F_S(\mathbf{Q},t)$. Figure 2.2 displays the total ISF, $F_S(\mathbf{Q},t)$, for the water in the mixtures at an experimentally relevant range of momentum transfer, Q.[27] We observe the slowest water dynamics as the mole fraction of DMSO approaches X_{DMSO} = 0.5. The same trend is observed for the center-of-mass ISF, $F_S^{CM}(\mathbf{Q},t)$ (Figure 2.3), and the rotational ISF, $F_S^R(\mathbf{Q},t)$ (Figure 2.4).

Closer inspection of traces in Figure 2.3 shows that translational motion appears nonexponential at short times (< 2 ps), which is due in part to the fact that short-time dynamics are mostly inertial or ballistic and to some extent to the microheterogeneity in the composition. At longer times, the decay is exponential, indicating that motion is diffusive. Experimental data[25,27] suggest the slowest translational diffusion should be observed in the X_{DMSO} = 0.33 mixture. If water-DMSO interactions are stronger than water-water interactions, we expect that water motion will become slower as the water molecules interact more with DMSO molecules. The Skaf group has noted[19] that the SPC/E water and P2-DMSO mixture may overestimate the interactions between water and DMSO, leading to stiff bonds; their simulations indicate the maximum interaction between water and DMSO molecules, hence the slowest translational motion, for the

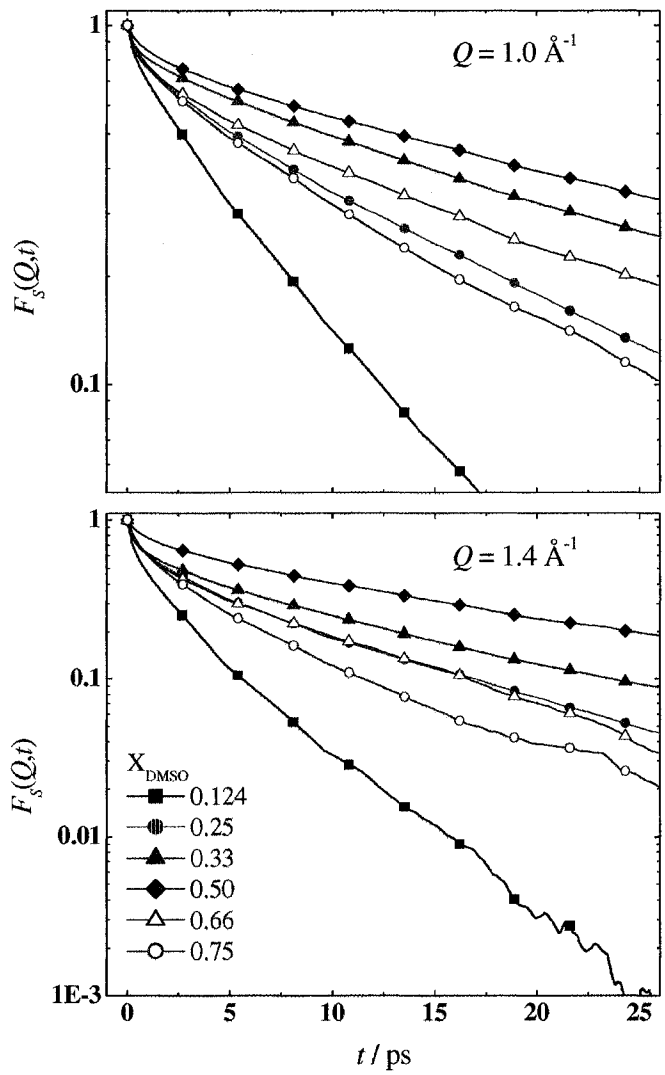


Figure 2.2. Water total intermediate scattering functions $F_S(\mathbf{Q},t)$ at Q = 1.0 Å^{-1} (top) and Q = 1.4 Å^{-1} (bottom), for water-DMSO mixtures of compositions X_{DMSO} = 0.124 (squares), 0.25(circles), 0.33(triangles), 0.5(diamonds), 0.66(open triangles), and 0.75(open circles).

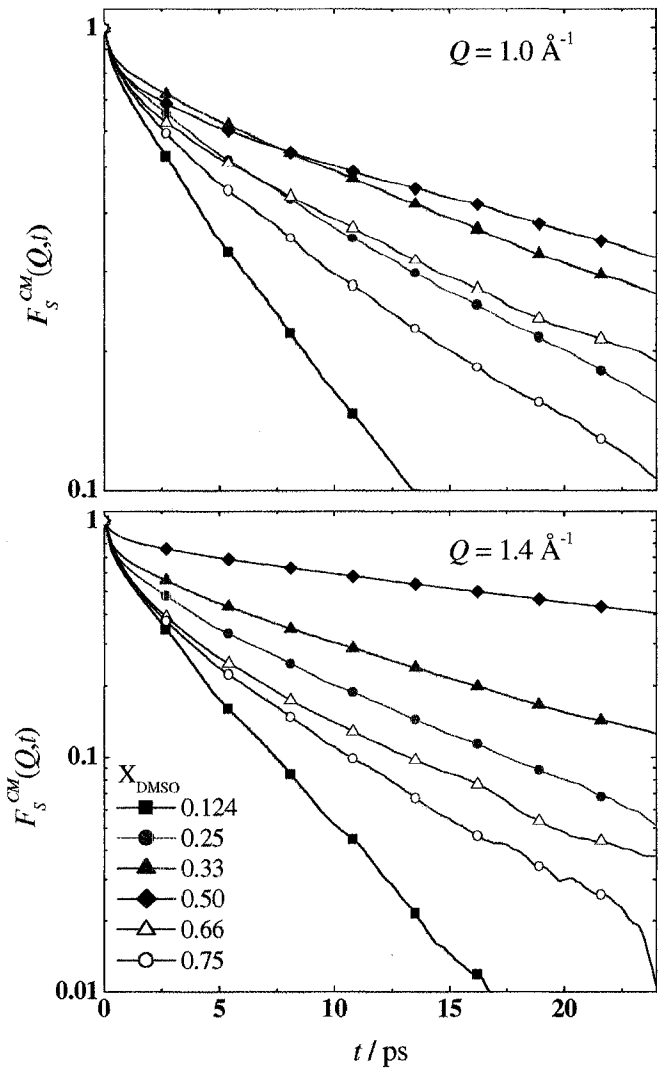


Figure 2.3. Water translational intermediate scattering functions $F_S^{CM}(\mathbf{Q},t)$ at Q = 1.0 Å^{-1} (top) and Q = 1.4 Å^{-1} (bottom), for water-DMSO mixtures of compositions X_{DMSO} = 0.124 (squares), 0.25(circles), 0.33(triangles), 0.5(diamonds), 0.66(open triangles), and 0.75(open circles).

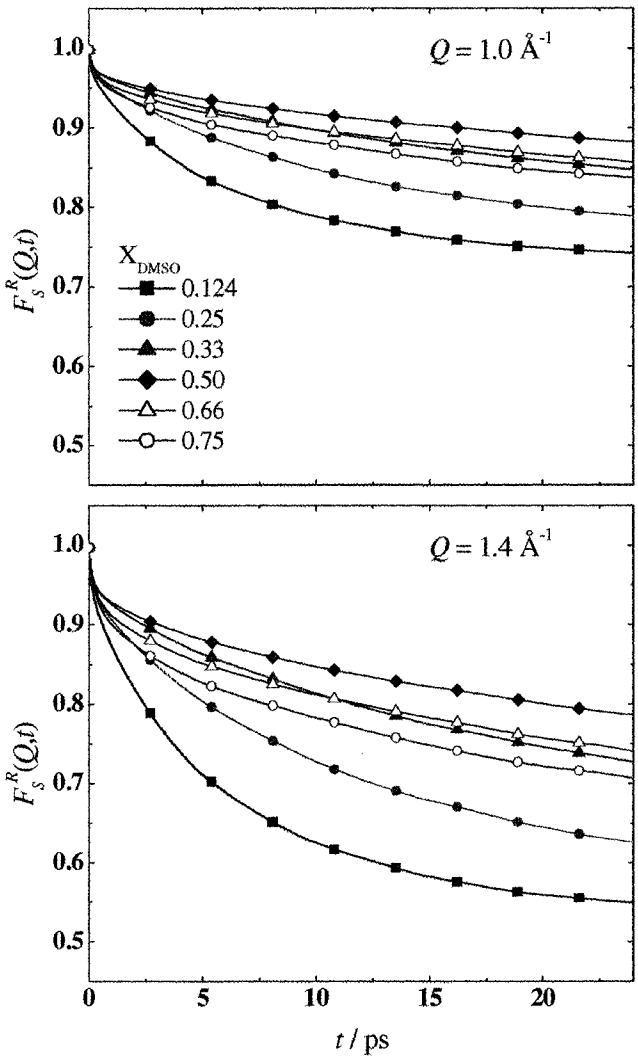


Figure 2.4. Water rotational intermediate scattering functions $F_S^R(\mathbf{Q},t)$ at $Q = 1.0$ Å^{-1} (top) and $Q = 1.4$ Å^{-1} (bottom), for water-DMSO mixtures of compositions X_{DMSO} = 0.124 (squares), 0.25(circles), 0.33(triangles), 0.5(diamonds), 0.66(open triangles), and 0.75(open circles).

X_{DMSO} = 0.5 mixture. At the same time, the P2-DMSO molecule shows greater mobility than is experimentally observed,[37,38] which leads to an overall increase in the mobility of the system at higher X_{DMSO}.

Unlike the translational ISF, which in bulk liquid eventually decays to zero, the rotational ISF decays to a plateau value that is a function of Qd, as shown in Eq. (2.9). Specifically, when the orientational time correlation functions $C_l(t)$ have decayed to zero at long time, then the rotational ISF is given by

$$F_S^R(Q,t\to\infty)\approx\left[j_0(Qd)\right]^2=\left[\frac{\sin(Qd)}{Qd}\right]^2 \tag{2.16}$$

For an SPC/E water molecule, $d = 0.964\,\text{Å}$, and the plateau values should be 0.73 and 0.52 for Q = 1.0 and 1.4 $Å^{-1}$, respectively. Figure 2.4 shows that rotational relaxation is not complete within 24 ps for any of the water/DMSO mixtures studied at either Q value. Previous simulations[39] indicate that bulk SPC/E water relaxes within this time window, and in the X_{DMSO} = 0.124 mixture, $F_S^R(\mathbf{Q},t)$ is relatively close to full relaxation at t = 24 ps. These simulation results imply that there may exist a component to the rotational motion in mixtures with $X_{DMSO} > 0.124$ that cannot be resolved through the QNS experiments of Bordallo et al., [27] in which the QENS instrument resolution was on the order of 50 ps.

We can use our MD data to determine the extent of rotation-translation coupling present in the water ISFs. Figure 2.5 compares the full ISF, Eq.(2.5), to the product approximation $F_P(\mathbf{Q},t)$, Eq. (2.6) for several compositions and Q values. The results displayed in Figure 2.5 reveal that the decoupling approximation is not consistently applicable within the Q range relevant to the experiment, which is

$0.38\,\text{Å}^{-1} < Q < 2.63\,\text{Å}^{-1}$. From Eq. (2.15) , we expect that the effects of coupling between translational and rotational motion on $F_S(\mathbf{Q},t)$ will increase with Q, and this result is seen in the MD data. In addition, the coupling between translation and rotation increases as the $X_{DMSO} = 0.5$ mixture is approached from either water-poor or water-rich mixtures. The coupling between water translational and rotational motions should increase as the hydrogen bonding network becomes stronger and rotational motion becomes more hindered. Because typical experimental QNS data are analyzed by assuming that translational and rotational motion are decoupled, our results showing that rotation and translation are coupled suggest that the assumption is not entirely correct at high-Q and is least valid for the $X_{DMSO} = 0.5$ mixture.

We further explore the rotational dynamics of water in these mixtures and evaluate the effectiveness of the isotropic rotational diffusion model by inspection of the rotational time correlation function $C_l(t)$, given by Eq.(2.10). In the isotropic rotational diffusion model, molecular reorientation is assumed to take place through random, small-angle rotations. On a time average, this model assumes that there is no preferred orientation of a given molecule. As water molecules are not spherical rotors, we anticipate that rotational relaxation might not be isotropic. Anisotropy in rotational relaxation can be tested by comparing the relaxation properties of different unit vectors embedded in the molecule. In this case we compare the relaxation rate of the vector $\mathbf{d}$ to other vectors related to the orientation of water molecules, such as the unit vector $\hat{\mathbf{u}}$ along the molecular dipole.

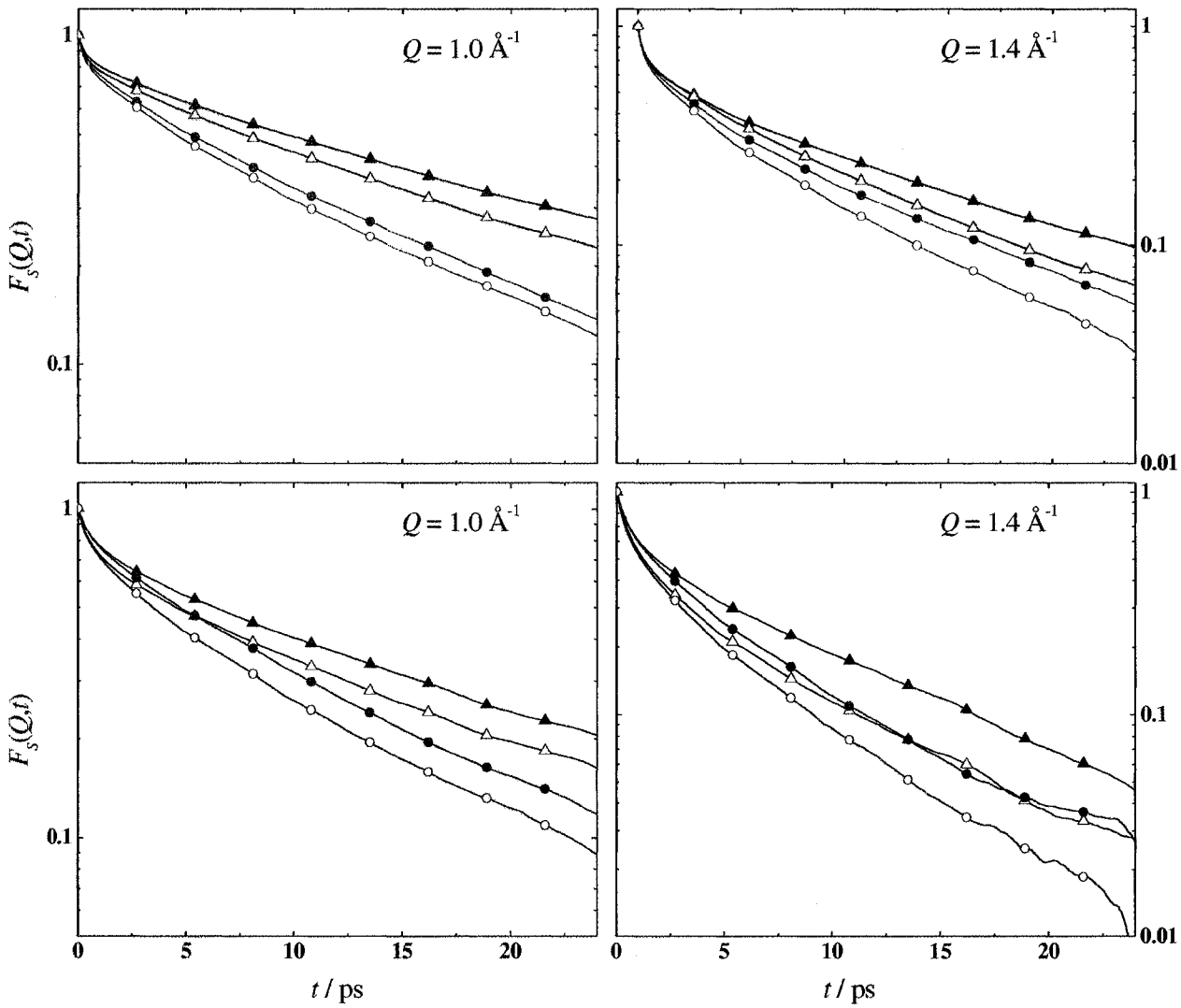


Figure 2.5. Water product and total intermediate scattering functions $F_S^P(\mathbf{Q},t)$ and $F_S(\mathbf{Q},t)$ at Q = 1.0 Å^{-1} (left) and Q = 1.4 Å^{-1} (right), for water-DMSO mixtures of compositions X_{DMSO} = 0.25(circles), 0.33(triangles) in the top graph, and 0.66(triangles), and 0.75(circles). Filled symbols represent total intermediate scattering functions, open symbols represent product intermediate scattering functions.

Comparison of $C_l(t)$ with

$$C_{l,\mu}(t) = \left\langle P_l\left[\hat{\mathbf{u}}(0)\cdot\hat{\mathbf{u}}(t)\right]\right\rangle \tag{2.17}$$

allows us to determine the extent of rotational anisotropy of the system. From the rotational diffusion model, $C_l(t)$ is assumed to consist of a single exponential decay as in Eq.(2.12). Figure 2.6 shows that at a given X_{DMSO}, the bond vector relaxation, $C_1(t)$, is slower than the dipole vector relaxation, $C_{1,\mu}(t)$, and that both relaxations become slower as X_{DMSO} approaches 0.5. This indicates that rotational relaxation of water in mixtures with DMSO is not isotropic. Reports in the literature[13,16,18,19] indicate that hydrogen bonds formed between water and DMSO are stronger than hydrogen bonds between water molecules. This should lead to slower relaxation for the bond vector than for the dipole vector because relaxation of the bond vector necessarily involves breaking these stronger hydrogen bonds. As the water dipole vector can rotate through a relatively large angle before the hydrogen bond needs to break, the dipole vector relaxation exhibits faster dynamics than the hydrogen bond vector relaxation does. As we approach the equimolar water-DMSO mixture, rotational dynamics become more hindered, suggesting that the interactions between water and DMSO molecules have the largest effect on H-bond network strength and connectivity in the X_{DMSO} = 0.5 mixture. This result agrees with the MD simulation study performed by Borin and Skaf.[19]

We expect that at short times, we will not see rotational diffusion in $C_1(t)$, as inertial as well as librational motion produce oscillatory features within the first picosecond. At longer times, we may see rotational diffusion, which will manifest as a

single exponential decay if there is one type of rotational process. Close inspection of Figure 2.6 reveals evidence of librational motion within the oscillatory features of the first 200 femtoseconds of $C_1(t)$. At longer times, $C_1(t)$, does not appear to decay as a single exponential. This, in conjunction with differences in bond vector and dipole vector relaxation differences noted above, challenges the applicability of the isotropic rotational diffusion model for this system.

To further explore the applicability of the isotropic rotational diffusion model to water in mixtures with DMSO, we investigate the l-dependence of the $C_l(t)$ time correlation functions. According to the isotropic rotational diffusion model, $C_l(t)$ is given by Eq. (2.12), so a plot of $\ln C_l(t) / [l(l+1)]$ as a function of t should be linear and independent of l. However, we observe that the rotational relaxation becomes slower as l increases, as shown in Figure 2.7. This behavior is indicative of a high-torque liquid, and is expected for water.[40] At short times, we can see that the time-evolution is nonexponential. Oscillations in the traces reveal water librational motion. We see an increase in the amplitude of the oscillations and an increase in the number of oscillation periods as X_{DMSO} increases, which indicates that the strength of the hydrogen bonds increases, decreasing the damping of water librational motion at high X_{DMSO} to a greater extent than it does at low X_{DMSO}. This is a consequence of the replacement of water-water H-bonds with the stronger water-DMSO H-bonds.

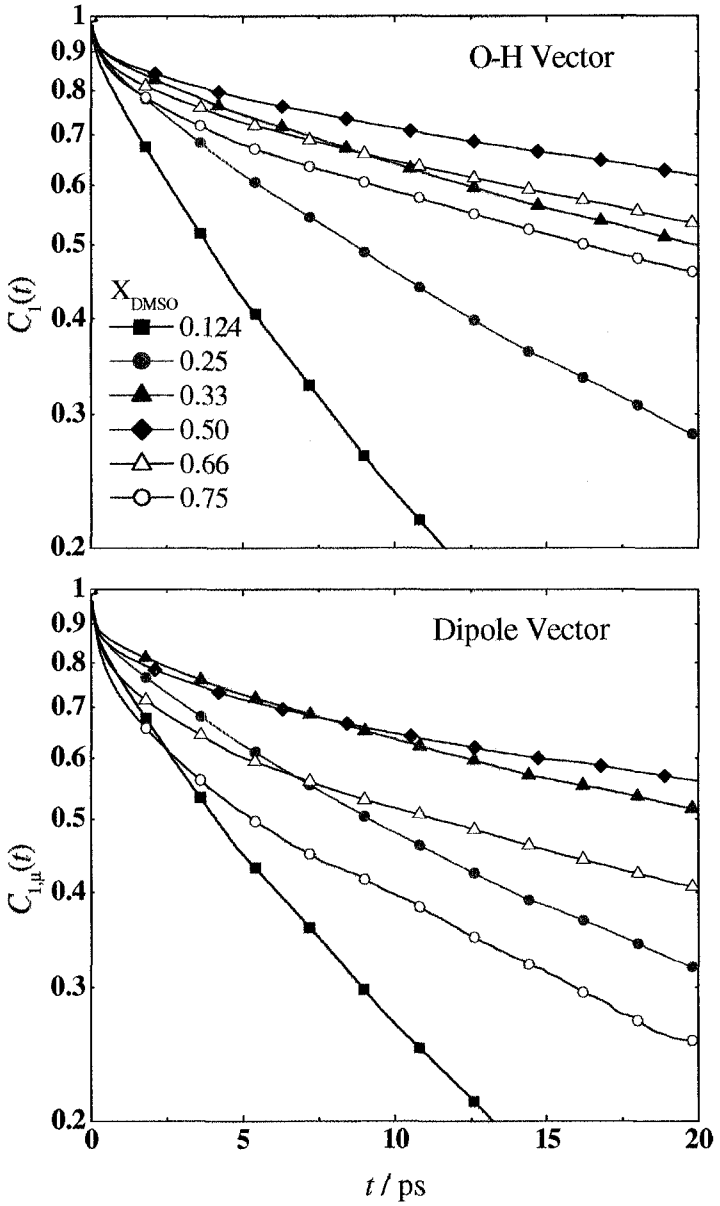


Figure 2.6. Time correlation functions $C_1(t)$ (top) and $C_{1,\mu}(t)$ (bottom) for water-DMSO mixtures of compositions X_{DMSO} = 0.33 (triangles), 0.5 (diamonds), and 0.66(open triangles).

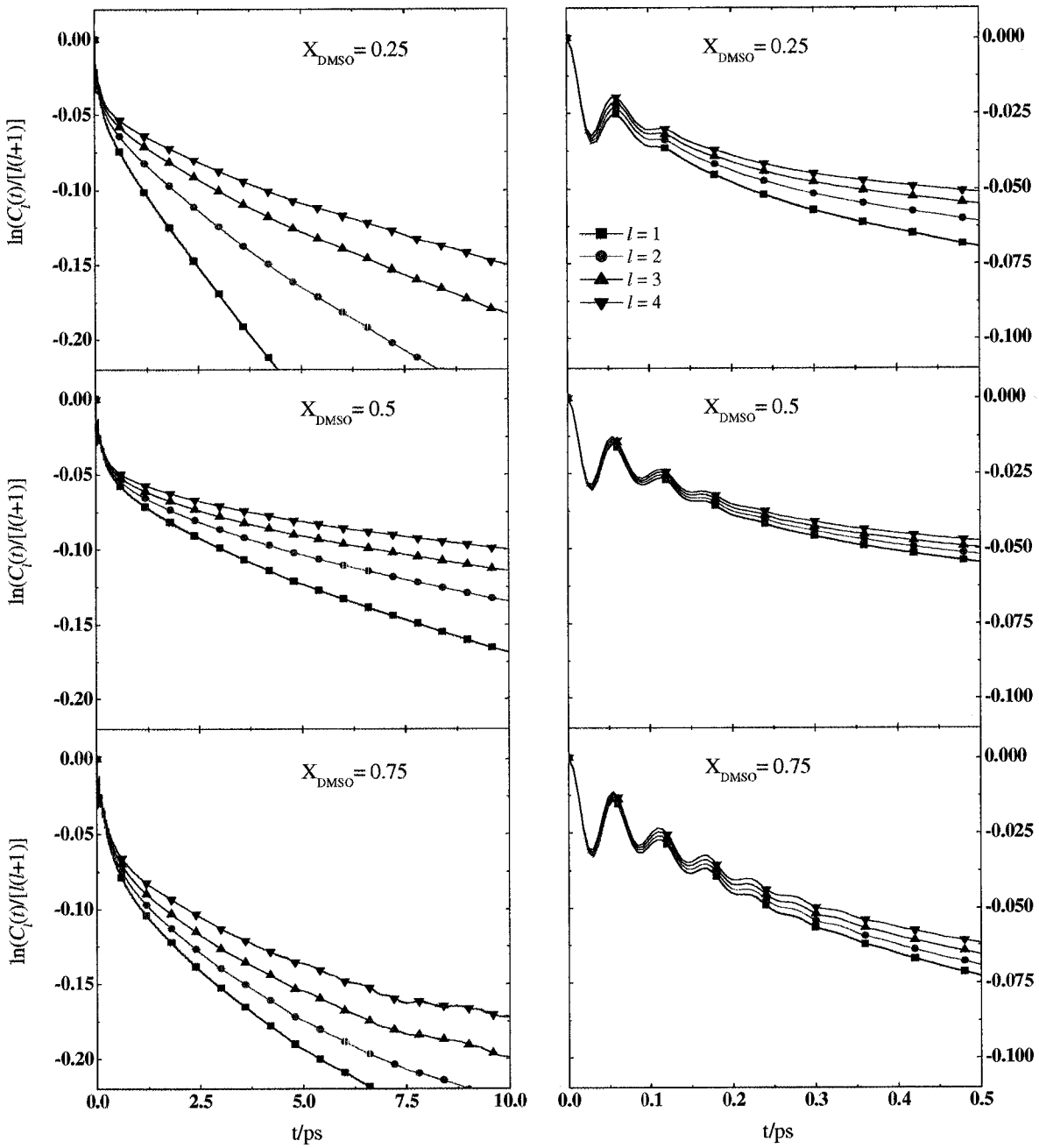


Figure 2.7. First 10.0 ps (left) and 0.5 ps (right) of time correlation function $C_l(t)$ for l=1-4 for water-DMSO mixtures of compositions X_{DMSO} = 0.25 (top), 0.5 (center), and 0.75 (bottom).

Laage and Hynes recently proposed an extended-jump rotational diffusion model[41] to interpret the rotational relaxation of liquid water. The extended-jump model is based on the Ivanov jump-rotation model, [42] in which a molecule rotates in jumps rather than continuously. The molecule has a particular orientation for a time τ_0, and then rotates instantaneously through an angle θ_0. In the Ivanov model, $C_l(t)$ decays exponentially with the rotational relaxation time

$$\tau_l^{jump} = \tau_0 \left(1 - \frac{\sin\left[(l+1/2)\theta_0 \right]}{(2l+1)\left[\sin(\theta_0/2) \right]} \right)^{-1} \tag{2.18}$$

In the case of water $1/\tau_0$ corresponds to the rate of the hydrogen bond switching. This can be envisioned as a specific water molecule switching hydrogen bonding partners by rotating by θ_0 after a period τ_0. The Ivanov model[42] assumes that orientational motion is frozen between angular jumps. The extended jump model [41] adds an additional relaxation process which corresponds to reorientation of the oxygen-oxygen vector of a pair of hydrogen-bonded water molecules. In this model, the time-dependence of the single exponential decay associated with Legendre polynomial P_l of the O-H vector rotational motion is given as

$$\frac{1}{\tau_l} = \left(\frac{1}{\tau_l^{jump}} + \frac{1}{\tau_l^{OO}} \right) \tag{2.19}$$

where τ_l^{OO} is the reorientation time for P_l of oxygen-oxygen vector, and $\tau_{l,jump}$ is given by Eq.(2.18). Laage and Hynes applied[41] the extended-jump model to SPC/E water at room temperature, and found that the model is consistent with experimentally-observed reorientation times, and is also supported by their MD simulation data.

In the case of aqueous mixtures, generalization of the Ivanov and extended jump models would involve taking into account the fact that water forms hydrogen bonds of different strengths with water and the other mixture component. Applying the model to water-DMSO mixtures would involve determining the parameters associated with both types of hydrogen bond. This type of analysis is beyond the scope of our study. As a step in this direction, we determine the *l*-dependence of the exponentially-decaying portion of $C_l(t)$ for the range of compositions covered by our MD simulations. As this model is not appropriate for short times,[41] where relaxation is inertial or librational, we limit our fitting to a time window from 5-120 ps. Figure 2.8a displays the *l*-dependence of the inverse of the relaxation time, $1/\tau_l$ at several X_{DMSO} values. We see a minimum in the magnitude of $1/\tau_l$ for X_{DMSO} = 0.5, which again indicates that the water is rotationally hindered in the equimolar mixture. In the water-rich X_{DMSO} = 0.124 mixture, we see a maximum in both slope and magnitude of $1/\tau_l$, indicating that the water has an increased reorientational mobility, relative to the other mixtures studied. Figure 2.8b displays the ratio τ_1/τ_l plotted against l for each of the water-DMSO mixtures we investigate. We see that τ_1/τ_l for water-DMSO mixtures deviates significantly from the ratio predicted by rotational diffusion, which further challenges the applicability of the isotropic rotational diffusion model. The water-rich mixtures X_{DMSO} = 0.124, 0.25, and 0.33 appear to deviate more from rotational diffusion than the X_{DMSO} = 0.5 and 0.75 mixtures do.

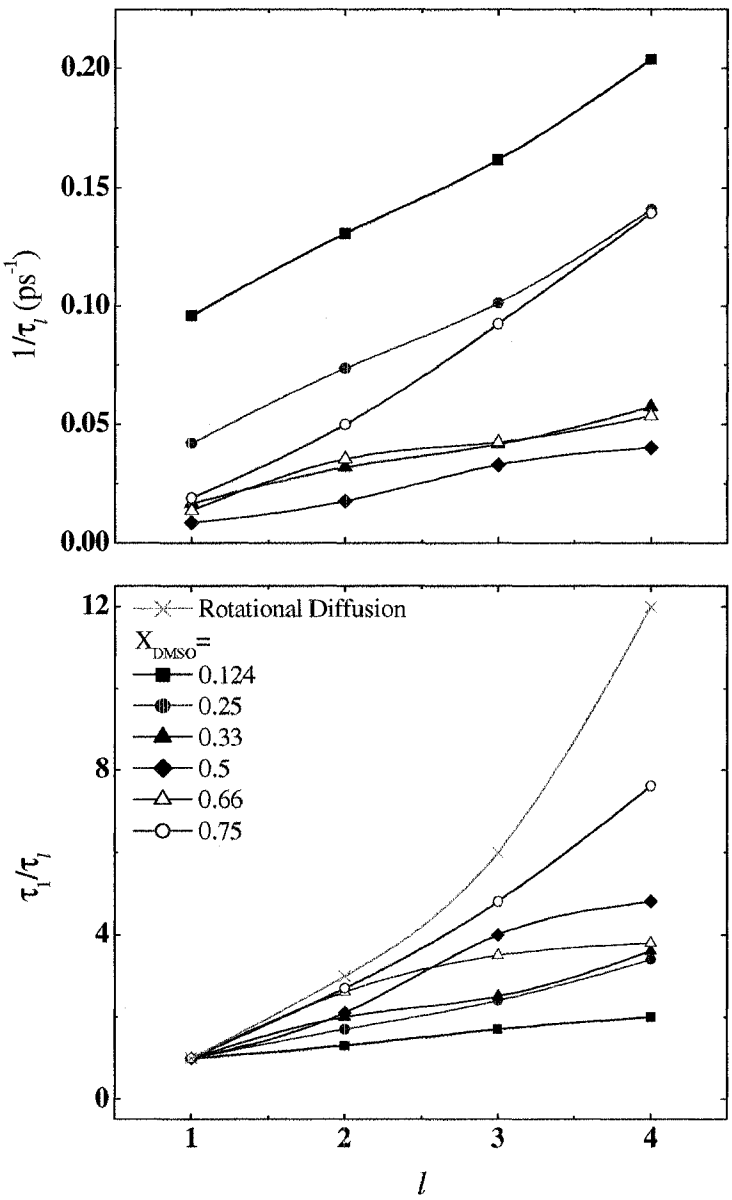


Figure 2.8. Results from fits of rotational time correlation function $C_l(t)$ for l=1-4 to a single exponential decay. The relaxation period $1/\tau_l$ is given for water-DMSO mixtures of compositions X_{DMSO} = 0.124 (squares), 0.25(circles), 0.33(triangles), 0.5(diamonds), 0.66(open triangles), and 0.75(open circles) in the top panel (2.8a). The bottom panel (2.8b) contains τ_1/τ_l plotted against l for the water-DMSO mixtures, with diffusive rotations (X) as a comparison.

Table 2.3. Evaluation of the diffusive character of rotational relaxation times

X_{DMSO}	τ_1 (ps)	τ_2 (ps)	τ_3 (ps)	τ_4 (ps)	τ_1/τ_2	τ_1/τ_3
0.124	10	7.7	6.2	4.9	1.3	1.7
0.25	24	14	10	7.1	1.7	2.4
0.33	61	31	24	17	2.0	2.5
0.50	120	57	30	25	2.1	4.0
0.66	73	28	21	19	2.6	3.5
0.75	53	20	11	7.2	2.7	4.8

Table 2.3 displays the relaxation times τ_l for l=1-4 within these water-DMSO mixtures. We see that τ_l reaches a maximum in the X_{DMSO} = 0.5 mixture. Considering τ_l as a function of l at constant X_{DMSO}, reveals a deviation from the $1/\tau_l = l(l+1)D_R$ relationship expected in the case of diffusive reorientations. As can be seen from the table and Fig. 2.8b, the ratios τ_1/τ_l are always smaller than $l(l+1)/2$ expected on the basis of the rotational diffusion model. Laage and Hynes observed[41] similar results in the case of bulk SPC/E water.

2.III.B. Comparisons to Experiment

Direct comparison of our simulations to experimental results allows us to determine if the interaction models for water and DMSO are adequate to represent the translational and rotational dynamics observed in QNS. Convoluting our MD simulations with the experimental QNS resolution function $Res(\mathbf{Q},\omega)$, obtained by Bordallo et al.,[27]

permits direct comparison of simulation with experiment. Representative plots of $S(\mathbf{Q},\omega)\otimes Res(\mathbf{Q},\omega)$ at different Q and X_{DMSO} shown in Fig. 2.9 display excellent agreement of our simulation with the experimentally obtained spectra[27] for $X_{DMSO} = 0.33$ and 0.5 at low-Q, but all mixtures exhibit deviation from experiment at mid- to high- Q, where the rotational contribution to $S(\mathbf{Q},\omega)$ is non-negligible. The agreement at low-Q can be explained by the convolution with $Res(\mathbf{Q},\omega)$; slow dynamics cannot be seen if the Lorentzian linewidth is narrower than the instrument resolution function.

We apply the jump-translational diffusion model, Eq.(2.13), to analyze the intermediate scattering functions we have generated with our MD simulations. Because the jump-translational diffusion model assumes that single timescale for the translational motion, and the MD simulation shows more than one time component, we needed to consider how to fit the simulated data. Furthermore, experimentally, the temporal range probed by QNS is limited by the resolution of the instrument. Long-time dynamics would be experimentally inaccessible, while dynamics are ballistic at very short times and therefore have a different functional form than the medium time data does. Thus we fit only the intermediate time data of our simulations, typically a window from 5 to 80 ps, of the MD simulation to a single exponential decay with a relaxation time $1/\Gamma$, where Γ would be the Lorentzian linewidth in the energy domain of a QNS scan. Figure 2.10 shows that, within these assumptions, the jump-translational diffusion model (Eq.(2.13)) fits the data reasonably well. A Lorentzian linewidth of less than $\sim 2\times10^{-2}\,ps^{-1}$ would be narrower than the instrument resolution function in the experiments of Bordallo et al.[27]

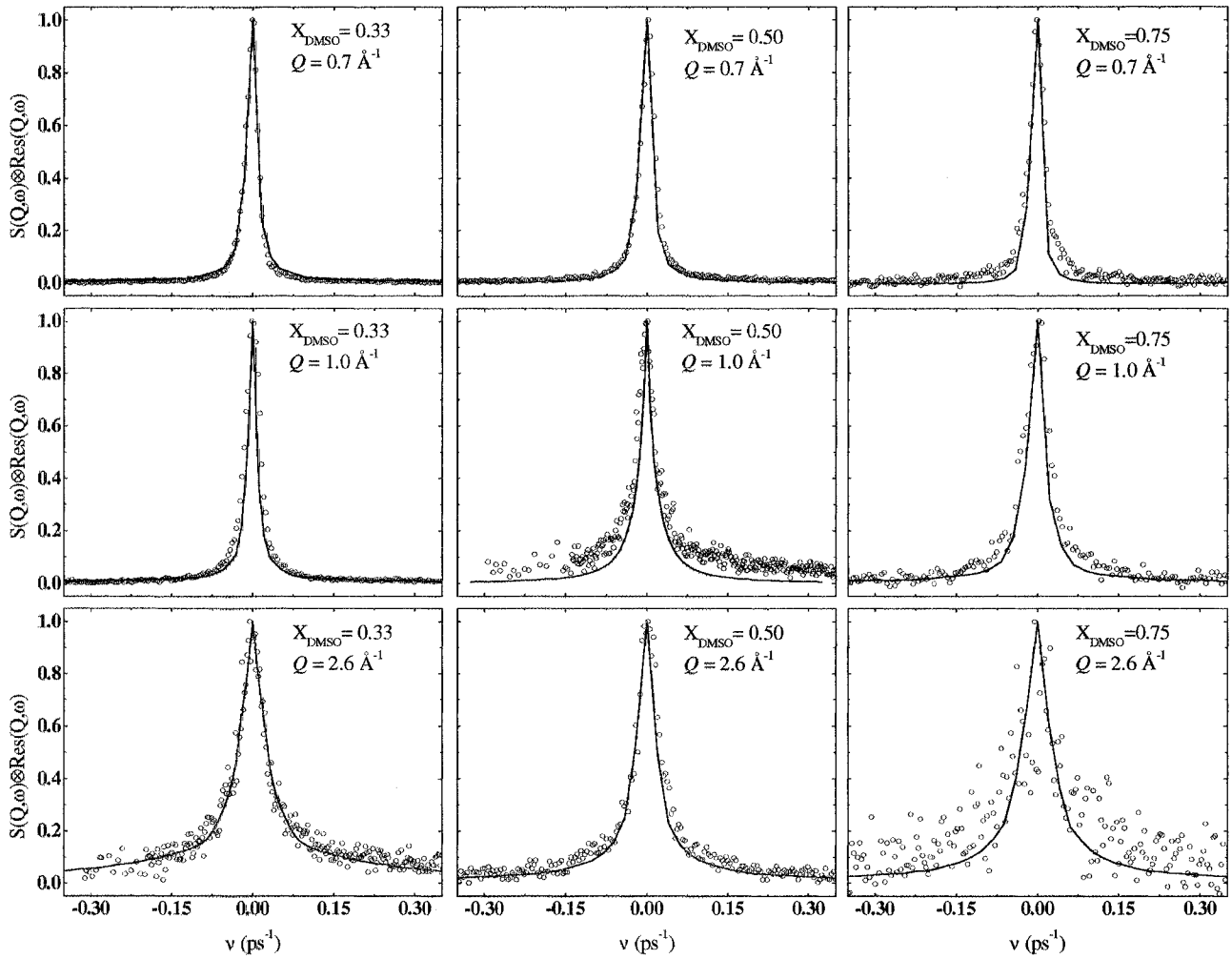


Figure 2.9. Comparisons between QNS spectra obtained from our MD simulations (lines) and QNS experiments (points) performed by Bordallo et al. [*J. Chem. Phys.* **2004**, *121*, 12457] at Q = 0.7 (left), 1.0 (middle), and 2.6Å (right) for mixtures X_{DMSO} = 0.33 (top), 0.5 (middle), and 0.75 (bottom).

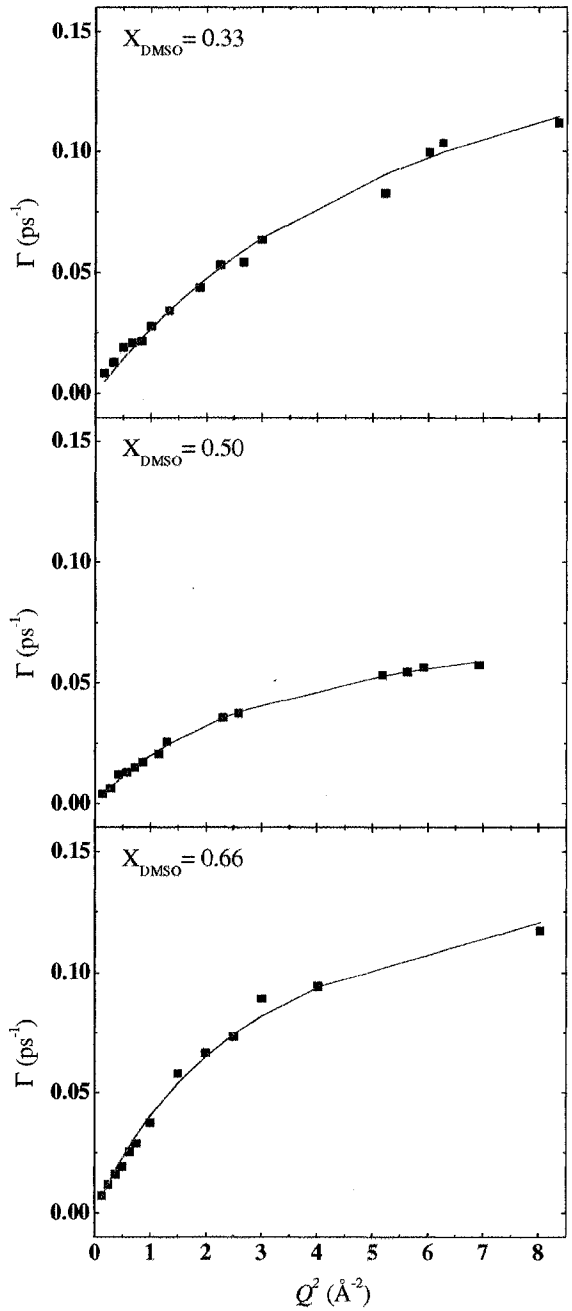


Figure 2.10. Example fits of translational intermediate scattering function to the jump-diffusion model. Experimental data are shown as points, line is the fit.

We see that for the X_{DMSO} = 0.33 and 0.5 mixtures at low-Q, the Lorentzian linewidth is less than $2\times10^{-2}ps^{-1}$, meaning that the agreement between experiment and simulation observed in Fig. 2.9 is due to the convolution of the data with the instrument resolution. Our simulation data should disagree with experimental data, but the discrepancy is obscured by the instrumental resolution function.

Translational mobility can also be quantified in terms of mean-squared displacements (MSDs) of the molecules in the mixture. At long times, MSDs can be related to the translational diffusion coefficient, D_{eff}, by the Einstein relationship:

$$D_{eff} = \lim_{t\to\infty} \frac{\left\langle \left| \mathbf{r}_{CM}(t) - \mathbf{r}_{CM}(0) \right|^2 \right\rangle}{6t} \tag{2.20}$$

From the slope obtained by a linear fit to the MSDs in a time window from 5 to 15 ps we extract the translational diffusion coefficient D_{eff}, so named to preserve the nomenclature used by Borin and Skaf[19] in their MD simulations of the water-DMSO system. D_{eff} can be compared to D_T, obtained by the jump-diffusion model.

The resulting translational diffusion constants predicted from the mean-squared displacements and jump-translation fit of the MD data, are plotted in Figure 2.11 along with the experimental results from Bordallo et al.[27] The translational diffusion constant predicted by the jump-diffusion analysis of the MD data agrees with that produced by a linear fit of the water mean-squared displacements of our data. Both the experimental data and our MD data also agree with simulations reported by Borin and Skaf.[19] We find that the translational diffusion constant predicted by MD simulations qualitatively agrees

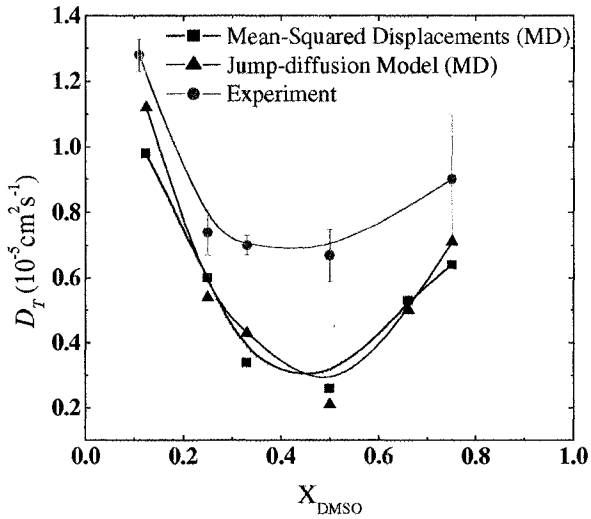


Figure 2.11. Water translational diffusion constants predicted from our fits of simulation mean-squared displacements (squares) and intermediate scattering functions (circles), and QNS experiments (triangles) performed by Bordallo and co-workers.[*J. Chem. Phys.* **2004**, *121*, 12457] Lines are drawn to guide the eye.

with that found experimentally by Bordallo et al.[27] The experimental trends appear to be matched quite well by the MD simulations, although the values appear to be offset. Residence times, shown in Figure 2.12, reveal very good agreement between MD simulations and experiments.

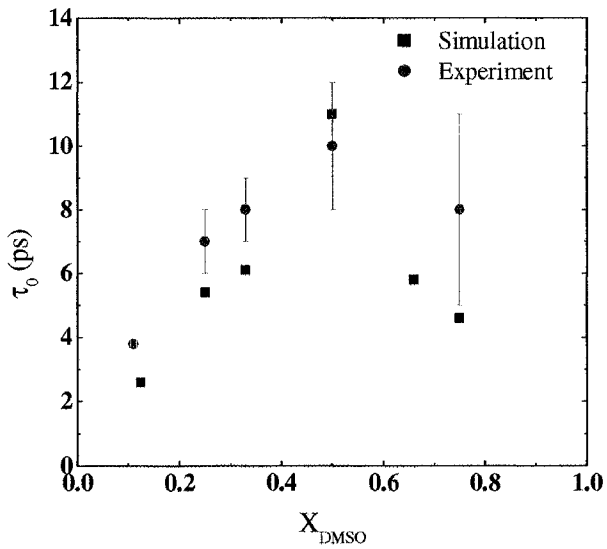


Figure 2.12. Water residence time as predicted by our simulations (circles) and QNS experiments (triangles) performed by Bordallo et al. [*J. Chem. Phys.* **2004**, *121*, 12457]

Although the translational motion of water is effectively captured by the jump diffusion model, our results indicate that the isotropic rotation model is not applicable for water dynamics in water-DMSO mixtures. NMR experiments[43] and MD simulations,[41,43-45] using the SPC/E water model, suggest that the rotational diffusion of bulk water is anisotropic. Ropp et al.[43] used NMR techniques to measure the $\tau_{2,int}$ integrated rotational correlation times of the OD bond vector and out-of-plane vector of bulk $D_2{}^{16}O$ and

$D_2{}^{17}O$, respectively, over a series of temperatures. The results of their experiments showed that the bond vector P_2 correlation time was about 1.33 times the out-of-plane vector correlation time, and this ratio was temperature independent. Since the correlation times of these orthogonal unit vectors were not equal, Ropp et al.[43] concluded that rotational motion of water is anisotropic. Simulations performed on several models of bulk water[41,43-45], including SPC/E, and on isotopically-substituted SPC/E water[43,44] lend additional support to the experimental observation, and suggest that the anisotropy is not an artifact of the isotopic substitutions made in either experiment or simulation, but is instead an inherent property of bulk water. The analysis of our data by the extended jump rotation model[41] in the preceding section, should more accurately capture bulk water rotational motion than does the traditional isotropic rotational model.

The rotational relaxation times of water molecules predicted by our MD simulations challenge the applicability of the isotropic rotational diffusion model to water dynamics in binary mixtures of water and DMSO. We see similarity in the overall trend of the water rotational relaxation predicted by our MD simulations and that obtained by the experiments of Bordallo et al.[27] Other MD simulations[20] show that the dielectric relaxation time of P2 DMSO agrees well with experimental data[8] within most water-DMSO mixtures, but deviates from experimental data in the X_{DMSO} = 0.33 and 0.5 mixtures, where the simulation predicts[18,19] a much slower relaxation than that which is observed experimentally. We also note that, based on $C_l(t)$ analysis described above and in the previous section, we would not expect the isotropic rotational diffusion model to accurately represent the data.

2.IV. Summary and Conclusions

This chapter presents the single-particle dynamics of water in water/DMSO mixtures calculated from a set of MD simulations in a composition range relevant for comparisons to QNS experiments performed by Bordallo et al.[27] The SPC/E water[46] and the P2 DMSO[16] models were used to carry out the simulation. MD simulations performed indicate that the translational and rotational motion of water molecules is slowest at the 1:1 mixture of water and DMSO, that is, X_{DMSO} = 0.5, which agrees with other simulations of this system.[20] The combination of SPC/E water and P2 DMSO models may overestimate the water-DMSO interactions,[19] resulting in a growing discrepancy between diffusivities predicted by MD and extracted from experimental data as the composition approaches the X_{DMSO}= 0.5 mixture.

To compare our MD simulations with QNS experiments of Bordallo et al.,[27] translational ISFs were analyzed using the jump-translational diffusion model. This model fit the simulations quite well over much of the time scale relevant to QNS, and gave results that qualitatively matched experimental results. The quantities predicted by MD data only appear to be offset from experimental data; the trends are reproduced by simulations, but the values appear shifted. This implies that the potentials used in the MD simulations predict uniformly slower motion than what is observed experimentally.

Analysis of the simulated rotational time correlation functions indicate that the water rotation is nonexponential, is not isotropic, and is not continuous. Rotational relaxation times for water OH bonds reach a maximum in the X_{DMSO} = 0.5 mixture, which is in line with other MD simulations of this system.[19] The strongest hydrogen bonding is observed in the 1:1 (X_{DMSO}=0.5) mixture. The dipole vector relaxation is faster than the

bond vector relaxation of water molecules in the mixtures, and we attribute this to the hydrogen bonding; stiffer hydrogen bonds affect the bond vector correlation functions to a greater extent than they affect the dipole vector correlations. We find that the rotational relaxation times do not exhibit the l-dependence expected for molecules relaxing according to the isotropic rotational diffusion model. Water reorientations in bulk water have been reported[43-45] to be anisotropic due to the highly directional hydrogen bond, so we would expect anisotropy in the water/DMSO hydrogen-bonding situation. Our data suggests that the jump-rotation model[41] may be useful in the interpretation of the water rotational time correlation functions. However, this model will need to be modified to include the variety of bonding arrangements that exist within the water-DMSO mixtures.

We find that the validity of the decoupling approximation for rotations and translations applied in the analysis of QNS data by Bordallo et al.[27] varies by mixture composition and by Q. At low-Q, where rotational effects are negligible, the approximation seems reasonable in most samples. However, at high-Q, the approximation breaks down. The extent of hydrogen bonding also affects the coupling between translation and rotation. Stronger hydrogen bonding leads to stronger coupling between water rotational and translational motions. This means that the decoupling approximation is least applicable at $X_{DMSO} = 0.5$, and most reasonable in low-Q and water-rich or water-poor samples. Experimental results[27] for translational motion are based on the fitting of low-Q data, where the decoupling approximation has negligible effect because the rotational contribution is relatively small, and is thus quite reasonable. However, rotational motion is analyzed at high-Q, where the decoupling approximation has the greatest effect, and thus is less reliable.

References and Notes

(1) *Biological Actions of Dimethyl Sulfoxide*; Jacob, S. W.; Herschler, R., Eds.; New York Acad. Sci.: New York, N. Y., 1975; Vol. 243, pp 508 pp.
(2) de la Torre, J. C. *Ann. NY Acad. Sci.* **1983**, *411*, 293.
(3) Yu, Z. W.; Quinn, P. J. *Biosci. Rep.* **1994**, *14*, 259.
(4) Hernandez-Perni, G.; Leuenberger, H. *Eur. J. Pharm. Biopharm.* **2005**, *61*, 201.
(5) Cowie, J. M.; Toporowski, P. M. *Can. J. Chem.* **1961**, *39*, 2240.
(6) Schickman, S. A.; Amey, R. L. *J. Phys. Chem.* **1971**, *75*, 98.
(7) Aminabhavi, T. M.; Gopalakrishna, B. *J. Chem. Eng. Data* **1995**, *40*, 856.
(8) Kaatze, U.; Pottel, R.; Schafer, M. *J. Phys. Chem.* **1989**, *93*, 5623.
(9) Baker, E. S.; Jonas, J. *J. Phys. Chem.* **1985**, *89*, 1730.
(10) Gordalla, B. C.; Zeidler, M. D. *Mol. Phys.* **1986**, *59*, 817.
(11) Gordalla, B. C.; Zeidler, M. D. *Mol. Phys.* **1991**, *74*, 975.
(12) Cabral, J. T.; Luzar, A.; Teixeira, J.; Bellissent-Funel, M. C. *J. Chem. Phys.* **2000**, *113*, 8736.
(13) Soper, A. K.; Luzar, A. *J. Chem. Phys.* **1992**, *97*, 1320.
(14) Soper, A. K.; Luzar, A. *J. Phys. Chem.* **1996**, *100*, 1357.
(15) Vaisman, II; Berkowitz, M. L. *JACS* **1992**, *114*, 7889.
(16) Luzar, A.; Chandler, D. *J. Chem. Phys.* **1993**, *98*, 8160.
(17) Liu, H.; Mueller-Plathe, F.; van Gunsteren, W. F. *JACS* **1995**, *117*, 4363.
(18) Borin, I. A.; Skaf, M. S. *Chem. Phys. Lett.* **1998**, *296*, 125.
(19) Borin, I. A.; Skaf, M. S. *J. Chem. Phys.* **1999**, *110*, 6412.
(20) Skaf, M. S. *J. Phys. Chem. A* **1999**, *103*, 10719.
(21) Benjamin, I. *J. Chem. Phys.* **1999**, *110*, 8070.
(22) Chalaris, M.; Samios, J. *J. Mol. Liq.* **2002**, *98-9*, 399.
(23) Mancera, R. L.; Chalaris, M.; Samios, J. *J. Mol. Liq.* **2004**, *110*, 147.
(24) Nieto-Draghi, C.; Avalos, J. B.; Rousseau, B. *J. Chem. Phys.* **2005**, *122*.
(25) Packer, K. J.; Tomlinson, D. J. *Trans. Farad. Soc.* **1971**, *67*, 1302.
(26) Cabral, J. T.; Luzar, A.; Teixeira, J.; Bellissent-Funel, M. C. *Physica B* **2000**, *276*, 508.
(27) Bordallo, H. N.; Herwig, K. W.; Luther, B. M.; Levinger, N. E. *J. Chem. Phys.* **2004**, *121*, 12457.
(28) Di Cola, D.; Deriu, A.; Sampoli, M.; Torcini, A. *J. Chem. Phys.* **1996**, *104*, 4223.
(29) Teixeira, J.; Bellissent-Funel, M. C.; Chen, S. H.; Dianoux, A. J. *Phys. Rev. A* **1985**, *31*, 1913.
(30) Berendsen, H. J. C.; Grigera, J. R.; Straatsma, T. P. *J. Phys. Chem.* **1987**, *91*, 6269.
(31) Allen, M. P.; Tildesley, D. J. *Computer Simulation of Liquids*; Clarendon: Oxford, U.K., 1987; Vol. 1987.
(32) Verlet, L. *Phys. Rev.* **1967**, *159*, 98.
(33) Evans, D. J. *Mol. Phys.* **1977**, *34*, 317.

(34) Evans, D. J.; Murad, S. *Mol. Phys.* **1977**, *34*, 327.
(35) Sears, V. F. *Can. J. Phys.* **1967**, *45*, 237.
(36) Egelstaff, P. A. *An Introduction to the Liquid State*; Academic: London, 1967.
(37) Skaf, M. S. *J. Chem. Phys.* **1997**, *107*, 7996.
(38) Skaf, M. S. *Mol. Phys.* **1997**, *90*.
(39) Harpham, M. R.; Ladanyi, B. M.; Levinger, N. E.; Herwig, K. W. *J. Chem. Phys.* **2004**, *121*, 7855.
(40) Lynden-Bell, R. M.; Steele, W. A. *J. Phys. Chem.* **1984**, *88*, 6514.
(41) Laage, D.; Hynes, J. T. *Science* **2006**, *311*, 832.
(42) Ivanov, E. N. *Soviet Physics JETP* **1964**, *18*, 1041.
(43) Ropp, J.; Lawrence, C.; Farrar, T. C.; Skinner, J. L. *JACS* **2001**, *123*, 8047.
(44) Svishchev, I. M.; Kusalik, P. G. *Journal Of The Chemical Society-Faraday Transactions* **1994**, *90*, 1405.
(45) van der Spoel, D.; van Maaren, P. J.; Berendsen, H. J. C. *J. Chem. Phys.* **1998**, *108*, 10220.
(46) Berendsen, H. J. C.; Grigera, J. R.; Straatsma, T. P. *J. Chem. Phys.* **1987**, *91*, 6269.

Chapter 3

The Effects of Size and Surfactant Counterion on the Dynamics of Water in Aerosol-OT Reverse Micelles

3.I. Introduction

In the appropriate concentration range, three-component systems of polar, nonpolar and amphiphilic molecules can form self-assembled structures in which isolated nanoscopic droplets of the polar phase are separated by an amphiphile layer from a continuous nonpolar phase, namely reverse micelles (RMs). RMs are used in a range of applications from templates for semiconductor and metal nanoparticles, [1-3] to restricted "nanobeakers" for chemistry, [4] and to models for biological membranes.[5] These small aggregates sequester polar solvents, usually water, in their interiors. When the RMs are approximately spherical, as is often the case, the diameter is proportional to the molar ratio of water to surfactant, $w_0 = [H_2O]/[\text{surfactant}]$. The properties of the water inside the RM differ significantly from those of the bulk liquid, both structurally and dynamically, due to the geometric size constraints of the environment and to intermolecular interactions at the micellar interface.[6-11]

We have launched a comprehensive program to explore how confinement impacts water sequestered in RMs. Our studies have focused on the dynamics of polar solvation by ultrafast time-resolved fluorescence spectroscopy, [12-18] molecular dynamics (MD) simulations, [19-23] and quasielastic neutron scattering (QNS). [22-24] These studies have allowed us to understand how the confined environment impacts both water structure and dynamics and how this heterogeneous environment affects solvation.

While QNS is a fairly nonstandard method for exploring water motion in RMs, this technique has been applied to studies of water motion in other confined systems, most frequently porous silica and alumina, [25-43] and to a smaller extent in other porous materials[44-46] and gels. [47,48] Several MD simulation studies modeling QNS in cylindrical pores representing Vycor glass have been carried out.[17,32,49-52] These studies indicate significant changes in water mobility due to confinement and also due to differences in mobility of water in different interfacial layers.

Confinement in RMs differs from confinement in pores due to differences in the geometry of the confining region (approximately, sphere vs. cylinder). Furthermore, varying the water content, w_0, makes it possible to vary the volume of the confining region as well as the properties of the interface, given that the surface area per surfactant head group increases with w_0. Our QNS experiments and MD simulations of $F_S(\mathbf{Q},t)$ of water hydrogen atoms in RMs of varying size provide information on the effects of changing core-interface mole ratios and interface properties on the water mobility.

Faeder et al. studied[19] differences between Na^+ and K^+ counterions in AOT, using the same model that we use in this work. That work focused on structural and dynamical differences between the RMs for w_0 values of 1, 2, 4, and 7.5. In terms of structure,

radial density profiles, pair density functions, and coordination numbers as a function of distance from the interface were calculated. For dynamics, the water dipole time autocorrelation and mean-squared displacements were calculated. This study showed greater interfacial mobility and weaker water-counterion coordination for water in K-AOT than water in Na-AOT. In time-resolved fluorescence Stokes shift experiments, Riter et al.[16,17,53] and Pant et al.[14,15] measured the impact of varying counterions on solvation dynamics in AOT reverse micelles. They found that $Ca(AOT)_2$ led to reduced solvation dynamics while K-AOT and NH_4AOT showed progressively faster solvation response than Na-AOT.

QNS is discussed in detail in Chapter 1, Section *II.A.* of this dissertation, so a brief discussion will be presented here. Quasielastic neutron scattering measures the dynamic structure factor $S(\mathbf{Q},\omega)$ [54] as a function of momentum transfer $\mathbf{Q}$ and frequency ω. If we use deuterated surfactant and deuterated nonpolar solvent in our system, the only hydrogen-containing species that remains is water, so the majority of the scattering occurs due to translational, rotational and vibrational motions of water hydrogen atoms. Specifically, $S(\mathbf{Q},\omega)$ is the frequency Fourier transform of the self-intermediate scattering function, $F_S(\mathbf{Q},t)$

$$S(\mathbf{Q},\omega)=\frac{1}{2\pi}\int_{-\infty}^{\infty}F_S(\mathbf{Q},t)e^{i\omega t}dt. \tag{3.1}$$

$$F_S(\mathbf{Q},t)=\frac{1}{N}\sum_{j=1}^{N}\left\langle \exp\left\{i\mathbf{Q}\cdot\left[\mathbf{r}_j(0)-\mathbf{r}_j(t)\right]\right\}\right\rangle, \tag{3.2}$$

where N is the number of water hydrogen atoms and $\mathbf{r}_j(t)$ is the position of the jth hydrogen atom.

MD simulation provides a means of obtaining $F_S(\mathbf{Q},t)$ from molecular trajectory data. By comparing the experimental QNS results with the MD simulations, we can increase our level of understanding of the environment found inside the RMs. Agreement between the experimental and simulation results gives us confidence to extend the simulations to explain the intramicellar environment beyond what is possible through experiments alone. We use the MD simulations to compare interfacial and core water motion. As a result, these studies begin to provide a comprehensive view of how the RM milieu acts to confine and perturb normal water motion.

3.II. Quasielastic Neutron Scattering Experiments

3.II.A. Experimental

For studies of the effect of varying w_0 on the dynamics, we made use of custom synthesized AOT-d34 (sodium bis(2-d-ethylhexyl) sulfosuccinate, Figure 3.1) with 98% deuteration of the ethylhexyl chains, which was purchased from Prof. Robert Thomas at Oxford University. Perdeuterated 2,2,4-trimethylpentane (d-isooctane) and D_2O were obtained from Cambridge Isotope Labs and used as received. For samples probing H_2O, ultrapure water (Milli-Q, 18.2MΩ-cm) was used for the samples. AOT-d34/H_2O/d-isooctane samples were prepared at w_0 = 1, 2.5, 5. Samples used to collect backgrounds at the same w_0 were prepared with AOT-d34/D_2O/d-isooctane.

The QNS experiments were carried out at the Argonne National Lab (ANL), Intense Pulsed Neutron Source (IPNS) division, using the QENS instrument. The energy resolution of this instrument is 85 μeV. The Q range was 0.36 - 2.53 $Å^{-1}$. Data were accumulated for 15-24 hours for each sample and background, at a fixed temperature of

300 K. Because d-AOT was only available in limited supply, higher w_0 were obtained by adding the requisite amount of water to the lower-w_0 samples. Further instrument details are given in Chapter 1, section *IV.A*.

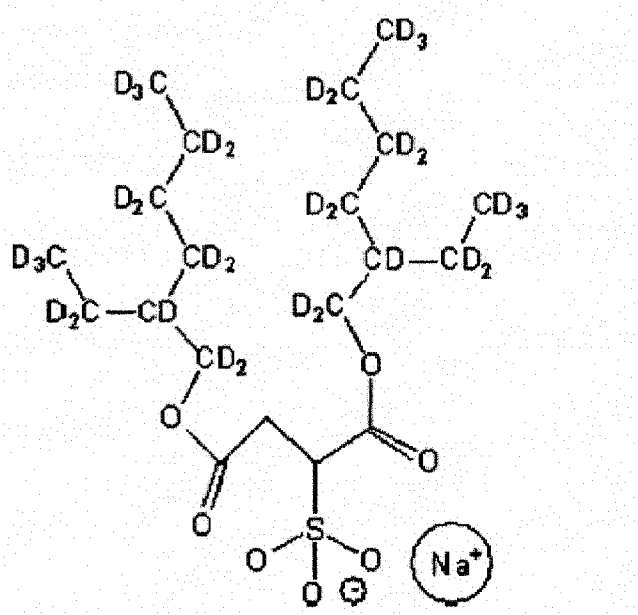


Figure 3.1. Structure of deuterated sodium bis(2-d-ethylhexyl) sulfosuccinate.

We explored the effect of the counterion on water dynamics using protonated AOT. We performed counterion exchange on dried, purified NaAOT using a liquid-liquid exchange method described by Eastoe et al.[55] We modified this procedure slightly, introducing additional purification steps before and after counterion exchange. We present the details of these modifications in Appendix 3.1 of this work.

We investigated the effect of exchanging the sodium counterion of AOT with potassium (KAOT), calcium ($CaAOT_2$), and copper[II] ($CuAOT_2$). In these samples, we used cyclohexane-*d12*, obtained from Cambridge Isotope Laboratory, for the nonpolar phase. Since these samples were prepared using protonated AOT, we investigated

counterion effects at higher w_0, where we would have more signal. $Ca(AOT)_2$ and $Cu(AOT_2)$ were prepared at w_0=10, and KAOT was prepared at w_0=5. For purposes of comparison, we analyzed w_0=5 NaAOT samples, with isooctane-*d34* and cyclohexane-*d12*. This allows the comparison of the counterion effects in a common solvent (cyclohexane), and a comparison to the isooctane data.

Counterion samples were analyzed at ANL as well as at the National Institute of Standards and Technology Center for Neutron Research (NCNR). Details of the ANL instrument have been discussed previously in this report. The instrument employed at NCNR is the disk-chopper spectrometer (DCS), which was set to give a resolution of 35 μeV, and a Q-range between 0.08 and 1.5 $Å^{-1}$. Sample data were accumulated for 8-12 hours with the temperature fixed at 300 K. Further details about the DCS instrument are presented in Chapter 1, section *IV.B.*

3.II.B. Models

To analyze QNS data, we assume that translation and rotation are decoupled, and that vibration can be approximated using a Debye-Waller factor.[54] Within the decoupling approximation,

$$S(\mathbf{Q},\omega) \cong V(\mathbf{Q},\omega) \otimes T(\mathbf{Q},\omega) \otimes R(\mathbf{Q},\omega) \tag{3.3}$$

where ⊗ denotes a convolution in ω. V, T and R represent contributions from vibrational, translational and rotational degrees of freedom, respectively. Furthermore, it is reasonable to assume that vibrations in the quasielastic region contribute only through the frequency-independent Debye-Waller factor:

$$V(Q) \cong \exp\left(\frac{-Q^2\langle u^2\rangle}{3}\right) \tag{3.4}$$

where $\langle u^2\rangle$ represents the mean squared vibrational displacement.[54] The decoupling approximation between rotational and translational degrees of freedom is not always accurate for bulk water[57] and will need to be tested by analysis of MD data for water within RMs.

If the decoupling assumed in Eq. (3.3) holds, $F_S(\mathbf{Q},t)$ is a product of the Debye-Waller factor and the center-of-mass, $F_S^{CM}(\mathbf{Q},t)$, and rotational, $F_S^R(\mathbf{Q},t)$ intermediate scattering functions :

$$F_S(\mathbf{Q},t) \cong e^{-\langle u^2\rangle Q^2/3} F_S^{CM}(\mathbf{Q},t) F_S^R(\mathbf{Q},t). \tag{3.5}$$

$F_S^{CM}(\mathbf{Q},t)$ and $F_S^R(\mathbf{Q},t)$ are, respectively, Fourier transforms of $T(\mathbf{Q},\omega)$ and $R(\mathbf{Q},\omega)$.

For bulk liquid water, a combination of jump diffusion model for translation and isotropic rotational diffusion model for $F_S^R(\mathbf{Q},t)$ fits reasonably well the experimental data.[57] This model has proved somewhat less successful for water confined in cylindrical pores. Nonetheless, due to its simplicity, we use it here as our starting point for the analysis of QNS data.

The jump-diffusion model[58] assumes that a molecule remains in a given position for a residence time τ_0, after which time it jumps a length l to another position. It has been shown[59] that for $Qa > \pi$, where a is the radius of confinement, the Lorentzian linewidth predicted from continuous diffusion is virtually indistinguishable from the

more accurate diffusion within a sphere model.[59] If we assume that the cavity radius appropriate for RMs is the radius of the water pool, then $a \geq$ 8 Å and the condition $Qa > \pi$ is met for all Q values in the w_0 = 2.5, 5, and 10 samples, and for most values of Q in the w_0 = 1 case. It should be noted, however, that we expect that water mobility will be reduced in the vicinity of the interface, so for interfacial water, the actual radius of confinement might be smaller than this estimate.

In the jump-diffusion model,

$$T(Q,t) = \frac{1}{\pi} \frac{\Gamma(Q)}{\omega^2 + \Gamma(Q)}, \tag{3.6}$$

where the Lorentzian linewidth is given by:

$$\Gamma(Q) = \frac{D_{T,jump} Q^2}{1 + D_{T,jump} Q^2 \tau_{jump}} \tag{3.7}$$

and $D_{T,jump}$ is the translational diffusion constant, which is related to L and τ_{jump} by:

$$D_{T,jump} = \frac{\langle L^2 \rangle}{6\tau_0} \tag{3.8}$$

The isotropic rotational diffusion model[60] accounts for molecular rotational motion. This model assumes that rotation takes place through a series of infinitely small angular changes, and that there is no preferred orientation of the water molecule. The functional form is:

$$R(Q,\omega) = [j_0(Qd)]^2 \delta(\omega) + \frac{1}{\pi} \sum_{l=1}^{\infty} (2l+1)[j_l(Qd)]^2 \left[\frac{l(l+1)D_R}{[l(l+1)D_R]^2 + \omega^2} \right], \tag{3.9}$$

where d is the distance of the H atoms from the water center-of-mass in equilibrium molecular geometry, and j_0 is the spherical Bessel function of order l and D_R is the rotational diffusion constant.

Within the models for translational and rotational relaxation given in Eqs. (3.6) and (3.9), $S(\mathbf{Q},\omega)$ is a sum of Lorentzians:

$$S(\mathbf{Q},\omega)e^{Q^2\langle u^2\rangle/3} = [j_0(Qd)]^2 \frac{1}{\pi}\frac{\Gamma(Q)}{\Gamma^2(Q)+\omega^2} + \sum_{l=1}^{\infty}(2l+1)\frac{[j_l(Qd)]^2}{\pi}\frac{\Gamma(Q)+l(l+1)D_R}{[\Gamma(Q)+l(l+1)D_R]^2+\omega^2} \quad (3.10)$$

The above equation shows that the low-Q spectrum is dominated by the rotational contribution, given that $[j_l(Qd)]^2 \propto (Qd)^{2l}$ at small Qd. At higher Q, the D_R-dependent terms become important and increase the width and intensity of the spectral wings.

3.IIC. Results and Discussion

Our first set of QNS experiments was performed to investigate how the radius of confinement affects the dynamics of water within the RMs. Figure 3.2 depicts the convolution between experimentally-determined QNS spectra $S(\mathbf{Q},\omega)$ the energy resolution function, $Res(\omega)$, for a $w_0 = 5$ NaAOT RM. $Res(\omega)$ is slightly asymmetric about $\omega = 0$, giving rise to the asymmetry in the depicted convolutions, $S(\mathbf{Q},\omega) \otimes Res(\omega)$.

At Q < 0.52 Å^{-1}, the $S(\mathbf{Q},\omega) \otimes Res(\omega)$ for $w_0 = 1$ and 2.5 (not shown) is dominated by the instrument resolution function $Res(\omega)$, while in the case of the largest RM, $w_0 = 5$, we can see in Figure 3.2 that some additional intensity is apparent in the spectral wings. At larger Q values, all three RM sizes exhibit appreciable intensity in the wings. The Lorentzian intensity increases with Q, as expected from Eq.(3.10).

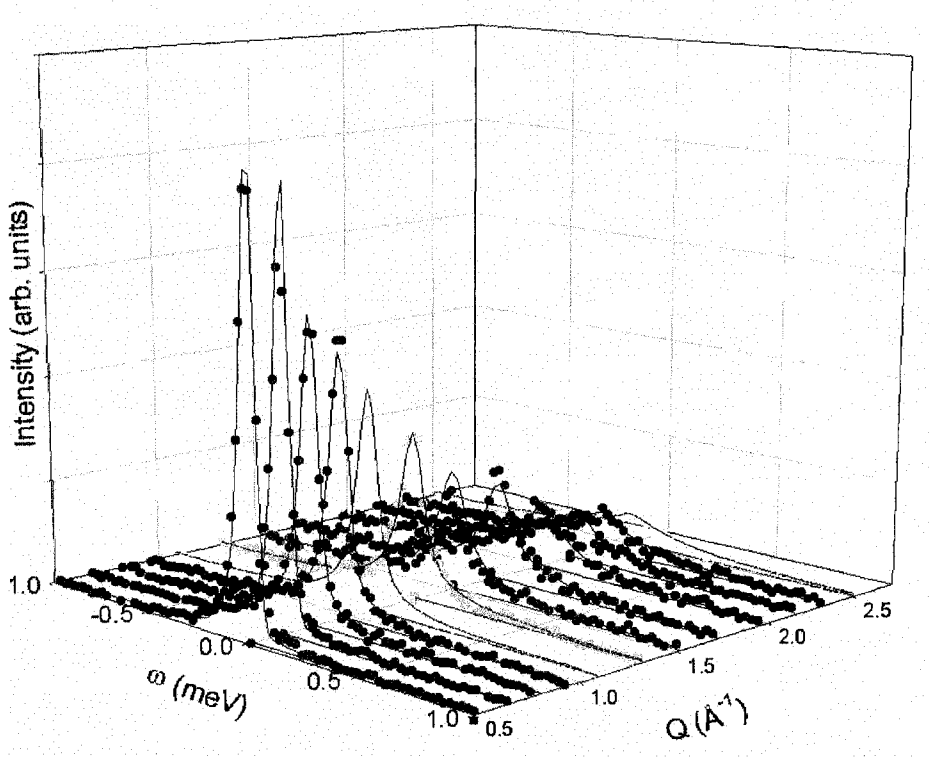


Figure 3.2. The incoherent structure factor $S(\mathbf{Q},\omega)\otimes Res(\omega)$ of $w_0 = 5$ NaAOT RMs plotted against ω and Q.

We determine the translational diffusion constant from the variation of the Lorentzian linewidth as a function of Q^2 (Eq.(3.7)) at low Q. Results from analysis of the QNS data with the models given in Section II.B are given in Table 3.1. The jump translational diffusion-isotropic rotational diffusion parameters for bulk water are also included.[61]

Table 3.1 Translational jump-diffusion/isotropic rotational diffusion parameters determined from analysis of QNS data.[a]

System	$D_{T,jump}$ (10^{-5} cm^2/s)	τ_{jump} (ps)	$\tau_1^{diff} = 1/(6D_R)$ (ps)
$w_0 = 1$	b	b	2.6 ± 0.1
$w_0 = 2.5$	b	b	1.5 ± 0.1
$w_0 = 5$	0.5 ± 0.1	12 ± 4	1.6 ± 0.1
bulk water	2.5	1.25	1.16

[a]RM data – our measurements; bulk water – Ref. [61] results for 293 K.
[b]For $w_0 = 1$ and 2.5, fits to this model revealed no translational component.

Translational diffusion rate in RMs is clearly lower than in the bulk. For the two smaller RM sizes corresponding to $w_0 = 1$ and 2.5, the width of $Res(\omega)$ obscures much of the spectral linewidth for $Q < 1$ Å^{-1} and we are unable to estimate the translational jump-diffusion parameters. These parameters can be estimated for the largest RM, $w_0 = 5$, but may not accurately reflect the translational diffusion rate in this system. Given that we are unable to determine translational diffusion parameters for $w_0 = 1$ and 2.5, the Q-dependent spectral lineshapes in these RMs are ascribed exclusively to rotational diffusion. This accounts for the fact that we find a slightly shorter rotational correlation

time, $\tau_1^{jump} = 1/(6D_R)$, in w_0 = 2.5 than in w_0 = 5, despite the fact that rotational motion is expected to be faster in larger RMs.

Despite these uncertainties, it is clear that rotational relaxation, as quantified by τ_1^{diff}, is lower than in bulk water. However, decrease in the rotational mobility resulting from confinement within RMs appears to be considerably more modest than the change in the translational mobility. Because of the difficulties associated with obtaining parameters related to translational diffusion, our values of the rotational diffusion times τ_1 carry some uncertainties associated with ascribing too much of the total spectral width to rotation, especially in the case of the smaller RMs. We rely on the results of MD simulation to remove some of these uncertainties and to supplement the information obtained from fitting of the QNS spectra.

Our second set of QNS experiments focuses on the effect of the surfactant counterion on water dynamics within reverse micelles. For these experiments, deuterated AOT was unavailable, so we were forced to use protonated AOT, which significantly increases the difficulty of the experiments, as the signal fraction produced by water hydrogen atoms is very small (< 5% of the total signal). We performed the analysis as in the previous set of experiments; subtract the background, perform a global fit of the spectra to the models. Results are given in Table 3.2.

Table 3.2 Translational jump-diffusion/isotropic rotational diffusion parameters determined from analysis of QNS data.

System	$D_{T,jump}$ (10^{-5} cm^2/s)	τ_{jump} (ps)	$\tau_1^{diff} = 1/(6D_R)$ (ps)
$w_0 = 10$ $Ca(AOT)_2$	0.9 ± 0.1	23 ± 5	1.2 ± 0.1
$w_0 = 10$ $Cu(AOT)_2$	1.2 ± 0.1	12 ± 3	1.1 ± 0.1
$w_0 = 5$ KAOT	0.4 ± 0.1	17 ± 4	1.9 ± 0.2
$w_0 = 5$ NaAOT	0.5 ± 0.1	12 ± 4	1.6 ± 0.1

We find that the water dynamics within the $w_0 = 5$ KAOT/cyclohexane RMs are slightly slower than those of water within $w_0 = 5$ NaAOT/isooctane RMs. We expect to see competing effects here; cyclohexane forms smaller RMs than does isooctane, but K^+ ions are larger than Na^+ ions, and therefore K^+ ions make larger RMs than do Na^+ ions. We would expect that water dynamics in $w_0 = 10$ $Ca(AOT)_2$ and $Cu(AOT)_2$ to be faster than the smaller $w_0 = 5$ NaAOT or KAOT RMs, but was not intrinsically clear which of the divalent cation RMs would hinder water motion the most. It is possible that the difference in symmetry between the spherical $Ca(AOT)_2$ RM and the ellipsoidal $Cu(AOT)_2$ RM could affect the dynamics. Water may be less hindered when moving along the long axis of the $Cu(AOT)_2$ RM than it is along the short axis. These QNS results indicate that the water dynamics are slightly slower in $Ca(AOT)_2$ RMs than $Cu(AOT)_2$ RMs.

We attempted to perform QNS experiments on these counterion-exchanged samples using the DCS at NCNR, however, there were problems in the data collection due to a slight misalignment of the sample. This misalignment rendered a portion of the low-Q data unusable and reduced the overall signal to noise ratio. Because of this, we do

not attempt to fit the data to the models discussed in the preceding section. Ideally, we would fit the low-Q portion of the spectra to a translational model, use the parameters from that as known values, and then fit the high-Q data to a rotational model. Because of the problems with the low-Q data, we fit to a convolution of a delta function plus an unconstrained Lorentzian with the instrument resolution. Lorentzian linewidths produced by these fits are given in Figure 3.3. Because of the scatter in these data, we are unable to extract reliable results.

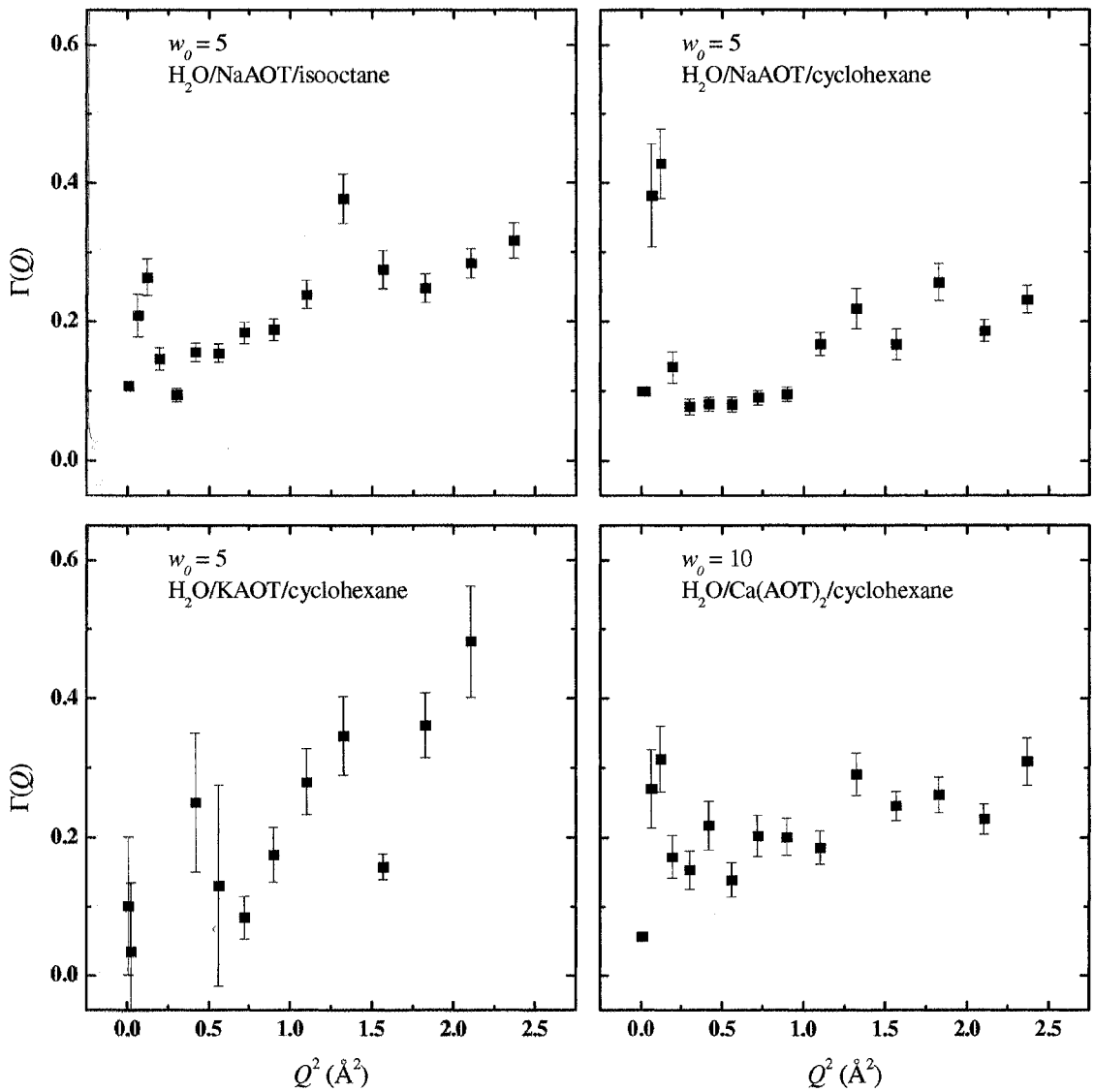


Figure 3.3. Lorentzian linewidth as a function of momentum transfer for $M^{n+}(AOT)_n$ RMs.

3.III. Molecular Dynamics Investigations

3.III.A. Simulation Model

A detailed description of the model for the RM, the method used for constructing RMs corresponding to a given w_0 value and of the simulation procedures used to produce equilibrated RMs have been presented elsewhere.[20] Briefly, the surfactant tails and the nonpolar phase of the RM are modeled as a rigid spherical cavity, treating the only the interior at the molecular level. As we assume a large excess of the nonpolar phase, intermicellar interactions are neglected. The surfactant headgroups protrude from the cavity boundary and are tethered only in the radial direction. This allows non-uniform lateral spacing between the headgroups. The radial positions of the headgroups are restricted using a harmonic potential given by:

$$u(b) = \frac{1}{2} k_e (b - b_e)^2 \tag{3.11}$$

where b is the distance from the cavity wall, b_e = 2.5 Å is the equilibrium distance between the headgroup and the cavity wall, and k_e = 600 kcal mol^{-1} Å^{-2} is the force constant. The interaction potential between the free molecules (water and counterions) and the cavity is determined by assuming the region outside the cavity is composed of a uniform continuum of nonpolar molecules, with potential parameters appropriate for alkanes,[62] that interact with the interior molecules through a Lennard-Jones (LJ) 6-12 potential. The form of this potential, averaged over a spherical cavity, is given in Ref. 20. Molecular interactions within the cavity are described in terms of a LJ + Coulomb form, with the interactions between sites α and β on different molecules given as:

$$u_{\alpha\beta}(r) = 4\left(\varepsilon_\alpha \varepsilon_\beta\right)^{1/2}\left[\left(\frac{\sigma_\alpha + \sigma_\beta}{2r}\right)^{12} - \left(\frac{\sigma_\alpha + \sigma_\beta}{2r}\right)^{6}\right] + \frac{q_\alpha q_\beta}{4\pi\varepsilon_0 r} \qquad (3.12)$$

where ε_α and σ_α are the LJ well-depth and diameter and q_α the (partial) charge of site α. The headgroup, denoted as Z^-, is represented as a single site carrying a charge of *-e*. We use the SPC/E model[63] for water. Na^+ parameters are taken from a previous simulation of sodium dodecyl sulfate in water[64] with the addition of K^+ and Cs^+ parameters.[65] The only change in going from Na^+ to K^+ and Cs^+ was to increase the LJ diameter by an amount determined from a previous parameterization of the counterion/water potential.[65] The potential parameters for the RM components are summarized in Table 3.3.

The number of water molecules and surfactant headgroups in Na^+ RMs of a given w_0 were taken from estimates based on light scattering data by Eicke and Rehak.[66] Table 3.4 lists the cavity radii and compositions of the simulated RMs, which were adjusted to give pressures similar to previous simulations.[19-22] K^+ and Cs^+ RMs were assumed to have the same composition as the Na^+ RMs of a given w_0. The nonpolar cavity sizes for KAOT and CsAOT were calculated based on scaling the RM volume based on the volume occupied by the K^+ or Cs^+ ion to the volume of the Na^+ ion.[19].

Integration of the trajectories was accomplished using the velocity-Verlet algorithm[67] with a 2 fs time step size and appropriate bond length and bond angle constraints[67,68] for the water molecule (the molecular geometry of SPC/E water is given in Table 2.3). Temperature was regulated using the Berendsen thermostat[69] with a time constant of 0.4 ps during equilibration periods and 2 ps during the 500 ps production runs. The average temperature was set at 300 K during the production runs.

Table 3.3 – Potential and molecular structural parameters

A. Lennard-Jones + Coulomb potential parameters

interaction site	σ_α (Å)	ε_α / k (K)	q_α (e)
wall[a]	2.500	231.55	0
O	3.166	78.24	-0.8476
H	–	–	0.4238
Z^-	6.0	251.58	-1.0
Na^+	2.275	58.01	1.0
K^+	3.023[b]	58.01	1.0
Cs^+	3.884[b]	58.01	1.0

[a] See Ref. 20 for the functional form of the potential between the cavity and interior sites.
[b] LJ parameters taken from Ref. 65

B. Molecular structure and harmonic potential parameters

Headgroup-wall: $b_e = 2.5$ Å, $k_e = 600$ kcal mol^{-1}Å^{-2}, **Water:** $R_{OH} = 1.0$ Å, $\angle_{HOH} = 109.5°$.

Table 3.4. Composition and size of model reverse micelles

Counterion	w_0	n_{water}	$n_{surfactant}$	$n_{counterion}$	Radius (Å)
Na^+	1	21	21	21	10.25
Na^+	2.5	73	29	29	12.5
Na^+	5	211	42	42	15.40
K^+	5	211	42	42	15.41
Cs^+	5	211	42	42	15.55

3.III.B. Evaluation of Intermediate Scattering Functions

We use the RM model to calculate $F_S(\mathbf{Q},t)$, which is simply related to the signals that we can measure in a QNS experiment through Eq.(3.1) . Our calculations focus on the Q values within the experimentally-accessible range. In addition to a direct comparison with experiment, we can use the simulation to explore details of the water dynamics that are not readily accessible from experiments. For example, we can evaluate the contributions to $F_S(\mathbf{Q},t)$ from translational and rotational dynamics separately by calculating the intermediate scattering functions for the center of mass (F_S^{CM}) of water molecules and for hydrogen motion about the center of mass (F_S^R). The expressions for these functions can be written by defining the position **r** of a water hydrogen atom as:

$$\mathbf{r} = \mathbf{r}_{CM} + \mathbf{d}, \tag{3.13}$$

where **d** is the distance of the hydrogen atom from the molecular center-of-mass, $\mathbf{r}_{CM}$. In a rigid water model such as SPC/E, the molecular geometry is fixed and **d** can therefore change only through molecular rotation.

The intermediate scattering function for a hydrogen atom in a water molecule is then

$$F_S(\mathbf{Q},t) = \left\langle \exp\left\{i\mathbf{Q}\cdot[\mathbf{r}_{CM}(0)-\mathbf{r}_{CM}(t)]\right\} \exp\left\{i\mathbf{Q}\cdot[\mathbf{d}(0)-\mathbf{d}(t)]\right\}\right\rangle. \tag{3.14}$$

In the analysis of experimental data, the decoupling approximation is usually assumed to apply. In this approximation,

$$F_S(\mathbf{Q},t) \cong F_S^P(\mathbf{Q},t) = F_S^{CM}(\mathbf{Q},t) F_S^R(\mathbf{Q},t), \tag{3.15}$$

where the translational and rotational intermediate scattering functions are given by

$$F_S^{CM}(\mathbf{Q},t) = \left\langle \exp\left\{i\mathbf{Q}\cdot[\mathbf{r}_{CM}(0)-\mathbf{r}_{CM}(t)]\right\}\right\rangle \tag{3.16}$$

and

$$F_S^R(\mathbf{Q},t)=\left\langle \exp\left\{i\mathbf{Q}\cdot\left[\mathbf{d}(0)-\mathbf{d}(t)\right]\right\}\right\rangle. \tag{3.17}$$

From a comparison between the product approximation, $F_S^P(\mathbf{Q},t)$, and $F_S(\mathbf{Q},t)$ calculated from Eq.(3.15), we can determine the range of validity of this approximation for the $\mathbf{Q}$ and t values relevant to QNS experiments.

The rotational intermediate scattering function, $F_S^R(\mathbf{Q},t)$ can be written in terms of the Rayleigh expansion the exponential[60]

$$F_S^R(\mathbf{Q},t)=\left[j_0(Qd)\right]^2+\sum_{l=1}^{\infty}(2l+1)\left[j_l(Qd)\right]^2C_l(t), \tag{3.18}$$

where $C_l(t)$ is the time correlation for the Legendre polynomial P_l of order l of the change in orientation of vector $\mathbf{d}$:

$$C_l(t)=\left\langle P_l\left[\hat{\mathbf{d}}(0)\cdot\hat{\mathbf{d}}(t)\right]\right\rangle, \tag{3.19}$$

where $\hat{\mathbf{d}}=\mathbf{d}/d$ is a unit vector along $\mathbf{d}$.

MD simulation can be used to evaluate $F_S^R(\mathbf{Q},t)$ directly from Eq. (3.18) or by evaluating the individual orientational time correlations (TCFs) $C_l(t)$ and then using Eq.(3.19). In the relevant range of Q ($\leq 3\text{Å}^{-1}$), a small number of terms in the sum over l is needed for convergence.

The behavior of $C_l(t)$'s is easier to interpret and analyze in terms of rotational relaxation models than that of $F_S^R(\mathbf{Q},t)$. For example, it allows us to test to what extent rotational relaxation is diffusive and to what extent the relaxation rate of the vector $\mathbf{d}$ resembles that other vectors related to the orientation of water molecules. One such

vector is the unit vector $\hat{\mathbf{u}}$ along the molecular dipole. We use a comparison of $C_l(t)$ with

$$C_{l\mu}(t) = \left\langle P_l\left[\hat{\mathbf{u}}(0)\cdot\hat{\mathbf{u}}(t)\right]\right\rangle \tag{3.20}$$

to investigate the extent of anisotropy of water orientational relaxation.

Experimental QNS results represent an average over all hydrogen atoms within the RMs of a given w_0 value, as indicated by Eq.(3.2) . The dynamics within the RMs for w_0 = 5 on the 15 ps time scale are heterogeneous, with the mobility of water molecules in the interfacial region reduced relative to the core.[20] We use our MD data to determine how these differences in mobility affect $F_S(\mathbf{Q},t)$ by separating it into contributions from different concentric shells.

IIIC. Effects of RM Size on Water Dynamics

Our QNS data are analyzed using the assumption that rotational and translational contributions to $F_S(\mathbf{Q},t)$ are not correlated, i.e., that the product approximation applies. Our MD data can be used to test the limitations of the product approximation, Eq.(3.15) . The results of this test for w_0 = 2.5 (top) and 5 (bottom) at Q = 1, $\sqrt{2}$ and $\sqrt{3}$ Å^{-1} are shown in Fig. 3.4. As can be seen from the figure, the overall agreement is very good. Differences between $F_S(\mathbf{Q},t)$ and $F_S^P(\mathbf{Q},t)$ increase with increasing Q and w_0. The trends with Q parallel those found for bulk water.[57] and the trends with w_0 show that the decoupling approximation is somewhat better in a more confined system with a larger fraction of molecules near the interface. Our tentative explanation for this is that water-water hydrogen bonding is suppressed near the interface,[20] allowing greater freedom of

rotational motion of the O-H vectors, while translational mobility is impeded by electrostatic interactions of water with the AOT head groups and Na^+ ions.

Given that $F_S(\mathbf{Q},t)$ is, to a good approximation, a product of translational and rotational contributions, we proceed now to illustrate how these contributions, $F_S^{CM}(\mathbf{Q},t)$ and $F_S^R(\mathbf{Q},t)$, vary with Q and w_0. Figures 3.5, 3.6 and 3.7 depict, respectively the results for $F_S^R(\mathbf{Q},t)$, $F_S^{CM}(\mathbf{Q},t)$ and $F_S(\mathbf{Q},t)$ at Q values of 1, $\sqrt{2}$ and $\sqrt{3}$ Å^{-1} for w_0 = 1 (top), 2.5 (middle) and 5 (bottom).

As expected, the intermediate scattering functions decay more rapidly at higher Q values which sample hydrogen atom motion on a smaller distance scale. At each Q, faster relaxation is seen for water in larger RMs. This is also expected from experimental data on real RMs and on the basis of the results on water mobility in the present RM model obtained by Faeder and Ladanyi.[20] $F_S^{CM}(\mathbf{Q},t)$, depicted in Fig. 3.6, does not decay exponentially. $F_S^{CM}(\mathbf{Q},t)$ decays to a constant, which is a consequence of the fact that translational diffusion restricted to a finite nanoscopic volume.[59] Nonexponential decay that we observe for $F_S^{CM}(\mathbf{Q},t)$ is to some extent a consequence of this fact. As we shall see, nonexpronential decay is also related to the differences in water mobility in interfacial and core regions of the reverse micelles.

Because the hydrogen atoms in $F_S^R(\mathbf{Q},t)$, Eq.(3.18) , are constrained to a sphere of radius d, the rotational intermediate scattering function does not decay to zero, but instead reaches the value,

$$F_S^R(Q,\infty) = [j_0(Qd)]^2 \tag{3.21}$$

in the long time limit.

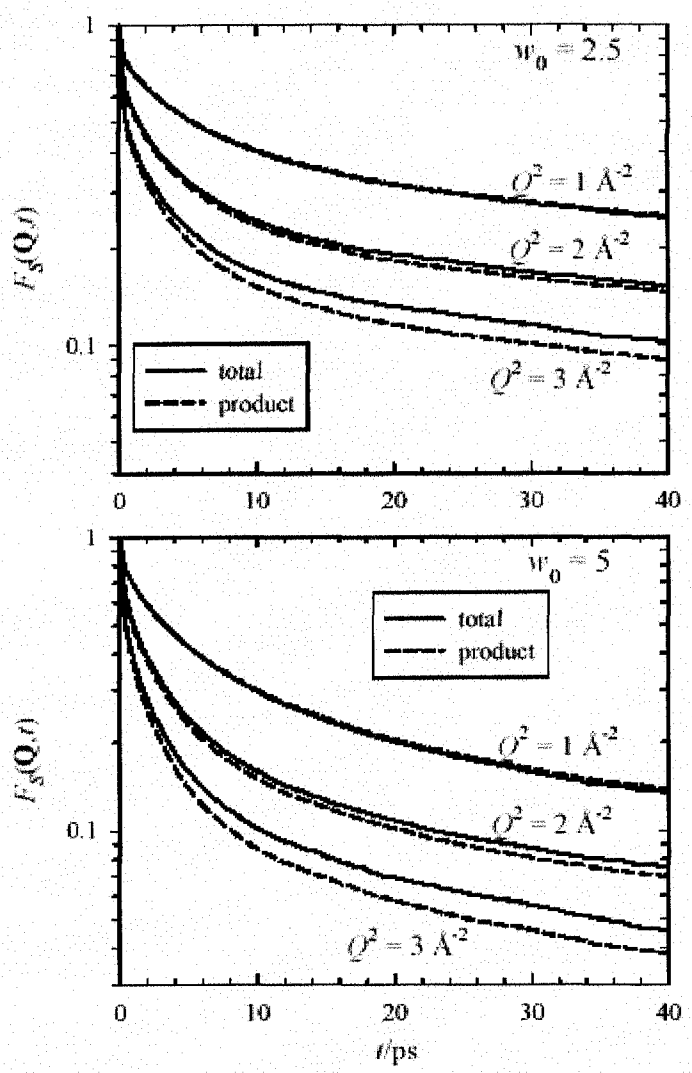


Figure 3.4. The comparison of the MD results for the intermediate self-scattering function, $F_S(\mathbf{Q},t)$ (full line) and the approximate form of this function obtained by representing it as the product $F_S^P(\mathbf{Q},t) = F_S^{CM}(\mathbf{Q},t)F_S^R(\mathbf{Q},t)$ (dashed line) of the rotational and translational ISF's at three Q values, shown as labels of line pairs. The results for w_0 = 2.5 are depicted in the top panel and the results for w_0 = 5 in the bottom panel.

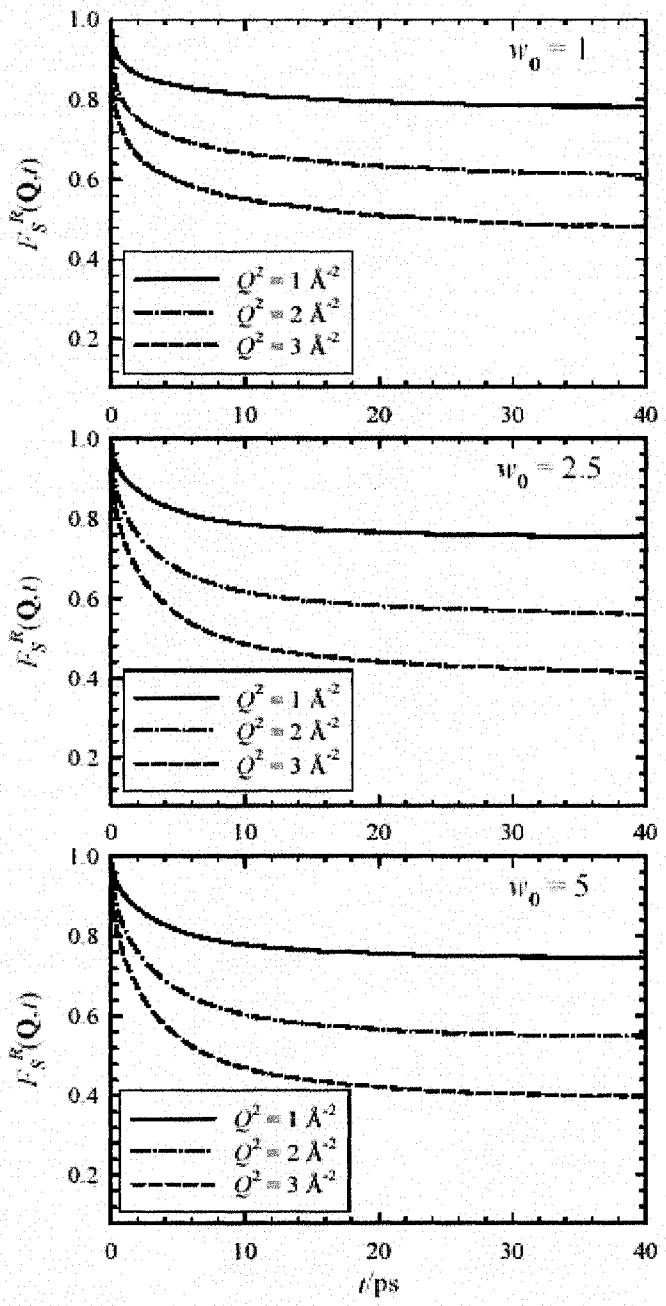


Figure 3.5. The MD results for the rotational component of the intermediate self-scattering function, $F_S^R(\mathbf{Q},t)$, at three Q values, depicted as different line styles, for each of the three RM sizes, depicted in different panels $w_0 = 1$ (top), $w_0 = 2.5$ (middle) and $w_0 = 5$ (bottom).

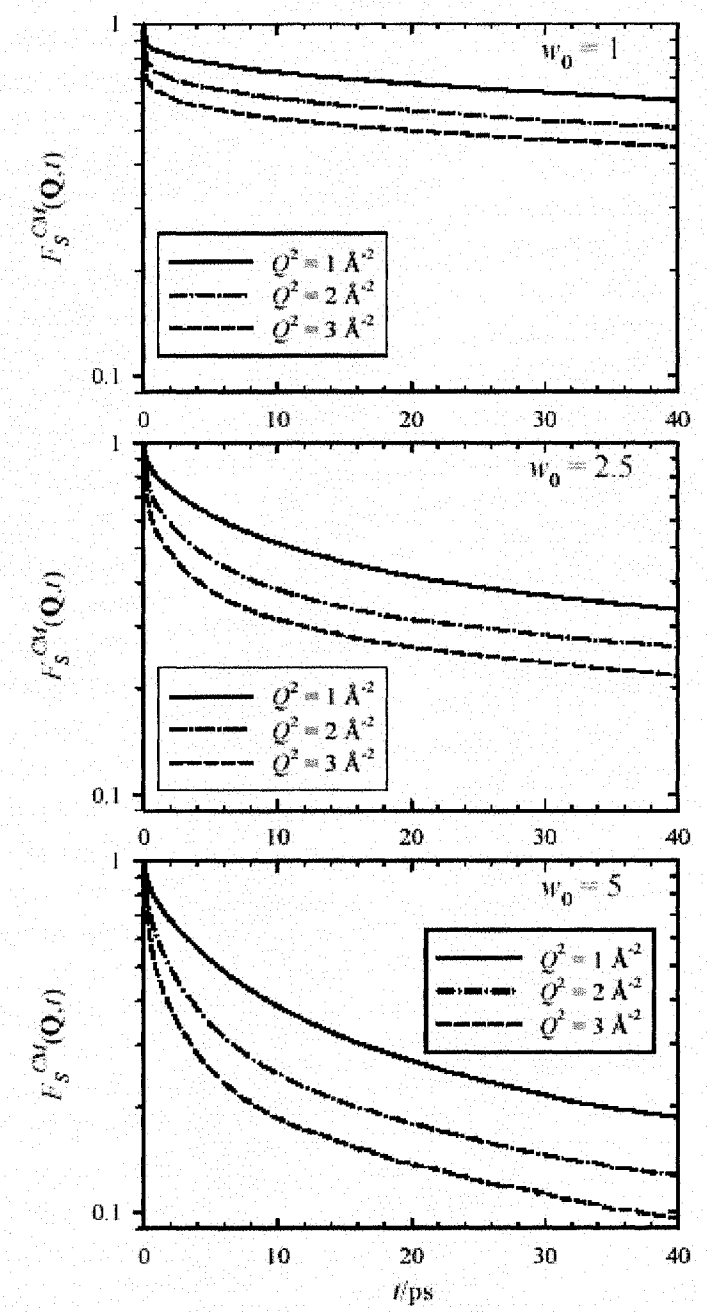


Figure 3.6. The MD results for the translational component of the intermediate self-scattering function $F_S^{CM}(\mathbf{Q},t)$ at three Q values, depicted as different line styles, for each of the three RM sizes, depicted in different panels: $w_0 = 1$ (top), $w_0 = 2.5$ (middle) and $w_0 = 5$ (bottom).

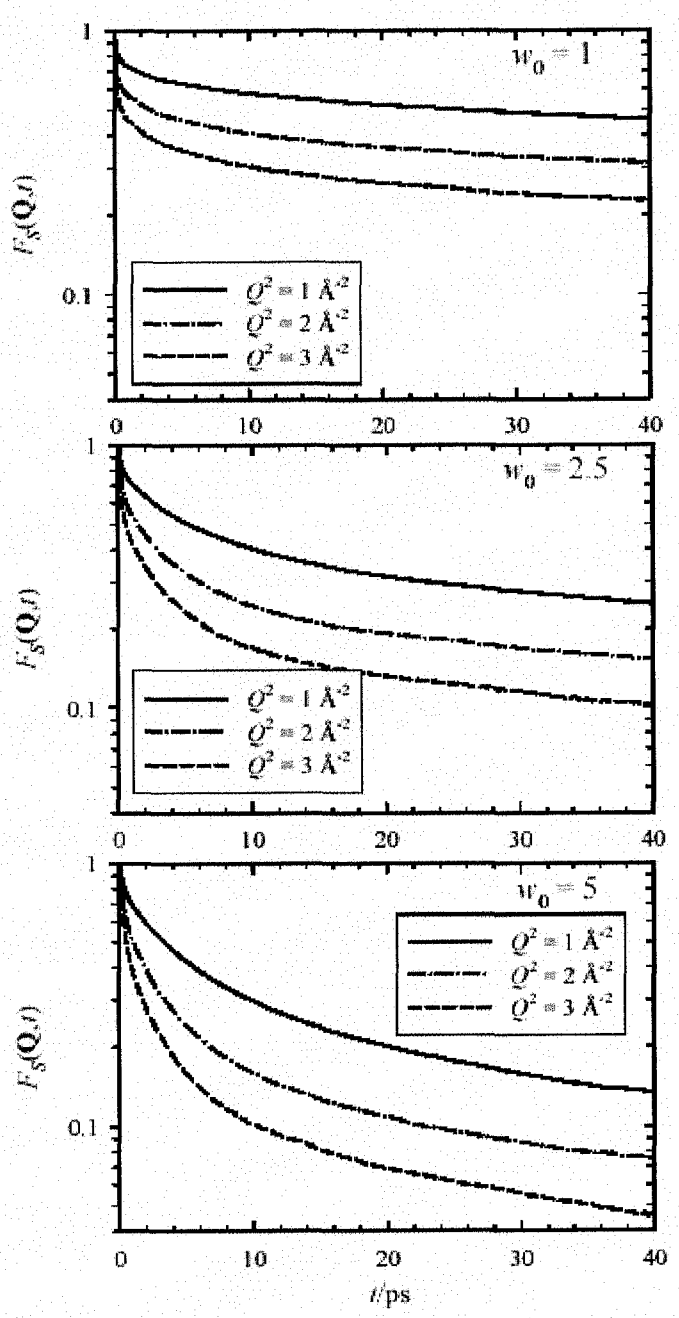


Figure 3.7. The MD results for the total intermediate self-scattering function, $F_S(\mathbf{Q},t)$, at three Q values, depicted as different line styles, for each of the three RM sizes, depicted in different panels: $w_0 = 1$ (top), $w_0 = 2.5$ (middle) and $w_0 = 5$ (bottom).

Given that d = 0.9642 Å in the SPC/E water model, $F_S^R(\mathbf{Q},\infty)$ values at Q = 1, $\sqrt{2}$ and $\sqrt{3}$ Å^{-1} are 0.7261, 0.5151 and 0.3551. We see from Fig. 3.5 that for w_0 = 2.5 and 5, $F_S^R(\mathbf{Q},t)$ has essentially reached this limit by t = 40 ps, while for w_0 = 1, the values of $F_S^R(\mathbf{Q},t)$ are quite different from their $t \to \infty$ limits, indicating that in this smallest RM the rotational mobility of water is substantially reduced compared with the bulk and even with the two larger RMs.

The results for the total $F_S(\mathbf{Q},t)$, shown in Fig. 3.7, display the trends with increasing Q and w_0 that can be anticipated from the behavior of $F_S^{CM}(\mathbf{Q},t)$ and $F_S^R(\mathbf{Q},\infty)$, given that $F_S(\mathbf{Q},t)$ is approximately their product, although the product approximation is not used in the results displayed in Fig. 3.7.

The mobility of water near hydrophilic surfaces is often found to be reduced relative to the bulk.[49,70] For the present RM model, water mobility has been shown to depend on its proximity to the interface as well as on the water content, since the surface area per head group increases with w_0.[20] Water oxygen density profile relative to the interface, i.e., as a function of D, the distance from the average headgroup locations, shown in Fig. 3.8, exhibits structural features which allow us to define several interfacial layers. The peak at around D = 0 corresponds to water trapped in-between the headgroups, the peak at around 3 Å corresponds to water bound to the inner surface of the headgroups. Structure at larger D values is w_0-dependent, becoming more diffuse as w_0 increases. The heights of the first and second peaks increase with w_0, indicating higher extent of water penetration into the interface as the surface area per headgroup increases.

Since 'trapped' water corresponds to a structural feature that is well-separated from the remainder of the density profile and since water molecules in this region do not exchange readily with water in the core regions,[20] we consider separately the contributions to $F_S(\mathbf{Q},t)$ from water molecules that are initially within $D \leq 1.5\text{Å}$ (interface) and $D > 1.5\text{Å}$ (core). These regions are illustrated in Fig. 3.8. The results of this layer analysis for $F_S^{CM}(\mathbf{Q},t)$ and $F_S^{R}(\mathbf{Q},t)$ at Q = 1 Å^{-1} and all three w_0 values are shown in Figs. 3.8 and 3.9. In the bottom panel of Fig. 3.9, $F_S^{CM}(\mathbf{Q},t)$ for bulk water is displayed in addition to the w_0 = 5 data.

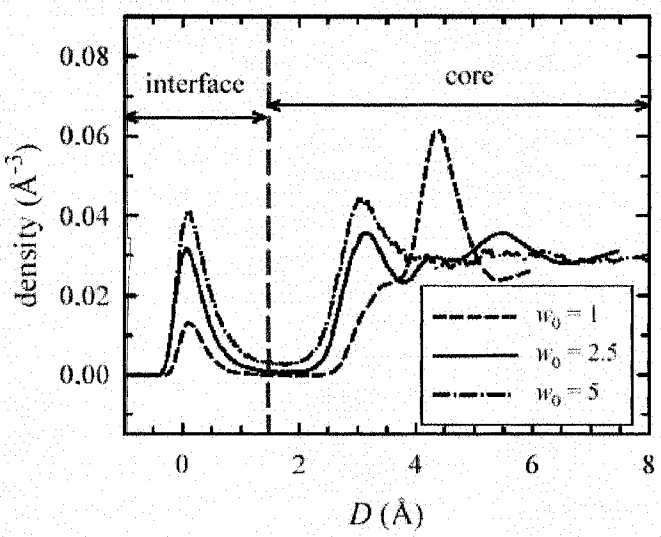


Figure 3.8. Water oxygen radial density profiles vs. D, the distance from the average position of the head groups, for the three RM sizes, depicted using different line styles. The definition of the interface and core regions used in our subsequent calculations is indicated on the graph.

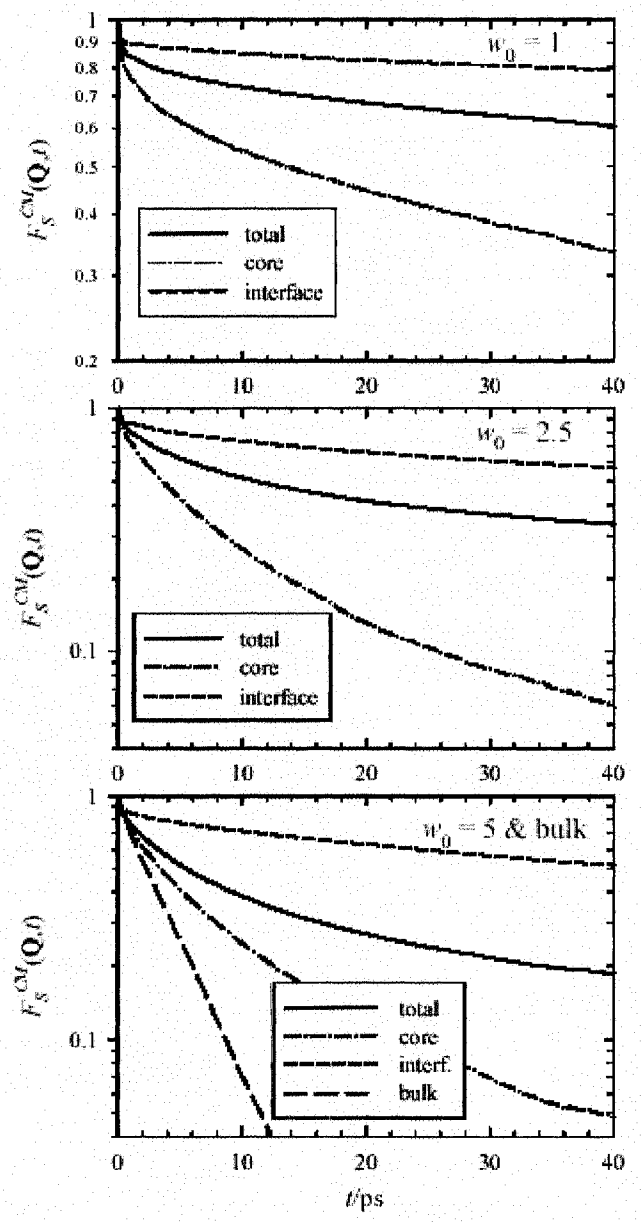


Figure 3.9. The total (full line) translational component $F_S^{CM}(\mathbf{Q},t)$ of the intermediate scattering function at $Q = 1.00$ Å^{-1} and the contributions to it from the core (dash-dotted line) and interface (dashed line) regions (see Fig. 3.8 for the definitions of these regions). for each of the three RM sizes, depicted in different panels: $w_0 = 1$ (top), $w_0 = 2.5$ (middle) and $w_0 = 5$ (bottom). The bottom panel also includes the $F_S^{CM}(\mathbf{Q},t)$ results at $Q = 1.02$ Å^{-1} for bulk water (long dashes).

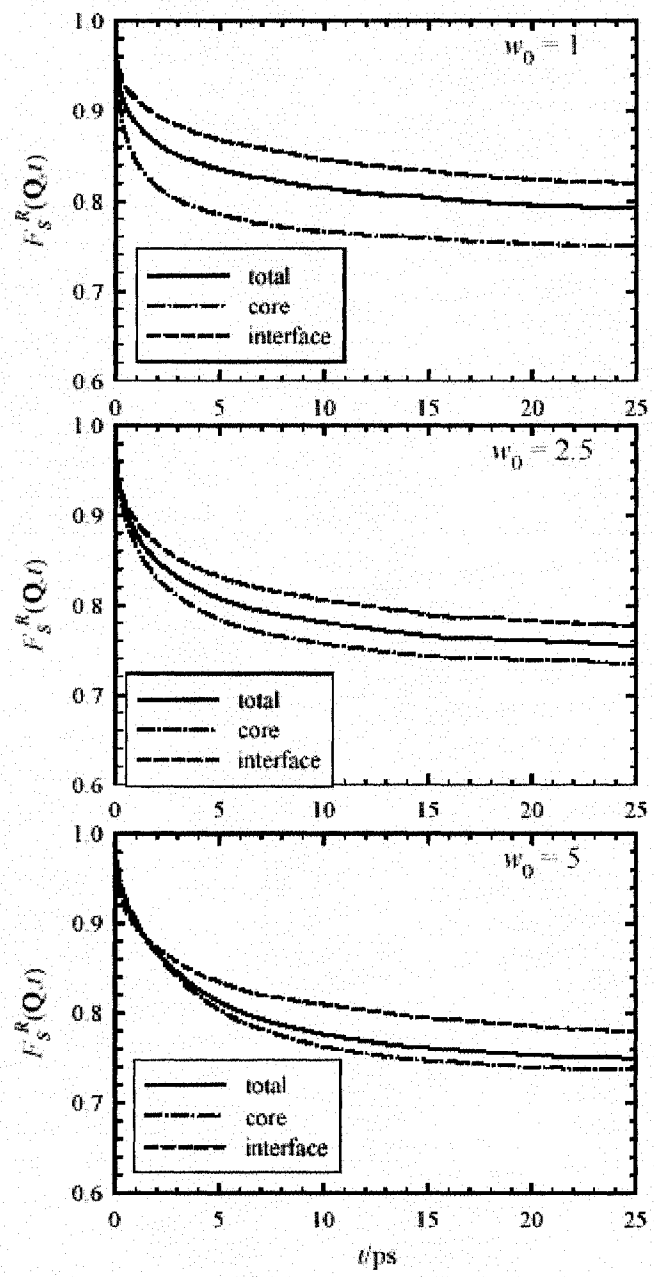


Figure 3.10. The total (full line) rotational component $F_S^R(\mathbf{Q},t)$ of the intermediate scattering function at $Q = 1.00$ Å^{-1} and the contributions to it from the core (dash-dotted line) and interface (dashed line) regions (see Fig. 3.8 for the definitions of these regions). for each of the three RM sizes, depicted in different panels: $w_0 = 1$ (top), $w_0 = 2.5$ (middle) and $w_0 = 5$ (bottom).

Considering first Fig. 3.9, we see that water mobility in each of the two RM regions (interface and core) increases with increasing w_0. In all three RMs, the mobility is considerably higher in the core than in the interface region. However, even in the largest of the three RMs, the core translational mobility is considerably lower than in bulk water. Considering the w_0-dependence of $F_S^{CM}(\mathbf{Q},t)$ in a given region, the differences are smaller for w_0 = 2.5 and 5 than when one compares w_0 = 1 with w_0 = 2.5.

In addition to the mobilities in each of the two regions, the overall translational mobility within the RM, represented by the total $F_S^{CM}(\mathbf{Q},t)$, depends on the relative populations of water molecules in the interface and core regions. Thus $F_S^{CM}(\mathbf{Q},t)$ decays considerably more slowly in w_0 = 2.5 than in w_0 = 5, despite the fact that translational mobilities in each region are only slightly lower in the w_0 = 2.5 RM.

Turning now to $F_S^R(\mathbf{Q},t)$, shown in Fig. 3.10, we see that for all RMs, the interface contribution decays more slowly than the core contribution. It also appears that the interface component is appreciably further than the core component from reaching the $t \to \infty$ plateau value (Eq.(3.21)), which is 0.7261 at Q = 1 Å^{-1}, within the time interval of 25 ps shown in the figure. In all three RMs, $F_S^R(\mathbf{Q},t)$ in the core is quite close to this limit at t = 25 ps. As w_0 increases, the relative population of water molecules increases. This is reflected in the fact that the total $F_S^R(\mathbf{Q},t)$ becomes more rapidly relaxing and core-like as w_0 increases.

As Eq. (3.18) indicates, $F_S^R(\mathbf{Q},t)$ represents a weighted sum of a baseline and orientational time correlations, $C_l(t)$. In order to determine to what extent the isotropic

rotational diffusion model applies to rotational relaxation of water within RMs, we need to examine the behavior of $C_l(t)$'s. We first test the assumptions that $C_l(t)$ decays exponentially and that rotational diffusion is isotropic, concentrating on leading term in series, $C_1(t)$, which is displayed on semi-log scale in the top panel of Fig. 3.11, for bulk water and RMs corresponding to $w_0 = 2.5$ and 5.

To test the extent of anisotropy in water reorientation, we display in the bottom panel of Fig. 3.11, $C_{1,\mu}(t)$ for the same three systems. Comparing the data for the RMs to those for bulk water, we see that the RM data for both $C_1(t)$ and $C_{1,\mu}(t)$ deviate strongly from exponential decay, while the bulk water data decay exponentially for most of the time interval displayed, i.e., for $t > 0.2$ ps. In the case of bulk water, the decay rates are quite close for the center-of-mass to H unit vector $\hat{\mathbf{d}}$, which is nearly parallel to the O-H bond vector, and the dipole unit vector $\hat{\mathbf{u}}$. However, for the two RMs, the decay rates for $C_1(t)$ and $C_{1,\mu}(t)$ are quite different, with $C_{1,\mu}(t)$ exhibiting markedly slower relaxation. Thus rotational relaxation within RMs is considerably less isotropic than in bulk water. While $C_{1,\mu}(t)$ in RMs is slower than in bulk water over the entire time interval displayed, $C_1(t)$ decays faster in RMs than in the bulk at short times, eventually becoming slower at longer times.

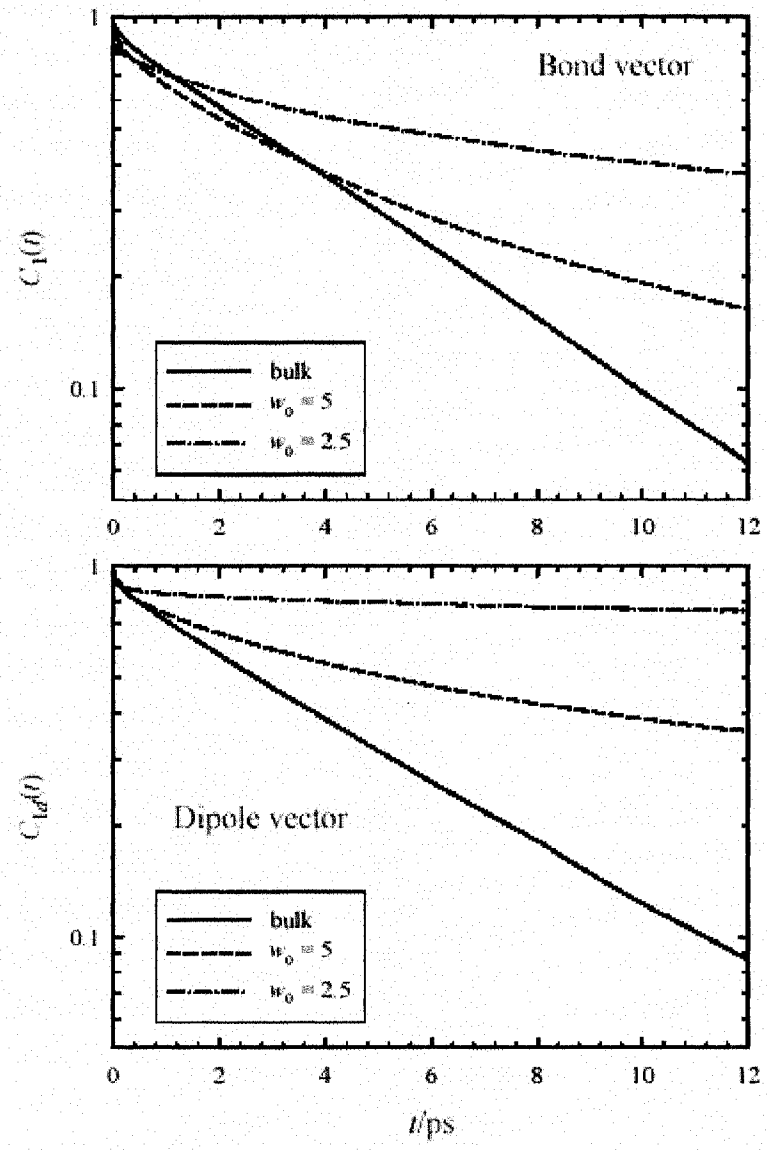


Figure 3.11. Orientational time correlations of the 1st order Legendre polynomial of the vector connecting the H atom to the molecular center-of-mass $C_1(t)$ and the molecular dipole $C_{1d}(t)$ for bulk water (full line) and water in reverse micelles, w_0 =5 (dashed line) and $w_0 = 2.5$ (dash-dotted line).

The most likely reason for the relatively fast decay $C_1(t)$ at short times and for the of more anisotropic rotational relaxation in RMs than in the bulk is that water molecules near the RM interface participate in fewer water-water H-bonds than do water molecules in the core.[20] Furthermore, these molecules are instead coordinated to the Na^+ counterions and, to a smaller extent to the headgroups.[20] Coordination to the counterions slows down reorientation of the water dipoles. However, for the same water molecules, motion of the O-H vectors around a fixed dipole orientation is less hindered than in the bulk in view of the fact that waters coordinated to counterions participate in fewer H-bonds.

Another aspect of the rotational diffusion model is tested in Figure 3.12. The figure depicts our MD results for $\ln C_l(t)/[l(l+1)]$ at l = 1, 2 and 3 for bulk water (top) the $w_0 = 5$ RM (bottom). According to the rotational diffusion model, $C_l(t)$ is given by Eq.(2.18), which predicts that $\ln C_l(t)/[l(l+1)]$ should be an l-independent straight line. Fig. 3.12 indicates that $\ln C_l(t)/[l(l+1)]$ is l-dependent in both systems, exhibiting slower decay as l increases. This behavior is typical of high-torque liquids[71] and is expected for water. In bulk water, $\ln C_l(t)/[l(l+1)]$ becomes linear beyond the short inertial-librational time scale, depicted at higher resolution in the insert. In the RM, the long-time decay rate of $\ln C_l(t)/[l(l+1)]$ decreases with increasing t, typical of stretched-exponential decay. Thus in this respect the rotational diffusion assumption is worse for RMs than for bulk water. Note that the w_0 = 5 RM has the largest core region of the three RMs studied here and that we therefore expect it to be the most bulk-like.

Inserts in Fig. 3.12 depict the behavior of $\ln C_l(t)/[l(l+1)]$ at short times, $0 \leq t \leq 0.4\,\text{ps}$, emphasizing the inertial-librational portion of $C_l(t)$. The comparison of the RM and bulk results shows that the initial inertial portion of $C_l(t)$ is larger and the amplitude of the subsequent oscillatory feature, due to O---H-O libration is smaller in RMs than in bulk water, leading to an overall faster decay of $C_l(t)$ in this time interval. This lends further support to our earlier explanation for the faster short-time decay of unit vector $\hat{\mathbf{d}}$ orientational correlations in RMs than in the bulk.

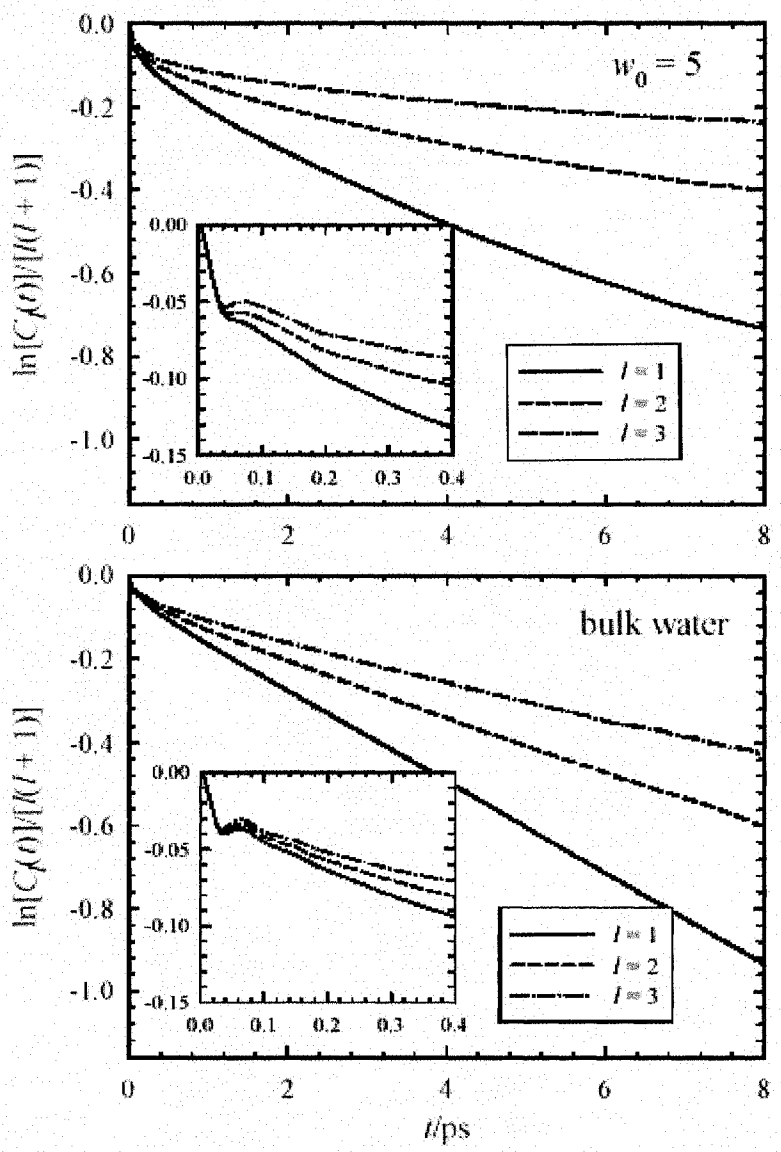


Figure 3.12. Orientational time correlations of l^{th} order Legendre polynomials ($l = 1$ (full line, $l = 2$ (dashed line) and $l = 3$ (dash-dotted line)) of the vector connecting the H atom to the molecular center-of-mass for water in the $w_0 = 5.0$ reverse micelle (top) and bulk liquid (bottom). Depicted are $\ln C_l(t)/[l(l+1)]$ vs t, which would be l-independent straight lines within the rotational diffusion model.

3.III.D.Effects of Surfactant Counterion on Water Structure and Dynamics

Figure 3.13 shows representative snapshots of three w_0=5 reverse micelles with Na^+, K^+, and Cs^+ counterions. These snapshots show that the majority of counterions reside on the surface of the RMs, where they have favorable interactions with the negatively charged surfactant headgroups. The counterions at the interface also appear to be highly coordinated with negatively charged headgroups, yielding a lattice-like structure, which will affect the mobility of water in the region. The large Cs^+ counterions occupy more surface area and exclude water molecules from the interface to a greater extent than smaller K^+ and Na^+ counterions.

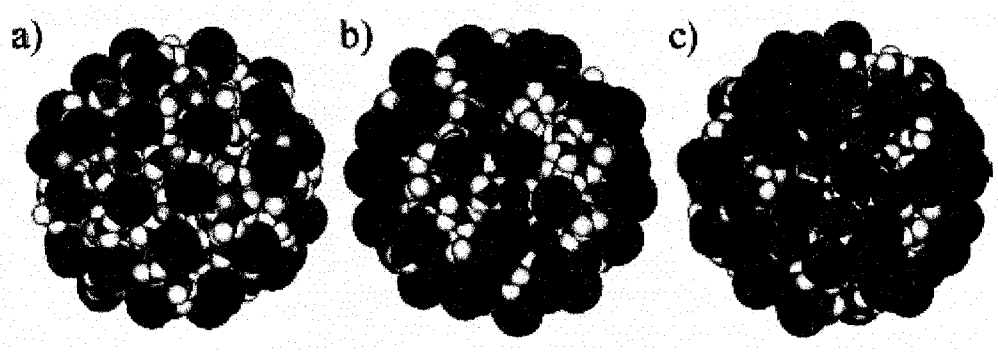


Figure 3.13. Snapshots of w_0 = 5 RMs with Na^+ (a), K^+ (b), and Cs^+ (c) AOT reverse micelles. Surfactant headgroups are shown in dark blue, water hydrogen atoms in white, and water oxygen atoms in red. Na^+, K+, and Cs^+ counterions are shown in yellow, green, and turquoise, respectively.

Radial density profiles of water oxygen atoms (left) and surfactant counterions (right) are shown in Figure 3.14. The density profiles are plotted as a function of the distance D from the equilibrium position of the surfactant headgroups. The average headgroup positions are at a distance of 2.5 Å from the nonpolar cavity wall. These figures show that the basic structural features for water and counterions are similar in all three RMs, but that their intensities and shapes in the interfacial region are quite different. In the case of water, the first peak height decreases with increasing counterion size, indicating that the larger counterions displace water molecules to a greater extent from the interfacial region. This decrease in the height of the first peak of water density is accompanied by an increase in the first peak in the counterion density at the interface. Also, the two larger counterions exhibit a single interfacial peak, while Na^+ exhibits a split peak. The peak splitting is due to ion-water coordination,[19,20] which is stronger for this than the other two counterions.

In our previous study,[22] we identified two regions in the RMs based on the water oxygen atom density profile. The first region, which we denote as *interfacial*, extends from the wall to the first minimum in the density profile, at ~1.5Å. The second region, which we refer to as *core*, extends from this first minimum to the center of the RM. As the counterion size increases, the density of interfacial water decreases with a concomitant increase in the density of core water. The counterion density profile of Na^+ was previously characterized[19] by a sharp peak at 0 Å, corresponding to counterions that are coordinated to surfactant headgroups, and a broader peak near 1 Å corresponding to counterions that are coordinated to water. In contrast to the Na^+ counterions, density profiles of Cs^+ and K^+ exhibit a single peak near ~0.25 Å. This peak is more intense for

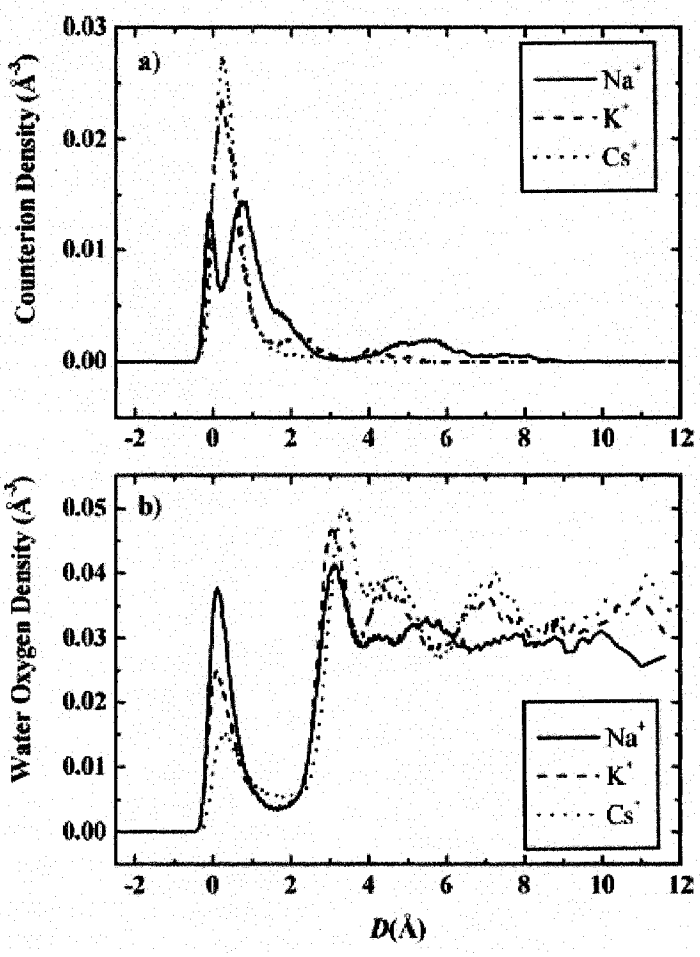


Figure 3.14. Radial density profiles of counterions (a) and water oxygen atoms (b) for $w_0 = 5$ Na^+ (solid), K^+ (dashes), and Cs^+ (dots) AOT RMs. Distance D is relative to the equilibrium position of the surfactant headgroups, which is 2.5 Å from the wall of the cavity.

the large Cs^+ counterion than for K^+. Additionally, the density of counterions dissolved in the water pool decreases as the counterion size increases.

Figure 3.15 displays the average pair density functions (PDF) for water with counterions and headgroups. The pair density $\rho_{\alpha\beta}(r)$ for sites α and β is defined as

$$\rho_{\alpha\beta}(r) = \rho_{\beta} g_{\alpha\beta}(r), \tag{3.22}$$

where ρ_{β} is the average number density of sites of type β, r is the scalar distance between the sites, and $g_{\alpha\beta}(r)$ is the site-site pair distribution function.[20] The figure displays $\rho_{M^+O}(r)$ (M = Na, K, Cs) in the top panel and $\rho_{Z^-H}(r)$ in the bottom panel. As can be seen from the figure, the first peak of $\rho_{Na^+O}(r)$ is more intense and narrower than the corresponding peaks of the K^+-O and Cs^+-O pair densities. This indicates the water coordinates more strongly to the smaller counterions than it does to larger ones. We also see this peak shift to larger separation as the counterion size increases, as expected on the basis of the increase in the effective LJ diameter $\sigma_{M^+O} = (\sigma_{M^+} + \sigma_O)/2$.

In the bottom panel of Fig. 3.15, the intensity of the first peak of $\rho_{Z^-H}(r)$ decreases as the counterion size increases. Hydrogen-headgroup bonding appears strongly suppressed in the systems containing large counterions, even though the water-surfactant potential is unchanged. This is a consequence of the fact, already pointed out in connection with Fig. 3.14, that the large cations pack more efficiently than small cations at the surface of the RM, and hence exclude water to a greater extent.

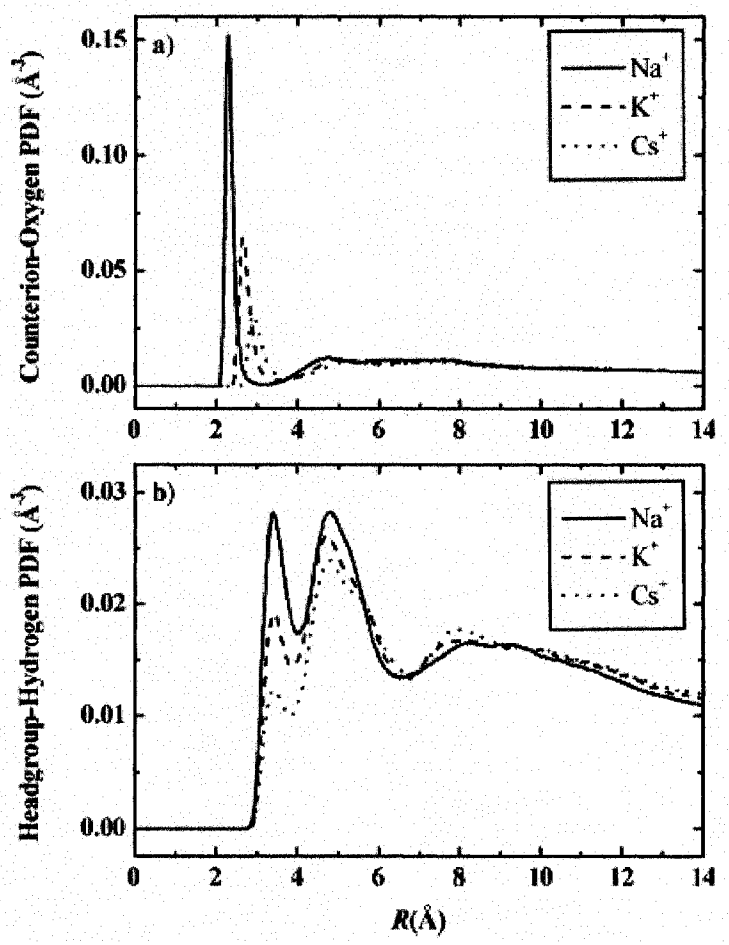


Figure 3.15. Counterion-oxygen (a) and headgroup-hydrogen (b) pair density functions for $w_0 = 5$ Na^+(solid), K^+(dashes), and Cs^+(dots) AOT RMs.

We explore the dynamics of these systems through simulations of the self-intermediate scattering function, $F_S(\mathbf{Q},t)$, of water hydrogen atoms in RMs of varying counterion. Figure 3.17 displays the total intermediate scattering function, $F_S(\mathbf{Q},t)$, for core (a) and interfacial (b) water in Na^+ , K^+, and Cs^+ RMs at selected momentum transfer, Q, values in the range relevant for QNS. At each value of Q, the water mobility decreases with decreasing counterion size, $Cs^+ > K^+ > Na^+$. In addition, the dynamics are faster for water molecules in the RM core than for those residing in the interfacial region. We observe the same trend for the center of mass intermediate scattering function, $F_S^{CM}(\mathbf{Q},t)$, plotted in Figure 3.18, which follows translational motion of the water molecules. We expect that water mobility would increase with increasing counterion size based on previous work[19,22] and because the density profiles suggest that there is relatively less interfacial water as the counterion size increases. Since interfacial water is more hindered that core water, the dynamics should be faster as the relative fraction of core water increases.

Figure 3.19 shows the rotational intermediate scattering function, $F_S^R(\mathbf{Q},t)$, for core (Fig. 3.19a) and interfacial (Fig. 3.19b) water in Na^+, K^+, and Cs^+ RMs at selected Q's. Core dynamics appear faster than interfacial dynamics, but it appears that counterions have a much smaller effect on the rotational motion of water than they do on the translational motion.

As we have seen in the previous section, once the molecule has rotationally relaxed, i.e., the orientational TCFs $C_l(t)$ have decayed to zero, $F_S^R(\mathbf{Q},t)$ will decay to a constant which depends Q and d (Eq.(2.21)). For a SPC/E water molecule, d is equal to

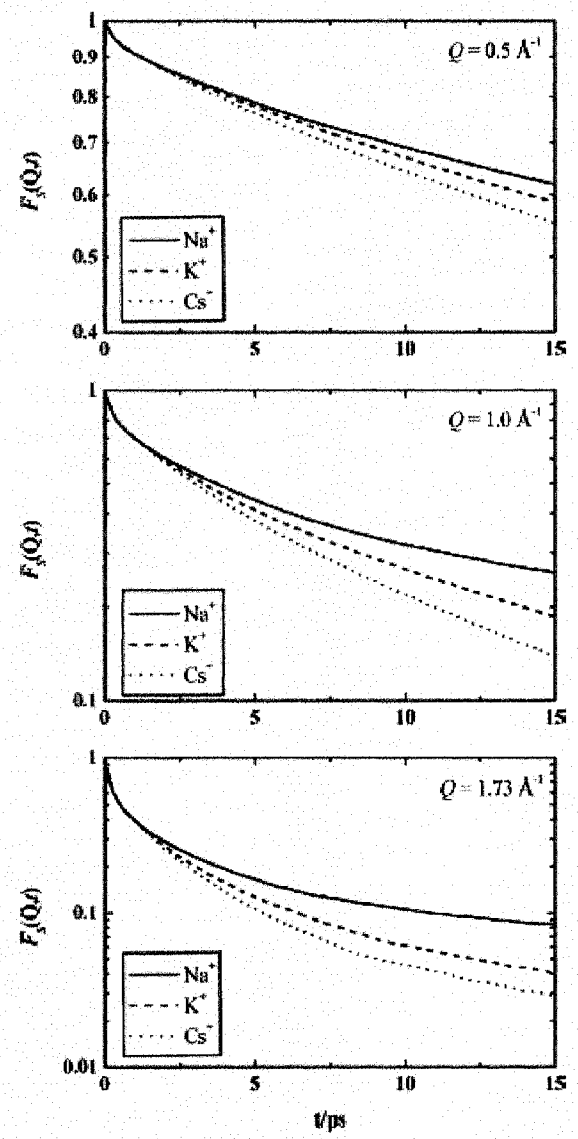


Figure 3.16. Total intermediate scattering function $F_S(\mathbf{Q},t)$ for water in w_0 = 5 Na^+ (solid), K^+ (dashes), and Cs^+ (dots) AOT RMs. Representative Q values 0.5 (top), 1.0 (middle), and 1.73 (bottom) $Å^{-1}$ are shown.

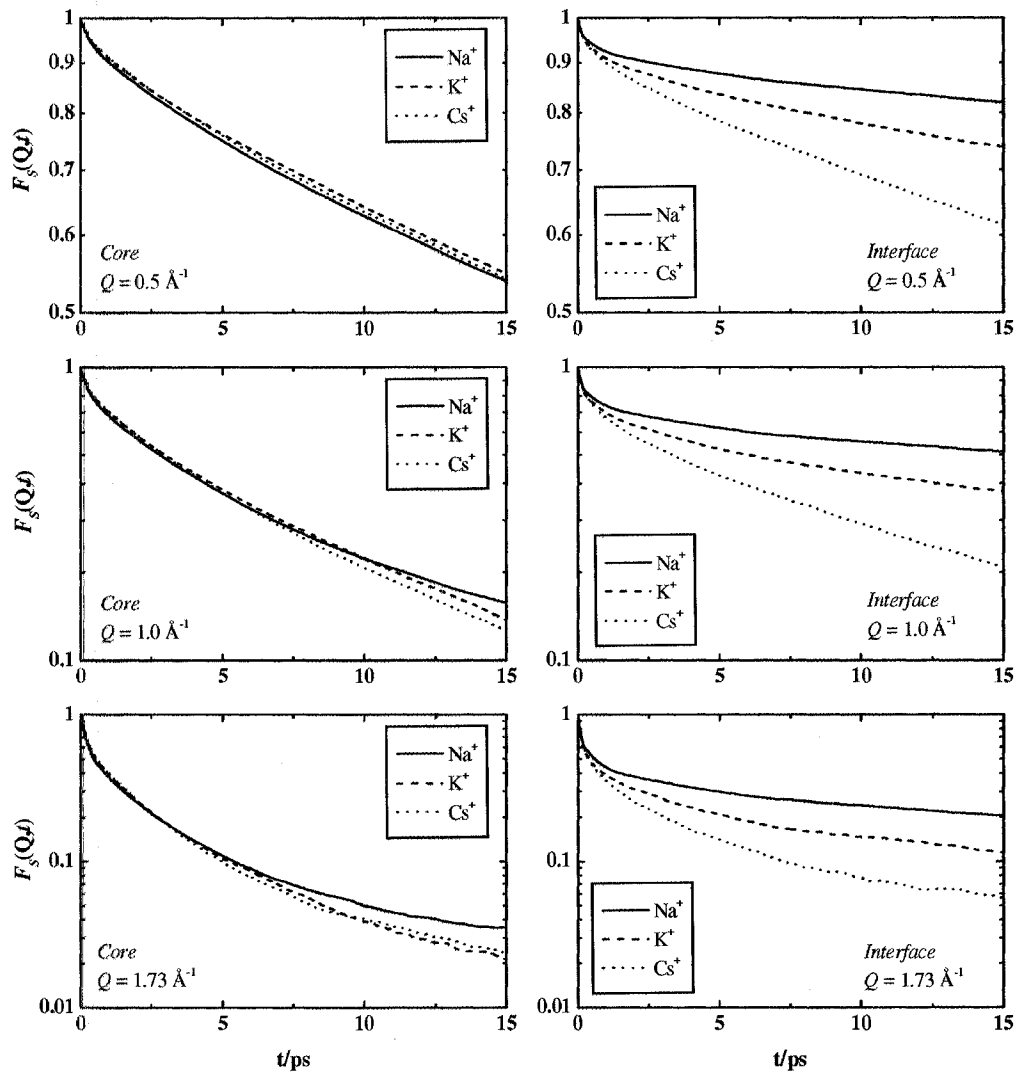


Figure 3.17. Total intermediate scattering function $F_S(\mathbf{Q},t)$ for core (left) and interfacial (right) water in $w_0 = 5$ Na^+(solid), K^+(dashes), and Cs^+(dots) AOT RMs. Representative Q values 0.5 (top), 1.0 (middle), and 1.73 (bottom) $Å^{-1}$ are shown.

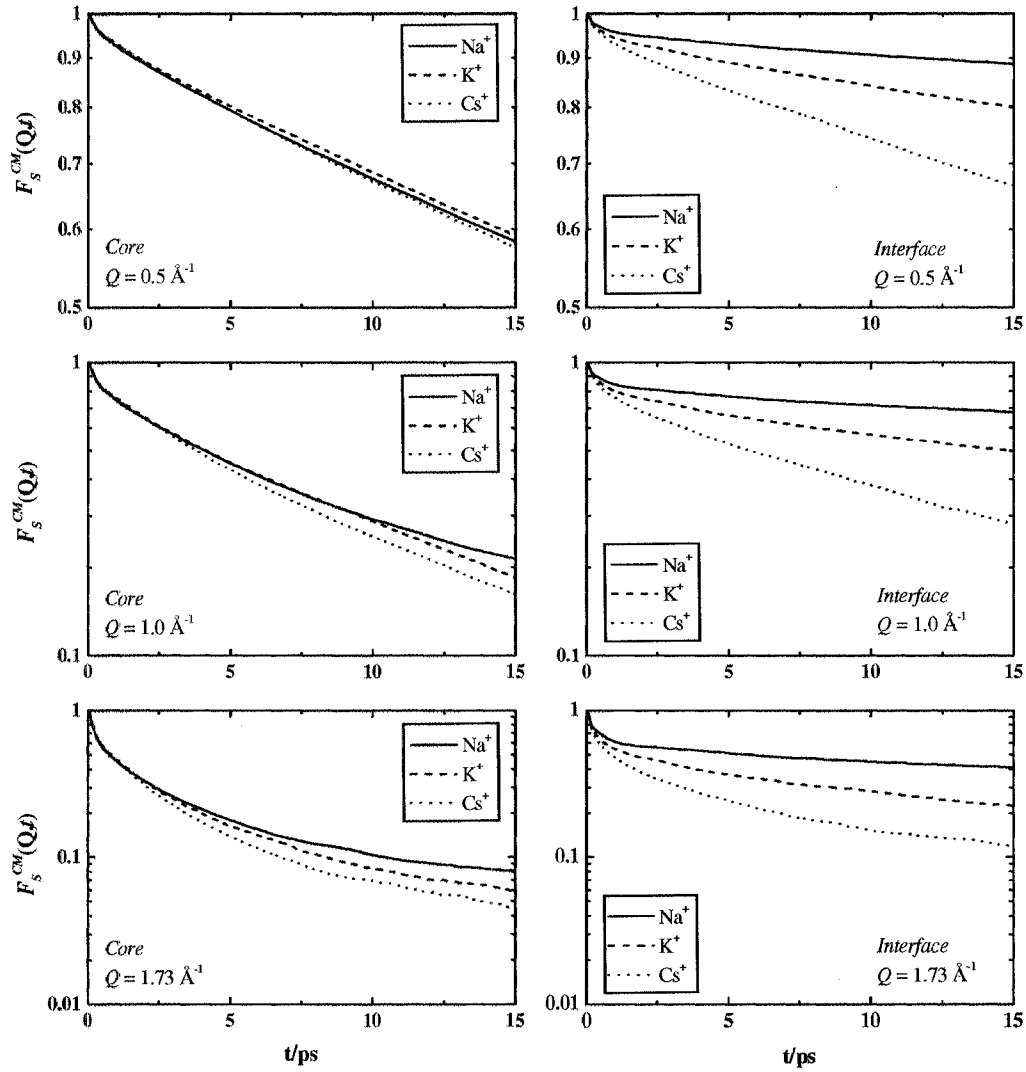


Figure 3.18. Translational intermediate scattering function $F_s^{CM}(Q,t)$ for core(left) and interfacial(right) water in w_0 = 5 Na^+(solid), K^+(dashes), and Cs^+(dots) AOT RMs. Representative Q values 0.5(top), 1.0(middle), and 1.73(bottom) $Å^{-1}$ are shown.

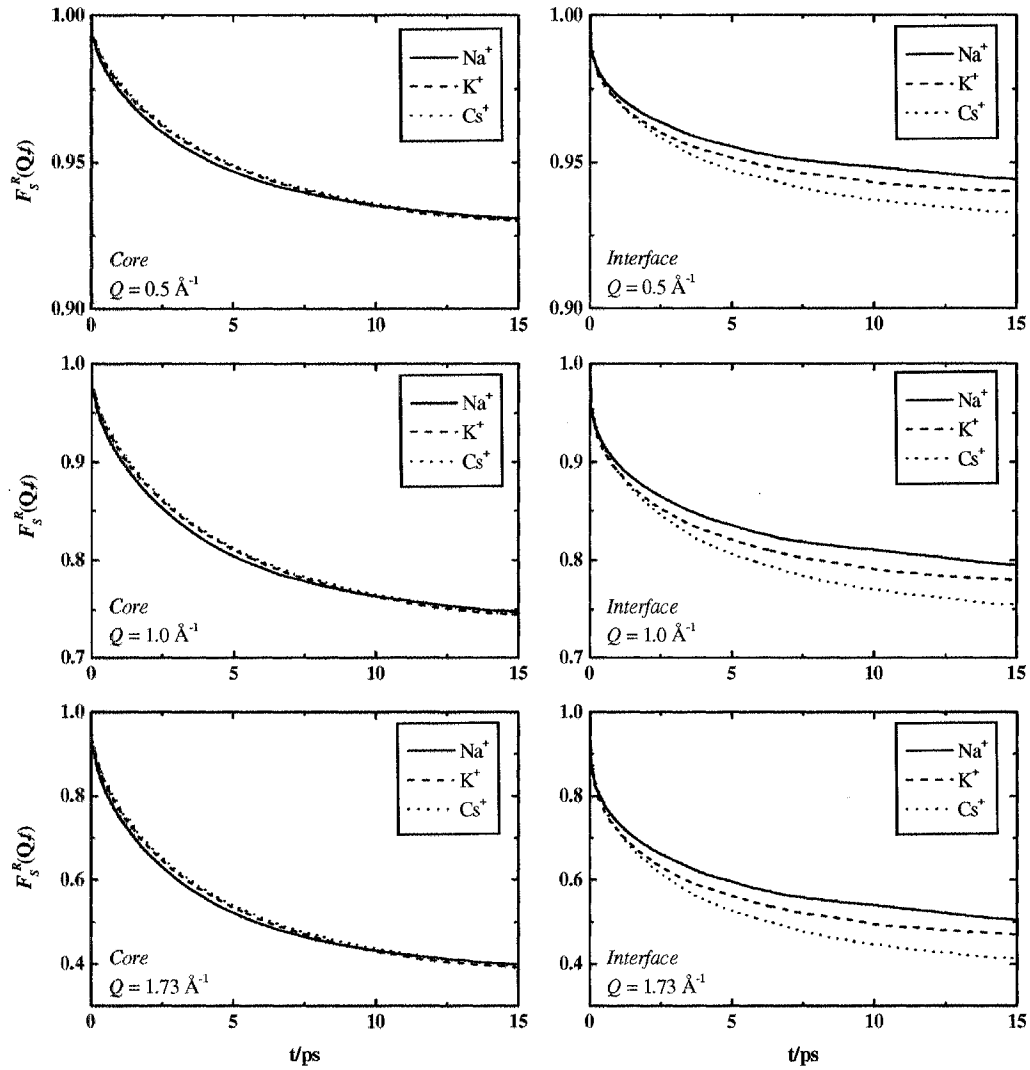


Figure 3.19. Rotational intermediate scattering function $F_S^R(\mathbf{Q},t)$ for core (left) and interfacial (right) water in $w_0 = 5$ Na^+ (solid), K^+ (dashes), and Cs^+ (dots) AOT RMs. Representative Q values 0.5 (top), 1.0 (middle), and 1.73 (bottom) $Å^{-1}$ are shown.

the H-to-CM distance of 0.964 Å. From this, we expect $F_S^R(\mathbf{Q},\infty)$ to reach a plateau of 0.93, 0.73, and 0.36, respectively, for Q = 0.5, 1.0, and 1.73 Å^{-1}. We see in Fig. 3.19 that the core water appears to be close to reaching these plateau values at the end of the 0 – 15 ps interval displayed. The interfacial water, however, is not as close to reaching this plateau value within the time interval plotted.

When we extend the time interval to 30 ps, close to the resolution limit of the QNS spectra, the interfacial $F_S^R(\mathbf{Q},t)$ arrive closer to these values, but do not reach them. The actual values reached depend on the counterion type. Following Di Cola et al.,[57] we calculate the effective values of the H-CM distance d that we would get from Eq. (3.21) by assuming that $F_S^R(\mathbf{Q},30\,\text{ps})$ corresponds to $F_S^R(\mathbf{Q},\infty)$. We use seven $\mathbf{Q}$ vectors between 0.5 and 2.23 Å^{-1}. For core water, we find that d is approximately 0.91 Å for all counterions studied. This is significantly less than the expected value of 0.964 Å. For interfacial water, where we expect the largest difference in water mobility in these RMs, we find a d of 0.80, 0.82, and 0.87 Å for Na^+, K^+, and Cs^+ RMs, respectively.

In a study of supercooled water, Di Cola et al.[57] calculated the value of d from their MD simulations of SPC/E water for three temperatures. They found that d was 0.84, 0.90, and 0.91 Å for SPC/E water at 264, 280, and 292 K, respectively at t = 80 ps. Di Cola et al. explained this lower than expected value of d as being due to hindered rotation of the water molecules. As the temperature increased, the water rotation became less hindered, and the value of d approached the expected H-to-CM distance. Our data indicates that the rotations of the core water molecules are approximately equally hindered in the RMs of the three counterions studied. However, the rotations of the interfacial water molecules become less hindered as the counterion size increases.

Comparing core water to interfacial water, the core water has a larger d than interfacial water does, indicating that rotational motion is less hindered in the core than in the RM interface.

Because experimental QNS data are frequently interpreted in terms of an isotropic rotational model, we test the degree of rotational anisotropy for water molecules in the RMs by evaluating the orientational TCFs of the Legendre polynomials of the (unit) bond and dipole vectors, $C_l(t)$ and $C_{l\mu}(t)$, given respectively, by Eqs. (3.19) and (3.20). While this anisotropy could be observed for Legendre polynomials of any order, we concentrate on the $l = 1$ because it provides the information that we seek. These results are presented in Figures 3.20 and 3.21, with $C_l(t)$, the bond vector TCF, shown in the top and $C_{l\mu}(t)$, the dipole vector TCF, in the bottom panel of these figures. Figure 3.20 displays these functions over the 0-15 ps time interval, while Figure 3.21 focuses on their short-time portion, 0-1 ps. In both figures, the TCFs for molecules initially in the core region are plotted on the left and for those in the interfacial region on the right. The results for bulk SPC/E water are added to the plots of the TCFs for the core water.

As in our previous study,[22] we find that RM data deviate strongly from exponential decay, and that the bond vector relaxation is faster than dipole vector relaxation. Comparing the results for the three counterions, we find that the differences between the decay rates of $C_1(t)$ and $C_{1,\mu}(t)$ diminish as the counterion size increases, but even for CsAOT it remains appreciable in the interfacial region. Counterion effects on $C_1(t)$ and $C_{1,\mu}(t)$ are much weaker in the core than at the interface, which is understandable in view of the much lower concentration of ionic species in this region. We note, however, that even in the core, water reorientation differs considerably from

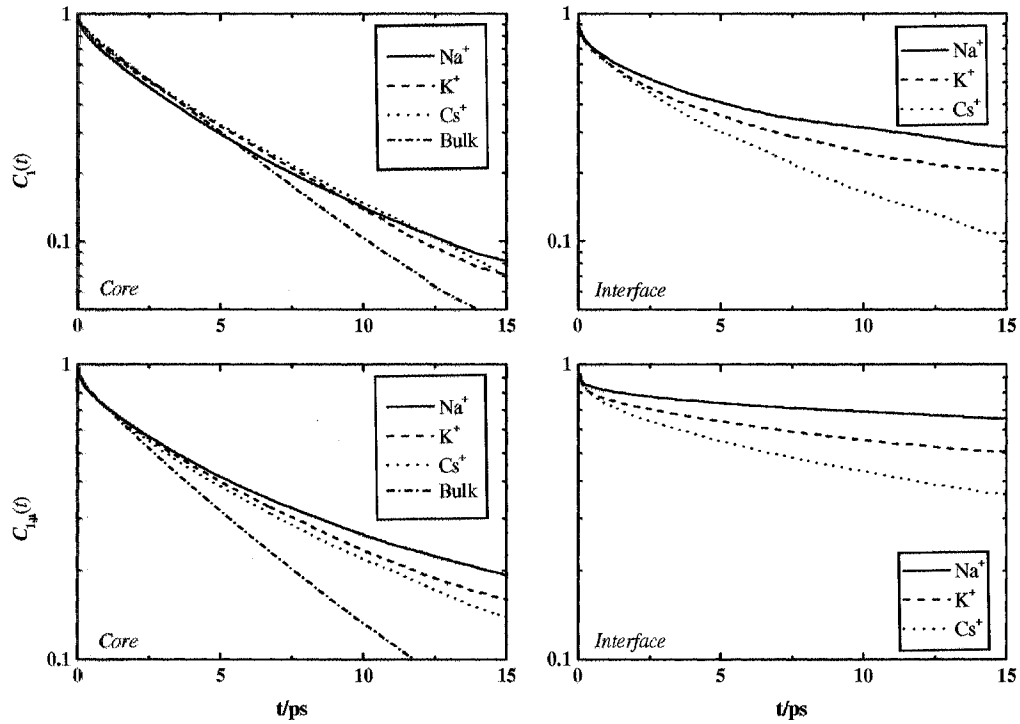


Figure 3.20: $C_1(t)$ (top) and $C_{1,\mu}$(t) (bottom) time correlation functions shown for core(left) and interfacial(right) water in w_0 = 5 Na^+(solid), K^+(dashes), and Cs^+(dots) AOT RMs.

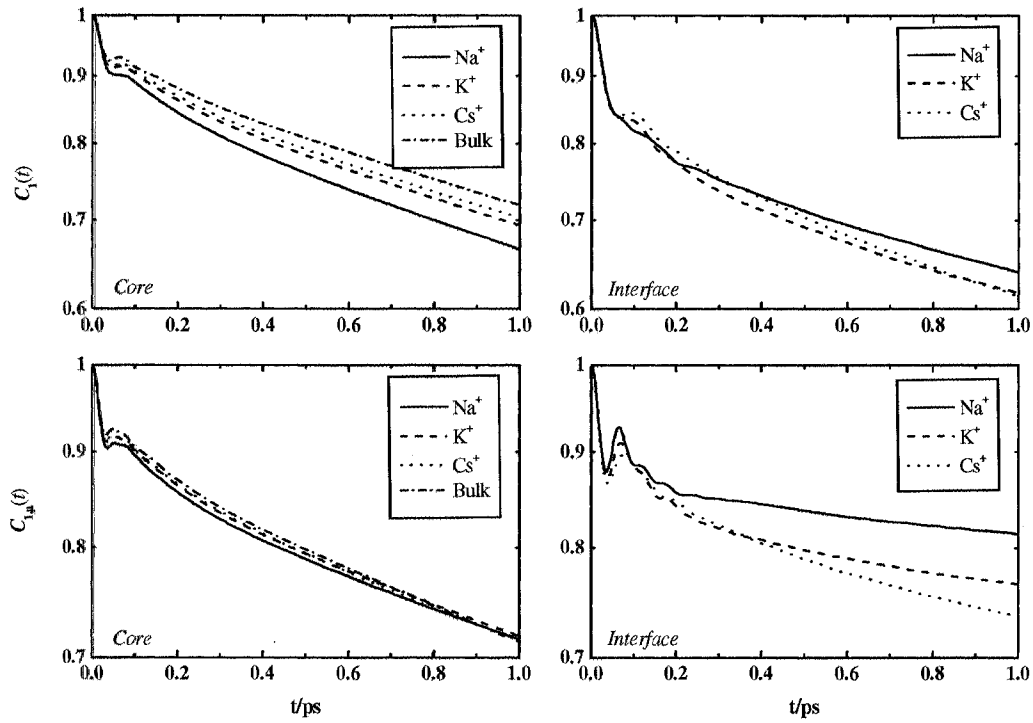


Figure 3.21. First 1.0 ps of the time correlation functions $C_1(t)$ (top) and $C_{1,\mu}$(t) (bottom) shown for core (left) and interfacial (right) water in $w_0 = 5$ Na^+ (solid), K^+ (dashes), and Cs^+ (dots) AOT RMs. Bulk SPC/E water results (dash-dot-dot) are shown in the left panels.

that for bulk liquid water, mainly in that it slows down at longer times, as can be seen from Figure 3.20, while in the bulk, the decay after about 0.5 ps is exponential.

The relatively fast decay of $C_l(t)$ at short time (Figure 3.21) has been explained previously[22] and attributed to disruption of water-water hydrogen bonds for water molecules residing at the RM interface compared to bulk water or water in the RM core region. In previous work,[20] it was found that the average number of water-water hydrogen bonds was greater for K^+ RMs than it was in Na^+ RMs, with the largest difference occurring for water molecules at the interface. Interfacial water molecules sacrifice hydrogen bonds to other water molecules in order to coordinate with the counterions and charged headgroups. The coordination of water to these ionic species slows down dipole reorientation. It also allows the O-H bond vector for water molecules coordinated to the counterions to reorient relatively freely over a limited angular range, while slowing down their overall rotational relaxation.

In contrast, rotations about the dipole vector appear less hindered in a RM than in bulk water. This reflects reduced rigidity of the water-water hydrogen bonding network at the interface of the RM compared to bulk water. Since Na^+ coordinates more strongly to water than K^+ and Cs^+ do, we expect and find faster bond vector relaxation and more anisotropy in the Na^+ system than in the K^+ and Cs^+ systems, and faster dipole vector relaxation in Cs^+ than in K^+ and Na^+. The P_1 bond vector correlations for core water over a ~10 ps interval show a trend where the water bond vector relaxation is faster for small counterions than for large counterions. Since smaller counterions dissolve more readily in the water pool than large counterions do, the water-water hydrogen bond network in the

core is more disrupted by Na^+ than K^+ and Cs^+, causing less-hindered librations for small counterions.

3.IV. Comparisons Between Simulations and Experiments

A comparison between the experimental and MD data for the QNS structure factors for NaAOT RMs, with Q = 1.0, 1.4 and 2.0 $Å^{-1}$ for each w_0, is presented in Fig. 3.22. As in Fig. 3.2, the QNS results correspond to the signal from which the background has been subtracted, i.e., to $S(\mathbf{Q},\omega) \otimes Res(\omega)$. The MD data for $F_S(\mathbf{Q},t)$ do not decay to zero as $t \rightarrow \infty$, but to a Q-dependent baseline.

In order to perform the time-to-frequency Fourier transform needed to construct $S(\mathbf{Q},\omega) \otimes Res(\omega)$, we fit the long-time ($t$ > 30 ps) portion of $F_S(\mathbf{Q},t)$ to a decaying exponential plus a baseline:

$$\Delta F_S(\mathbf{Q},t) = F_S(\mathbf{Q},t) - y_0 - ae^{-bt} \tag{3.23}$$

The parameters of this fit at several Q values are listed in Table 3.5. The baseline, y_0 is a product of the rotational baseline given in Eq. (3.18) and a translational baseline which is expected to decrease with increasing QR, where R is the radius of the confinement sphere[59], related to the RM radius, given in Table 3.3. The data in Table 3.5 obey this trend. Furthermore, the exponential decay constants, b, increase with increasing w_0, exhibiting faster relaxation, due primarily to translational diffusion, in larger RMs.

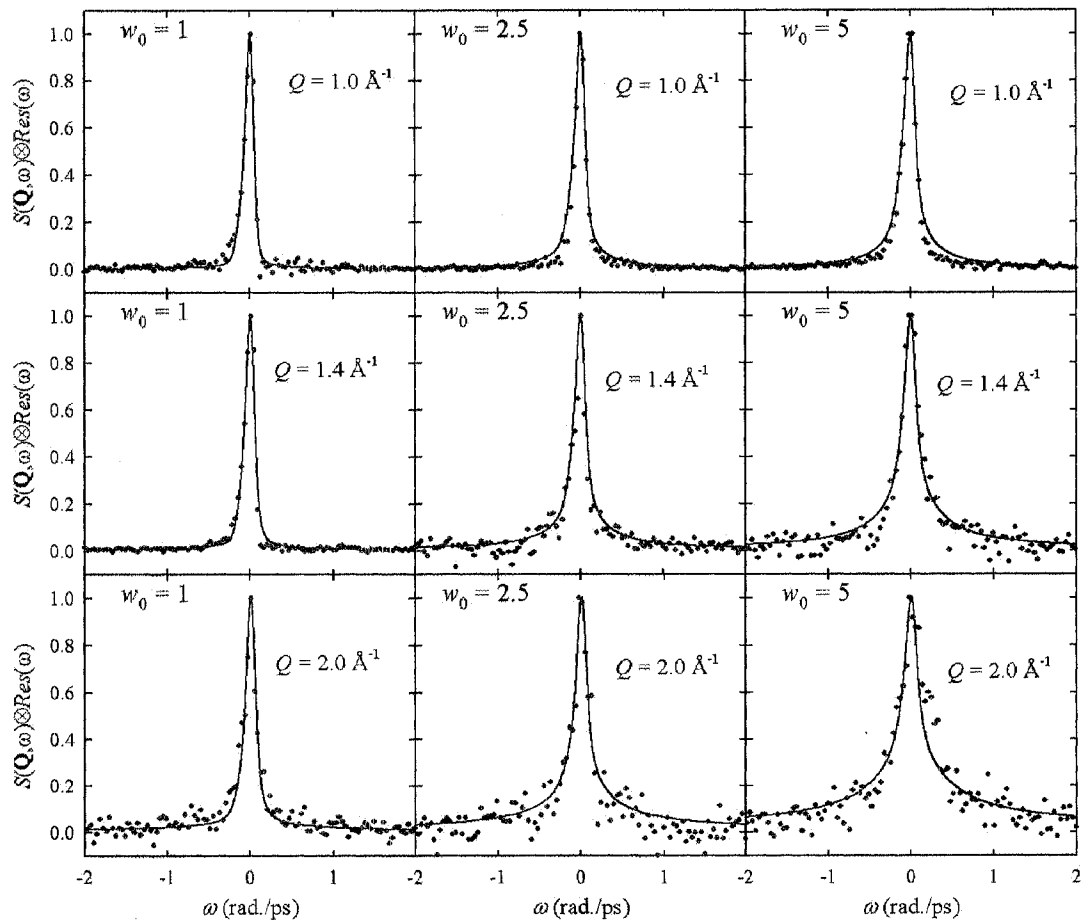


Figure 3.22. Comparisons between experimental and MD data for QNS structure factors. Experimental and MD data are given by points and lines, respectively.

Fourier transform of

$$\Delta F_S(\mathbf{Q},t) = F_S(\mathbf{Q},t) - y_0 - ae^{-bt} \tag{3.24}$$

is performed numerically and the Lorentzian corresponding to the transform of ae^{-bt} is added to the result. The experimental resolution function $Res(\omega)$ is fit to a sum of two Gaussians and the convolution with the MD spectrum minus the baseline contribution is performed numerically. The contribution due to the baseline ($y_0 Res(\omega)$) is then added.

The comparison of the MD and QNS data for Q = 1.0, 1.4 and 2.0 Å^{-1} indicates that there is very good agreement between simulation and experiment. The MD data show a marked difference in the line shapes and widths for the three RM sizes. The differences are more pronounced at higher Q at which for with the convolution is not dominated as strongly by the width of the resolution function. It signals, as do the experimental QNS data, that there is strong w_0-dependence of the mobility of water inside reverse micelles.

Given that the MD and QNS data agree quite well with each other, our analysis of the self-intermediate scattering functions, $F_S(\mathbf{Q},t)$, to obtain more detailed information on translational and rotational dynamics within RMs than was accessible through the interpretation of the fits of the spectra was quite reasonable.

Table 3.5. Fits of long-time portion of $F_S(Q,t)$ to $y_0 + a \exp(-bt)$

w_0	Q^2 (Å^{-2})	y_0	a	b (ps^{-1})
1	1	0.2687	0.3176	0.0125
	2	0.1634	0.2342	0.0108
	3	0.1022	0.1944	0.0110
	4	0.0904	0.1457	0.0161
2.5	1	0.0878	0.2563	0.0116
	2	0.0349	0.1811	0.0107
	3	0.0281	0.1225	0.0124
	4	0.0227	0.0907	0.0138
5	1	0.0406	0.1959	0.0177
	2	0.0180	0.1152	0.0174
	3	0.0134	0.0778	0.0209
	4	8.3283e-3	0.0532	0.0203

3.V. Summary

3.V.A. Effects of RM Size on Water Dynamics within NaAOT RMs

In Sections *IIC* and *IIIC*, we presented the results of QNS experiments on water/AOT/isooctane RMs. The nonpolar phase (isooctane) and the AOT alkyl chains were deuterated in order to focus on the mobility of water hydrogen atoms. To aid in the QNS data analysis, we performed MD simulations of single-particle dynamics of SPC/E water confined in model RMs[20] in the size range investigated by QNS.

In the QNS experiment, we studied water within RMs at w_0 values of 1, 2.5 and 5 to obtain information on translational and rotational mobilities of water in RMs of different water content. Analysis of these data was carried out first by using the jump translational diffusion – isotropic rotational diffusion model[58] to fit the experimental dynamic structure factor, $S(\mathbf{Q},\omega)$, and then by comparison to the MD simulation results. Our results show that water mobility in RMs is strongly w_0-dependent and quite different than bulk water mobility. Although the jump translational diffusion – isotropic rotational diffusion model fits the QNS data well, the resulting parameters (Table 3.1) are not entirely reasonable, so we have placed greater weight on the interpretation of the data on the basis of the MD simulation results, which also fit very well the experimental $S(\mathbf{Q},\omega)$ results.

The MD simulation results indicate that the decoupling approximation used to separate the translational and rotational contributions to the QNS spectrum is quite accurate in the Q and w_0 range of our experiments. Therefore it is reasonable to approximate the total self ISF, $F_S(\mathbf{Q},t)$, as a product of $F_S^R(\mathbf{Q},t)$ and $F_S^{CM}(\mathbf{Q},t)$. However, the MD results also show that rotational and translational dynamics in RMs are

rather more complicated than in bulk water and that neither $F_S^{CM}(\mathbf{Q},t)$ nor the rotational time correlations $C_l(t)$ can be well represented as simple exponential decays. Water-surfactant interactions and confinement both contribute to this more complicated behavior.

By tagging water molecules in our model RMs according to their initial location relative to the interface, we were able to determine the contributions to $F_S^R(\mathbf{Q},t)$ and $F_S^{CM}(\mathbf{Q},t)$ from molecules in the core and interfacial regions. Our MD results show that molecules near the interface are much less mobile than those in the core and that these differences persist over tens of picoseconds. The suppression of water mobility near the interface is stronger for smaller w_0 values due to the higher density of surfactant head groups. This and the fact that a smaller fraction of the molecules are in the more mobile core region lead to the decrease in water mobility as w_0 decreases.

The decrease in translational mobility is more pronounced than the rotational mobility decrease. We find that the lower density of water-water H-bonds near the interface leads to faster relaxation of $F_S^R(\mathbf{Q},t)$ at short times, compensating to some extent for the slowing down of diffusive reorientation rate as w_0 decreases.

We find that confinement has a strong influence on the behavior of $F_S^{CM}(\mathbf{Q},t)$, resulting in a decay to a Q and w_0 dependent baseline $y_c(Q,w_0)$ as $t \to \infty$. Taking into account the existence of $y_c(Q,w_0)$ as well as of the fact that translational diffusion rates in interfacial and core regions are different appears necessary in order to obtain information on translational mobility in RMs from QNS data.

3.V.B. Effects of the Surfactant Counterion on the Structure and Dynamics of AOT RMs

Section *II.D.* of this work discusses experimental QNS results for w_0 =5 KAOT, w_0 = 10 $Cu(AOT)_2$ and $Ca(AOT)_2$ RMs. We used the standard, hydrogen-containing, AOT for these experiments, which meant that the fraction of the signal coming from water hydrogen atoms was quite low. This, unfortunately, means that we are not terribly confident in the parameters obtained by the analysis of these data. The experimental QNS data suggest that water motion is less-hindered in w_0 =5 KAOT RMs than in w_0 =5 NaAOT RMs, which is consistent with ultrafast experiments[15-17] and MD simulations.[19] This can be attributed to the difference in size between KAOT and the slightly smaller NaAOT RMs, as well as the difference in the interactions between Na^+ and K^+ ions with water molecules. We explore this in further detail with MD simulations. Comparing w_0 =10 Cu^{2+} and Ca^{2+} (AOT_2) RMs, we find that the water dynamics are faster in the ellipsoidal $Cu(AOT_2)$ RMs than in the $Ca(AOT_2)$ RMs. We hypothesize that this is due to the difference in symmetry, as well as the interaction between water molecules and the counterions. When we compare water dynamics in w_0 =10 Cu^{2+} and Ca^{2+} (AOT_2) RMs with those of the smaller w_0 =5 RMs, we find that the water dynamics in the Cu^{2+} and Ca^{2+} (AOT_2) RMs than those in the smaller RMs.

In Section *III.D* , we present results which indicate that the large Cs^+ counterions pack more efficiently and exclude water more readily at the interface than the smaller K^+ and Na^+ counterions do. As a result, interfacial water molecules coordinate less strongly to Cs^+ counterions than they do to the smaller K^+ and Na^+ counterions. The weaker

coordination of water to Cs^+ than K^+ and Na^+ results in less disruption of the water structure by the larger cations, and hence water molecules have higher mobility in Cs^+ RMs than in K^+ and Na^+ RMs. We see the largest difference in water mobility at the interface, where these systems' structures differ most significantly. At short times, the dynamics of water are *faster* for smaller counterions than larger counterions, which is the opposite of the long-time behavior. We attribute this effect to the increased disruption of the hydrogen bonding network by small counterions; more disruption causes more effective librational motion. Translational and rotational time correlations appear to be highly non-exponential. Stronger coordination between interfacial water molecules and Na^+ counterions results in slower dipole vector relaxation and initially faster bond vector relaxation in Na^+ RMs than in K^+ and Cs^+ RMs, which have increasingly less coordination to water molecules. In addition, as the counterion size increases, the difference in relaxation rate between bond and dipole vectors decreases, meaning that the rotation is becoming more isotropic.

References and Notes

(1) Calandra, P.; Longo, A.; Liveri, V. T. *J. Phys. Chem. B* **2003**, *107*, 25.
(2) Cason, J. P.; Miller, M. E.; Thompson, J. B.; Roberts, C. B. *J. Phys. Chem. B* **2001**, *105*, 2297.
(3) Pavel, F. M.; Mackay, R. A. *Langmuir* **2000**, *16*, 8568.
(4) Pileni, M. P. *J. Phys. Chem.* **1993**, *97*, 6961.
(5) *Reverse Micelles: Biological and Technological Relevance of Amphiphilic Structures in Apolar Media*; Luisi, P. L.; Straub, B. E., Eds.; Plenum Press: New York, 1984.
(6) Amararene, A.; Gindre, M.; LeHuerou, J. Y.; Nicot, C.; Urbach, W.; Waks, M. *J. Phys. Chem. B* **1997**, *101*, 10751.
(7) Amico, P.; D'Angelo, M.; Onori, G.; Santucci, A. *Il Nuovo Cimento* **1995**, *17*, 1053.
(8) Boicelli, C. A.; Giomini, M.; Giuliani, A. M. *Appl. Spec.* **1984**, *38*, 537.
(9) Brubach, J. B.; Mermet, A.; Filabozzi, A.; Gerschel, A.; Lairez, D.; Krafft, M. P.; Roy, P. *J. Phys. Chem. B* **2001**, *105*, 430.
(10) D'Angelo, M.; Fioretto, D.; Onori, G.; Palmieri, L.; Santucci, A. *Phys. Rev. E* **1996**, *54*, 993.
(11) Venables, D. S.; Huang, K.; Schmuttenmaer, C. A. *J. Phys. Chem. B* **2001**, *105*, 9132.
(12) Corbeil, E. M.; Levinger, N. E. *Langmuir* **2003**, *19*, 7264.
(13) Corbeil, E. M.; Riter, R. E.; Levinger, N. E. *J. Phys. Chem. B* **2004**, *108*, 10777.
(14) Pant, D.; Levinger, N. E. *Langmuir* **2000**, *16*, 10123.
(15) Pant, D.; Riter, R. E.; Levinger, N. E. *J. Chem. Phys.* **1998**, *109*, 9995.
(16) Riter, R. E.; Undiks, E. P.; Levinger, N. E. *J. Am. Chem. Soc.* **1998**, *120*, 6062.
(17) Riter, R. E.; Willard, D. M.; Levinger, N. E. *J. Phys. Chem. B* **1998**, *102*, 2705.
(18) Willard, D. M.; Riter, R. E.; Levinger, N. E. *J. Am. Chem. Soc.* **1998**, *120*, 4151.
(19) Faeder, J.; Albert, M. V.; Ladanyi, B. M. *Langmuir* **2003**, *19*, 2514.
(20) Faeder, J.; Ladanyi, B. M. *J. Phys. Chem. B* **2000**, *104*, 1033.
(21) Faeder, J.; Ladanyi, B. M. *J. Phys. Chem. B* **2001**, *105*, 11148.
(22) Harpham, M. R.; Ladanyi, B. M.; Levinger, N. E.; Herwig, K. W. *J. Chem. Phys.* **2004**, *121*, 7855.
(23) Harpham, M. R.; Ladanyi, B. M.; Levinger, N. E. *J. Phys. Chem. B* **2005**, *109*, 16891.
(24) Herwig, K. W.; Dozier, W. D.; Huang, J. S. *Mater. Res. Using Cold Neutrons Pulsed Neutron Sources, [Proc.]* **1999**, 95.
(25) Bellissent-Funel, M. C. *J. Mol. Liq.* **1998**, *78*, 19.
(26) Bellissent-Funel, M. C.; Chen, S. H.; Zanotti, J. M. *Phys. Rev. E* **1995**, *51*, 4558.

(27) Crupi, V.; Dianoux, A. J.; Majolino, D.; Migliardo, P.; Venuti, V. *PCCP* **2002**, *4*, 2768.

(28) Crupi, V.; Majolino, D.; Migliardo, P.; Venuti, V. *J. Phys. Chem. B* **2002**, *106*, 10884.

(29) Crupi, V.; Majolino, D.; Migliardo, P.; Venuti, V. *Philosophical Magazine B-Physics of Condensed Matter Statistical Mechanics Electronic Optical and Magnetic Properties* **2002**, *82*, 425.

(30) Crupi, V.; Majolino, D.; Migliardo, P.; Venuti, V.; Bellissent-Funel, M. C. *Mol. Phys.* **2003**, *101*, 3323.

(31) Faraone, A.; Chen, S. H.; Fratini, E.; Baglioni, P.; Liu, L.; Brown, C. *Phys. Rev. E* **2002**, *65*.

(32) Faraone, A.; Liu, L.; Chen, S. H. *J. Chem. Phys.* **2003**, *119*, 6302.

(33) Faraone, A.; Liu, L.; Mou, C. Y.; Shih, P. C.; Brown, C.; Copley, J. R. D.; Dimeo, R. M.; Chen, S. H. *Eur. Phys. J. E* **2003**, *12*, S59.

(34) Faraone, A.; Liu, L.; Mou, C. Y.; Shih, P. C.; Copley, J. R. D.; Chen, S. H. *J. Chem. Phys.* **2003**, *119*, 3963.

(35) Mitra, S.; Mukhopadhyay, R.; Pillai, K. T.; Vaidya, V. N. *Journal of Non-Crystalline Solids* **1998**, *235*, 229.

(36) Mitra, S.; Mukhopadhyay, R.; Pillai, K. T.; Vaidya, V. N. *Solid State Communications* **1998**, *105*, 719.

(37) Mitra, S.; Mukhopadhyay, R.; Tsukushi, I.; Ikeda, S. *Journal of Physics-Condensed Matter* **2001**, *13*, 8455.

(38) Takahara, S.; Nakano, M.; Kittaka, S.; Kuroda, Y.; Mori, T.; Hamano, H.; Yamaguchi, T. *J. Phys. Chem. B* **1999**, *103*, 5814.

(39) Teixeira, J. *Nuovo Cimento* **1994**, *16*, 1433.

(40) Venturini, F.; Gallo, P.; Ricci, M. A.; Bizzarri, A. R.; Cannistraro, S. *Philosophical Magazine B-Physics of Condensed Matter Statistical Mechanics Electronic Optical and Magnetic Properties* **2002**, *82*, 507.

(41) Venuti, V.; Crupi, V.; Galli, G.; Majolino, D.; Migliardo, P. *J. Mol. Struct.* **2002**, *615*, 83.

(42) Venuti, V.; Crupi, V.; Magazu, S.; Majolino, D.; Migliardo, P.; Bellissent-Funel, M. C. *Journal De Physique Iv* **2000**, *10*, 211.

(43) Zanotti, J. M.; Bellissent-Funel, M. C.; Chen, S. H. *Phys. Rev. E* **1999**, *59*, 3084.

(44) Crupi, V.; Majolino, D.; Migliardo, P.; Venuti, V. *J. Phys. Chem. A* **2000**, *104*, 11000.

(45) Fratini, E.; Chen, S. H.; Baglioni, P.; Bellissent-Funel, M. C. *J. Phys. Chem. B* **2002**, *106*, 158.

(46) Fratini, E.; Chen, S. H.; Baglioni, P.; Cook, J. C.; Copley, J. R. D. *Phys. Rev. E* **2002**, *65*.

(47) Farrer, R. A.; Fourkas, J. T. *Accounts of Chemical Research* **2003**, *36*, 605.

(48) Fouzri, A.; Dorbez-Sridi, R.; Oumezzine, M. *J. Chem. Phys.* **2002**, *116*, 791.

(49) Gallo, P. *PCCP* **2000**, *2*, 1607.

(50) Gallo, P.; Rapinesi, M.; Rovere, M. *J. Chem. Phys.* **2002**, *117*, 369.

(51) Gallo, P.; Ricci, M. A.; Rovere, M. *Physica A* **2002**, *304*, 53.
(52) Gallo, P.; Ricci, M. A.; Rovere, M. *J. Chem. Phys.* **2002**, *116*, 342.
(53) Riter, R. E.; Undiks, E. P.; Kimmel, J. R.; Levinger, N. E. *J. Phys. Chem. B* **1998**, *102*, 7931.
(54) Bee, M. *Quasielastic Neutron Scattering*; Hilger: Bristol, 1988.
(55) Eastoe, J.; Towey, T. F.; Robinson, B. H.; Williams, J.; Heenan, R. K. *J. Phys. Chem.* **1993**, *97*, 1459.
(56) Vaisman, II; Berkowitz, M. L. *JACS* **1992**, *114*, 7889.
(57) Di Cola, D.; A., D.; Sampoli, M.; Torcini, A. *J. Chem. Phys.* **1996**, *104*, 4223.
(58) Egelstaff, P. A. *An Introduction to the Liquid State*; Academic: London, 1967.
(59) Volino, F.; Dianoux, A. J. *Mol. Phys.* **1980**, *41*, 271.
(60) Sears, V. F. *Can. J. Phys.* **1967**, *45*, 237.
(61) Texeira, J.; Bellisent-Funel, M.-C.; Chen, S. H.; Dianoux, A. J. *Phys Rev A* **1985**, *31*, 1913.
(62) Lee, C. Y.; McCammon, J. A.; Rossky, P. J. *J. Chem. Phys.* **1984**, *80*, 4448.
(63) Berendsen, H. J. C.; Grigera, J. R.; Straatsma, T. P. *J. Chem. Phys.* **1987**, *91*, 6269.
(64) Schweighofer, K. J.; Essmann, U.; Berkowitz, M. J. *J. Phys. Chem. B* **1997**, *101*, 3793.
(65) Dang, L. X. *J. Am. Chem. Soc.* **1995**, *117*, 6954.
(66) Eicke, H.-F.; Rehak, J. *Helv. Chim. Acta* **1976**, *59*, 2883.
(67) Allen, M. P.; Tildesley, D. J. *Computer Simulations of Liquids*Clarendon: Oxford,U.K., 1987; Vol. 1987.
(68) Ciccotti, G.; Ryckaert, J.-P. *Comput. Phys. Rep.* **1986**, *4*, 345.
(69) Berendsen, H. J. C.; Postma, J. P. M.; van Gunsteren, W. F.; DiNola, A.; Haak, J. R. *J. Chem. Phys.* **1984**, *81*, 3684.
(70) Tarek, M.; Tobias, D. J. *Biophysical Journal* **2000**, *79*, 3244.
(71) Lynden-Bell, R. M.; Steele, W. A. *J. Phys. Chem.* **1984**, *88*, 6514.

Appendix 3.I: Methods for Purifying, Drying, and Counterion-exchanging Aerosol OT

This section explains the steps involved in the purification, drying, and counterion exchange of sodium bis-(2-diethylhexyl) sulfosuccinate (Aerosol OT, NaAOT). We apply the purification and drying procedure to all surfactant used in our QNS experiments, regardless of whether it was NaAOT purchased from Aldrich, deuterated AOT (d-AOT) custom-synthesized by Prof. Robert K. Thomas, or $M^{+n}(AOT)_n$ counterion exchanged as above. The purification and drying consists of four steps – two separations, and one lyophilization. Our counterion exchange procedure is a combined and modified version of the liquid-liquid exchange procedures detailed by Eastoe et al.[1] Crupi et al. [2] We perform the counterion exchange procedure on purified and dried NaAOT to produce M^{n+} $(AOT)_n$, where M^{n+} is Cu^{2+}, Ca^{2+}, or K^+.

The first step is to remove any organic impurities. We combine a methanolic solution of NaAOT with petroleum ether in a 1:1 ratio within a glass separatory funnel. For 2 grams of NaAOT, we use approximately 30 mL each of petroleum ether and methanol. For approximately five minutes, we stopper the separatory funnel, shake, invert, and remove the stopper at 10-20 second intervals. After this, we discard the petroleum ether phase, and repeat the process with twice more with additional petroleum ether. We then drain the methanol phase, which contains NaAOT, into a beaker and evaporate to dryness on a hot plate, which we set at about 15K above room temperature.

The next separation involves removing any hydrophilic contaminants.We dissolve the relatively dry NaAOT in 30 mL of benzene and add this to 30 mL of nanopure water in a separatory funnel. Again, we alternate between shaking, inverting, and venting at intervals over about five minutes, drain the aqueous phase, and repeat with additional

aliquots of water. We then drain the benzene phase into a 250 mL round-bottom flask, and perform a vacuum distillation to remove the benzene.

Before lyophilization, we dissolve the purified AOT in approximately 100 mL of water. We prepare an ice bath using crushed dry ice and acetone, and use this to freeze-coat the solution of NaAOT in water to the inside of the round-bottom flask. We then attach the frozen round-bottom flask to the lyophilizer, and after 24 hours of lyophilization, we have dried, purified NaAOT for use in NaAOT samples or for counterion exchange.

We use a modified version of the liquid-liquid exchange procedure[1,2] on NaAOT to replace the sodium ion with calcium, copper[II], or potassium ions. For exchange with potassium or calcium, we mix equal volumes of 1 M solution of NaAOT in heptane with a saturated methanolic solution of $M^{n+}(NO_3)_n$ in a separatory funnel. After five minutes of agitation and venting of the contents of the separatory funnel, we allow the mixture to phase-separate, and discard the methanol layer. To increase the efficiency of the counterion exchange, we wash thrice more with $M^{n+}(NO_3)_n$ solution.

We then perform a vacuum distillation of the $M^{n+}(AOT)_n$ / heptane solution, dissolve the solid $M^{n+}(AOT)_n$ in benzene, and transfer the solution to a separatory funnel. We then add an aliquot of water, and the mixture phase separates; the benzene layer contains $M^{n+}(AOT)_n$, while the aqueous layer contains any remaining salts. We wash thrice with water, discarding the aqueous layer after agitating the mixture for a period of five minutes. We perform another vacuum distillation to remove the benzene, and finally dry the pure $M^{n+}(AOT)_n$ by lyophilization.

Counterion exchange with copper[3] is slightly more interesting than with the other ions mentioned above; in addition to having more colorful solutions, we form a three-phase system during the exchange. We perform the initial steps as outlined above; a 1:1 mixture of a methanolic solution of copper[3] nitrate is combined with a 1M solution of NaAOT in heptane. We discard the methanol layer, making sure that we do not discard the cloudy methanol/heptane/AOT layer above it. We wash several more times with methanolic metal nitrate solution, discarding the lower layer each time, and then perform a vacuum distillation on the cloudy layer and heptane layer. At this point, we perform the purification procedure described towards the beginning of this Appendix; a methanol/ether separation to remove organic contaminants, followed by a water/benzene separation to remove excess sodium and nitrate ions. Finally, we perform a vacuum distillation of the $Cu(AOT)_2$/benzene solution, followed by a lyophilization.

References and Notes

(1) Eastoe, J.; Towey, T. F.; Robinson, B. H.; Williams, J.; Heenan, R. K. *J. Phys. Chem.* **1993**, *97*, 1459.

(2) Crupi, V.; Magazu, S.; Maisano, G.; Majolino, D.; Migliardo, P. *Phys. Scripta* **1994**, *50*, 200.

Appendix 3.II: Source Code Used for the Analysis of Intermediate Scattering Functions and Legendre Polynomials

This Appendix gives the C-source code used for calculating intermediate scattering functions and Legendre polynomials. The code presented here was added to the analysis procedures within the *newmd* program written by Faeder,[1] and will not compile on its own. The procedures added to the analysis are called within the main loop of the code, which was left unmodified, save for the addition of these procedure calls.

The main section of the analysis program is included in this dissertation because it a loop over configurations, which is necessary for calculating the time correlation functions.

References and Notes

(1) Faeder, J. newmd Classical MD Version, 2002.

//Main

```
#include "newmd.h"
#include "interface.h"
#include "analyze.h"

static const int LINE_LENGTH=400;
static const int LINE_TOKENS=70;

extern void read_config_search_time( double t);
int fragment_analysis( Config *cp, char *prefix, int init);
void do_analyze(){
   struct keylist list_local;
   int status;
   char buf[400];
   FILE *outfile;
   char *prefix=0x0, *config_file=0x0;
   io_handle *config_handle=0x0, *print_handle=0x0;
   Config *cp=0x0;
   int *mol_layers;
   int err=0,nspec;
   int do_rdp=0;
   int do_pdf=0;
   int do_msd=0;
   int do_vacf=0;
   int layers=0;
   int ctlayers=0;
   int ltype=0;
   int do_cisf=0;
   int do_risf=0;
   int do_tisf=0;
   int do_ct=0;
```

```
int do_dv=0;
int do_tcf=0;
int do_M=0;
int do_acf=0;
int do_frag=0;
int do_h2o=0;
int do_h2o_energy=0, energy_int=1;
int LeP;
double Qfac;
int do_print=0;
int do_dip_avg=0;
int do_dip_dens=0;
int print_binary=1;
int nbins=200;
double R_max=30, R_min, dR_radial=1.0;
int corr_interval=10;
int corr_length=100;
double t_start=-1.0e99, t_end=1.0e99;
Config *origin_list=0x0;
int nconfig, n_expected=0, index;
int n_types, n_tri;
double *boundaries=0x0;
double dt, dt_last, t_last;
acf *acfs=0x0;
h2o_props *h2o=0x0;
int max_configs=-1;
char *solv_type=0x0;
char *pos_type=0x0;
char *neg_type=0x0;
/* Cutoff distances in Angstroms */
double R_nn_solv= 3.5;
```

```
double R_nn_pos_solv= 3.5;
double R_nn_neg_solv= 5.0;
double R_nn_pos_neg= 4.8;
/* Hydrogen bond energy threshold in kJ/mol */
double hbond_thresh= 16.0;

solv_type= strdup("H2O");
pos_type= strdup("Na+");
neg_type= strdup("Z-");

init_keylist(&list_local);

/* set local keywords */
add_keyword(&list_local, "config_file",
            0x0,
            &config_file,  "string",
            "Name of file from which to read types and configurations. \
Defaults to global config_file.",
            0);
add_keyword(&list_local, "prefix",
            0x0,
            &prefix,  "string",
            "Prefix for output data files.  Defaults to config_file prefix.",
            0);
add_keyword(&list_local, "print",
            0x0,
            &do_print, "opt",
            "Print configs.  Useful for converting old file formats.",
            0);
add_keyword(&list_local, "boundaries",
            0x0,
```

```
                &boundaries, "darray",
                "Boundaries used in defining regions in Angstroms. \
Must be in ascending order and must be > 0.0.",
                0);
  add_keyword(&list_local, "dip_avg",
                0x0,
                &do_dip_avg, "opt",
                "Calculate (water) average dipole moment as function of R.",
                0);
  add_keyword(&list_local, "dip_dens",
                0x0,
                &do_dip_dens, "opt",
                "Calculate (water) dipole density as function of R.",
                0);
  add_keyword(&list_local, "acf",
                0x0,
                &do_acf, "opt",
                "Calculate various angular correlation functions.",
                0);
  add_keyword(&list_local, "fragment_analysis",
                0x0,
                &do_frag, "opt",
                "Analyze cluster fragments.",
                0);
  add_keyword(&list_local, "h2o",
                0x0,
                &do_h2o, "opt",
                "Analyze various structural properties of H2O.",
                0);
  add_keyword(&list_local, "h2o_energy",
                0x0,
```

```
                &do_h2o_energy, "opt",
                "Compute energy of H2O interactions.",
                0);
add_keyword(&list_local, "Ltype",
        0x0,
        &ltype, "int",
        "Type of layer analysis (R=1, Tot=2) Defaults to zero.",
        0);
add_keyword(&list_local, "ctlayers",
        0x0,
        &ctlayers, "int",
        "How many layers for the analysis? Defaults to zero.",
        0);
add_keyword(&list_local, "layers",
        0x0,
        &layers, "int",
        "How many layers for the analysis? Defaults to zero.",
        0);

add_keyword(&list_local, "energy_int",
                0x0,
                &energy_int, "int",
                "How often to compute H2O energy.  Defaults to 1.",
                0);
add_keyword(&list_local, "rdp",
                0x0,
                &do_rdp, "opt",
                "Calculate the radial density profile for all types.",
                0);
add_keyword(&list_local, "pdf",
                0x0,
```

```
                &do_pdf, "opt",
                "Calculate the pair radial distribution functions for all types.",
                0);
add_keyword(&list_local, "msd",
                0x0,
                &do_msd, "opt",
                "Compute the mean square displacements for all types.",
                0);
add_keyword(&list_local, "cisf",
        0x0,
        &do_cisf, "opt",
        "Compute c-of-m ISF for water molecules.",
        0);

add_keyword(&list_local, "risf",
        0x0,
        &do_risf, "opt",
        "Compute rotational ISF for water molecules.",
        0);
add_keyword(&list_local, "tisf",
        0x0,
        &do_tisf, "opt",
        "Compute total ISF for water molecules.",
        0);
add_keyword(&list_local, "dv",
        0x0,
        &do_dv, "opt",
        "Compute C(t)for dipole vector of water molecules.",
        0);
add_keyword(&list_local, "ct",
        0x0,
```

```
        &do_ct, "opt",
        "Compute C(t)for water molecules.",
        0);
add_keyword(&list_local, "Qfac",
        0x0,
        &Qfac, "double",
        "Use a specific multiplier.",
        0);
add_keyword(&list_local, "LeP",
        0x0,
        &LeP, "int",
        "Calc Specific Legendre Polynomial",
        0);
  add_keyword(&list_local, "nspec",
        0x0,
        &nspec, "int",
        "specify # species",
        0);
add_keyword(&list_local, "dipole",
            0x0,
            &do_M, "opt",
            "Compute the total system dipole for each config.",
            0);
add_keyword(&list_local, "vacf",
            0x0,
            &do_vacf, "opt",
            "Compute the velocity autocorrelation functions for all types.",
            0);
add_keyword(&list_local, "corr_length",
            0x0,
            &corr_length, "int",
```

```
                "Maximum distance between configurations to correlate. Defaults \
to 100.",
                0);
  add_keyword(&list_local, "corr_interval",
                0x0,
                &corr_interval, "int",
                "Interval between configurations to use as origins in computing tcfs. \
Defaults to 10.",
                0);
  add_keyword(&list_local, "R_max",
                0x0,
                &R_max, "double",
                "Maximum R to use in calculating radial density (in Ang.).",
                1);
  add_keyword(&list_local, "dR_radial",
                0x0,
                &dR_radial, "double",
                "Bin sizes to use when computing nn interactions by region(in Ang.).",
                1);
  add_keyword(&list_local, "n_bins",
                0x0,
                &nbins, "int",
                "Number to use in calculating distribution functions.",
                1);
  add_keyword(&list_local, "t_start",
                0x0,
                &t_start, "double",
                "Earliest time (in fs) to begin analysis",
                0);
  add_keyword(&list_local, "t_end",
                0x0,
```

```
                &t_end, "double",
                "Last time (in fs) to perform analysis ",
                0);
  add_keyword(&list_local, "max_configs",
                0x0,
                &max_configs, "int",
                "Maximum number of configurations to read. Default is unlimited.",
                0);
  add_keyword(&list_local, "solv_type",
                0x0,
                &solv_type,  "string",
                "Solvent molecule type.",
                1);
  add_keyword(&list_local, "pos_type",
                0x0,
                &pos_type,  "string",
                "Positive ion molecule type.",
                1);
  add_keyword(&list_local, "neg_type",
                0x0,
                &neg_type,  "string",
                "Negative ion molecule type.",
                1);
  add_keyword(&list_local, "R_nn_solv",
                0x0,
                &R_nn_solv, "double",
                "Cutoff distance (Ang) to use for determining solvent-solvent \
nearest neighbors.",
                1);
  add_keyword(&list_local, "R_nn_pos_solv",
                0x0,
```

```
                &R_nn_pos_solv, "double",
                "Cutoff distance (Ang) to use for determining positive ion-solvent \
nearest neighbors.",
                1);
  add_keyword(&list_local, "R_nn_neg_solv",
                0x0,
                &R_nn_neg_solv, "double",
                "Cutoff distance (Ang) to use for determining negative ion-solvent \
nearest neighbors.",
                1);
  add_keyword(&list_local, "R_nn_pos_neg",
                0x0,
                &R_nn_pos_neg, "double",
                "Cutoff distance (Ang) to use for determining positive ion-negative \
nearest neighbors.",
                1);
  add_keyword(&list_local, "hbond_thresh",
                0x0,
                &hbond_thresh, "double",
                "Hydrogen bond energy threshold in kJ/mol.",
                1);
  add_keyword(&list_local, "help",
                do_help,
                0x0, 0x0,
                NULL,
                0);
  /* handle help request */
  if (input.status&STAT_HELP){
        status= process_keys(&list_local);
        goto exit;
  }
```

```
/* process other arguments to the function */
status= process_keys(&list_local);
if ( status&STAT_ERROR) goto exit;

/* set default config file */
if (!config_file){
    config_file= strdup(globals.config_file);
}

/* determine if tcf's are to be calculated */
do_tcf= do_msd || do_vacf|| do_cisf || do_risf|| do_tisf||do_ct||do_dv||layers;
if (!do_tcf && !(is_key_set("corr_length", &list_local))){
    corr_length=0;
    corr_interval=0;
}

/* define region boundaries */
if (boundaries){
    int i;
    for(i=0; i<darray_len(boundaries); ++i) boundaries[i] /= BOHR_TO_ANG;
    define_regions_radial( boundaries, darray_len(boundaries));
}

/* convert variables to atomic units */
R_max/= BOHR_TO_ANG;
dR_radial/= BOHR_TO_ANG;

/* END OF INPUT PROCESSING */

outfile=output.file;
```

```
/* clear current properties list */
clear_configs();

/* open config file */
if(!(config_handle=OPEN_HANDLE( config_file, "r", IO_FLAGS_DFLT))){
        sprintf(buf, "analyze: Couldn't open file %s", config_file);
        handle_error(buf);
        goto exit;
}

/* read types from config file */
if (read_types(config_handle->file) < 1){
        sprintf(buf, "analyze: No types could be read from file %s",
                config_file);
        ++err;
        handle_error(buf);
        goto exit;
}

/* determine default output file prefix */
if (!prefix){
        sprintf(buf, "%s", config_file);
        chop_suffix(buf, ".newmd");
        chop_suffix(buf, ".newmd.gz");
        prefix= strdup( buf);
}

/* open file to print configs */
if (do_print){
        sprintf(buf, "%s.bin.newmd", prefix);
```

```
        if(!(print_handle=OPEN_HANDLE( buf, "w", IO_FLAGS_DFLT))){
          goto exit;
        }
    }

    if ( do_rdp)
        if (init_rdp( outfile, prefix, R_max, nbins, 1)) goto exit;
    if ( do_pdf)
        if (init_pdf( outfile, prefix, R_max, nbins)) goto exit;
    if ( do_msd)
        if (init_msd( outfile, prefix, corr_length)) goto exit;
    if ( layers)
      if (init_layer_tisf( outfile, prefix, corr_length, Qsq,nspec,layers,mol_layers, ltype))
goto exit;
    if ( do_cisf)
      if (init_cisf( outfile, prefix, corr_length, Qsq,nspec)) goto exit;
    if ( do_tisf)
      if (init_tisf( outfile, prefix, corr_length, Qsq,nspec)) goto exit;
    if ( do_risf)
      if (init_risf( outfile, prefix, corr_length, Qsq,nspec)) goto exit;
    if ( do_ct||do_dv)
     if (init_ct( outfile, prefix, corr_length, nspec, ctlayers)) goto exit;
    if ( do_vacf)
        if (init_vacf( outfile, prefix, corr_length)) goto exit;
    if ( do_M)
        if (init_M( outfile, prefix)) goto exit;
    if ( do_acf)
        if (!(acfs=new_acf( ACF_RAD, n_regions(), nbins))) goto exit;
    if ( do_dip_avg)
        if (init_h2o_dip_avg( outfile, R_max, nbins)) goto exit;
    if ( do_dip_dens)
```

```
        if (init_h2o_dip_dens( outfile, R_max, nbins)) goto exit;

if (t_start>-1.0e98 || t_end < 1.0e98)
        read_config_search_time(t_start);
t_start /= AU_TO_FS;
t_end /= AU_TO_FS;

/* loop over configs in config file */
nconfig=0;
dt=0;
while (cp=read_config(config_handle->file)){
        int save_config=0;

        cp->index=nconfig;
        ++nconfig;

        if (cp->time < t_start) goto cleanup; /* shouldn't happen b/c of
                                                 read_config_search_time */
        if (cp->time > t_end) break;
        if (max_configs>0 && nconfig > max_configs) break;

        if (nconfig==1) fprintf(outfile, "Found config 1\n");

        /* calculate molecule COM positions */
        COM_positions(cp->molarray);

        /* Determine time spacing between configs */
        if (nconfig>1) dt= cp->time-t_last;
        if (nconfig>2){
           /* make sure time spacing between configs is constant */
           if (do_tcf && dt && fabs(dt-dt_last)/dt > 1.0e-6){
```

```
                sprintf(buf, "analyze: Configurations must be equally spaced \
in time.");
                handle_error(buf);
                goto exit;
            }
        }
        if (nconfig>1) dt_last= dt;
        t_last=cp->time;

        /* print running count of configs that have been processed */
        if ((nconfig) % 100 == 0){
            fprintf(outfile, "Reached config %5d\n", nconfig);
        }

        /* Classify molecules according to boundaries */
        if (boundaries) get_regions_radial( cp);

        /* Save current configuration for computing tcf's at interval
          corr_interval. */
        if (do_tcf && ((nconfig-1) % corr_interval==0)){
            add_config( cp, &origin_list);
            save_config=1;
        }

        /* save current config */
        if (do_print){
            FILE *file= print_handle->file;
            if (nconfig==1) print_types(file);
            if (print_binary)
                print_config_binary( file, cp, cp->flags);
            else
```

```
            print_config( file, cp, cp->flags);
        fflush(file);
    }

    /* calculate properties for this config */
    if (do_rdp) calc_rdp( cp);
    if (do_pdf) calc_pdf( cp);
    if (do_msd) calc_msd( &origin_list, cp);
if (do_cisf) calc_cisf( &origin_list, cp);
if (do_tisf) calc_tisf( &origin_list, cp);
if (do_risf) calc_risf( &origin_list, cp);
if (do_dv) {
     calc_dv( &origin_list, cp, ctlayers);
    };
if (do_ct) {
     calc_ct( &origin_list, cp, ctlayers);
    };
if (layers!=0) {
get_regions_radial( cp);
if (ltype==1) calc_layer_risf(&origin_list, cp, layers, mol_layers);
if (ltype==2) calc_layer_tisf(&origin_list, cp, layers, mol_layers);
if (ltype==0) calc_layer_cisf(&origin_list, cp, layers, mol_layers);
}; /* END if */

    if (do_vacf) calc_vacf(&origin_list, cp);
    if (do_M) calc_M( cp);
    if (acfs) calc_acf( acfs, cp, solv_type);
    if (do_frag){
        fragment_analysis( cp, prefix, (nconfig==1));
    }
    if (do_h2o){
```

```
        if (!h2o){
            h2o=new_h2o_props( cp, dR_radial, corr_length, corr_interval,
                               solv_type, pos_type, neg_type, R_nn_solv,
R_nn_pos_solv,
                               R_nn_neg_solv, R_nn_pos_neg, hbond_thresh);
        }
        calc_h2o_props( h2o, cp);
      }
      if (do_h2o_energy && ((nconfig-1)%energy_int==0)){
        calc_h2o_energy( cp, prefix, solv_type, pos_type, neg_type);
      }
      if (do_dip_avg){
        calc_h2o_dip_avg(cp);
      }
      if (do_dip_dens){
            calc_h2o_dip_dens(cp);
      }

  cleanup:
      if (!save_config){ free_config(cp); cp=0x0;}
  }

  /* cleanup after all configs have been processed */
  if (nconfig){
      fprintf(outfile, "Computed properties from %d configs in file %s.\n",
            nconfig, config_file);
  }

  if (do_rdp)  print_rdp( outfile);
  if (do_pdf) print_pdf( outfile, prefix);
  if (do_msd) print_msd( outfile, dt);
```

```
  if (do_cisf||do_risf||do_tisf) print_isf( outfile, dt, Qsq,nspec);
  if (layers) print_layer_isf( outfile, dt, Qsq, nspec, layers);
  if (do_ct||do_dv) print_ct( outfile, dt, nspec, Qsq , ctlayers);
  if (do_vacf) print_vacf( outfile, dt);
  if (do_M) print_M( outfile);
  if (acfs) print_acf( acfs, outfile, prefix);
  if (h2o) print_h2o_props( h2o, dt, outfile, prefix);
  if (do_dip_avg) print_h2o_dip_avg( outfile, prefix);
  if (do_dip_dens) print_h2o_dip_dens( outfile, prefix);
  if (do_h2o_energy) clear_h2o_energy();

exit:
  if (acfs) free_acf( acfs);
  if (h2o) free_h2o_props(h2o);
  if (cp) free_config(cp);
  if (origin_list) free_config_list(origin_list);
  free_keylist(&list_local);
  if (prefix) free( prefix);
  if (config_file) free( config_file);
  if (config_handle) close_handle( config_handle, IO_FLAGS_DFLT);
  if (print_handle) close_handle( print_handle, IO_FLAGS_DFLT);
  if (boundaries) FREE_VECTOR( boundaries);

  return;
}
```

//Layers

```
static struct {
  int ncorr;            /* correlation length (max time diff) */
  int nmols;
```

```
  double ***re;                /* array of real portion of Fs(Q,t) */
  int *nconfig;                /* array for number of configs at a gived time diff. */
  int **ndist;
  double qfac;
} layer_isf;

/* Clears memory for Fs(Q,t) before calc initializing */
int clear_layer_isf(){

  if (layer_isf_handle){
    close_handle( layer_isf_handle, IO_FLAGS_DFLT);
    layer_isf_handle = 0x0;
  }
    if (layer_isf.re){
    FREE_MATRIX_ARRAY(layer_isf.re,layer_isf.ncorr);
    layer_isf.re = 0x0;
  }
    layer_isf.nmols = 0;
    layer_isf.ncorr = 0;
    layer_isf.qfac=0;
  if (layer_isf.nconfig){
    free(layer_isf.nconfig);
    layer_isf.nconfig = 0x0;
  }
    if (layer_isf.ndist){
    IFREE_MATRIX(layer_isf.ndist);
    layer_isf.ndist = 0x0;
  }
  return(0);
}
```

```
/* allocates memory for the F(Q,t), trans, rot, or total*/
int init_layer_tisf( FILE *msgfile, char *prefix, int corr_length, int nspec, int layers, int
*mol_layers, int ltype, double qfac){
   char buf[1000], *layer_isf_file=0x0;
   int err=0, ntri, numQ = 0, i,j,k;

   clear_layer_isf();
   layer_isf.qfac=qfac;
   layer_isf.nmols =nspec;
/* set up isf file */
   if (ltype==2)
   sprintf( buf, "%s.ltisf", prefix);
   if (ltype==1)
   sprintf( buf, "%s.lrisf", prefix);
   if (ltype==0)
   sprintf( buf, "%s.lcisf", prefix);
   printf("%s\n", buf);
   layer_isf_file= strdup(buf);
   if(!(layer_isf_handle=OPEN_HANDLE( layer_isf_file, "w", IO_FLAGS_DFLT))){
sprintf(buf, "init_layer_isf: Couldn't create file %s", layer_isf_file);
            handle_error(buf);
            ++err;
            goto exit;
      };

   layer_isf.ncorr= corr_length+1;
   printf("Starting allocations\n");
```

```
mol_layers=IALLOC_VECTOR(nspec);
IINIT_VECTOR(mol_layers, 0, nspec);
  /* This next set of statements allocates memory for the arrays */
isf.re= ALLOC_MATRIX_ARRAY( corr_length+1, layers, 7);
printf("Allocated real portion...");

layer_isf.nconfig= IALLOC_VECTOR( corr_length+1 );
IINIT_VECTOR(layer_isf.nconfig, 0, corr_length+1);

layer_isf.ndist= IALLOC_MATRIX( corr_length+1, layers);
IINIT_VECTOR(layer_isf.ndist[0], 0, (corr_length+1)*layers);   printf("Finished
allocations...");
   printf("Finished initializations\n");
   printf("ncorr = %d\n", layer_isf.ncorr);

printf("Finished allocations...");
for (i=0;i < (corr_length+1);i++)
        for (j=0;j < layers;j++){
                layer_isf.ndist[i][j]=0;
            for (k=0;k <= 7;k++){
                layer_isf.re[i][j][k] = isf.im[i][j][k]=0.0;
            };
            };

layer_isf.nconfig= IALLOC_VECTOR( corr_length+1 );
IINIT_VECTOR(layer_isf.nconfig, 0, corr_length+1);

layer_isf.ndist= IALLOC_MATRIX( corr_length+1, layers);
IINIT_VECTOR(layer_isf.ndist[0], 0, (corr_length+1)*layers);   printf("Finished
allocations...");
   printf("Finished initializations\n");
```

```
    printf("ncorr = %d\n", layer_isf.ncorr);
exit:

  if (err) clear_layer_isf();
  if (layer_isf_file) free(layer_isf_file);
  return (err);
}



/* CALCULATE Rotational ISF */
int calc_layer_risf( Config **origin_list, Config *cp , int layers,int *mol_layers)
{   Config *cp0, *cp0_next, cp0_temp;                    /* pointers to config 0 */
  int n_types, n_reg, dt;   Molecule *molptr, *mol0ptr;     /* pointers to molecules
in config 1 and 0 */      ThreeVector dR;                 /* R(t)- R(0) */
    double QdR;
      double pi2=3.141592653928*3.141592653928;
    int n_primary, i,k, class;

    /* Loop over config cp0                                */
    /* Initial Condition: cp0 is at the beginning of list of   */
    /*                    configurations                   */
    /*          : cp0 advanced one time step               */
    /* Exit Condition: cp0 has no data (its previous value   */
    /*         was the last configuration in the data set)   */

  for (cp0=*origin_list; cp0!=NULL; cp0=cp0_next){
    cp0_next=cp0->next;
    dt = abs( cp->index - cp0->index);
    if (dt >= layer_isf.ncorr){
```

```
            drop_config( cp0, origin_list);
            continue;
        }

        /* Loop over Molecules                          */
/* Initial Conditions: i=0, mol0ptr has memory address of 1st molecule at t=0      */
/*                     molptr has mem address of 1st molecule at time t            */
/*                  : increment # of molecules, pick next molecule in configs      */
/* Exit condition: molptr=NULL, meaning that there is a molecule present          */

        for (i=0,mol0ptr=cp0->molarray, molptr=cp->molarray; mol0ptr!=NULL;
molptr=molptr->next, mol0ptr=mol0ptr->next, i++) {
            int j, my_layer0, my_layert;
                int xint,yint,zint;
                double
xdiff,ydiff,zdiff,csx,csy,csz,snx,sny,snz,csxy,csxz,csyz,snxy,snxz,snyz;
                double cs2x,cs2y,cs2z,sn2x,sn2y,sn2z,cs2xy,cs2xz,cs2yz;
            ThreeVector *R0, *R;
            ThreeVector d0, d;
            ThreeVector *R0_com, *R_com;            class =
get_molecule_type_index(molptr->typeptr);
            if (class == 0) {               R0_com= &mol0ptr->R_com;
              R_com = &molptr->R_com;
              my_layer0= (cp0->regions) ? cp0->regions[i] : 0;
              for (k=1;k<=2;k++){
                 R = &molptr->primary_atom.positions[k];
                 vector_sub (R_com, R, &d);
                 R0 = &mol0ptr->primary_atom.positions[k];
                 vector_sub (R0_com, R0, &d0);
                 vector_sub( &d0, &d, &dR );
```

```
xint=int(0.5*dR.x);
yint=int(0.5*dR.y);
zint=int(0.5*dR.z);

xdiff=layer_isf.qfac*pi2*(dR.x-2*xint);
ydiff=layer_isf.qfac*pi2*(dR.y-2*yint);
zdiff=layer_isf.qfac*pi2*(dR.z-2*zint);

csx=cos(xdiff);
csy=cos(ydiff);
csz=cos(zdiff);
snx=sin(xdiff);
sny=sin(ydiff);
snz=sin(zdiff);
csxy=csx*csy-snx*sny;
csxz=csx*csz-snx*snz;
csyz=csy*csz-sny*snz;
snxy=snx*csy+sny*csx;
snxz=snx*csz+snz*csx;
snyz=sny*csz+snz*csy;
cs2x=csx*csx-snx*snx;
cs2y=csy*csy-sny*sny;
cs2z=csz*csz-snz*snz;
sn2x=2*snx*csx;
sn2y=2*sny*csy;
sn2z=2*snz*csz;
cs2xy=cs2x*cs2y-sn2x*sn2y;
cs2xz=cs2x*cs2z-sn2x*sn2z;
cs2yz=cs2y*cs2z-sn2y*sn2z;
layer_isf.re[dt][my_layer0][0]+= csx+csy+csz;
layer_isf.re[dt][my_layer0][1]+= csxy+csxz+csyz;
```

```
                    layer_isf.re[dt][my_layer0][2]+= csxy*csxz-snxy-snz;
                    layer_isf.re[dt][my_layer0][3]+= cs2x+cs2y+cs2z;
                    layer_isf.re[dt][my_layer0][4]+= cs2x*csy-sn2x*sny+csx*cs2y-
                            snx*sn2y+cs2x*csz-sn2x*snz+csx*cs2z-snx*sn2z
                            +cs2y*csz-sn2y*snz+csy*cs2z-sny*sn2z;
                    layer_isf.re[dt][my_layer0][5]+= cs2x*csyz-sn2x*snyz +
                                    cs2y*csxz- sn2y*snxz+cs2z*csxy-sn2z*snxy;
                    layer_isf.re[dt][my_layer0][6]+= cs2xy+cs2xz+cs2yz;
                    layer_isf.re[dt][my_layer0][7]+= cs2x*cs2x-sn2x*sn2x +
                            cs2y*cs2y - sn2y*sn2y+cs2z*cs2z-sn2z*sn2z;
                     ++layer_isf.ndist[dt][my_layer0];

                };/* Loop over H's *//*          }; End if (same layer) */
            }; /* EnD if */
        };          /* End loop over molecules */
        ++layer_isf.nconfig[dt];
        }               /* End loop over cp0 */

        exit:

        return(0);
}

/* Center of mass intermediate scattering function */
int calc_layer_cisf( Config **origin_list, Config *cp , int layers,int *mol_layers)
{   Config *cp0, *cp0_next;                                 /* pointers to config 0 */
    int n_types, n_reg, dt;
    Molecule *molptr, *mol0ptr;          /* pointers to molecules in config 1 and 0 */
    ThreeVector dR;                          /* R(t)- R(0) */
    double QdR;
     int n_primary, i,k, class;
```

```
    /* Loop over config cp0                                    */
    /* Initial Condition: cp0 is at the beginning of list of */
    /*                    configurations                       */
    /*          : cp0 advanced one time step                   */
    /* Exit Condition: cp0 has no data (its previous value   */
    /*          was the last configuration in the data set)*/
  for (cp0=*origin_list; cp0!=NULL; cp0=cp0_next){
    cp0_next=cp0->next;
    dt = abs( cp->index - cp0->index);
    if (dt >= layer_isf.ncorr){
        drop_config( cp0, origin_list);
    continue;
      }

/* Exits loop over cp0 without calculating ISF's if time difference too large */

 /* Loop over Molecules                                                    */
/* Initial Conditions: i=0, mol0ptr has memory address of 1st molecule at t=0   */
 /*              molptr has mem address of 1st molecule at time t              */
/*          : increment # of molecules, pick next molecule in configs          */
/* Exit condition: molptr=NULL, meaning that there is a molecule present     */
for (i=0,mol0ptr=cp0->molarray, molptr=cp->molarray; mol0ptr!=NULL;
molptr=molptr->next, mol0ptr=mol0ptr->next, i++) {
        int j, my_layer0, my_layert;
        int xint,yint,zint;
        double xdiff,ydiff,zdiff,csx,csy,csz,snx,sny,snz,csxy,csxz,csyz,snxy,snxz,snyz;
        double cs2x,cs2y,cs2z,sn2x,sn2y,sn2z,cs2xy,cs2xz,cs2yz;
        ThreeVector d0, d;
        ThreeVector *R0_com, *R_com;

        class = get_molecule_type_index(molptr->typeptr);
```

```
if (class == 0) {
  R0_com= &mol0ptr->R_com;                R_com = &molptr->R_com;
  my_layer0= (cp0->regions) ? cp0->regions[i] : 0;
  vector_sub( R0_com, R_com, &dR );
  xint=int(0.5*dR.x);
  yint=int(0.5*dR.y);
  zint=int(0.5*dR.z);

  xdiff= layer_isf.qfac*pi2*(dR.x-2*xint);
  ydiff= layer_isf.qfac*pi2*(dR.y-2*yint);
  zdiff= layer_isf.qfac*pi2*(dR.z-2*zint);

  csx=cos(xdiff);
  csy=cos(ydiff);
  csz=cos(zdiff);
  snx=sin(xdiff);
  sny=sin(ydiff);
  snz=sin(zdiff);
  csxy=csx*csy-snx*sny;
  csxz=csx*csz-snx*snz;
  csyz=csy*csz-sny*snz;
  snxy=snx*csy+sny*csx;
  snxz=snx*csz+snz*csx;
  snyz=sny*csz+snz*csy;
  cs2x=csx*csx-snx*snx;
  cs2y=csy*csy-sny*sny;
  cs2z=csz*csz-snz*snz;
  sn2x=2*snx*csx;
  sn2y=2*sny*csy;
  sn2z=2*snz*csz;
  cs2xy=cs2x*cs2y-sn2x*sn2y;
```

```
            cs2xz=cs2x*cs2z-sn2x*sn2z;
            cs2yz=cs2y*cs2z-sn2y*sn2z;
            layer_isf.re[dt][my_layer0][0]+= csx+csy+csz;
            layer_isf.re[dt][my_layer0][1]+= csxy+csxz+csyz;
            layer_isf.re[dt][my_layer0][2]+= csxy*csxz-snxy-snz;
            layer_isf.re[dt][my_layer0][3]+= cs2x+cs2y+cs2z;
            layer_isf.re[dt][my_layer0][4]+= cs2x*csy-sn2x*sny+csx*cs2y-
                    snx*sn2y+cs2x*csz-sn2x*snz+csx*cs2z-snx*sn2z +cs2y*csz-
                    sn2y*snz+csy*cs2z-sny*sn2z;
            layer_isf.re[dt][my_layer0][5]+= cs2x*csyz-sn2x*snyz+cs2y*csxz-
                                            sn2y*snxz+cs2z*csxy-sn2z*snxy;
            layer_isf.re[dt][my_layer0][6]+= cs2xy+cs2xz+cs2yz;
            layer_isf.re[dt][my_layer0][7]+= cs2x*cs2x-sn2x*sn2x+cs2y*cs2y-
                                            sn2y*sn2y+cs2z*cs2z-sn2z*sn2z;

            ++layer_isf.ndist[dt][my_layer0];

             }; /* EnD if class==0*/          };        /* End loop over molecules */
          ++layer_isf.nconfig[dt];          }            /* End loop over cp0 */
    exit:
    return(0);
}



/* CALCULATE TOTAL ISF */int calc_layer_tisf( Config **origin_list, Config *cp , int
layers,int *mol_layers){   Config *cp0, *cp0_next;                              /*
pointers to config 0 */
  int n_types, n_reg, dt;
  Molecule *molptr, *mol0ptr;          /* pointers to molecules in config 1 and 0 */
 ThreeVector dR;                        /* R(t)- R(0) */
```

```
        double QdR;
        int n_primary, i,k, class;
        /* Loop over config cp0                                    */
        /* Initial Condition: cp0 is at the beginning of list of */      /*
configurations          */
        /*              : cp0 advanced one time step          */
        /* Exit Condition: cp0 has no data (its previous value   */
        /*          was the last configuration in the data set)*/   for (cp0=*origin_list;
cp0!=NULL; cp0=cp0_next){
        cp0_next=cp0->next;
        dt = abs( cp->index - cp0->index);
        if (dt >= layer_isf.ncorr){
                drop_config( cp0, origin_list);              continue;}  /* Exits loop over cp0
without calculating ISF's if time difference too large */

        /* Loop over Molecules                                    */
 /* Initial Conditions: i=0, mol0ptr has memory address of 1st molecule at t=0 */
/*                  molptr has mem address of 1st molecule at time t        */
/*              : increment # of molecules, pick next molecule in configs                */
/* Exit condition: molptr=NULL, meaning that there is a molecule present       */

for (i=0,mol0ptr=cp0->molarray, molptr=cp->molarray; mol0ptr!=NULL;
molptr=molptr->next, mol0ptr=mol0ptr->next, i++) {
                int j, my_layer0,my_layert;
                int xint,yint,zint;
                double xdiff,ydiff,zdiff,csx,csy,csz,snx,sny,snz,csxy,csxz,csyz,snxy,snxz,snyz;
                double cs2x,cs2y,cs2z,sn2x,sn2y,sn2z,cs2xy,cs2xz,cs2yz;
                ThreeVector *R0, *R, *R0_com, *R_com;

                class = get_molecule_type_index(molptr->typeptr);
                if (class == 0) {
```

```
R0_com= &mol0ptr->R_com;                R_com = &molptr->R_com;
my_layer0= (cp0->regions) ? cp0->regions[i] : 0;
        for (k=1;k<=2;k++){
                R = &molptr->primary_atom.positions[k];
        R0 = &mol0ptr->primary_atom.positions[k];
        vector_sub( R0, R, &dR );

        xint=int(0.5*dR.x);
        yint=int(0.5*dR.y);
        zint=int(0.5*dR.z);

        xdiff= layer_isf.qfac*pi2*(dR.x-2*xint);
        ydiff= layer_isf.qfac*pi2*(dR.y-2*yint);
        zdiff= layer_isf.qfac*pi2*(dR.z-2*zint);

        csx=cos(xdiff);
        csy=cos(ydiff);
        csz=cos(zdiff);
        snx=sin(xdiff);
        sny=sin(ydiff);
        snz=sin(zdiff);
        csxy=csx*csy-snx*sny;
        csxz=csx*csz-snx*snz;
        csyz=csy*csz-sny*snz;
        snxy=snx*csy+sny*csx;
        snxz=snx*csz+snz*csx;
        snyz=sny*csz+snz*csy;
        cs2x=csx*csx-snx*snx;
        cs2y=csy*csy-sny*sny;
        cs2z=csz*csz-snz*snz;
        sn2x=2*snx*csx;
```

```
                    sn2y=2*sny*csy;
                    sn2z=2*snz*csz;
                    cs2xy=cs2x*cs2y-sn2x*sn2y;
                    cs2xz=cs2x*cs2z-sn2x*sn2z;
                    cs2yz=cs2y*cs2z-sn2y*sn2z;
                     layer_isf.re[dt][my_layer0][0]+= csx+csy+csz;
                    layer_isf.re[dt][my_layer0][1]+= csxy+csxz+csyz;
                     layer_isf.re[dt][my_layer0][2]+= csxy*csxz-snxy-snz;
                     layer_isf.re[dt][my_layer0][3]+= cs2x+cs2y+cs2z;
                    layer_isf.re[dt][my_layer0][4]+= cs2x*csy-sn2x*sny+csx*cs2y-
                            snx*sn2y+cs2x*csz-sn2x*snz+csx*cs2z-snx*sn2z
                            +cs2y*csz-sn2y*snz+csy*cs2z-sny*sn2z;
                     layer_isf.re[dt][my_layer0][5]+= cs2x*csyz-sn2x*snyz
                            +cs2y*csxz-sn2y*snxz+cs2z*csxy-sn2z*snxy;
                     layer_isf.re[dt][my_layer0][6]+= cs2xy+cs2xz+cs2yz;
                    layer_isf.re[dt][my_layer0][7]+= cs2x*cs2x-sn2x*sn2x
                            +cs2y*cs2y-sn2y*sn2y+cs2z*cs2z-sn2z*sn2z;
                    ++layer_isf.ndist[dt][my_layer0];


                        ++layer_isf.ndist[dt][my_layer0];
                    };/* Loop over H's */      /*          }; END if (same region)? */
                  }; /* EnD if (class==0)*/                 };          /* End loop over
molecules */
          ++layer_isf.nconfig[dt];          }          /* End loop over cp0 */

     exit:

     return(0);
}
```

```
int print_layer_isf( FILE *msgfile, double dt, double qunit,int nspec, int layers){
   FILE *file=layer_isf_handle->file;
   register int i, j, k,mol, q;
   int n_types, n_reg, n_class;
   /* print properties key */
   print_regions(file);

   fprintf( file, "Q values\n");
   fprintf( file, "fs1=%d\t%fs2=%d\tfs3=%d\tfs4=%d\n",qunit,qunit*sqrt(2),
        qunit*sqrt(3),qunit*2);
   fprintf( file, "fs5=%d\t%fs6=%d\tfs7=%d\tfs8=%d\n",qunit*sqrt(5),qunit*sqrt(6),
        qunit*sqrt(8),qunit*4);



   fprintf( file, "%d species\n", nspec);    fprintf( file, "time(ps)\t");
   for (i=0;i<layers;i++){
        fprintf( file, "#layer %d\t",i);
        for (j=0;j<7;j++)
                fprintf( file, "fs%d\t",j);
        };
   fprintf(file, "\n");

   for(i=0; i<layer_isf.ncorr; ++i){
           double rv=0, iv=0;
           double dum=0,dum3=0;

           rv=iv=0;
```

```
            if (layer_isf.nconfig[i]==0) continue;
            if (dt){
              fprintf( file, "%9.3f\t", i*dt*AU_TO_FS*.001);
            } else {
            fprintf( file, "%15d", i);
            }

            for (j=0;j<layers;j++){
                        dum=layer_isf.ndist[i][j];
                  fprintf(file, "%6d\t",int(dum));
                        dum=1/dum;
                        dum3=dum/3;

                  fprintf(file, "%11.4e\t", layer_isf.re[i][j][0]*dum3);
                  fprintf(file, "%11.4e\t", layer_isf.re[i][j][1]*dum3);
                  fprintf(file, "%11.4e\t", layer_isf.re[i][j][2]*dum);
                  fprintf(file, "%11.4e\t", layer_isf.re[i][j][3]*dum3);
                  fprintf(file, "%11.4e\t", layer_isf.re[i][j][4]*dum3*0.5);
                  fprintf(file, "%11.4e\t", layer_isf.re[i][j][5]*dum3);
                  fprintf(file, "%11.4e\t", layer_isf.re[i][j][6]*dum3);
                  fprintf(file, "%11.4e\t", layer_isf.re[i][j][7]*dum3);
            }
            fprintf(file,"\n");
      }

exit:
      clear_isf();

      return(0);
}
```

```
//Legendre polynomials for unit dipole and unit bond vector time correlation functions

static struct {
   double ***u0ut;    /* 2d array of u0 dot ut [dt][mols][layer] */
   int ncorr;
   double ***u0ut2;   /* 2d array of u0 dot ut [dt][mols][layer] */
   double ***u0ut3;   /* 2d array of u0 dot ut [dt][mols][layer] */
   int *nconfig;      /* array for number of configs at a gived time diff. */
   int **ndist;
   int nmols;
} u0_ut;

/* Clears memory for Csubl(Q,t) before calc initializing */
int clear_ct(){

   if (ct_handle){       close_handle( ct_handle, IO_FLAGS_DFLT);
      ct_handle = 0x0;
   }
      if (u0_ut.u0ut){
      FREE_MATRIX_ARRAY(u0_ut.u0ut, u0_ut.ncorr);
      u0_ut.u0ut = 0x0;
   }
      if (u0_ut.u0ut2){
      FREE_MATRIX_ARRAY(u0_ut.u0ut2, u0_ut.ncorr);
      u0_ut.u0ut2 = 0x0;
   }
      if (u0_ut.u0ut3){
      FREE_MATRIX_ARRAY(u0_ut.u0ut3, u0_ut.ncorr);
      u0_ut.u0ut3 = 0x0;
   }
   if (u0_ut.nconfig){
```

```
    free(u0_ut.nconfig);
    u0_ut.nconfig = 0x0;
  }
    if (u0_ut.ndist){
    IFREE_MATRIX(u0_ut.ndist);
    u0_ut.ndist = 0x0;
  }
    u0_ut.nmols=0;
    u0_ut.ncorr=0;
  return(0);
}

/* allocates memory for the ct(t) */int init_ct( FILE *msgfile, char *prefix, int
corr_length, int nspec,int ctlayers){
  char buf[1000], *ct_file=0x0;
  int err=0, ntri, i,j,k;
  int layers;

  clear_ct();
  u0_ut.nmols = nspec;
  layers = ctlayers+1;
  printf("%d layers\n", layers);
  /* set up isf file */
  sprintf( buf, "%s.ct", prefix);
  ct_file= strdup(buf);
  if(!(ct_handle=OPEN_HANDLE( ct_file, "w", IO_FLAGS_DFLT))){
    sprintf(buf, "init_ct: Couldn't create file %s", ct_file);
        handle_error(buf);
        ++err;
        goto exit;
    };
```

```
u0_ut.ncorr = corr_length + 1;
printf("Starting allocations\n");

  /* This next set of statements allocates memory for the arrays */
u0_ut.u0ut = ALLOC_MATRIX_ARRAY((corr_length+1), nspec, layers);
printf("Allocated u0 dot ut array...");
u0_ut.u0ut3 = ALLOC_MATRIX_ARRAY((corr_length+1), nspec, layers);
u0_ut.u0ut2 = ALLOC_MATRIX_ARRAY((corr_length+1), nspec, layers);

u0_ut.nconfig = IALLOC_VECTOR((corr_length+1));
IINIT_VECTOR(u0_ut.nconfig, 0, (corr_length+1));

u0_ut.ndist = IALLOC_MATRIX((corr_length+1), layers);
IINIT_VECTOR(u0_ut.ndist[0], 0, ((corr_length+1)*layers));

printf("Finished allocations...");
for (k=0;k<=ctlayers;k++){
   for (i=0;i<u0_ut.ncorr;i++)
            for (j=0;j<nspec;j++){
                u0_ut.u0ut[i][j][k] = 0.0;
                u0_ut.u0ut2[i][j][k] = 0.0;
                u0_ut.u0ut3[i][j][k] = 0.0;
                };
   };

   printf("Finished initializations\n");
exit:

  if (err) clear_ct();
  if (ct_file) free(ct_file);
```

```
   return (err);
}


/* cl(t) */

int calc_dv( Config **origin_list, Config *cp , int ctlayers){
   Config *cp0, *cp0_next;                  /* pointers to config 0 */
   int n_types, n_reg, dt, my_layer0;
   Molecule *molptr, *mol0ptr; /* pointers to molecules in config 1 and 0 */
      ThreeVector dR;                       /* R(t)- R(0) */
      int n_primary, i,H,k, class;
      /* Loop over config cp0                                  */
      /* Initial Condition: cp0 is at the beginning of list of */
      /*                    configurations                     */
      /*          : cp0 advanced one time step                 */
      /* Exit Condition: cp0 has no data (its previous value   */
      /*         was the last configuration in the data set)*/
   for (cp0=*origin_list; cp0!=NULL; cp0=cp0_next){
      cp0_next=cp0->next;
      dt = abs( cp->index - cp0->index);
      if (dt >= u0_ut.ncorr){
            drop_config( cp0, origin_list);
            continue;}
/* Exits loop over cp0 without calculating ct's if time difference too large */
      /* Loop over Molecules                                   */
 /* Initial Conditions: i=0, mol0ptr has memory address of 1st molecule at t=0 */
/*                  molptr has mem address of 1st molecule at time t        */
 /*            : increment # of molecules, pick next molecule in configs                */
/* Exit condition: molptr=NULL, meaning that there is a molecule present       */
```

```
 for (i=0,H=0,mol0ptr=cp0->molarray, molptr=cp->molarray; mol0ptr!=NULL;
molptr=molptr->next, mol0ptr=mol0ptr->next,i++) {
        int j;
        ThreeVector *R0, *R, *R0_1, *R_1;
        ThreeVector d0,d0_1, d, d_1;
        ThreeVector *R0_com, *R_com;
        ThreeVector u0,u;
        double u0dotut;
        class = get_molecule_type_index(molptr->typeptr);
        if (class == 0) {
          R0= &mol0ptr->R_com;
          R = &molptr->R_com;
          my_layer0=(cp0->regions) ? cp0->regions[i] : 0;
          if (molptr->typeptr->primary_atom.n>1){

        /* H-H vectors --- Removed
          R0=&mol0ptr->primary_atom.positions[1];
          R0_1=&mol0ptr->primary_atom.positions[2];
          vector_sub(R0_1,R0,&d0);
          normalize_vector(&d0);

          R=&molptr->primary_atom.positions[1];
          R_1=&molptr->primary_atom.positions[2];
          vector_sub(R_1,R,&d);
          normalize_vector(&d);
          u0dotut=vector_product(&d,&d0); */

        /*Dipole Vector */
          if (!mol0ptr->axes) mol_axes(mol0ptr, 00);
          mol_axes(molptr, 00);
          u0dotut= vector_product(&mol0ptr->axes[0],
```

```
                                &molptr->axes[0]);


                u0_ut.u0ut[dt][H][my_layer0]+= u0dotut;
                u0_ut.u0ut2[dt][H][my_layer0]+= u0dotut*u0dotut;
                u0_ut.u0ut3[dt][H++][my_layer0]+= u0dotut*u0dotut*u0dotut;
                ++u0_ut.ndist[dt][my_layer0];
            };
            }; /* EnD if */
            };          /* End loop over molecules */

            ++u0_ut.nconfig[dt];
            }                   /* End loop over cp0 */

    exit:

    return(0);
}



/* cl(t) */
int calc_ct( Config **origin_list, Config *cp , int ctlayers){
   Config *cp0, *cp0_next;                  /* pointers to config 0 */
   int n_types, n_reg, dt, my_layer0;
   Molecule *molptr, *mol0ptr; /* pointers to molecules in config 1 and 0 */
     ThreeVector dR;                    /* R(t)- R(0) */
     int n_primary, i,H,k, class;
     /* Loop over config cp0                          */
     /* Initial Condition: cp0 is at the beginning of list of */
     /*                    configurations             */
```

```
    /*            : cp0 advanced one time step          */
    /* Exit Condition: cp0 has no data (its previous value   */
    /*          was the last configuration in the data set)*/
  for (cp0=*origin_list; cp0!=NULL; cp0=cp0_next){
    cp0_next=cp0->next;
    dt = abs( cp->index - cp0->index);
    if (dt >= u0_ut.ncorr){
        drop_config( cp0, origin_list);
        continue;}
/* Exits loop over cp0 without calculating ct's if time difference too large */
    /* Loop over Molecules                          */
/* Initial Conditions: i=0, mol0ptr has memory address of 1st molecule at t=0 */
/*              molptr has mem address of 1st molecule at time t        */
/*          : increment # of molecules, pick next molecule in configs                */
/* Exit condition: molptr=NULL, meaning that there is a molecule present       */
for (i=0,H=0,mol0ptr=cp0->molarray, molptr=cp->molarray; mol0ptr!=NULL;
molptr=molptr->next, mol0ptr=mol0ptr->next,i++) {
        int j;
        ThreeVector *R0, *R, *R0_o, *R_o;
        ThreeVector d0, d;
        ThreeVector *R0_com, *R_com;
        double u0dotut;
        double tmp;
        class = get_molecule_type_index(molptr->typeptr);
        if (class == 0) {


          R0_com= &mol0ptr->R_com;
          R_com = &molptr->R_com;
          my_layer0=(cp0->regions) ? cp0->regions[i] : 0;
```

```
        for (k=1;k<=2;k++){
          R = &molptr->primary_atom.positions[k];

          vector_sub (R_com, R, &d);

          normalize_vector(&d);
          R0 = &mol0ptr->primary_atom.positions[k];

          vector_sub (R0_com, R0, &d0);

          normalize_vector(&d0);

          u0dotut=vector_product(&d0,&d);


          u0_ut.u0ut[dt][H][my_layer0]+= u0dotut;
          u0_ut.u0ut2[dt][H][my_layer0]+= u0dotut*u0dotut;
          u0_ut.u0ut3[dt][H++][my_layer0]+= u0dotut*u0dotut*u0dotut;
          ++u0_ut.ndist[dt][my_layer0];
          };      /* End loop over H's */
        }; /* EnD if */
       };          /* End loop over molecules */

      ++u0_ut.nconfig[dt];
      }               /* End loop over cp0 */

exit:

return(0);
}
```

```
int print_ct( FILE *msgfile, double dt, int nspec, double Qsquare, int ctlayers){
  FILE *file=ct_handle->file;   register int i, j, mol, l, lcounter;
  int n_types, n_reg, n_class;      double ***Ct;
    double prefactor[6];
    double Q;
    double C[5];
    double jl;     /* jl represents the lth spherical bessel function */
    Q= 0.964*sqrt((double)(Qsquare));

    Ct=ALLOC_MATRIX_ARRAY(4,u0_ut.ncorr,(ctlayers+2));
    INIT_VECTOR(Ct[0][0],0,4*u0_ut.ncorr*(ctlayers+2));
  /* print properties key */
    fprintf( file, "%d species, %d corr\n", nspec, u0_ut.ncorr);
    print_regions(file);

    for (lcounter=0;lcounter<=ctlayers;lcounter++){
      fprintf(file, "\nlayer = %d\n", lcounter);      fprintf(file, "0th term = %11.5e\n",
      prefactor[0]=(sin(Q)/Q)*(sin(Q)/Q));
      fprintf( file, "layer %d\n", lcounter);
      fprintf( file, "#\tt/ps\tu0dotut\n", nspec);
       for (l=1;l<4;l++){
        prefactor[l]=(2*l+1);
        if (l==1)
            jl=(-cos(Q)/Q + sin(Q)/(Q*Q));
        else if (l==2)
            jl=(-sin(Q)/Q - 3*cos(Q)/(Q*Q) + 3*sin(Q)/(Q*Q*Q) );
        else if (l==3)
        jl=(cos(Q)/Q - 5*sin(Q)/Q - 12*cos(Q)/(Q*Q) + 12*sin(Q)/(Q*Q*Q));
        prefactor[l]*=(jl*jl);
        }; /*end for loop of prefactor */
```

```
        fprintf(file, "C(t) with Qd = %e Angstroms\n", Q);
        fprintf( file, "#\tt/ps\t\tC1(t)\t\tC2(t)\t\tC3(t)\t\tC(t)\n");
        for(i=0; i < u0_ut.ncorr; ++i){
           int dum;
           double temp;
           if (u0_ut.nconfig[i]==0) continue;
           if (dt){
             fprintf( file, "%d\t%9.3f\t", u0_ut.ndist[i][lcounter],i*dt*AU_TO_FS*.001);
            } else {
            fprintf( file, "%15d", i);
            }
            for (mol=0;mol<nspec;mol++){
             if (u0_ut.nconfig[i]){
                temp= (double)(u0_ut.ndist[i][lcounter]);
                Ct[1][i][lcounter]+=u0_ut.u0ut[i][mol][lcounter]/temp;
Ct[2][i][lcounter]+=(1.5*(u0_ut.u0ut2[i][mol][lcounter])/temp-0.5);
Ct[3][i][lcounter]+=0.5*(5*(u0_ut.u0ut3[i][mol][lcounter])/temp-
3*(u0_ut.u0ut[i][mol][lcounter])/temp);                    }
            } /* End loop over molecules */

        /*Ct[1][i][lcounter]/= temp;
        Ct[2][i][lcounter]/= temp;
        Ct[3][i][lcounter]/= temp; */       fprintf( file, "%11.5e\t%11.5e\t%11.5e\n",
Ct[1][i][lcounter],Ct[2][i][lcounter],Ct[3][i][lcounter]);
        } /*Corr*/
        } /* layer */
 exit:
   clear_ct();

   return(0);
} /* FCN */
```

Chapter 4

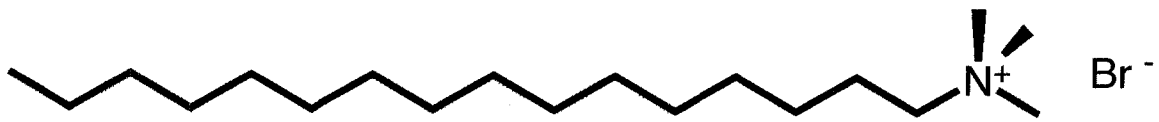
Effects of Surfactant Charge on Water Dynamics within Quaternary Reverse Micelles

4.I. Introduction

In reverse micelles (RMs) formed from single-tailed surfactants such as cetyl trimethylammonium bromide (CTAB, Fig 3.1a) and sodium dodecyl sulfate (SDS, Fig 3.1b), a second amphiphilic molecule, known as a cosurfactant, is necessary to stabilize the system. Typically, cosurfactants are long-chain alcohols such as pentanol, heptanol, or decanol. Cosurfactants are thought to minimize repulsive interactions between surfactants by intercalating between them, as illustrated in Fig. 3.2.

Quaternary reverse micellar systems have been studied primarily through ultrafast laser spectroscopy,[1,2] a technique which requires the addition of a fluorescent dye molecule to the RM system. Reports in literature show [1,2] that the dynamics of water within quaternary RMs are significantly hindered relative to those of bulk water. However, the location of the dye molecule within the RM system has not been unequivocally determined. In some cases, the general consensus is that the dye molecule resides in the interface of the RMs, in which case it would be sensitive to the motion of interfacial water, but not bulk-like water.

a)



b)

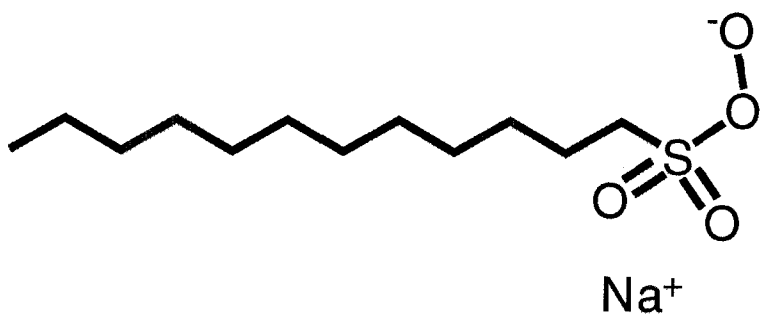


Figure 4.1. (a) CTAB and (b) SDS surfactants

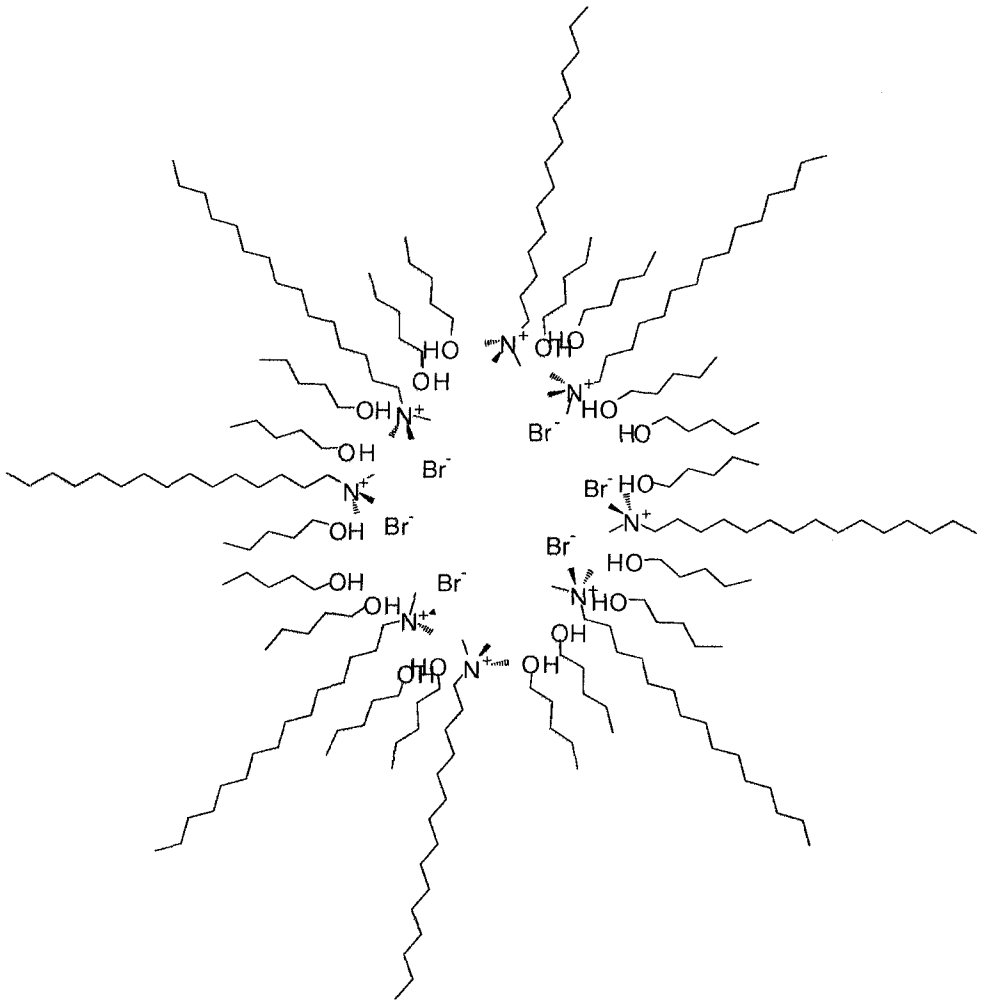


Figure 4.2. Illustration of a CTAB reverse micelle, with pentanol as a cosurfactant.

4.II. Experimental Methods

Quaternary RM samples were prepared at $w_0 = 5$, cosurfactant:surfactant ratio of 5:1, and surfactant concentration of 150mM. To study the effects of surfactant charge on water dynamics, we use deuterated surfactants, custom-synthesized to have the same chain length on their hydrocarbon tails. Deuterated sodium dodecyl sulfate (dSDS, 98% isotopic substitution) and deuterated cetyltrimethylammonium bromide (dC_{12}dTAB, 98% isotopic substitution) were purchased from Robert K. Thomas at Oxford University. The subscript on the dC_{12}dTAB indicates a 12-carbon chain, rather than the standard 16-carbon chain, on CTAB. With this modification, both deuterated surfactants will have the same chain length, making the surfactant headgroup the only difference between the two surfactants. Perdeuterated cyclohexane (cyclohexane-*d12*), deuterated pentanol (pentan-*d11*-ol), and D_2O were obtained from Cambridge Isotope Labs and used as received.

The surfactants dSDS and dC_{12}dTAB were purified by lyophilization before use in experiments. For samples measuring H_2O dynamics, ultrapure water (Milli-Q, 18.2MΩ-cm) was used for the samples. Samples measuring D_2O dynamics required deuteration of the alcoholic hydrogen on the pentan-*d11*-ol to produce pentanol-*d12*. This isotopic exchange was accomplished by repeatedly mixing D_2O in a 3:1 ratio with pentan-*d11*-ol at room temperature, separating at low temperature, and drying. Samples were prepared by mass, with H_2O sample and D_2O background compositions within 0.5%.

The QNS experiments on dSDS and dC_{12}dTAB RMs were carried out at the Argonne National Lab (ANL), Intense Pulsed Neutron Source (IPNS) division, using the QENS instrument. The energy resolution of this instrument is 85 μeV. The Q range was

0.36 - 2.53 $Å^{-1}$. Data were accumulated for 18 hours for each sample and background, at a fixed temperature of 300 K.

Experiments were also performed on dC_{12}dTAB RMs using the Disk-Chopper Spectrometer (DCS) at the NIST Center for Neutron Research (NCNR). The DCS was operated in high resolution mode, with an energy resolution of ~20 μeV, and a Q range between 0.08 – 1.92 $Å^{-1}$. Data were accumulated for 6 hours each for the sample and background, at a fixed temperature of 300 K.

4.III. Results and Discussion

It appears that the QENS instrument does not have the capability of resolving the slow translational motion of water within the quaternary reverse micelles. At low-Q, the spectra appear to be resolution-broadened delta functions. There is no quasielastic broadening of the QNS spectra until high-Q, where water translational and rotational motions may contribute to the signal.

These QNS results are consistent with those of ultrafast experiments, [1,2] in which the water dynamics in $w_0 = 5$ quaternary RMs are found to occur on the 10^2 ps timescale, which cannot be resolved by the QENS instrument. These ultrafast experiments relied on the use of a fluorescent probe molecule that was most likely located in the RM interface, where water dynamics would be the slowest.[3,4] Our QNS results suggest that the dynamics of core water are also very slow.

Analysis using the DCS instrument was problematic as well; the instrument itself may have been able to resolve the slow translational motion of water within these samples. However, the requirement of a cosurfactant creates several problems. First, the

cosurfactant:surfactant ratio is equal to the w_0 of the RM, meaning that the alcoholic proton accounts for 1/3 of the signal from the hydrogen atoms in the system, and this contribution to the signal is not removed by a background subtraction. Since the alcoholic proton is labile, we were forced to use perdeuterated pentanol for the background, and deuterated pentanol for the sample. Because of this, background-subtraction does not remove the contribution of the alcoholic hydrogen atom pentanol. We did not have enough instrument time remaining to compensate for the low signal to noise.

At this time, it is not clear exactly how pentanol partitions within the RM. Pentanol is slightly soluble in both water and cyclohexane, so it can exist in one of three places – the water pool (which is fairly unlikely), the interface, and the nonpolar phase. Most likely, the amount of pentanol that has dissolved in the nonpolar phase is non-negligible. [5] The pentanol in at least one of these environments may be moving on a timescale similar to that of the water dynamics we wish to measure, which increases the difficulty of the analysis of this experiment.

4.IV. Conclusions

This study of water dynamics in quaternary reverse micelles does not yield any hard results, but we confirm that water dynamics in quaternary RMs are uniformly slow, and we have a better idea about what is necessary to investigate this system. To gain information about the translational motion of water in these quaternary RM's, we need to use an instrument with a higher resolution than the QENS instrument at ANL. The signal to noise ratio on the DCS data indicates that the data should be accumulated for about 24 hours each for the sample and background set. Finally, the partitioning of the cosurfactant

in this system needs to be evaluated. It is possible to analyze a samples consisting of pentan-*d11*-ol in cyclohexane-*d12*, which would allow us to determine the timescale of the rotational and translational dynamics of the cosurfactant in the nonpolar solvent. Knowing about cosurfactant dynamics in the nonpolar phase may simplify the analysis of the data.

Based on the experiments presented in this chapter, Zhao[6] recently conducted further QNS experiments on dC_{12}dTAB samples, using the high resolution DCS instrument at NCNR and accumulating data for 24 hours each for sample and background. Although their data from $0.08 \leq Q \leq 0.4$ is dominated by the resolution function, their medium-Q data shows quasielastic broadening. They interpret their spectra to show translational motion of the cosurfactant on the same timescale as that of the water molecules within the reverse micelle, which makes the analysis nonintuitive.

Zhao[6] is currently analyzing experimental small angle neutron scattering (SANS) and neutron spin-echo (NSE) data which may aid in the analysis of the data presented in this chapter. A discussion of SANS and NSE is beyond the scope of this dissertation. Briefly, these techniques will allow Zhao and co-workers to more accurately determine the partitioning of the cosurfactant between the water pool and nonpolar phase, and the effect of the cosurfactant on the RM interface, respectively. SANS will supply structural information such as coherent structure factor, which is related to the pair correlation function $g(r)$ by a Fourier transform. By monitoring the shape fluctuations and modulus of curvature of the interface in the NSE experiments, Zhao et al. should be able to determine how much of the cosurfactant is embedded in and how it affects the rigidity of the RM interface.

References and Notes

(1) Corbeil, E. M.; Levinger, N. E. *Langmuir* **2003**, *19*, 7264.

(2) Corbeil, E. M.; Riter, R. E.; Levinger, N. E. *J. Phys. Chem. B* **2004**, *108*, 10777.

(3) Harpham, M. R.; Ladanyi, B. M.; Levinger, N. E. *J. Phys. Chem. B* **2005**, *109*, 16891.

(4) Harpham, M. R.; Ladanyi, B. M.; Levinger, N. E.; Herwig, K. W. *J. Chem. Phys.* **2004**, *121*, 7855.

(5) Palazzo, G.; Lopez, F.; Giustini, M.; Colafemmina, G.; Ceglie, A. *J. Phys. Chem. B* **2003**, *107*, 1924.

(6) Zhao, X., private communication.

Chapter 5

Concluding Remarks

Our goals were to investigate water dynamics in binary mixtures with dimethyl sulfoxide and within ionic reverse micelles (RMs), and through quasielastic neutron scattering (QNS) experiments and molecular dynamics simulations. Unlike traditional methods of analyzing water motion, such as steady-state or ultrafast spectroscopy, QNS experiments do not require a probe molecule. Probe molecules are typically moderately large dye molecules, whose location and influence on the system may not be fully understood. Molecular dynamics (MD) simulations provide another method of analyzing the system, and may be used to calculate properties which may not be experimentally observable.

MD simulations and QNS experiments each have a set of assumptions associated with them. In the case of MD simulation, assumptions are made in the functional form of the potentials used, as well as in the rigidity of the molecules simulated. Interpretation of QNS data relies on assumptions about the functional form of the model. In the case of the RM samples, the dynamics extracted from the experimental QNS data represent an average of the motion of all water molecules with a timescale of diffusion that is detectable by the instrument. When used in tandem, QNS measurements and MD

simulations can each be used to determine the validity of the assumptions made by the other and to assist in the interpretation of the other. We successfully investigated mixtures of water and dimethyl sulfoxide over a composition range from 0.124 to 0.75 mole fraction DMSO. One of the goals of this investigation was to evaluate the assumptions made in QNS experiments performed[1] recently. We found that the decoupling approximation made in the experimental analysis were less reasonable in this system than in the RM systems. We found that water motion was slower in our simulations than in the QNS experiments. The models we use are known to give slower water dynamics than are experimentally observed. Despite this, the simulation model does reproduce trends in water mobility quite well.

Our analysis of water rotational motion within the water-DMSO system shows that water rotation is neither isotropic not diffusive. Based on this, the isotropic rotational diffusion model used in the analysis of experimental data may not be the most useful model. We began a preliminary analysis of water rotational motion in binary mixtures using the extended jump rotational model suggested by Laage and Hynes.[2] Our initial results are promising, and this model may more accurately describe the water rotational motion within this system than do traditional rotational models. Based on our preliminary results, Ladanyi and Levinger groups may re-analyze the rotational motion of water in binary mixtures, as well as that of water in RMs, with this extended jump rotational model.

Our investigation of dynamics within Aerosol-OT reverse micelles was largely successful. Results from experiment and simulation showed similar trends in water mobility as a function of RM size and surfactant counterion. Water molecules within

larger RMs exhibited faster motion than did those in smaller RMs. Changing the surfactant counterion from Na^+ to K^+ also increased water mobility within the RMs.

The simulation model successfully predicted experimental spectra, and was used to explain experimental results and to investigate properties that could not be observed experimentally. We checked the experimental assumption that rotational and translational motions were decoupled, and found that this assumption is quite reasonable. We were able to perform a layer analysis of the water dynamics by location within the simulated RM, and came to several conclusions based on that. The simulations showed that the water dynamics varied by layer; water molecules closer to the interface were hindered, in translation and rotation, relative to water molecules further from the interface. Furthermore, the MD simulations determined that the QNS experiments were unable to resolve the slower motion of water molecules near the RM interface.

We were less successful in our efforts to explore the quaternary RM system. We found that the water translational dynamics were too slow to resolve with the QENS spectrometer at ANL. QNS experiments performed with the disk-chopper spectrometer had too low a signal to noise ratio for us to analyze the data. In working with this data, we began to think more about the location and effects of the cosurfactant on this system. The cosurfactant pentanol has very low solubility in water. However, pentanol is quite soluble in the cyclohexane that was used as a nonpolar phase in these experiments. Pentanol in water, cyclohexane, and in the interface may exhibit varying mobility. This would have no effect on the QNS analysis if the timescale of pentanol motion was different from that of water. However, we do not know for a fact that pentanol motion is on a significantly different timescale than that of water. Based on these results and

questions, a further investigation of this system is being performed by Zhao, Herwig, and Levinger to determine the location(s) of the cosurfactant, the effect of the cosurfactant on interfacial rigidity, as well as the timescale of cosurfactant motion.

Construction of a simulation model for this quaternary RM system is significantly more involved than that of the ternary RM system. At this point, we lack experimental knowledge. We do not know the location of the cosurfactant and the effects of the cosurfactant on structural and dynamical properties of the system.

Our investigation of water rotational motion was quite interesting. In the binary mixtures, water interacted with DMSO through hydrogen bonding. Because of this, time correlation functions for water OH vectors rotations experienced more hindrance than those for dipole vector rotations in mixtures rich in DMSO. This result was the opposite of what we saw in the aqueous RM system, in which the ionic surfactant and counterions severely hindered water dipole vector rotations, and sometimes enhanced water librational motion. When we changed the counterion on the surfactant, we found that the dipole vector reorientation was less hindered by the large Cs^+ counterions than by smaller K^+ and Na^+ counterions.

In conclusion, investigating water dynamics in reverse micelles and in binary mixtures through QNS experiments and MD simulations furthered our understanding of these systems, and gave us greater confidence in the validity of the assumptions used in either method.

References and Notes

(1) Bordallo, H. N.; Herwig, K. W.; Luther, B. M.; Levinger, N. E. *J. Chem. Phys.* **2004**, *121*, 12457.

(2) Laage, D.; Hynes, J. T. *Science* **2006**, *311*, 832.