Simulation Studies of Slow Dynamics of Hydration Water in Lysozyme: Hydration Level Dependence and Comparison with Experiment using New Time Domain Analysis

by

Chansoo Kim

B.S. in Computer Science and B.S. in Nuclear Science (2003)
Seoul National University

M.S. in Nuclear Science (2005)
Seoul National University

Submitted to the Department of Nuclear Science and Engineering
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Nuclear Science and Engineering

at the
Massachusetts Institute of Technology
[September]
August 2008

Signature of Author .....
Department of Nuclear Science and Engineering
August 27, 2008

Certified by .....
Sow-Hsin Chen
Professor of Nuclear Science and Engineering
Thesis Supervisor

Read by .....
Sidney Yip
Professor of Nuclear Science and Engineering
Thesis Reader

Accepted by .....
Jacquelyn C. Yanch
Chairman, Department Committee on Graduate Student

Simulation Studies of Slow Dynamics of Hydration Water in Lysozyme: Hydration Level Dependence and Comparison with Experiment using New Time Domain Analysis

by

Chansoo Kim

Abstract

A series of Molecular Dynamics (MD) simulations using the GROMACS® package has been performed in this thesis. It is used to mimic and simulate the hydration water in Lysozyme with three different hydration levels (h = 0.3, 0.45 and 0.6). In this thesis, GROMACS is used in an innovative way, because it is applied to investigate mainly behaviors of water molecules than those of biopolymers, which has been originally the simulation target of GROMACS package. The protein (Lysozme) – water system is simulated using TIP4P water potential to model the slow dynamics of the hydration water at low temperatures well. Besides the simulation works, a new time domain Relaxing-Cage Model (RCM) fitting methodology is introduced in the experiment part. We use the Gaussian functions to convert the Intermediate Scattering Functions (ISF) from Quasi-Elastic Neutron Scattering (QENS) experiments from frequency domain to time domain. Then, the Relaxing-Cage Model (RCM) fitting is performed on the converted ISF in time domain. The average translational relaxation time of the MD simulation is compared with the QENS experiment. Three different hydration levels are designed and used in the MD simulations. Other quantities, which can be used to observe the crossover phenomena of the hydration water, such as the number of hydrogen bonds, Mean Squared Displacement (MSD), the structure factors $S(Q)$ and the radial distribution functions $g(r)$ are compared at the different hydration levels. We have found that experiment and simulation agree well in terms of the crossover temperature T_L at hydration level 0.3: $T_L(experiment)$ is 226 K and $T_L(simulation)$ is 221 K, and those are in the crossover temperature range of 220 ± 10 K. The crossover temperature obtained from the average translational relaxation time increases as the hydration level becomes lower. The crossover phenomenon is also observed in the number of hydration bonds between water and water. It only appears in hydrogen bonds between water and water (not in bonds between water and Lysozyme case), so we can say that water can trigger the biomolecules' functionality. The main observations of this thesis is that the crossover temperature depends on the hydration level even though the crossover phenomenon occurs at any hydration level and water possibly triggers the biomolecules' functionality.

Thesis Supervisor: Sow-Hsin Chen
Title: Professor

Acknowledgment

I would first like to thank my supervisor, Professor Sow-Hsin Chen, for his guidance throughout my time at MIT. His integrated view on science made my experience rich and fruitful. I am fortunate to be one of the many students over the years to call him a supervisor. Many thanks should also go to Professor Sidney Yip for his advised and kind suggestions to my research and life. I would also like to thank Professor Francesco Mallamace in University of Messina, Italy for his advises and many support. Their enthusiasm and positive attitude toward science was always encouraging, and gave me great strength to overcome various hardships.

Experimental work in this thesis was done in NIST Center for Neutron Research (NCNR) at National Institute of Standards and Technology (NIST). I would like to thank Dr. John Copley, Dr. Antonio Faraone and Juscelino Leao at NIST for their great help on the QENS experiment at Disk-Chopper Time-of-Flight Spectrometer (DCS). Performing the experiment with Xiang-Qiang Chu was fruitful. Without help from them, this work cannot be done. I would like to thank Dr. Marco Lagi for his great help. He opened the door to Molecular Dynamics simulations for me by teaching me with deep understanding.

I would like to thank all the professors in the Department of Nuclear Engineering of the Seoul National University, because I could not have reached MIT without a sound undergraduate education provided from them.

My thanks should also go to my group members who spent valuable years with me. It was always happy to discuss with them. They are Professor Li Liu, Dr. Dazhi Liu, Xiang-Qiang Chu, Yang Zhang, Dr. Marco Lagi, Dr. Jianlan Wu and Dr. Matteo Broccio. I also thank all the supports from Korean students in our department. I feel fortunate to have lots of wonderful discussions with a Chemist, Dr. Jongnam Park about researches and life.

I thank my wife, Yeon-Joo (김연주) for her warm and sweet support for our Cambridge life. Without her, I even could not begin to build my life as a researcher. I truly love her. Our little girl, Soo-Yeon (김수연) has always been a strong motivation for my life.

Finally, I would like to thank my parents, Jeong-Ki Kim (김정기) and Sung-Sook Lee (이성숙) for their support throughout my entire life. No amount of success in my life would be possible if not for them. I love them sincerely.

I dedicate this thesis to my grandfather Sun-Base Kim (김선배) in heaven.

Table of Contents

Abstract 3

Acknowledgment 4

Chapter 1. Introduction 8

1.1. Background 8

1.2. Hydration Water in Biopolymers 10

1.3. Motivation 11

1.4. Computer Simulation Study: Molecular Dynamics (MD) 12

1.5. Experiments Study: Incoherent Neutron Scattering Spectroscopy 16

1.5.1. Elastic scattering 17

1.5.2. Inelastic scattering 17

1.5.3. Quasi-Elastic scattering 18

1.6. Analysis Model: Relaxing-Cage Model (RCM) 18

Chapter 2. Molecular Dynamics (MD) Simulation: GROMACS 24

2.1. GROMACS Package 24

2.1.1. Important GROMACS commands 25

2.1.2. GROMACS installing and running procedure 27

2.1.2.1. Installation with the required programs 27

2.1.2.2. Running MD simulations 28

2.2. Applications: Post-Processing the MD Simulation Trajectories File 33

2.2.1. Number of hydrogen bonds 34

2.2.2. Autocorrelation functions and relaxation time, τ_{relax} 35

2.2.3. Radial Distribution Function (RDF), $g(r)$ 36

2.2.4. Mean Square Displacement (MSD) and diffusion constant, D 37

2.2.5. Intermediate Scattering Functions (ISF) 37

2.2.6. Structure factor, $S(Q)$ 40

2.3. Simulation Configurations 41

2.3.1. Hydration water in Lysozyme 41

2.3.2. Three different hydration levels configurations 42

Chapter 3. Analysis Methodology for ISF Data 47

3.1. An Time Domain Analysis for Experimental ISF Data 47

3.1.1. Gaussian approximation for raw ISF data in frequency domain 49
3.1.2. Fourier Transform (FT) of the Gaussian approximated functions 54
3.1.3. RCM fitting analysis in time domain 57
3.2. MD Simulation ISF Data Analysis 62
3.2.1. Global least squares method for the ISF data from MD simulation 62
3.2.2. RCM fitting analysis in time domain 63
Chapter 4. Results – Comparison and Discussion 66
4.1. Average Translational Relaxation Time $\langle \tau_T \rangle$ and ISF 66
4.1.1. Comparison between the experiment and MD simulation 67
4.1.2. Comparison among MD simulation results of three different hydration levels 69
4.2. Mean Square Displacement (MSD) and diffusion constant, D 74
4.3. Number of hydrogen bonds 80
4.4. Structure factor, $S(Q)$ 85
4.5. Radial Distribution function, $g(r)$ 92
4.6. Autocorrelation functions and the hydrogen bond relaxation time 99
Chapter 5. Conclusion and Future Work 106
Chapter 6. Appendix 108
6.1. All the results of ISF in MD simulation 108
6.2. Publications 139
6.3. Matlab® Code for QENS ISF fitting by Gaussian Functions 139
Chapter 7. Bibliography 140

Chapter 1. INTRODUCTION

1.1. Background

The study of supercooled and glassy water is motivated by the well known observation of anomalous behavior in thermodynamic as well as transport properties in bulk liquid water, that at ambient temperature and pressure, although quantitatively small, becomes increasingly significant at supercooled temperatures [1, 2]. It has been found that the extrapolated thermodynamic response functions and characteristic relaxation times appear to diverge, according to power laws, at a singular temperature $T_s = 228$ K [3]. Although this anomaly has sparked an enormous interest in the scientific community, a coherent explanation of the apparent singularity in supercooled water has not yet emerged. The basic reason for this is the fact that T_s is buried below the homogeneous nucleation temperature of water, $T_H = 235$ K [4], in an inaccessible temperature range for bulk supercooled water. This hampers a direct experimental investigation of the thermodynamics and the dynamics in the critical region in order to confirm, or to rule out one of the proposed scenarios, for example, the liquid-liquid phase transition and the associated second (liquid-liquid) critical point in water [2, 5].

While many methods can be used to measure the macroscopic properties of water inside biological, geological and engineering systems, experimental techniques capable of determining the structure and dynamics of water molecules under nanometer-scale confinement are scarce. Neutron scattering is a method of choice because of the extraordinarily large neutron incoherent scattering cross section of hydrogen atoms, rendering high sensitivity to hydrogen motion unmatched by optical and X-ray spectroscopy [6]. Furthermore, judicious H-D substitution or application of high magnetic fields and neutron polarization analysis can enhance significantly the contrast between targeted hydrogen groups against the host medium for structural determination. The spatio-temporal range that neutron scattering method probes encompasses the 0.1 ~ 100 Å and 10^{-4} ~ 20 *ns* realm that matches well the length and time scale of short-to-long range order structure, molecular diffusion and atomic vibrations in water. Additionally, the measured neutron

spectra, expressed as the time Fourier transform of the Intermediate Scattering Function (ISF), can be quantitatively compared with those calculated by computer Molecular Dynamics (MD) simulations or theoretical modeling.

Besides being relevant for many industrial and biological applications, water confined in nanoporous matrices and forming the hydration layer on the surface of biopolymers allow us to enter into the inaccessible temperature range for supercooled bulk water. Therefore, both the structure and dynamics of water in confined geometries have been studied using MD simulations and different experimental techniques [7]. In particular, previous neutron scattering experiments [8, 9] clearly showed that the ISF of water in vycor glass exhibits the α-relaxation at long time at a lower equivalent temperature, much the same as supercooled bulk water, as shown in an MD simulation of SPC/E model [10].

Search for the predicted [5] first-order liquid-liquid transition line and its end point, the second low-temperature critical point [1, 2] in water, has been hampered by intervention of the homogeneous nucleation process. However, by hydrating water on the surface of biomolecules, such as Lysozyme which used in this report, we have been able to study the dynamical behavior of water in a temperature range down to 160 K, without crystallization and it can be related to the functionality and transformation of that biopolymer. Using high-resolution QENS method and Relaxing-Cage Model (RCM) [11] for the data analyses, we determine the temperature and pressure dependencies of the average translational relaxation time $< \tau_T >$ for the confined supercooled water [12-14].

The target system of the MD simulation, here, is the Lysozyme hydration water, which is one of the 2-D confined hydration water. In reality, thanks to Lysozyme, water becomes hydration water around its surface and will not be crystallized. In addition, one can focus on the relationship between the protein and water. There exists a temperature called the glass transition temperature of biopolymers, in which the biopolymers sets the limit of biological activities via conformational flexibility. MD simulations can simulate this situation clearly and correctly with an appropriate potential selection for water molecules.

I show a series of MD simulations with three different hydration levels predict that crossver phenomenon occurs clearly. These studies achieve deep supercooling without crystal nucleation due to the small system size and short observation time explored compared with experimental result. Here, four-site transferable interaction potential (TIP4P) for water molecule is considered to be the most appropriate model for simulating and mimicking neutron experiments when used with a simple spherical cutoff for the long-ranged electrostatic interactions [48].

This thesis summarizes all of the simulation results which come from three different hydration levels, h = 0.3, 0.45 and 0.6. Finally crossover phenomenon shown by the average translational relaxation time is given. Comparison among all hydration levels results is given with many post-processed quantities such as hydrogen bonds. Using the results of number of hydrogen bonds, it is possible to show that water acts important role in the glass transition temperature of biopolymers. I also summarize a newly suggested time-domain analysis method to fit the QENS ISF spectra. In addition, comparison between experiment and MD simulation of h = 0.3 case is described.

1.2. Hydration Water in Biopolymers

Water molecules in a protein solution may be classified into three categories:

(1) the bound internal water;

(2) the surface water, i.e., the water molecules that interact with the protein surface strongly; and

(3) the bulk water.

The bound internal water molecules, which occupy internal cavities and deep clefts, are extensively involved in the protein-solvent H-bonding and play a structural role in the folded protein itself. The surface water, which is usually called the hydration water, is the first layer of water that interacts with the solvent-exposed protein atoms of different chemical character, feels the topology and roughness of the protein surface, and exhibits the slow dynamics. Finally, water that is not in direct contact with the protein surface but

continuously exchanges with the surface water has properties approaching that of bulk water.

The hydration water is believed to have an important role in controlling the biofunctionality of the protein. In this article, we shall present some neutron scattering results of hydrated protein powder. In this case, the hydration water represents the water in category (1) and (2) mentioned above.

Functions of many globular proteins generally show sharp slowing-down around the temperatures between 200 and 240 K [15-16]. Experimental [17, 18, 55] and computational [19-21] results show a sharp increase of the Mean Square Displacement (MSD) $< x^2 >$ of hydrogen atoms in proteins at about T_C = 220 K, which suggests that the dynamic transition (sometime called the glass transition) may be occurring in the proteins at this temperature. There is strong evidence that this dynamic transition of protein is solvent-induced, since the hydration water of a protein also shows a kind of dynamic transition around similar temperature [22-24].

1.3. Motivation

Since water is the most important substance in the world and the one of the best subjects to investigate the complex systems, many researchers are interested in water. To draw an extensively detailed phase diagram with investigating the 'no man's land', we need to collect various data.

What have been motivating me to be involved in this research and to do the complex liquids research are to answer the following questions. Since our group has been using proteins to make water hydration water to prevent its freezing, I want to ask a question: is there any difference in properties such as the average translational relaxation time between hydration levels? What are different aspects between QENS experiment and MD simulations in terms of the average translational relaxation time, which shows the crossover of water in cusp-like behavior changes in the temperature changes? Does any other way exist to fit the ISF data

come from the QENS experiments in the frequency domain with an improvement of computation speed and reliability?

1.4. Computer Simulation Study: Molecular Dynamics (MD)

Molecular Dynamics (MD) is a sort of computer simulation. Different from Monte Carlo (MC) simulations, which is purely random based one, it is a deterministic simulation using potential function to compute the future configuration of the system at next time step. In a simulation configuration atoms and molecules interact with other for a given period of time by approximations of known physics, giving a view of the motion of the atoms. Since normal molecular system consists of a large number of particles (atoms or molecules), it is impossible to analytically solve a given system to obtain properties of such complex system. However, this MD simulation is easily attack this problem through numerical methods of Physics. MD probes the relationship between molecular structure, movement and function.

MD gives researchers a powerful way by providing good relation between laboratory experiments and theory, which is directly related to the MD simulations. Therefore, the MD simulations can be called as a 'virtual experiments'. MD techniques allow detailed time and space resolution into representative behavior in phase space. In this thesis one will see that Quasi-Elastic Neutron Scattering (QENS) experiments on hydration water in Lysozyme as the laboratory experiments and a series of the MD simulations as theory. As you will see, QENS has limitation of time resolutions, represented by frequency, while MD does not. One problem of MD that I point out here is that "the longer simulation time, the longer trajectories given".

Actually, before the advent of good computers having great computational capabilities, MD has been developed very slowly. In the first time of MD generation physicist only can imagine its picture of configuration consisting of molecules without any mathematical calculations [25]. Thanks to the computers, we can track of those particles' movement with ease (but still with long simulation time).

MD simulation was originated from Physics area in the late 1950s [26], but it is also possibly applied to other fields such as Biology, Economics, and Sociology with an appropriate modification of potential field (interaction term). It is applied today mostly in Material Science and Biology. This MD simulation could also be used as Agent-Based Model (ABM), which is a popular simulation approach in Sociology and Economics, since those are all computer simulations. In other words, many physics concept in simulation area, which basically is based on Statistical Mechanics, can be usefully applied to the Economics and Sociology (in the sociological application, one molecule can be treated one person) [27]. This can explain why many physicists (focusing Statistical Physics and complex system) such as Professor Eugene Stanley and Professor Barabasi are able to research Economics and Sociology.

As slightly touched on before, MD simulations actually stem from a hypothesis of Statistical Mechanics, the ergodic hypothesis: "statistical ensemble averages are equal to time averages of the system". This is the reason that people refer MD as "Statistical Physics by Numbers". It predicts the future position (motion) of every particle by computing nature's forces among those particles [28]. Figure 1 simply depicts the concept of its computation.

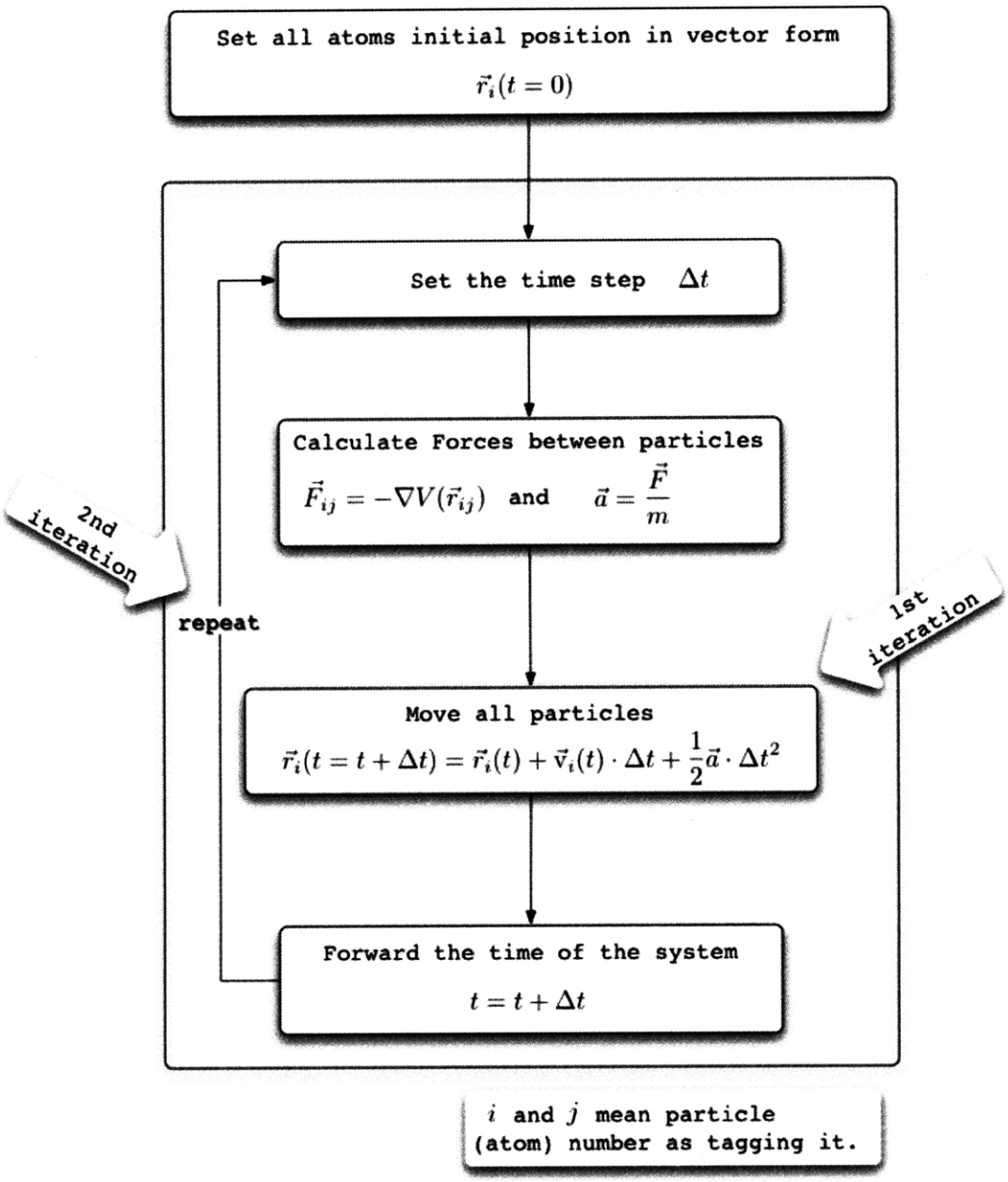


Figure 1. Using Gaussian Approximation to remove the asymmetry before the Fourier Transform

MD requires the definition of a potential function, which can allow the code to calculate interaction among particles and to move those particles for the next time step (means future). Various potential fields, which can be empirical or theoretical, are able to define the forces used in MD simulations as potential functions.

Most force fields in chemistry are empirical and consist of a summation of bonded forces associated with chemical bonds, bond angles, and bond dihedrals, and non-bonded forces associated with van der Waals force and electronic charge. Those can be treated as parameters to control the force fields. Some experimental physical properties such as elastic constancs, lattice parameters and spectroscopic measurements can also be used. In Chemistry force fields usually use preset bonding arrangements, while potentials in Physics can vary system coordinates and bond breaking.

As a simple choice, one can imagine that the total potential energy comes from the sum of energy contributions between only "pairs of atoms": we call this "pair potential", because only pairs can interact with each other based on a given potential. Lennard-Jones potential is the good example for this pair potential to be used for calculating van der Waals forces. For the ionic lattice, Born model is used as a pair-potential as another example. It has Coulomb's law for pair ions, a short-range repulsion by Pauli's exclusion principle and dispersion. In this case the non-boded energy can also be calculated by summing over interactions between the particles of the system. However, many-body potential computes the effects of three or more particles interacting with each other. In many-body potentials, one cannot simply sum over all pairs of atoms to obtain the energy, because this type of many-body interactions are calculated explicitly as higher-order terms.

Because of the non-local nature of non-bonded interactions, all of the possible weak interactions between all particles in the system are included. Its calculation is the bottleneck in the speed of MD. To achieve a higher computation speed, MD usually has an option to set cutoff radii to ignore bonds shorter than that. If one needs more accurate and finer levels of detail regardless of computation time, Quantum Mechanics potentials can be alternatively used to MD. For example, in a simulation configuration, a bulk of the system is

basically treated classically, but a small region is treated as a quantum system, usually undergoing a chemical transformation.

As mentioned above, in this thesis MD is extensively used to simulate three different configuration of hydration water in Lysozyme with varying its hydration level.

1.5. Experiments Study: Incoherent Neutron Scattering Spectroscopy

QENS and Inelastic Neutron Scattering (INS) techniques offer many advantages for the study of single particle dynamics of confined water. The main reason is that the total scattering cross section of hydrogen is much larger than that of atoms in for example silica or carbon, composed of oxygen and silicon or carbon in the confined substrates. Furthermore, the neutron scattering of hydrogen atoms is mostly incoherent so that QENS and INS spectra reflect, essentially, the self-dynamics of the hydrogen atoms in water. Combining this dominant cross section of hydrogen atoms with the use of spectrometers having different energy resolutions, we can study the molecular dynamics of water in a wide range of time-scale, encompassing picoseconds to tens of nanoseconds. In addition, investigating different Q values (Q being the magnitude of the wave vector transfer in the scattering) in the range from $0.2\ \text{Å}^{-1} \le Q \le 2.0\ \text{Å}^{-1}$, the spatial characteristics of water dynamics can be investigated at the sub-nanometer level.

It can be shown generally [29] that the double differential scattering cross section is proportional to the self-dynamic structure factor of hydrogen atoms $S_H(Q,E)$ through the following relation:

$$\frac{d^2\sigma_H}{d\Omega dE} = N\frac{\sigma_H}{4\pi\hbar}\frac{k_f}{k_i}S_H(Q,E) \qquad \text{(1.1)}$$

where $E = E_i - E_f = \hbar\omega$ is the energy transferred by the neutron to the sample; $\hbar\vec{Q} = \hbar\vec{k}_i - \hbar\vec{k}_f$, the momentum transferred in the scattering process; and N, the number of scattering centers in the scattering volume. The self-dynamic structure factor, $S_H(Q,E)$

embodies the elastic, quasielastic and inelastic scattering contributions. It can be expressed as a Fourier transform of the self-ISF of a typical hydrogen atom according to:

$$S_H(Q,E) = \frac{1}{2\pi\hbar}\int_{-\infty}^{\infty} dt e^{-i\omega t} F_H(Q,t) \quad \text{(1.2)}$$

$F_H(Q,t)$ is the density-density time correlation function of the tagged hydrogen atom being measured by the neutron scattering. It is, thus, the primary quantity of theoretical interest related to the experiment. It can be calculated by a model, such as the RCM, and by an MD simulation based on a phenomenological potential model of water.

1.5.1. Elastic scattering

For analysis of the elastic incoherent scattering intensity from hydrogen atoms when they are bound in space, it can be shown that

$$S_H(Q,0) = B\exp\left(-Q^2\left\langle u_H{}^2\right\rangle\right) \quad \text{(1.3)}$$

where $\left\langle u_H{}^2\right\rangle$ is the projection of the mean-square displacement of the hydrogen atoms in the direction of Q vector, and B, a constant. Therefore, $\left\langle u_H{}^2\right\rangle$ can be determined experimentally by measuring the peak height of $S_H(Q,0)$ as a function of Q.

1.5.2. Inelastic scattering

From the inelastic scattering intensity dominated by incoherent scattering from hydrogen atoms, the Q-dependant vibrational Density-Of-States of hydrogen atoms can be obtained by

$$G_H(Q,E) = \frac{2M_H}{\hbar^2}\frac{E}{n(E)+1}\left\langle \frac{\exp\left(Q^2\left\langle u_H{}^2\right\rangle\right)}{Q^2} S(Q,E)\right\rangle \quad \text{(1.4)},$$

where M_H is the mass of hydrogen atom and $n(E)$ is the Bose-Einstein distribution function, and $\langle\cdots\rangle$ represents the average over all observed Q values.

The true hydrogen DOS is obtained in the $Q \to 0$ limit of the $G_H(Q,E)$. In practice, $Q \to 0$ limit means $Q < 1\text{Å}^{-1}$ in the case of water:

$$G_H(E) = \lim_{Q=0} G_H(Q,E) \quad \text{(1.5)}.$$

1.5.3. Quasi-Elastic scattering

In principle, the single-particle dynamics of bulk or confined water should include both the translational and the rotational motions of a rigid water molecule. Given the fact that in the process of QENS data analysis, we only focus our attention to ISF with $Q \leq 1.1\text{Å}^{-1}$, we can safely neglect the contribution of the rotational motion to the total dynamics [30], which means $F_H(Q,t) \approx F_T(Q,t)$, where $F_T(Q,t)$ is the translational part of the ISF.

1.6. Analysis Model: Relaxing-Cage Model (RCM)

During the past several years, we have developed the RCM for the description of the translational and the rotational dynamics of water at supercooled temperatures. This model has been tested with MD simulations of SPC/E water, and has been found to be accurate. It has been used to analyze many QENS data from supercooled bulk water as well as interfacial water [31-35].

On lowering the temperature below the freezing point, around a given water molecule, there is a tendency to form a hydrogen-bonded, tetrahedrally coordinated first and second neighbor shells (we call it cage). At short times, less than 0.05 *ps*, the center of mass of a

water molecule performs vibrations inside the cage. At long times, longer than 1.0 *ps*, the cage eventually relaxes and the trapped particle can migrate through the rearrangement of a large number of particles surrounding it. Therefore, there is a strong coupling between the single particle motion and the density fluctuations of the fluid. The mathematical expression of this physical picture is the so-called RCM.

The RCM assumes that the short-time translational dynamics of the tagged (or the trapped) water molecule can be treated approximately as the motion of the center of mass in an isotropic harmonic potential well provided by the mean field generated by its neighbors. We can, then, write the short time part of the translational ISF in the Gaussian approximation, connecting it to the velocity auto-correlation function, $\langle \vec{v}_{CM}(t) \cdot \vec{v}_{CM}(0) \rangle$, in the following way,

$$F_T^s(Q,t) = \exp\left(-\frac{Q^2}{2}\left\langle r_{CM}^2(t)\right\rangle\right) = \exp\left(-Q^2\left[\int_0^t (t-\tau)\langle \vec{v}_{CM}(0)\cdot\vec{v}_{CM}(\tau)\rangle d\tau\right]\right) \quad \text{..... (1.6).}$$

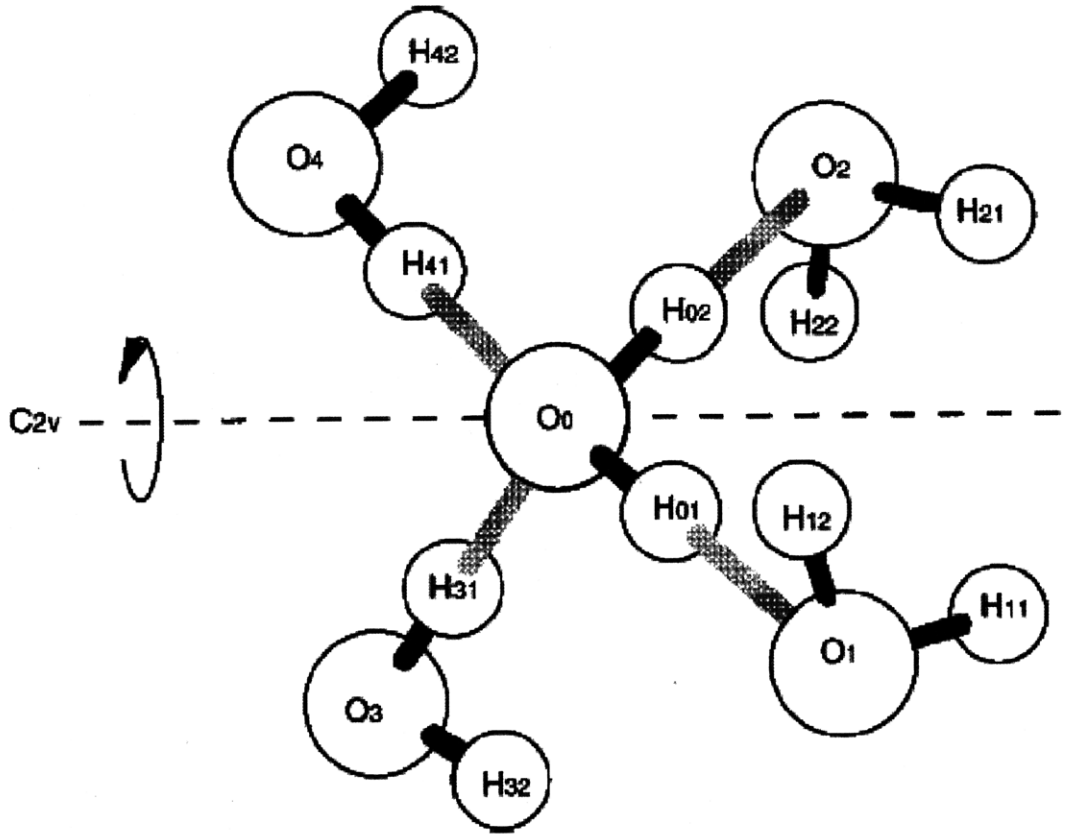


Figure 2. Relaxing-Cage Model (RCM) - One water molecule is confined in the cage which formed by its neighbors through H-atoms at the supercooled temperatures. For short time, it acts like an harmonic oscillations and vibrations inside the cage. However, for long time regime, the cage begins to relax and the molecule escapes (α-relaxation).

Since the translational density of states, $Z_T(\omega)$, is the time Fourier transform of the normalized center of mass velocity auto-correlation function, one can express the mean squared deviation, $\langle r_{CM}^2(t)\rangle$ as follows,

$$\langle r_{CM}^2(t)\rangle = \frac{2}{3}\langle v_{CM}^2\rangle \int_0^\infty d\omega \frac{Z_T(\omega)}{\omega^2}(1-\cos\omega t) \quad \text{(1.7)}.$$

Here, $\langle v_{CM}^2\rangle$ is defined as $\langle v_x^2\rangle + \langle v_y^2\rangle + \langle v_z^2\rangle = 3v_0^2 = 3\frac{k_B T}{M}$. It means the average center of mass square velocity, and M is the mass of water molecule.

Experiments and MD results show that the translational harmonic motion of a water molecule in the cage gives rise to two peaks in $Z_T(\omega)$ at about 10 to 30 meV, respectively [36]. Thus, the following Gaussian functional form is used to represent approximately the translational part of the density of states,

$$Z_T(\omega) = \frac{(1-C)\omega^2}{\omega_1^2\sqrt{2\pi\omega_1^2}}\exp\left[-\frac{\omega^2}{2\omega_1^2}\right] + \frac{C\omega^2}{\omega_2^2\sqrt{2\pi\omega_2^2}}\exp\left[-\frac{\omega^2}{2\omega_2^2}\right] \quad \text{(1.8)}.$$

Moreover, the fit of MD results using Eq. (1.7) gives $C = 0.44$, $\omega_1 = 10.8$ THz, and $\omega_2 = 42.0$ THz. Using Eqs. (1.6-1.7), one can finally get an explicit expression for $F_T^s(Q,t)$,

$$F_T^s(Q,t) = \exp\left\{-Q^2 v_0^2\left[\frac{(1-C)}{\omega_1^2}\left(1-\exp(-\frac{\omega_1^2 t^2}{2})\right) + \frac{C}{\omega_2^2}\left(1-\exp(-\frac{\omega_2^2 t^2}{2})\right)\right]\right\} \quad \text{(1.9)}.$$

The above equation is the short-time behavior of the translational ISF. It starts from unity at $t = 0$ and decays rapidly to a flat plateau determined by an incoherent Debye-Waller factor $A(Q)$, given by

$$A(Q) = \exp\left\{-Q^2 v_0^2\left[\frac{(1-C)}{\omega_1^2} + \frac{C}{\omega_2^2}\right]\right\} = \exp\left[-Q^2 a^2/3\right] \quad \text{(1.10)}.$$

In the above equation 1.10, a is the root mean square vibrational amplitude of the water molecules in the cage, in which the particle is constrained during its short-time movements. According to MD simulations, $a \approx 0.5$ Å is fairly temperature independent [37].

On the other hand, the cage relaxation at long-time can be described by the standard α-relaxation model, according to the Mode-Coupling Theory (MCT), with a stretched exponential having a structural relaxation time τ_T and a stretch exponent β. Therefore, the translational ISF, valid for the entire time range, can be written as a product of the short time part and a long time part,

$$F_T(Q,t) = F_T^s(Q,t)\exp\left[-\left(\frac{t}{\tau_T}\right)^{\beta}\right] \quad \text{(1.11)}.$$

The fit of the MD generated $F_T(Q,t)$ using Eq. (10) shows that τ_T is Q-dependent, obeying the power-law. Therefore, one can see the formula,

$$\tau_T = \tau_0 (aQ)^{-\gamma} \quad \text{(1.12)}$$

where γ is ≤ 2, with a slight dependency on Q, and $\beta < 1$ is slightly Q dependent as well. In the $Q \to 0$ limit, one should approach the diffusion limit, where $\gamma \to 2$ and $\beta \to 1$. Thus the translational ISF can be written as: $F_T(Q,t) = \exp\left[-DQ^2 t\right]$, D being the self-diffusion coefficient. In QENS experiments, this low Q limit is not usually reached, and both β and γ can be considered Q-independent in the limited Q range of $0 \leq Q \leq 1$ [33, 34].

Using RCM one is able to define a Q-independent average translational relaxation time $\langle \tau_T \rangle$,

$$\langle \tau_T \rangle = (\tau_0/\beta)\Gamma(1/\beta) \quad \text{(1.13)}.$$

$\langle \tau_T \rangle$ is a convenient quantity to be extracted from the experimental data by the fitting process of RCM. This quantity can be identified to be proportional to the α-relaxation time which dominates the long-time decay of the ISF in low temperature water. Combining Eqs. (1.1), (1.8), and (1.10), we can calculate the theoretical values of $I_H(Q,\omega)$ and compare it directly with its experimental spectral data.

In actual QENS experiment, one have to take into account the signal coming from the bound hydrogen atoms in $Si(OH)_4$ on the pore surfaces of the silica sample. Denoting the fraction of the elastic scattering coming from the bound hydrogen atom by p we can analyze the experimental data according to the following model,

$$I(Q,\omega) = pR(Q_0,\omega) + (1-p)FT\{F_H(Q,t)R(Q_0,t)\} \quad \text{(1.14).}$$

Here, $F_H(Q,t) \approx F_T(Q,t)$ is the ISF of hydrogen atoms which defines the quasi-elastic scattering, $R(Q_0,t)$ is the experimental resolution function, and the symbol FT denotes the Fourier transform from time t to frequency ω. In the above equation 1.14, one can use four parameters, p,β,γ,τ_0 to extract the information on the average translational relaxation time, τ.

Chapter 2. MOLECULAR DYNAMICS (MD) SIMULATION: GROMACS

2.1. GROMACS Package

GROMACS® (GROningen Machine for Chemical Simulations, GROMACS. '®' will not appear every time after this) is a MD simulation tools originally developed in the University of Groningen [38].

This package is very well known to the people for the following facts, which are just referred from the reference [39].

> "(1) computation of the virial in a single, rather than in a double sum over particles,
> (2) generic representation of all possible periodic box types as triclinic,
> (3) optimized handling of the neighbor list by storage of translation vectors to the nearest neighbor in a periodic system,
> (4) a specialized routine for the calculation of the inverse square root,
> (5) the use of cubic spline interpolation from tabulated values for the evaluation of force/energy."

Actually GROMACS is very popular in protein related research area, so it is now jointly attached to a code, Folding@Home, which is mainly used for protein folding. As one of the most famous MD codes, GROMACS is originally designed to investigate biopolymers' behavior by molecular dynamics simulations, which mainly using classical mechanics. One can say without any difficulty that GROMAS is the code mainly targeted to protein research area.

I want to stress that I have used the GROMACS in an innovative way, because I applied this code mainly to water molecules behaviors. In my research works, I have used the code for simulating water molecules' collective behaviors. Therefore, one can see another useful aspect of the code: GROMACS, which focuses on the proteins, can be possibly used for

simulation for water that has originally been treated as a just solvent. This concept change is one of the 'most important' and the 'most innovative' parts of the thesis.

2.1.1. Important GROMACS commands

An MD simulation needs a configuration of a target system including molecular positions, potential functions, and protein structures. A series of simulations can be started based on that configuration, and this calculation procedure is the essential of the MD simulation approach. Upon being done with the simulation calculations, one can see the trajectories of all molecules in a user-given system. Finally, those trajectories would be converted to any forms that researchers want to have.

One can understand that this is a sort of normal procedure of GROMACS simulations (even MD simulations, also). If one wants to use GROMACS, he needs to know some important commands. Those commands are very basic and essential to follow the procedure written in the above.

Before doing simulations, there should exist a configuration of the target system. For that purpose, GROMACS provide `pdb2gmx`, `genbox`, `editconf`, `genion`, `make_ndx`, and `ffscan`. They are mainly designed for generating topologies and coordinates. `pdb2gmx` converts a Protein Data Bank (PDB) files, which describe protein(s), to configuration files, so that GROMACS can proceed to make an appropriate target system. However, the result file of the `pdb2gmx` does not have solvent, water (H_2O) in the system. `genbox` is a tool that adds water molecules to the target system, which only has protein(s) inside. When the tool solvates the system, setting its density can control the number of solvent (water) molecules. `editconf` edits the target system, so that one can modify the target system with ease. Since GROMACS is not able to begin its calculation for an electrically unstable target system, one cannot avoid modifying system's ionic states. Therefore, `genion` is a required tool to make the system has zero (0) ionic state. For example, for the Lysozyme case, which has been used in this thesis, nine (9) chlorides ions (Cl^-) are substituted with water's oxygen (O) atoms for that purpose. `make_ndx` is the tool that makes index lists of the atoms in the

target system. It can control and name atoms in the target system, so one can select specific atoms to get their characteristics such as mass distribution properties or distances between structures in the post-processing programs. Using `ffscan`, one can check its potential functions, scan, and modify force field data. Actually the force filed data gives a single point energy calculation, which is the essential to the MD calculation, so that it can be modified to get better results.

To run series of the simulations, one has to be familiar with `grompp` and `mdrun`. `grompp` makes a binary file as a run input file from the input files of configurations, parameters, and topologies (force files). All of the input files can be generated by the commands introduced in the above. As one can easily guess from the name, `mdrun` is the main calculation procedure of GROMACS. It performs a simulation based on the binary file generated from `grompp`. As one understand from the Chapter 1, `mdrun` generates simulation results, trajectories files (trr files) by calculating energies for trajectory frames, finding a potential energy minimum and moving molecules using the potential energy values, and calculating the Hessian matrixes. After running `mdrun`, what one gets are trajectories files, which is the direct MD calculation results.

For the post-processing trajectories files to get application values from the simulation results, one has to know one more command. It is also strongly related to viewing trajectories. `trjconv` is the one to do both of jobs. It converts and manipulates trajectories files to other formats, as well as converts trajectories to PDB files, which can be views with many other general MD codes. Moreover, to calculate $S(Q)$ or ISF of water molecules, which is not the main target of the GROMACS, a series of PDB files is required. Reason can be explained by (1) a series of PDB files can be treated as many snapshots of trajectories at each (given) time step, and (2) PDB files are ASCII-type files, while trajectories are binaries.

Important commands can be summarized to

(1) pre-processing and configuration generation: `pdb2gmx`, `genbox`, `editconf`, `genion`, `make_ndx`, and `ffscan`,

(2) MD simulations: `grompp` and `mdrun`, and

(3) post-processing and trajectories conversion: `trjconv`.

Visual Molecular Dynamics (VMD) is a visualization program developed by Theoretical and Computational Biophysics Group at the University of Illinois at Urbana-Champagne leaded by Professor Kalus Schulten [40]. It has a powerful 3-D graphical ability to show the configuration of a target system and to generate a sort of movie showing molecules' movement with simulation time based on trajectories files. This should also be very useful before the simulation, because one can check the system by their eyes.

2.1.2. GROMACS installing and running procedure

2.1.2.1. Installation with the required programs

Since GROMACS is designed to use many CPUs in a parallel way, it has an option if it will use many CPUs simultaneously. Also, if one wants to use GROMACS in that mode, there is a required program called mpi. Including this, installing procedure will be briefly described here.

Since I have been using Apple Macintosh® for my simulation machines, their developer tools package Xcode should be installed from the beginning. It is using Graphical User Interface (GUI) installation package, one can just download and click sometime to set it up on the machine.

Fotran77 compiler is also required, so g77 v.3.4 intel version can be downloaded from http://hpc.sourceforge.net/ installed with the simple commands in the source file directory,

```
./configure
make
make install
```

lam-mpi is a package that allows Operating Systems (OS) use multiple processors in parallel way, so it should be installed upon deciding if parallel computing is required. Its source code, lam-7.1.3.tar.gz can be downloaded from http://www.lam-mpi.org/7.1/download.php . Its installation is performed with

```
./configure
make
make install
```

Fast Fourier Transform is also required for doing Discrete Fourier Transform (DFT), because there is a time- and frequency- domains transformation calculations. It is developed by the MIT researchers, so it is for me to contact them with ease. [41] fftw v.3.1.2 is a good tool for doing this work, and project website locates at http://www.fftw.org/ . Commands for installation would be

```
./configure --enable-float --enable-threads
make
make install
```

After allowing the machine prepared for MD simulations, it is time for one to install GROMACS package. Its official package site is http://www.gromacs.org/ . Its recent source code for Mac, gromacs v.3..3.3 can be installed in a following way,

```
./configure --enable-mpi
make -j 8
make install
make links
```

For VMD, one could probably visit the project webpage, http://www.ks.uiuc.edu/Research/vmd/ . They provide compiled binaries for many platforms, one can just download and use it with ease [42].

2.1.2.2. Running MD simulations

Based on the above section 2.1.1., one is already aware of the procedure of a MD simulation. It may be better to explain running procedure with an example of a similar case to the simulations in this thesis.

One can assume that there is one relatively large sized protein, Lysozyme in the center of the target configuration box. Its configuration box size is 5 by 5 by 5 (all nm) cubic and density of the box does not matter in this simple example.

First step that one should follow is to get the protein, Lysozyme from the Protein Data Bank (PDB) website, http://www.rcsb.org/pdb/home/home.do . Lysozyme that one is using here has PDB name, 1AKI. This `1AKI.pdb` file has only protein (Lysozyme) molecules inside and does not have any potential or energy-related information inside.

Next step is to get a topology file from the `1AKI.pdb file`. One can use **pdb2gmx** here,

```
pdb2gmx -f 1AKI.pdb -o lyso.pdb -p lyso.top \
        -ff ooplsad -water TIP4P
```

Here, one can focus on '`-ff ooplsad`' and '`-water TIP4P`'. '`-ff ooplsad`' is related to calculate potential and force field of the protein, Lysozyme. This force field decides the parameters K_b and b_0 of a potential equation $K_b(b-b_0)^2$ in the topology file. '`-water TIP4P`' means that the GROMACS treats the water molecules as TIP4P water. TIP4P water structure is more realistic and good, because TIP4P water model can calculates the free binding energies of inhibitors of the protein in a well-defined way thanks to its unique potential structure to maintain a clear tetrahedral form of one water molecule [38].

One has to put water (solvent) to the configuration that only has protein inside, so that one can see protein's behavior with the solvent. (This is original purpose of GROMACS code, but it can be used to focus on water's behavior. Therefore, this is the innovative approach of this thesis, as mentioned above.) When one solvates the target system, **genbox** is used in an appropriate way. In other words, command would be

```
genbox -cp lyso.pdb -cs tip4p.gro -o solution.pdb \
       -box 5 5 5
```

Basically this without any specific options makes Periodic Boundary Conditions (PBC), which allows GROMACS to see this box consists virtually and identically large blocks such as LEGO® blocks. Therefore, every molecule can go and cross over from left to right or

from bottom to upper side. This is also a good reason that there exist two Lysozymes in the configuration box in my actual simulations.

Or, alternatively, if one wants to make protein locate in the center position in the configuration box, one should gives commands,

```
editconf -f lyso.pdb -o lyso2.pdb -box 5 5 5 -c
genbox -cp lyso2.pdb -cs tip4p.gro -o solution.pdb
```

Now one needs to modify the topology file to add solvent (water) molecules to the topology. One can just add '`SOL  # molecules`' in the topology file, `lyso.top`.

To control simulation's output, GROMACS require mdp file, which means MD procedure. For this simple case, one can just use the default file that GROMACS provides.

Before running `grompp` to generate tpr file, which is the input for MD calculation program, mdrun, one has collected all files for `grompp`'s input so far: (1) pdb file (configuration, here `solution.pdb`), (2) mdp file (simulation parameters, here `mdout.mdp`), and (3) top file (topology and force field file, here `lyso.top`).

However, one could not run `mdrun`, because the configuration does have charges inside, which is +8e. To solve this problem, one has to add -8e charges to the configuration and it would not be a problem. For this, GROMACS provides `genion`. This time `em.mdp` should be used instead of `mdout.mdp`, because `mdout.mdp` is intended for actual simulation and `em.mdp` is written for minimizing energy. One has to give commands such as

```
grompp -f em.mdp -c solution.pdb -p lyso.top \
     -o system_em.tpr
genion -s system_em.tpr -o sol_4em.pdb -nn 8 -nname CL-
```

Here one may understand that '`-nn 8`' is means the number of negative charged atoms.

Since one will not have any problem because of the charge, one can run **grompp** to generate tpr file to be used as an input file for **mdrun**. Energy minimization procedure should be done first, and then actual simulation will be begun. Therefore, to minimize energy, one may want to give commands with **grompp** and **mdrun**,

```
grompp -f em.mdp -c sol_4em.pdb -p lyso.top \
      -o system_em.tpr
mdrun -s system_em.tpr -o system_em.trr \
      -c system_4md.pdb -np 1
```

Here '**-np #**' means the number of processor that the machine will use later for the actual simulations.

A series of actual MD simulation is now being done with **mdrun** command after excuting grompp again with **mdout.mdp** paramter file.

```
grompp -f mdout.mdp -c system_4md.pdb -p lyso.top \
      -o system_md.tpr
mdrun -s system_md.tpr -o system_md.trr -np 1
```

After sometime, one can see the result file of **system_md.trr** as the simulation output. Upon manipulating this file, one can calculate various applied quantities as a post-processing of the simulations with converting trajectories by **trjconv**. These applications will be reviews in the following section, and almost all of the results in this thesis have been calculated by the approached with the current and the following sections.

The following picture, figure 3 conceptually summarizes this running procedure and it shows how one can calculate various application quantities using GROMACS MD simulations.

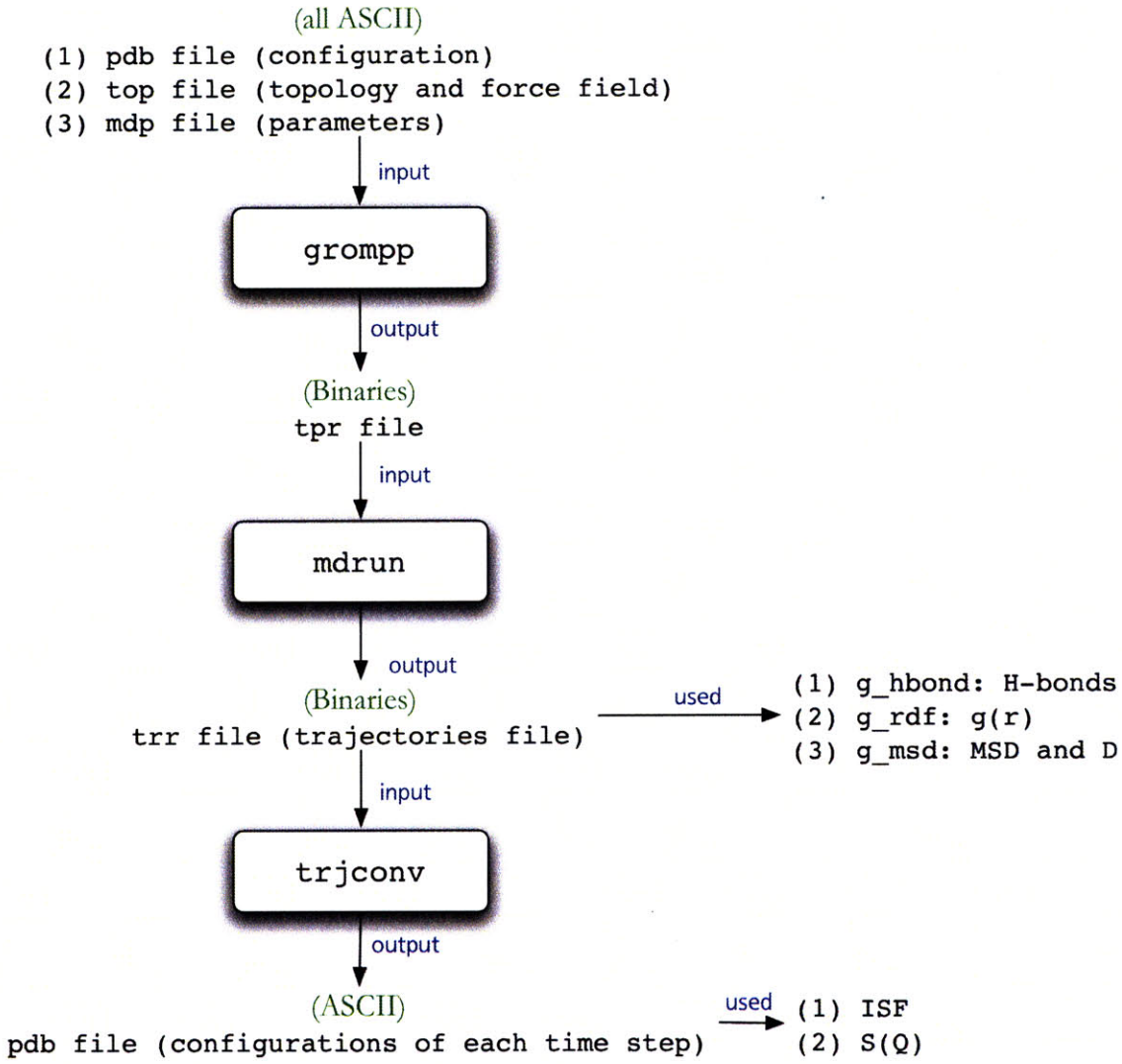


Figure 3. Using Gaussian Approximation to remove the asymmetry before the Fourier Transform

2.2. Applications: Post-Processing the MD Simulation Trajectories File

To confirm and show dynamic crossover in the hydration water around Lysozyme, I have tried to calculate many quantities from the MD simulation trajectories. Those quantities, which has been calculated in this thesis work for three difference hydration levels, are:

(1) Intermediate Scattering Functions (ISF),

(2) Number of hydrogen bonds between water and water,

(3) Number of hydrogen bonds between water and Lysozyme,

(4) Autocorrelation functions of hydrogen bonds between water and water,

(5) Autocorrelation functions of hydrogen bonds between water and Lysozyme,

(6) Hydrogen bonds relaxation time between water and water,

(7) Hydrogen bonds relaxation time between water and Lysozyme,

(8) Radial distribution function $g(r)$ for oxygen atoms,

(9) Structure factor $S(Q)$ for oxygen atoms,

(10) Mean Square Displacement (MSD) of water's hydrogen atoms
and diffusion constants D from it.

(11) Mean Square Displacement (MSD) of protein's (Lysozyme's) hydrogen atoms
and diffusion constants D from it.

Some of them such as number of hydrogen bonds are calculated by GROMACS' own functions, and some are obtained by Fortran codes, which are not the part of GROMACS. Some of them clearly show the crossover around the temperature $T_L = 220 \pm 10K$ [43], while it has not appeared in some quantities. This section discusses how one can calculate those quantities using GROMACS and its MD simulation results, trajectories file. It helps us to understand where those quantities are originated as shown figure 3. In the Chapter 4, comparison between simulation and experiment as well as comparison among simulation results of three different hydration levels is described.

2.2.1. Number of hydrogen bonds

Number of hydrogen bonds can be treated as one important aspect of the structural properties of the target configurations. Therefore, one may use one of the commands that GROMACS provide to calculate structures of the target system based on its simulated trajectories. Among many commands such as `g_saltbr`, `g_sas`, `g_hbond`, `g_clustsize`, `g_sgangle`, and etc. for calculating structures, the command, `g_hbond` is an appropriate one to get number of hydrogen bonds in the target system. It generates the number of hydrogen bonds in the system based on the trajectories files (`trr` file) as one can see in the figure 3.

Basically `g_hbond` computes and analyzes hydrogen bonds of the target system. GROMACS detects hydrogen bonds in the system using cutoff values, which are related to: (1) the angle among acceptor (A), donor (D), and hydrogen atom (H) (A-D-H in order) and (2) distance between acceptor and hydrogen atom (A-H). This program treats dummy hydrogen atoms as being connected to the first preceding non-hydrogen atom. According to the GROMACS manuals, one can appreciate the acceptor and donor as follows,

(1) Donors: OH- and NH- groups

(2) Accepter: O (always), N (default) [38].

`g_hbond` needs user to select two groups that make hydrogen bonds. Thanks to this ability of the code, I am able to count two types of hydrogen bonds in two different groups:

(1) Number of hydrogen bonds between water and water and

(2) Number of hydrogen bonds between water and Lysozyme.

Commands for those calculations of an example case of hydration level 0.6 and temperature 180 K are given to the program as follows,

(1) Number of hydrogen bonds between water and water:

```
g_hbond -f system_md.trr -s system_md.tpr \
      -b 15000 -e 50000 \
       -num T180.h06.hbond.sol.sol
```

Then, after giving the above commands, the code asks user to select target molecules using index file (`ndx` file). One selects `SOL` (water) and `SOL` (water).

(2) Number of hydrogen bonds between water and Lysozyme:

```
g_hbond -f system_md.trr -s system_md.tpr \
      -b 15000 -e 50000 \
       -num T180.h06.hbond.sol.lyso
```

Here, in the selection screen, one selects `SOL` (water) and `PROTEIN` (Lysozyme).

2.2.2. Autocorrelation functions and relaxation time, τ_{relax}

Since `g_hbond` computes and analyzes hydrogen bonds, one can obtain autocorrelation function $C(t)$ using this program [38]. From the function $C(t)$, one can also decide the relaxation time by finding the time τ_{relax} when $C(t) = 1/e$.

As the same as in the number of hydrogen bonds case, one can specify two groups for analysis. Those must be either identical or non-overlapping.

Using the same program, `g_hbond` with different input options, one can get autocorrelation function for each temperature and each hydration level. Then, relaxation time for that condition will be extracted.

Therefore, to get relaxation times, two autocorrelation functions should be calculated:

(1) Autocorrelation functions of hydrogen bonds between water and water,

(2) Autocorrelation functions of hydrogen bonds between water and Lysozyme,

For the same example as in the above section (hydration level 0.6 and temperature 180 K), one can give the following commands.

(1) Autocorrelation functions of hydrogen bonds between water and water:

```
g_hbond -f system_md.trr -s system_md.tpr \
      -b 15000 -e 50000 \
       -ac T180.h06.autofunc.sol.sol
```

Then, after giving the above commands, the code asks user to select target molecules using index file (`ndx` file). One selects `SOL` (water) and `SOL` (water).

(2) Autocorrelation functions of hydrogen bonds between water and Lysozyme:

```
g_hbond -f system_md.trr -s system_md.tpr \
      -b 15000 -e 50000 \
       -ac T180.h06.autofunc.sol.lyso
```

Here, in the selection screen, selects `SOL` (water) and `PROTEIN` (Lysozyme).

As mentioned in the first paragraphs, relaxation time is easily extracted from the autocorrelation functions by comparing the function with $1/e$. Then, one can record

(1) Hydrogen bonds relaxation time between water and water,

(2) Hydrogen bonds relaxation time between water and Lysozyme.

2.2.3. Radial Distribution Function (RDF), $g(r)$

RDF is the most popular, common, and the easiest way to reveal liquid structure. Neuron scattering methods can show this function, but it is not easy to calculate. That means MD simulations can be powerful to imagine liquid structure.

GROMACS approach to obtain this value is based on a viewpoint of mass distribution properties over time. In other words, it calculates the center of mass of a set of particles and generates RDF [38]. For the results in this thesis, $g(r)$s are calculated based on the interaction between oxygens.

Command that I have used for a case of hydration level 0.45 and temperature 180 K is,

```
g_rdf -f system_md.trr -s system_md.tpr -n index.ndx \
      -o T180.h0.45.g.xvg -b 40000 -e 50000
```

After giving the above command, `g_rdf` asks user to select target molecules using index file (`ndx` file). Selects `SOL & O*` (oxygens) and `SOL & O*` (oxygens). Or,

alternatively one can change its name `SOL & O*` to `Oxygens` by modifying the index file (`ndx` file).

2.2.4. Mean Square Displacement (MSD) and diffusion constant, *D*

When one wants to calculate MSD, `g_msd` is the correct procedure to help. This program and GROMACS calculates MSD value of atoms using mass distribution over time with their initial positions. This result is connected to diffusion constant *D* through Einstein's relation between two quantities. The diffusion constant comes from (least squares) fitting a straight line to the MSD result values. Then, *D* can be automatically calculated by this command, `g_msd`, because it has fitting procedure inside the code [38].

Using one command `g_msd`, one can get MSD values and diffusion constant. Since one can choose its target atoms from the trajectories file, I have calculated two cases,

(1) MSD of water's hydrogen atoms and diffusion constants *D* from it and

(2) MSD of protein's hydrogen atoms and diffusion constants *D* from it.

Commands for each above case under the situation hydration level 0.6 and temperature 240 K are given as

(1) MSD of water's hydrogen atoms and diffusion constants *D*:

```
g_msd -f system_md.trr -s system_md.tpr \
-o T240.h06.msd.sol-h.xvg -b 10000
```

(2) MSD of protein's hydrogen atoms and diffusion constants *D*:

```
g_msd -f system_md.trr -s system_md.tpr \
-o T240.h06.msd.lyso-h.xvg -b 10000
```

2.2.5. Intermediate Scattering Functions (ISF)

GROMACS does not provide a program or built code to calculate ISF. Therefore, a specific procedure should be written by a researcher, who wants to obtain. Since we have to

calculate the Van Hove correlation function $G_s(\vec{r},t) = \frac{1}{N}\left\langle \sum_{i=1}^{N} \delta\left[\vec{r} + \vec{r}_j(0) - \vec{r}_j(t)\right] \right\rangle$ here to perform the ISF calculation, single atom's position in every time step is required (this means actually that ISF calculation needs two iteration blocks: one for atoms, and the other for time steps). However, as mentioned above, the trajectories file is written in binaries type and does not show each time step's configuration. Therefore, one should convert its from to a series of ASCII files containing each time step's configuration, which can be read and show atoms' positions.

`trjconv` is the exact tool required for the above process. Basically it converts trajectories to pdb which can be viewed with VMD. It can convert trajectories file from one to another format, which can also be ASCII. It also has ability to reduce the number of time frames, so that one can reduce its result trajectories keeping the detailed simulation with small time step. These are main work for `trjconv` before doing ISF calculation. In addition, this `trjconv` is used to obtain a more detailed structure factor than one that GROMACS provides.

For the purpose of getting configurations of each time step, command is given as to the machine,

```
trjconv -f system_md.trr -s system_md.tpr \
      -o pdbs/system_md.pdb \
      -b 10000 -sep -pbc nojump
```

In the above `trjconv` command, there should be explanation for paramters.

(1) `-sep` : to write every time step frame to a seperate `pdb` file, and

(2) `-pbc nojump` : to make the PDC routine check
if atoms jump across the box and then puts them back.

Actually `-pbc nojump` option makes all molecules remain as is and ensures that the trajectories remain continuous.

As known well, ISF has a meaning of correlation of molecules in the time domain and is the Fourier Transformed result of the Van Hove correlation function $G_s(\vec{r},t)$. After getting all the **pdb** files from the **trjconv** procedure, one can get all the properties, which can be probed by QENS experiemtns, to calculate ISF. Hydrogen positions are recorded in a tetrahedral representation with time step. Then, based on those series of **pdb** files, the Van Hove correlation function $G_s(\vec{r},t)$ describing the diffusion is calculated [44].

$$G_s(\vec{r},t) = \frac{1}{N}\left\langle \sum_{i=1}^{N} \delta\left[\vec{r} + \vec{r}_j(0) - \vec{r}_j(t)\right] \right\rangle \quad \text{(2.1)}$$

where δ is the Dirac delta function, $\vec{r}_j(\cdot)$ is the position vector of hydrogen atom number j, and N is the total number of hydrogen atoms tracked in the simulation. This factor means the conditional probability to find a H-atom within a displacement $\vec{r}$ within time t. As one can read from the equation 2.1, the calculation procedure needs two iterations: one for atoms and the other for time steps.

The ISF $I(\vec{Q},t)$ comes from the spatial Fourier Transform of $G_s(\vec{r},t)$, so

$$I(\vec{Q},t) = \int G_s(\vec{r},t) e^{i\vec{q}\cdot\vec{r}} d^3\vec{r} \quad \text{(2.2)}$$

If one is able to treat the situation (or assume) that the displacements are isotropic, results become losing vector properties, so that $I(\vec{Q},t) = I(Q,t)$ and $G_s(r,t) = G_s(\vec{r},t)$. One can see that the MD has powerful ability with ease, because MD simulations results can explicitly provide the ISF for each temperature for hydrogen atoms in the water by calculating the Van Hove correlation functions. This calculation is done through tracking the hydrogen atoms' mobility in the system written in a series **pdf** file. Therefore, by computing the above equations using the positions of only hydrogen atoms in the simulation configuration, one can get the ISF.

Then, the ISF results of simulations are used to obtain the average relaxation time using RCM fitting. Finally the average relaxation time results are compared to the experimental data.

2.2.6. Structure factor, $S(Q)$

Even though GROMACS provides a way to calculate the structure factor $S(Q)$ using `g_rdf` function with the option of Fast Fourier Transform (FFT) (`-sq`), it is not so satisfying methodology because it is just domain-transformed quantity. Therefore, a Fortan program to obtain this value has been used. It basically uses `pdb` files converted from the trajectories file. Using each time step configuration of the simulation result, the program calculates

As one has done in the ISF case, `trjconv` should be performed before calculating the structure factor. For the purpose of getting configurations of each time step to get the structure factor, one do not need a detailed trajectory (which can be made by `–sep` option). Command looks like

```
trjconv -f system_md.trr -s system_md.tpr -n index.ndx \
     -o pdbs_sk/system_md.pdb \
     -sep -skip 500 -b 10000
```

In the above `trjconv` command, `-skip 500` is used to sample the configuration with 500 time step intervals as `pdb` format [38].

After giving the above command, `trjconv` asks user to select target molecules using index file (`ndx` file), which one already gives to the command as a parameter. Then, selects `SOL & O*` (oxygens). Or, alternatively one can change its name `SOL & O*` to `Oxygens` by modifying the index file (`ndx` file). `trjconv` generates `pdb` files containing only oxygen atoms information. Reason one should choose oxygen is that the structure factor is calculated focusing on oxygen atoms here and it is usual calculation.

Using the generated **pdb** files, the structure factor $S(Q)$ is obtained. It is calculated by summing three oxygen atom's position of a specific atom at specific time step and three random Q in the three iterations under (1) all of the number of time steps of the all trajectories written in **pdb** format, (2) average number of all Q vectors in the MD powder sample, and (3) the number of oxygen atoms inside the simulation configuration system. Therefore, one can use the following equation for the calculation,

$$S(Q) = \frac{\iiint \begin{bmatrix} S(Q,t,q) \\ +\cos^2\left(q_{r1}\cdot r(a,t,q_{r1}) + q_{r2}\cdot r(a,t,q_{r2}) + q_{r3}\cdot r(a,t,q_{r3})\right) \\ +\sin^2\left(q_{r1}\cdot r(a,t,q_{r1}) + q_{r2}\cdot r(a,t,q_{r2}) + q_{r3}\cdot r(a,t,q_{r3})\right) \end{bmatrix} dadtdq}{a\times t\times q} \quad \text{.....(2.3).}$$

In the above equation, Q here means virtual Q-vector to be used for describing $S(Q)$ as its independent variable and has values from 1 to 100. a is the total number of atoms (here, oxygen), t is the time steps, and q means the number of average Q-vector of the powder sample, which I have used for the simulations. q_{r1}, q_{r2}, and q_{r3} mean the random Q values to be used for obtaining its position $r(\cdot,\cdot,\cdot)$ value of a specific atom at specific time step, and three random Q.

As similar to the ISF case, the MD simulation shows its easy approach for computing $S(Q)$. It can explicitly give us the structure factor at each temperature. If one would selects any other kind of atom during the `trjconv` procedure, one can get $S(Q)$ for that atom case. The computation is done by following the mobility of the target atoms (here, oxygen).

2.3. Simulation Configurations

2.3.1. Hydration water in Lysozyme

Our group has chosen the 'hydrated powder protein model' to simulate in a better way. This model has been verified by Tarek and Tobias, and it agrees more with the experimental data

[45]. Since a single protein configuration with water molecules cannot reflect Lysozyme molecules' motion as a powder type and then cannot mimic the experimental data of powder sample, we have to put more Lysozyme to make it look like a powder. This idea was verified by previous researches [46, 47]. such that two proteins or eight proteins case can model in an realistic way. Therefore, two Lysozymes are put in the target configuration box.

A force field should be decided in a serious manner, because it affects a lot to the simulation results and make the result agrees well with the experimental data. As described as an innovative way above, my research has focused on water molecules than protein, so water model is crucial to simulate the real system in the best way. TIP4P water model is one of the best choices, because it shows a good agreement with experiments in terms of self-diffusion constant computation [48]. For Lysozyme's force field, the OPLS-AA field is implemented [49]. A previous research [50] informs that when one uses OPLS-AA field for protein and TIP4P for water, MD simulation generates the satisfactory result in a viewpoint of free energy of binding of inhibitors on a protein.

Figure 4 shows this configuration in stable status upon maintaining hydration level 0.3 with 484 water molecules.

2.3.2. Three different hydration levels configurations

In this thesis, I compare three different hydration levels of the target system containing hydration water. Hydration level h is set by the water mass (g) per the solution mass (g), so its unit is g/g (gram per gram).

Based on the configuration described in the previous section, one thing has to decided in this section. It is the hydration level of the hydrated powder sample configuration. Three cases are

(1) hydration level h =0.3 : total 484 water molecules,

(2) hydration level h =0.45 : total 726 water molecules, and

(3) hydration level h =0.6 : total 968 water molecules.

As mentioned above, each case has two Lysozymes in the configuration box. It makes the PDC work better. Moreover, two Lysozymes configuration can reveal how water molecules and their hydrogen bonds act on the two Lysozymes and how they make those two move closer to each other and collapse toward the center between the two.

I have performed total 33 MD simulations:

(1) 180 K to 280 K with 10 K difference (11 runs) at hydration level h = 0.3,

(2) 180 K to 280 K with 10 K difference (11 runs) at hydration level h = 0.45, and

(3) 180 K to 280 K with 10 K difference (11 runs) at hydration level h = 0.6.

MD simulations in my thesis work have used parallel CPUs to improve its calculation speed. Each temperature one begins its simulation from the configuration of the one step before (below 10 K from the current temperature).

Figures 4~6 in following pages show these three configuration yet energy-minimized in a picture. In those pictures, Lysozyme is depicted in wired-frame style and water in CPK style.

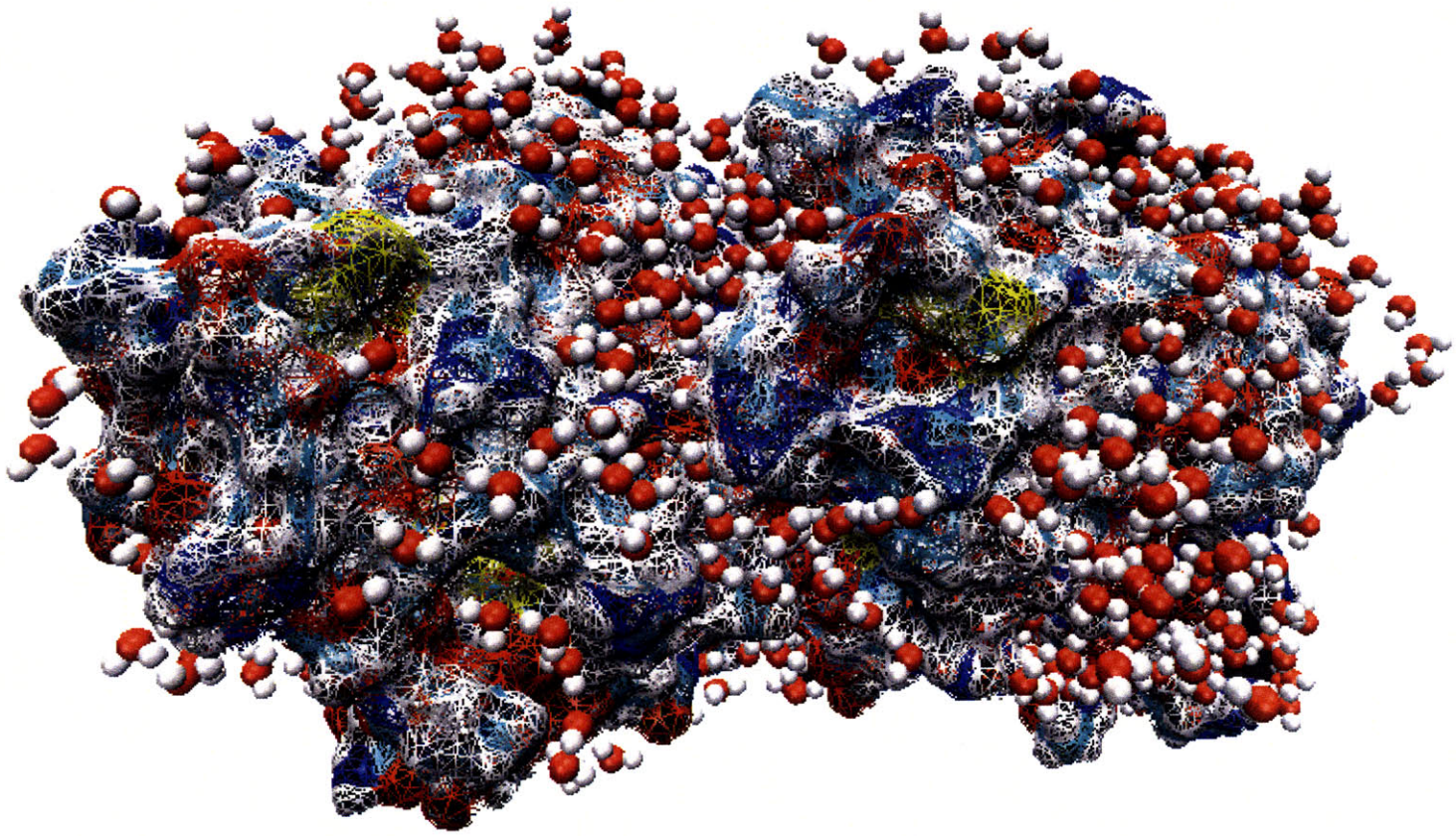


Figure 4. An MD simulation configuration for the Lysozyme hydration water having hydration level h = 0.3

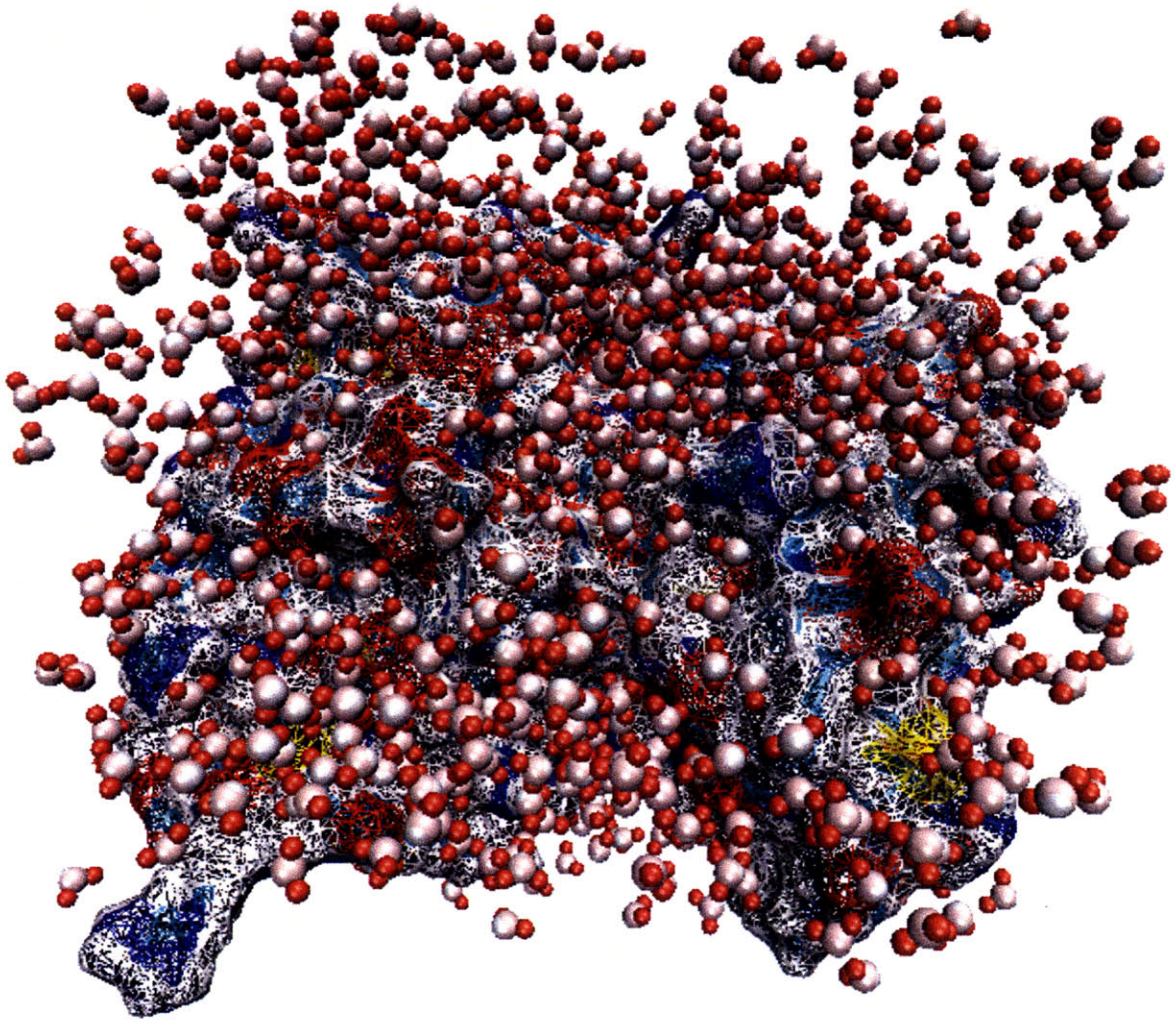


Figure 5. An MD simulation configuration for the Lysozyme hydration water having hydration level h = 0.45

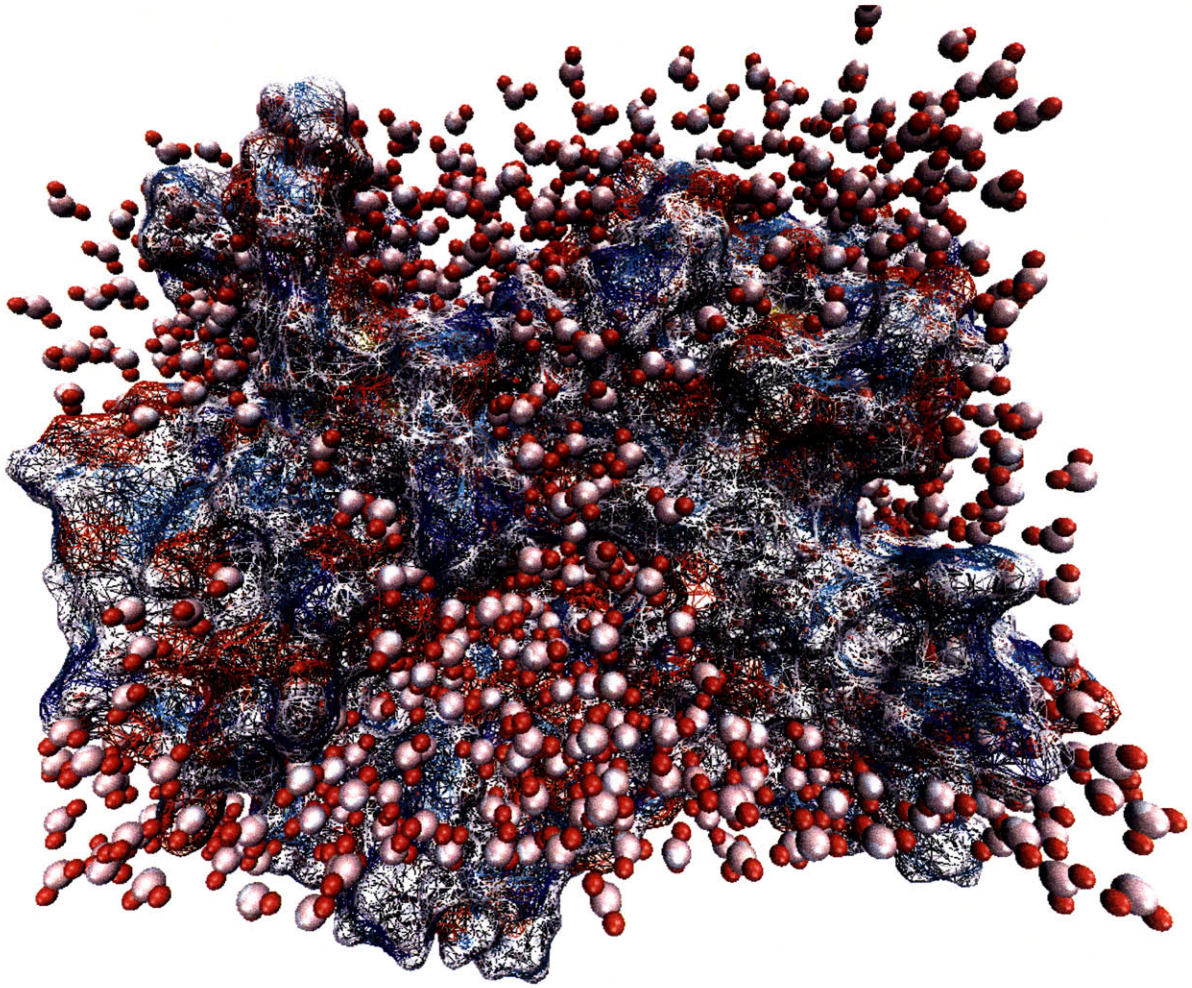


Figure 6. An MD simulation configuration for the Lysozyme hydration water having hydration level h = 0.6

Chapter 3. ANALYSIS METHODOLOGY FOR ISF DATA

3.1. An Time Domain Analysis for Experimental ISF Data

Normally, the experimental QENS data have been analyzed in the frequency-domain, which is the inelastic neutron scattering experiment domain. One cannot see the dynamics of a certain material easily, because the experimental results are obtained not in time-domain. The other weakness of the frequency-domain fitting is that it takes much more longer time than doing it in time-domain because of the convolution characteristics of the Fourier Transform.

I suggest changing the domain first, and for doing this, one should perform the Fourier Transform. This transform could be done after approximating experimental data using two or three Gaussian functions. There would be no need to use convolution to remove the resolution problem. Convolution before the Fourier Transform will be changed to just multiplication after the transform. Therefore, one can just divide the experimental data in time-domain by the resolution function values in time-domain. After that, fitting procedure will be performed in a timely manner. Fitting procedure will be done on this 'time-domain data' doing it simultaneously seven Q-values curves together. Levenberg-Marcus computation to have the smallest chi-square was used to the fitting procedure.

The experiments on the Lysozyme hydration water have been done in Backscattering Spectroscopy at NIST Center for Neutron Research, NCNR. Its condition is normal pressure 1 atm. The QENS experiments data used here as examples to explain the time domain fitting procedure has the case of hydration level h = 0.3 and it is the real data compared with MD simulations.

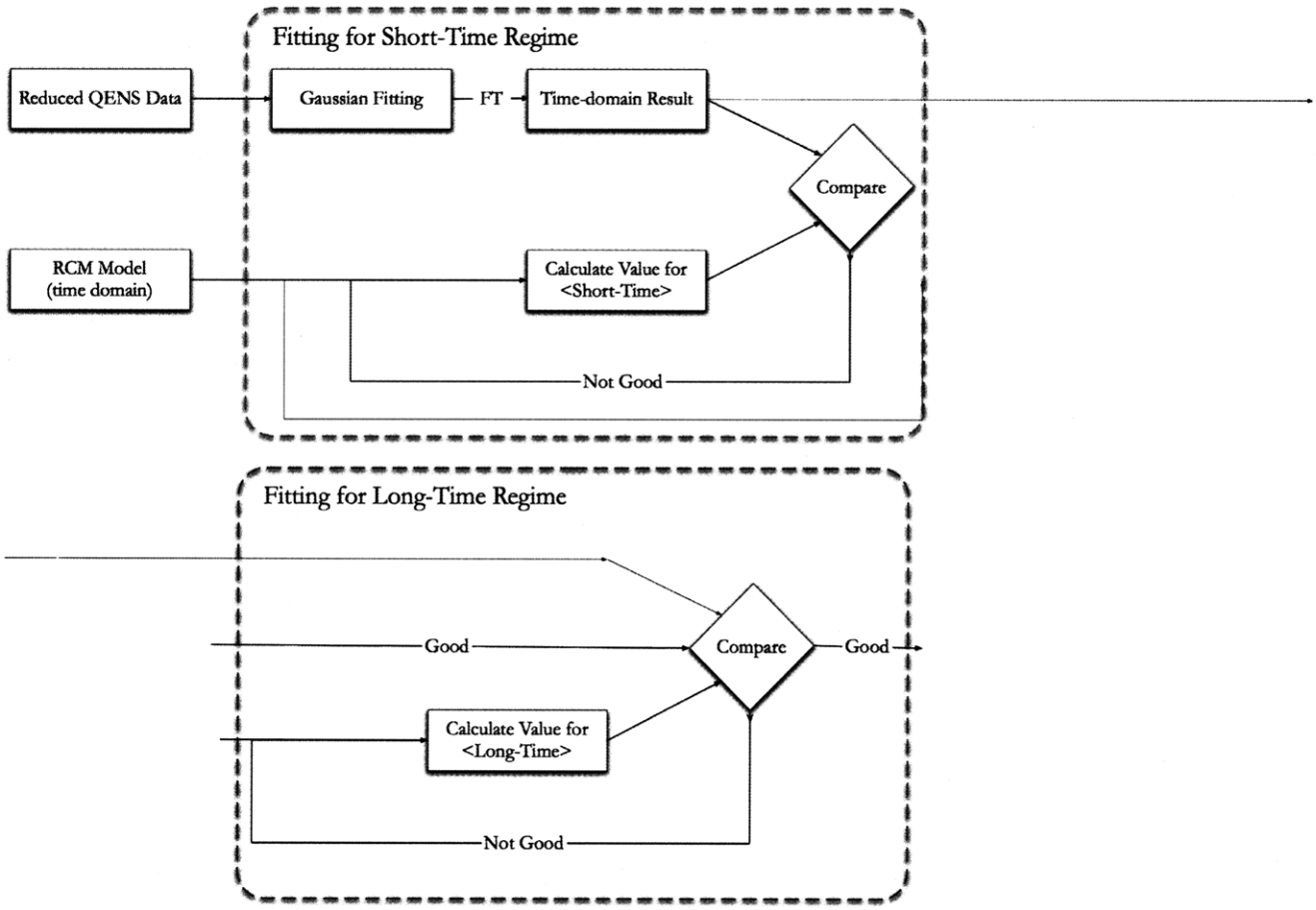


Figure 7. Conceptual diagram of the new time domain analysis method including the Gaussian functions approximation shows the Gaussian approximation to the raw data (in frequency domain), Fourier Transform, and the RCM fitting. Firs be centered at $\omega_{peak} = 0$. Based on the shifted ones, experimental data is converted to a functional form, the sum of functions. Fourier Transform is applied to that functional form and generated data in time domain. By dividing the da function, I get the pure time domain ISF data. Then, RCM fitting is performed with that data in time domain to finally g relaxation times. In the RCM fitting procedure the short time and the long time parts are separately analyzed, because which may occur in combining the short time RCM function and the long time RCM function.

3.1.1. Gaussian approximation for raw ISF data in frequency domain

To perform the Fourier Transform appropriately avoiding to generate the imaginary parts, we need to model the frequency domain data with a certain kind of symmetric function such as Gaussian. And, the transform requires that the area below the data should be calculated because there is integral inside the transform.

First, we perform the Gaussian approximation to remove the asymmetry of the raw data. Here three Gaussian functions are used to fit the experimental ISF data. The equation used for the fit is

$$I_{GaussianFit-Shifting}(\omega) = a_1 \cdot \exp\left(-\frac{(\omega-\omega_1)^2}{2b_1^2}\right) + a_2 \cdot \exp\left(-\frac{(\omega-\omega_2)^2}{2b_2^2}\right) + a_3 \cdot \exp\left(-\frac{(\omega-\omega_3)^2}{2b_3^2}\right) \quad (3.1).$$

Fitting parameters are a_i, b_i and ω_i (i=1, 2 and 3), and those actually has no serious meaning but finding ISF curve's peak position, ω_{peak}. After getting the peak position ω_{peak}, this approximation program automatically shifts the experimental data by the amount of ω_{peak} to make that shifted curve has centered at $\omega_{peak} = 0$. Then, the Fourier Transform of the shifted ISF does not generate imaginary part. This is the main reason that I added this procedure for the new time domain analysis. In the following page, one can see figure 8 and its shows a result of the first Gaussian fitting.

And then, it makes the symmetric data set to perform the Fourier Transform avoiding to have imaginary part in time domain. After that, we perform the Gaussian functions fitting again (call this one as the second Gaussian fitting) to get a kind of functional form to do the Fourier Transform with ease. Of course it ensures that the generated function can represent the experimental data correctly and perfectly: then, we can assume that the Gaussian function is identical to the experimental data. A sum of three Gaussian function sets, which are centered at 0, are used for this analysis,

$$I_{GaussianFit-DataIndentical}(\omega) = a_1 \cdot \exp\left(-\frac{\omega^2}{2b_1^2}\right) + a_2 \cdot \exp\left(-\frac{\omega^2}{2b_2^2}\right) + a_3 \cdot \exp\left(-\frac{\omega^2}{2b_3^2}\right) \quad \text{.....(3.2),}$$

where a_i, b_i and ω_i (i=1, 2 and 3) are parameter to mimic the shifted ISF data. One can see the result of this procedure in figure 9.

Figure 10 shows the Gaussian functions approximation results of all temperatures at a specific Q value (0.87 $Å^{-1}$). From this graph, I am strongly able to say that all the data can be completely fitted and represented with the sum of three Gaussian functions having centered at 0. Therefore, one can see that those functional results can be treated as the experimental data, which we manipulate for the next-step researches.

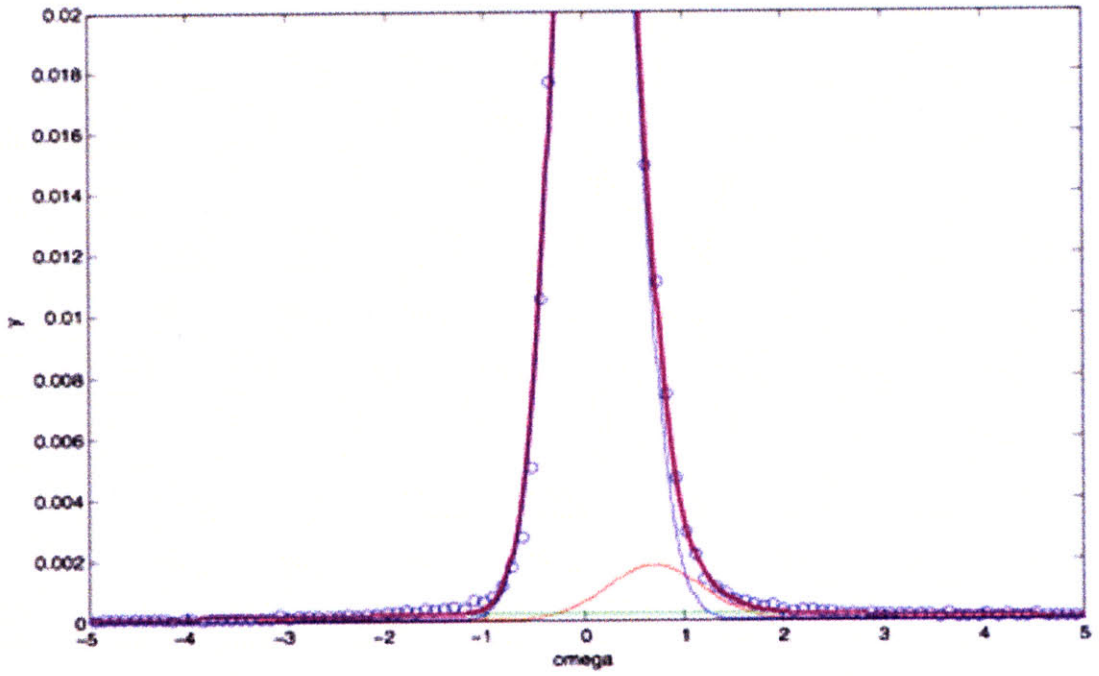


Figure 8. An example of the first Gaussian approximation: it uses the sum of three Gaussian functions to remove the asymmetry of the experimental ISF data. This is a case of 230 K, Q = 0.87 $Å^{-1}$ and hydration level h=0.3

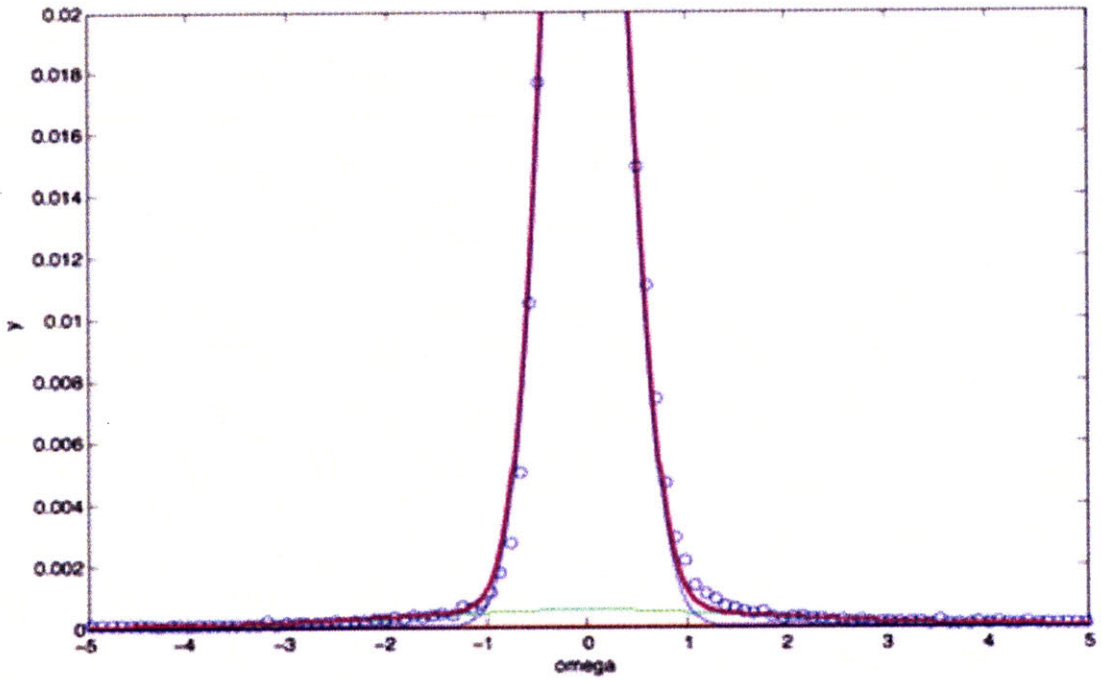


Figure 9. Gaussian functions approximation result. The purpose of this fitting is to obtain one kind of functional form to do the Fourier Transform in an appropriate way to get time domain data. This part is the essence of the new time domain fitting method. This is performed on the shifted datasets.

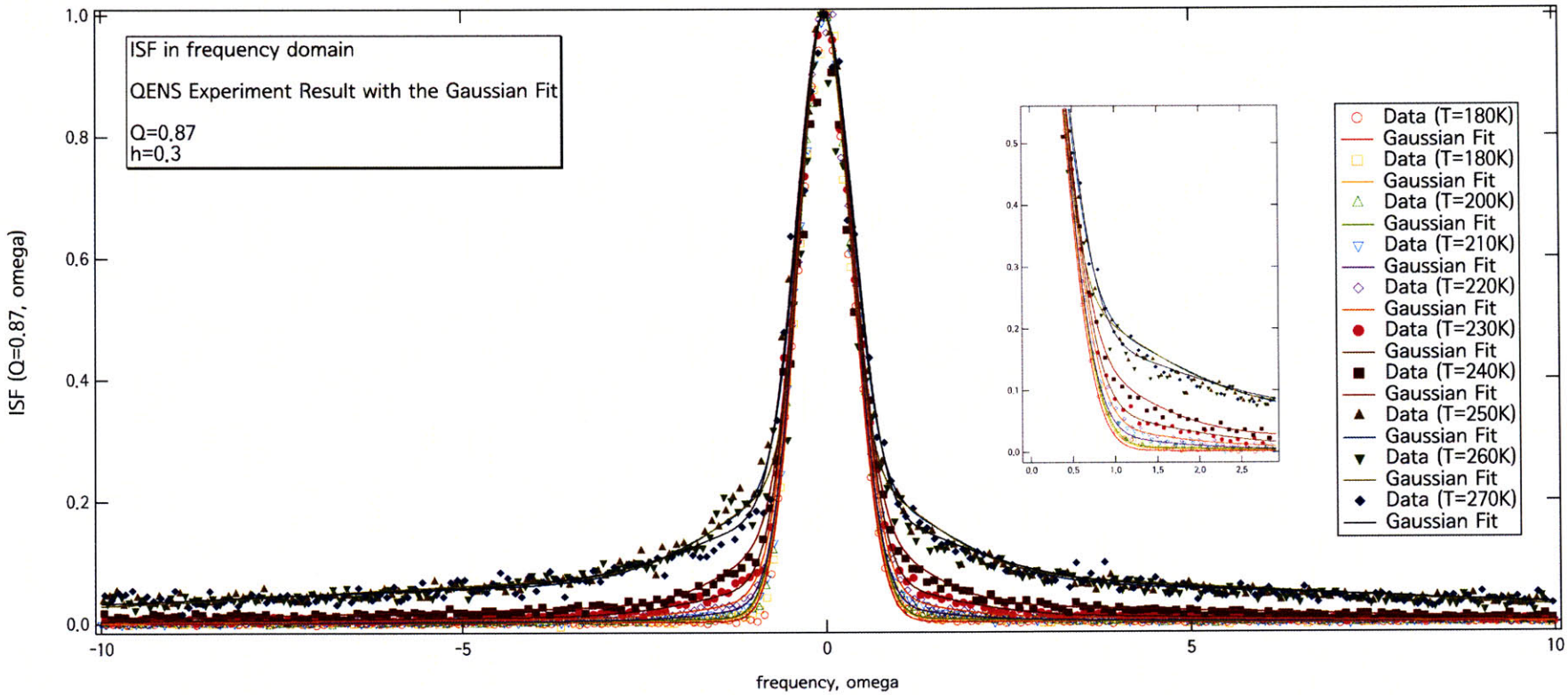


Figure 10. Gaussian functions approximation results for all temperatures from 180 K to 270 K at Q is 0.87 $Å^{-1}$. One can check it generates good fits.

3.1.2. Fourier Transform (FT) of the Gaussian approximated functions

Once we have functional forms of the raw data in frequency domain, we can perform the Fourier Transform with avoidance of getting imaginary terms. Since there is no need to manipulate unnecessary imaginary part, which is hard to provide physical meaning in time domain, one can use this result as an individual ISF function in the time domain. All of the results from the transform can be treated as time domain data.

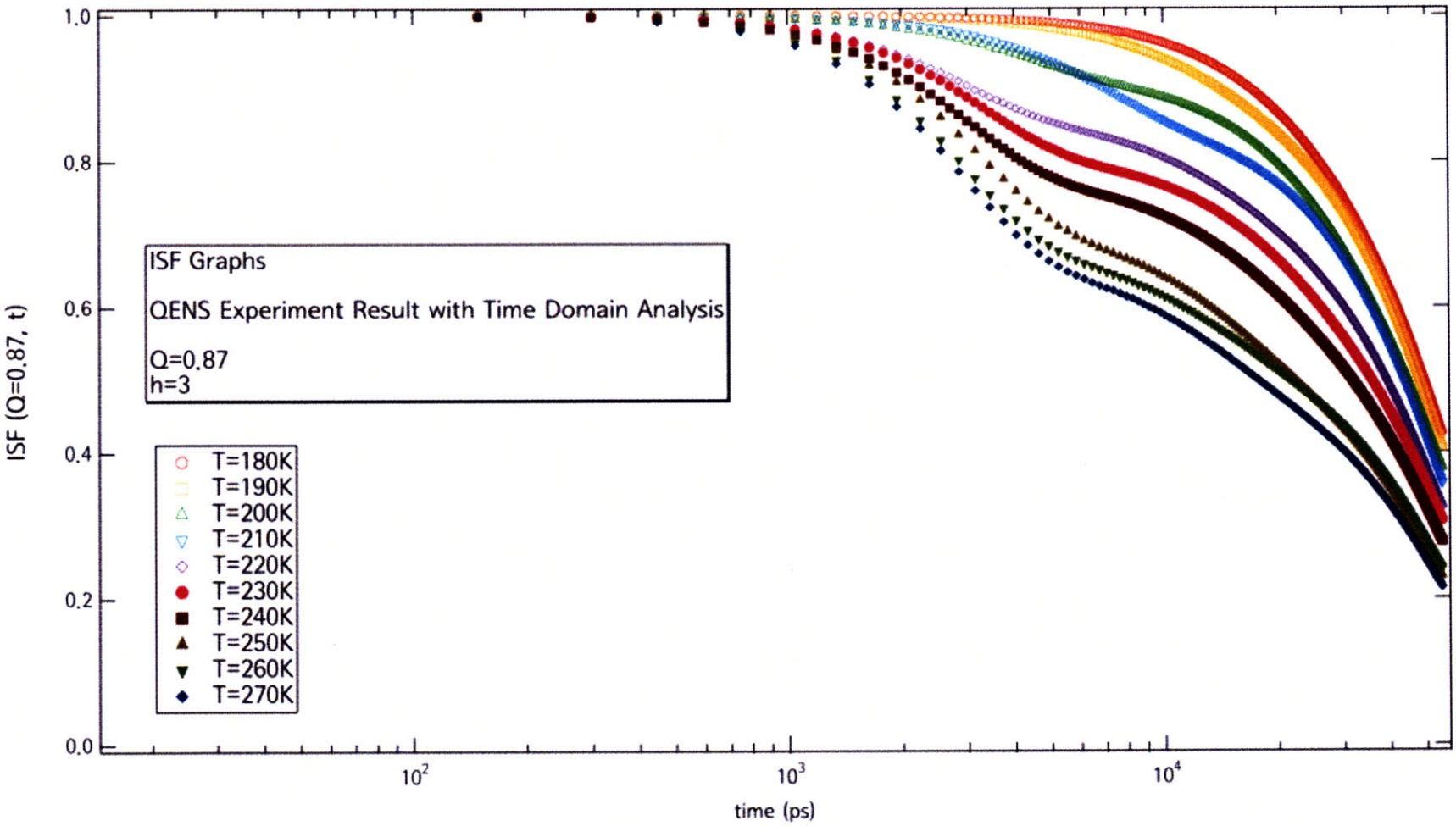


Figure 11. It shows time domain ISF datasets using Gaussian approximation. Curves vary with Temperature at a specific Q = 0.87 $Å^{-1}$.

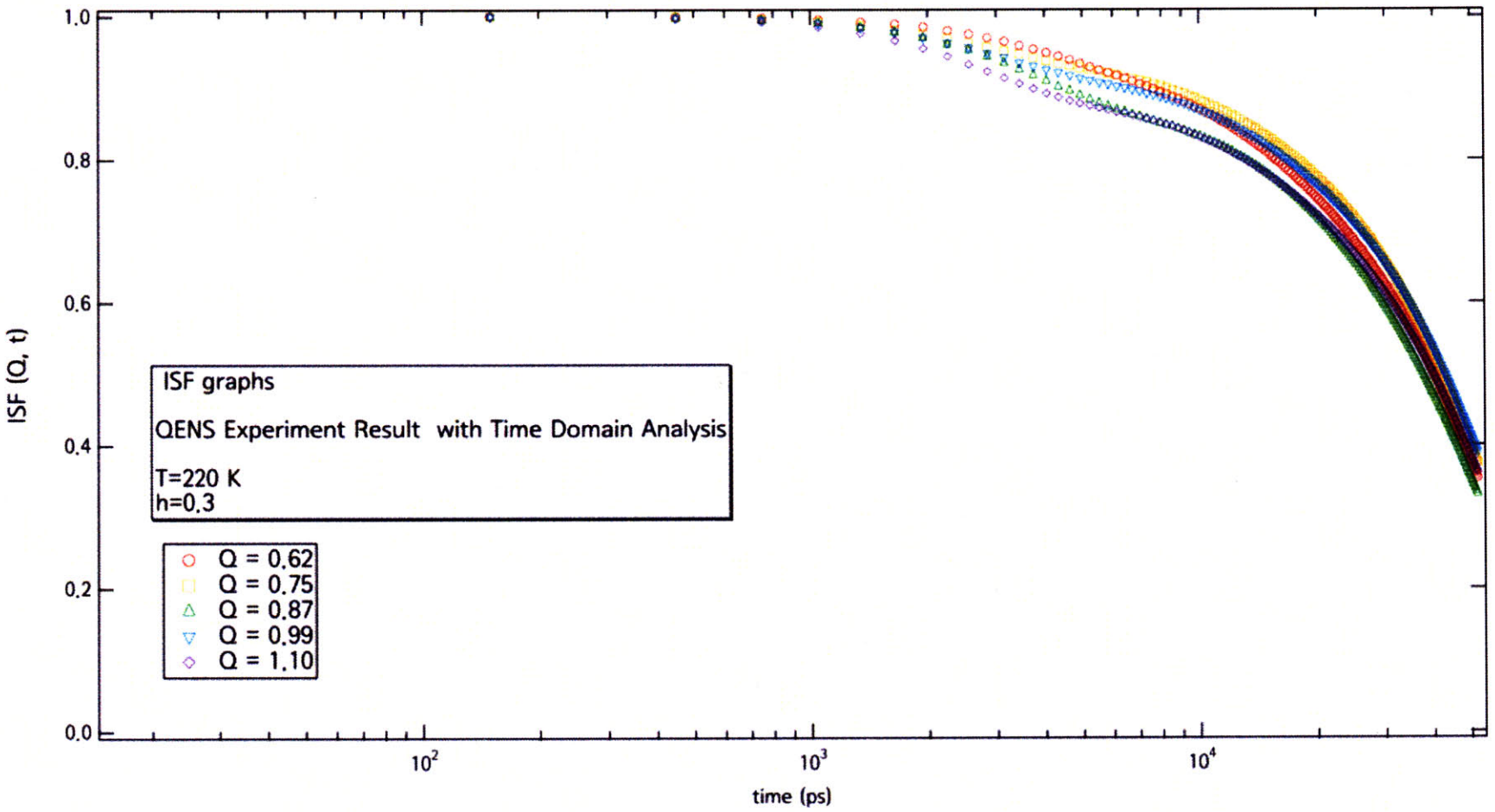


Figure 12. Converted ISF datasets using Gaussian approximation to time domain values. This has Q-dependent time domain ISF curves at a specific temperature 220 K.

3.1.3. RCM fitting analysis in time domain

RCM fitting has been done with the time-domain data. As written before, it was done with seven Q curves simultaneously to get common average translational relaxation time value. Two example temperatures, 190 K and 210 K of the experimental results using Gaussian are shown in this section.

For experimental results, seven ISF curves ($I(Q,t)$ functions for Q from 0.32 to 1.1 $Å^{-1}$) are fitted using RCM jointly. RCM model for fitting the ISFs given by MD simulation is given as,

$$I(Q,t) = p_Q + (1 - p_Q) \times \exp\left(-Q^2 \times b^{\frac{2}{3}}\right) \times \exp\left[-\left(\frac{t}{\tau_0 \times (a \times Q)^{\gamma}}\right)^{\beta_Q}\right] \quad \text{.....(3.3),}$$

where t is time of the ISF. The joint fitting parameters, which are the same for all five ISF curves, are b, τ_0, and γ among all six parameters. a is fixed as a constant 0.5 and used cancel Q's unit. The other two parameters having subscript Q are p_Q and β_Q. They can be different for every seven Qs. p_Q is related to the Elastic Incoherent Structure Factor (EISF). Since I do have time domain data, which are converted from QENS experimental data using the Gaussian functions approximation, there is no problem to apply the above equation 3.3 to fitting the ISF curves.

Fitting procedure is done by fitting seven curves (Q is 0.35, 0.49, 0.62, 0.75, 0.87, 0.99 and 1.1) together. From this RCM fitting results, the Q-dependent average translational relaxation time $\tau_T(Q)$ (ps) for each temperature is obtained via

$$\tau_T(Q) = \tau_0 (aQ)^{\gamma} \quad \text{.....(3.4).}$$

Since a is constant 0.5, the Q-dependent average translational relaxation time depends on two RCM fitting parameters, τ_0 and γ.

The Q-independent average translational relaxation time $\langle \tau_T \rangle$ (ps) for each temperature can be obtained with ease. It is computed as follows,

$$\langle \tau_T \rangle = \tau_0 \times \frac{\Gamma\left(\frac{1}{\beta}\right)}{\beta} \quad \text{.....(3.5).}$$

Then, one can obtain a result graph of the average translational relaxation time. From that graph of $\langle \tau_T \rangle (Temp.)$ *vs.* $\frac{1000}{Temp}$, I can use two laws: the Vogel-Fulcher-Tammann (VFT) law for high temperature larger than 220 K and the Arrhenius law for low temperatures. are applied to fit the average translational relaxation time values.

VFT law is written as $\langle \tau_T(\mathrm{T}) \rangle = \tau_0 \times \exp\left(D \times \frac{T_0}{(T - T_0)}\right)$, and the Arrhenius law $\langle \tau_T(\mathrm{T}) \rangle = \tau_0 \times \exp\left(\frac{E_a}{RT}\right)$. τ_0, D, and T_0 are fitting parameter in the VFT, and τ_0 and E_a for the Arrhenius. R is the Gas constant. T_0 is the ideal glass transition temperature, so it is reasonable that it is high when we give smaller hydration level to a sample.

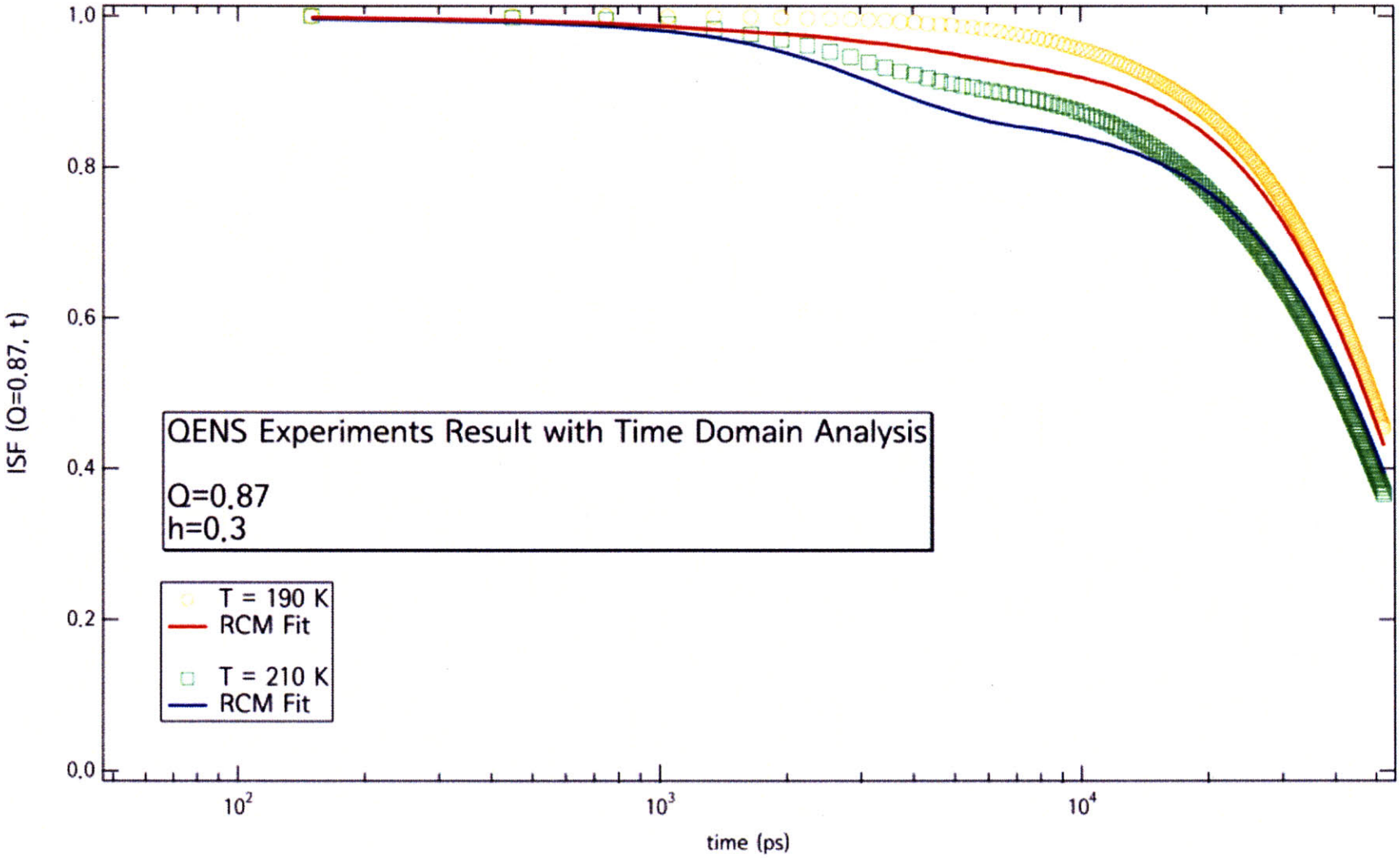


Figure 13. Comparison of RCM fitting results on the time domain ISF data of 190 and 210K cases Q = 0.87 $Å^{-1}$.

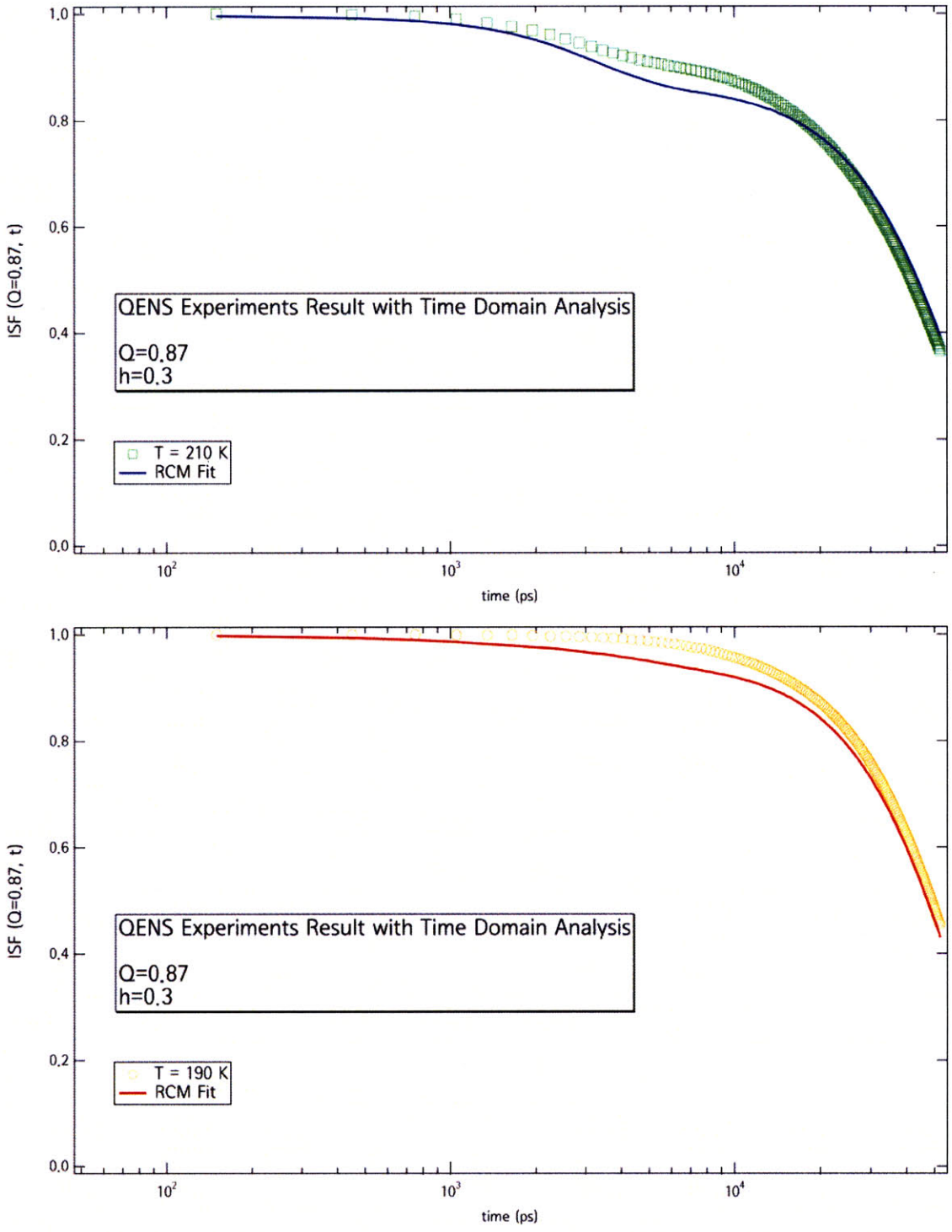


Figure 14. RCM fitting results on the time domain ISF data of 190 and 210K cases Q = 0.87 $Å^{-1}$.

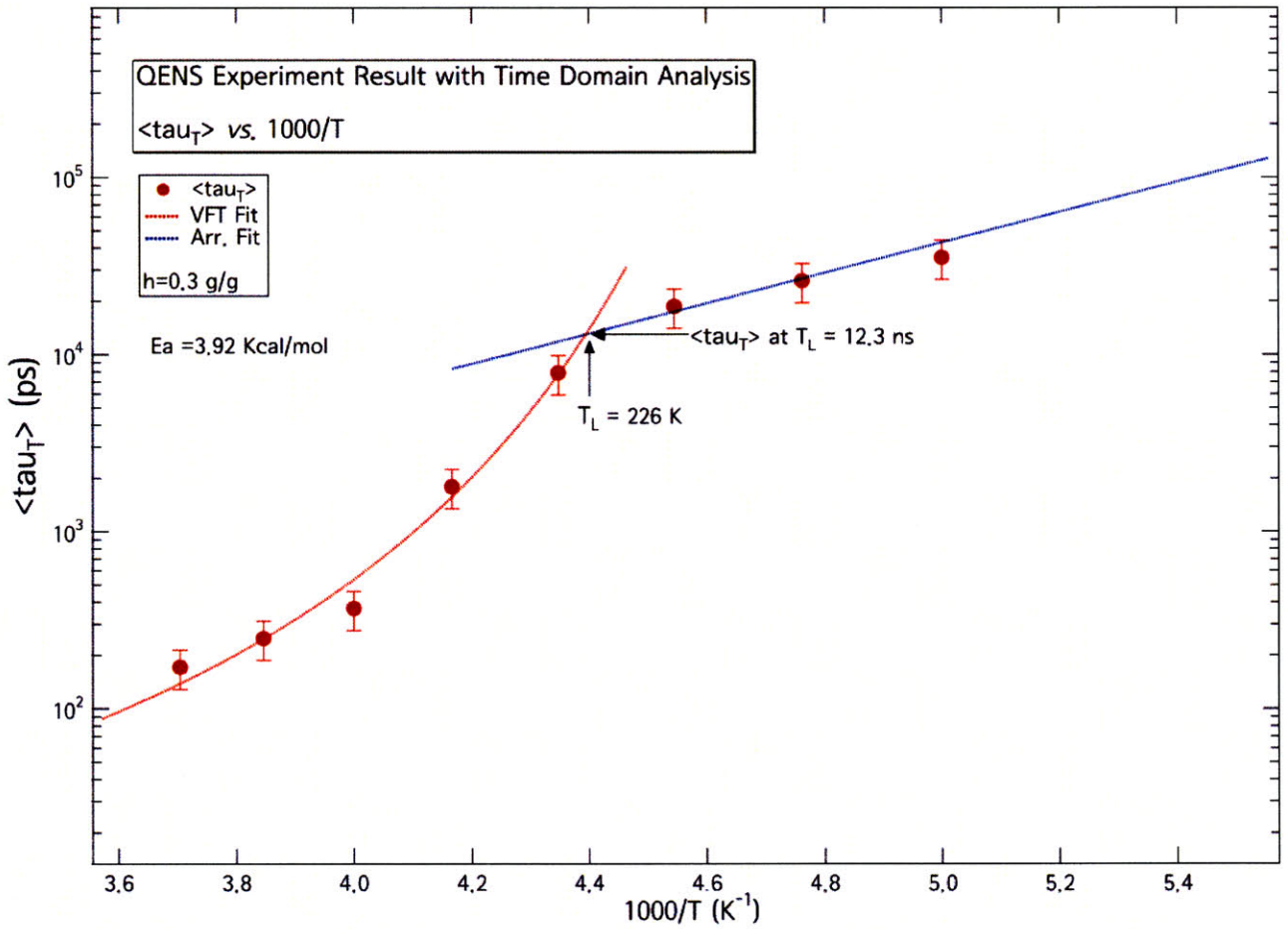


Figure 15. Average translation relaxation time graph vs. 1000/T (K^{-1}) is shown here. This shows clearly well defined cusp-like dynamic crossover behavior in each case. The red line represents fitted curves using the VFT law, while the blue line is the fitting according to the Arrhenius law. This is obtained from the time domain ISF data by fitting it with RCM. Fitting parameter shown here, E_a is the result for the Arrhenius fit of the experimental ISF curve in time domain.

3.2. MD Simulation ISF Data Analysis

3.2.1. Global least squares method for the ISF data from MD simulation

To analyze the ISF data obtained from the MD simulation results, one of the famous algorithm, the Levenberg-Marquardt algorithm is used to search for the coefficient values that minimize chi-square. Since I have to fit five ISF curves ($I(Q,t)$ functions for Q from 0.4 to 0.8 $Å^{-1}$), so-called global fitting, which uses the algorithm, is an appropriate way as I succeeded in last section.

Normally, in curve fitting we have raw data and a function with unknown coefficients. One wants to find values for the coefficients such that the function matches the raw data as well as possible. The "best" values of the coefficients are the ones that minimize the value of chi-square. The chi-square is defined as, with the meaning of "least-squares",

$$\sum\left(\frac{y-y_i}{\sigma_i}\right)^2 \quad \text{.....(3.6),}$$

where y is a fitted value for a given point, y_i is the measured data value for the point and σ is an estimate of the standard deviation for y_i.

The Levenberg-Marquardt algorithm is one of the iterative fitting methods, so its operation is iterative as the fit tries various values for the unknown coefficients. For each try, it computes chi-square searching for the coefficient values that yield the minimum value of chi-square. The Levenberg-Marquardt algorithm is used to search for the coefficient values that minimize chi-square. This is a form of nonlinear, least-squares fitting. As the fit proceeds and better values are found, the chi-square value decreases. The fit is finished when the rate at which chi-square decreases is small enough.

The Levenberg-Marquardt algorithm is used to search for the minimum value of chi-square. The chi-square defines a surface in a multidimensional error space. Starting from the initial guesses, the fit searches for the minimum value by traveling down hill from the starting point on the chi-square surface. The method wants to find the deepest valley in the chi-square surface. This is a point on the surface where the coefficient values of the fitting function minimize, in the "sense of least-squares." Some functions may have multiple valleys, places where the fit is better than surrounding values, but it may not be the best fit possible. When the fit finds the bottom of a valley it concludes that the fit is complete even though there may be a deeper valley elsewhere on the surface. Those are actually dependent on the initial guesses.

The fitting procedure will terminate after 40 passes in searching for the best fit, but will quit if 9 passes in a row produce no improvement in chi-square value. This may happen if the initial guesses are too good to start the fitting procedure for improving the minimum chi-square. It can also happen if the initial guesses are way off or if the function does not fit the data at all.

3.2.2. RCM fitting analysis in time domain

In my case, as desribed, five ISF curves ($I(Q,t)$ functions for Q from 0.4 to 0.8 Å^{-1}) are fitted using RCM simultaneously. RCM model for fitting the ISFs given by MD simulation is given as,

$$I(Q,t) = p_Q + (1 - p_Q) \times \exp\left(-Q^2 \times b^{\frac{2}{3}}\right) \times \exp\left[-\left(\frac{t}{\tau_0 \times (a \times Q)^{\gamma}}\right)^{\beta_Q}\right] \quad \text{.....(3.7),}$$

where t is time of the ISF. The joint fitting parameters, which are the same for all five ISF curves, are a, b, τ_0, and γ among all six parameters. The other two parameters having subscript Q are p_Q and β_Q. They can be different for every five Qs. p_Q here is the Elastic Incoherent Structure Factor (EISF).

Therefore, fitting procedure is done by fitting five curves together,

(1) $I(Q=0.4,t)$,

(2) $I(Q=0.5,t)$,

(3) $I(Q=0.6,t)$,

(4) $I(Q=0.7,t)$, and

(5) $I(Q=0.8,t)$.

simultaneously to minimize the global chi-square.

From this MD simulation generated ISF functions, the Q-dependent average translational relaxation time $\tau_T(Q)$ (ps) for each temperature is obtained using

$$\tau_T(Q) = \tau_0(aQ)^\gamma \quad \text{.....(3.8).}$$

Based on the ISF fitting result, one can also compute the Q-independent average translational relaxation time $\langle\tau_T\rangle$ (ps) for each temperature from 180 K to 280 K. It is defined as

$$\langle\tau_T\rangle = \tau_0 \times \frac{\Gamma\left(\frac{1}{\beta}\right)}{\beta} \quad \text{.....(3.9).}$$

Then, one can obtain a result graph of the average translational relaxation time that looks like in the section 4.1 and 4.2. From the graph of $\langle\tau_T\rangle(Temp.)$ *vs.* $\frac{1000}{Temp}$, the Vogel-Fulcher-Tammann (VFT) law and the Arrhenius law are applied to fit the average translational relaxation time values. For high temperature, VFT law equation $\langle\tau_T(\mathrm{T})\rangle = \tau_0 \times \exp\left(D \times \frac{T_0}{(T - T_0)}\right)$ can be applied to fit the data. However, one should use the Arrhenius law $\langle\tau_T(\mathrm{T})\rangle = \tau_0 \times \exp\left(\frac{E_a}{RT}\right)$ to fit with data but with the same prefactor τ_0. In the both of the equations to describe water's states, fitting parameters are τ_0, D, and T_0

for the VFT, and τ_0 and E_a for the Arrhenius. Here, T_0 is the ideal glass transition temperature. Parameters will be discussed in the following sections regarding their meaning.

Chapter 4. RESULTS – SOMPARISON AND DISCUSSION

In this Chapter, the experimental results and MD simulations result (in case of hydration level h = 0.3) are compared in terms of the average translational relaxation time.

Then, I compare three different hydration levels of the target system containing hydration water. Total 33 MD runs are performed in four Apple Macintosh workstations which have eight (8) processors respectively.

Simulation runs are,

(1) 180 K to 280 K with 10 K difference (11 runs) at hydration level h = 0.3 (totally 484 water molecules),

(2) 180 K to 280 K with 10 K difference (11 runs) at hydration level h = 0.45 (totally 726 water molecules),and

(3) 180 K to 280 K with 10 K difference (11 runs) at hydration level h = 0.6 (totally 968 water molecules).

4.1. Average Translational Relaxation Time $\langle \tau_T \rangle$ and ISF

In the average translational relaxation time $\langle \tau_T \rangle$ graph it is strongly encouraged to describe its analysis procedure again because its deep importance. At high temperatures, above $T_L \approx 220$ K, $\langle \tau_T \rangle$ obeys the VFT law, namely, $\langle \tau_T \rangle = \tau_0 \exp\left[DT_0/(T - T_0)\right]$, where D is a dimensionless parameter providing the measure of fragility and T_0 is the ideal glass transition temperature. Below T_L, the temperature dependence of $\langle \tau_T \rangle$ switches to an Arrhenius behavior, which is written as $\langle \tau_T \rangle = \tau_0 \exp(E_a/RT)$, where E_A is the activation energy for the relaxation process and R is basically the gas constant. This dynamic crossover from a super-Arrhenius (the VFT law) to the Arrhenius behaviors is cusp-like and thus it sharply defines the crossover temperature T_L.

4.1.1. Comparison between the experiment and MD simulation

QENS studies have been made on hydration water of Lysozyme. As mentioned before, Lysozyme hydration sample has the hydration level h = 0.3. Using the new time domain analysis, I can extract the average translational relaxation time, $\langle \tau_T \rangle$ for temperatures from 200 K to 270 K with good fitting result. In figure 16, one has already seen the $\log(\langle \tau_T \rangle)$ *vs.* $1000/T$ plot of the QENS results.

That result are compared with the MD simulation result that is also the Lysozyme hydration water having the hydration level h = 0.3. MD results are available from 180 K to 280 K as I expected from designing simulation configurations.

Experiments and MD show a slight difference in $\langle \tau_T \rangle$ value. The Fragile-to-Strong Crossover (FSC) temperature for experimental case $T_L(\textit{experiment})$ is obtained as 226 K, and for simulations $T_L(\textit{simulations})$ is 222 K. Within the experimental and simulation error, namely, 10K, those actually argree well. In addition, both of them are in the range of the expected FSC temperature 225 ± 10 K. Even though there is difference in the average translational relaxation time and in the crossover temperature, this comparison can tell us that this could be reasonable. Since one can check the Arrhenius parts of both are parallel, one can say those could possibly slightly shifted by some factors, which could be the difference between 'real' hydration water and the simulated hydration water. This reasonable difference is easily accepted to researchers, because simulations cannot generate any unexpected situation changes and T_L difference is within the experimental error.

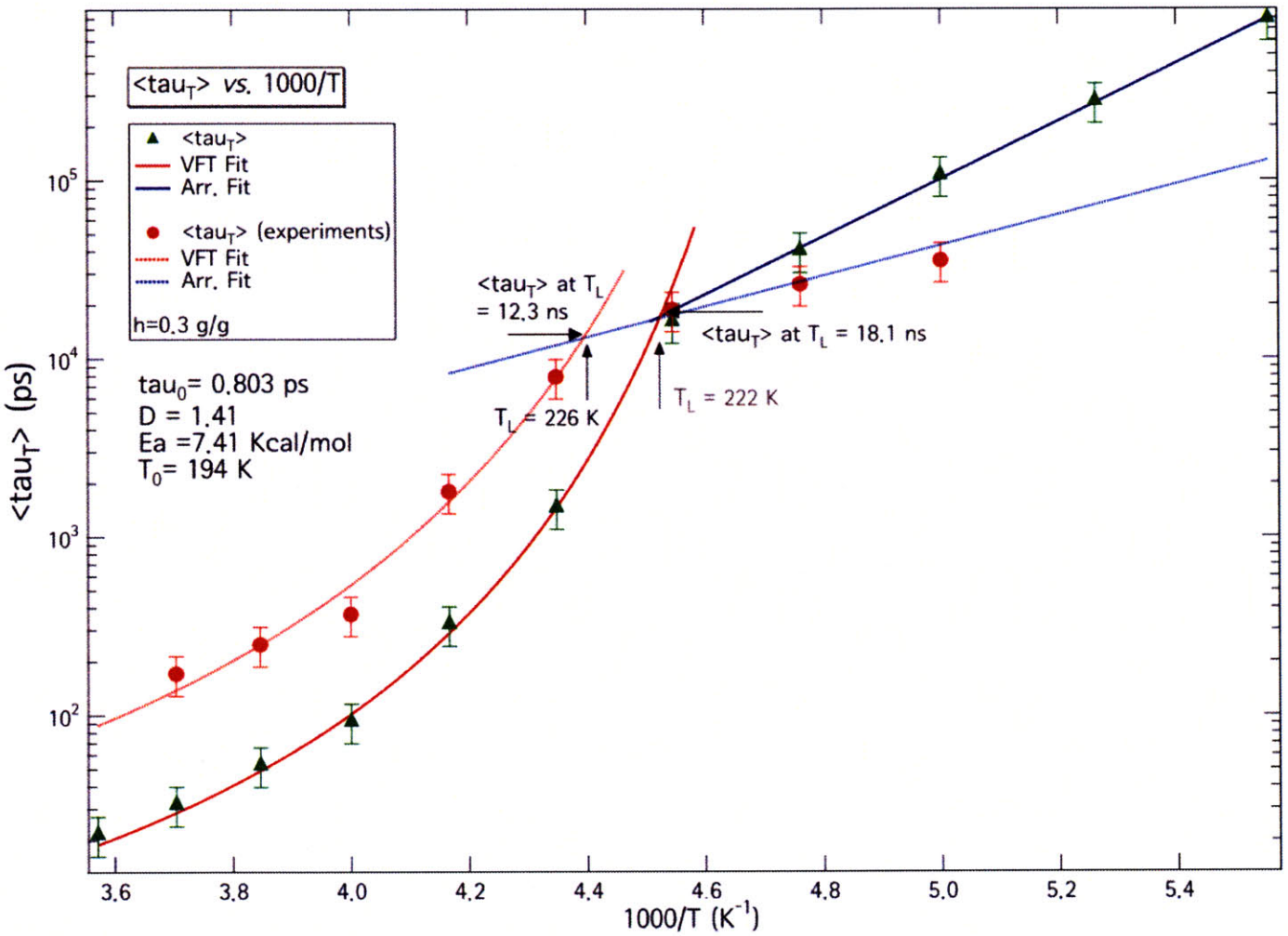


Figure 16. The extracted $< \tau_T >$ from fitting of QENS spectra by RCM plotted in the log scale *vs.* $1/T$. Both of results from the experiments and simulations have the FSC temperatures around 225 ± 10 K. I can say those agree well with each other accompanying a slight difference. In the Arrhenius region, two results are parallel. All of the fitting parameters shown here are its results for the Arrhenius fit of the MD ISF.

4.1.2. Comparison among MD simulation results of three different hydration levels

This section shows a series of the average translational relaxation time *vs.* 1000/T graph with a specific hydration level. Hydration levels for the samples are 0.3, 0.45 and 0.6.

As one can see in the figure 20, the FSC temperature is decreasing as I increase the hydration level. That means if we have more water in hydration water around the Lysozyme (or a sort of biopolymers), crossover phenomenon occurs at higher temperature. T_0 increases when the sample hydration level is going lower. Since T_0 means the ideal glass transition temperature, it is reasonable that it becomes high when the sample has lower hydration level. Compared the our group's previous research [51], T_0 of the case having hydration level h = 0.3 agrees with it, as T_0 is around 200 K.

As mentioned above, figure 20 shows the comparison chart of three different hydration levels. In VFT region, which is high temperature ones, the exponent of the three cases is increasing when we lower the hydration level. In other words, the average translational relaxation time becomes increasing more faster when we have lower hydration level. It can be seen reasonable, because it is strongly related to the number of hydrogen atoms.

The dynamic crossover observed in experiments can be attributed only to the crossover phenomenon by evaluating the average translational relaxation time by analyses of the long-time decays of the ISF of the hydrogen atoms attached to a typical water molecule [52, 53]. This means that even though MD simulations can analyze quantities for other hydrogen atoms in other molecule, such as Lysozyme, I have focused on water, which has been treated just solvents in the simulation. Then, ISF can be obtained in a clear way.

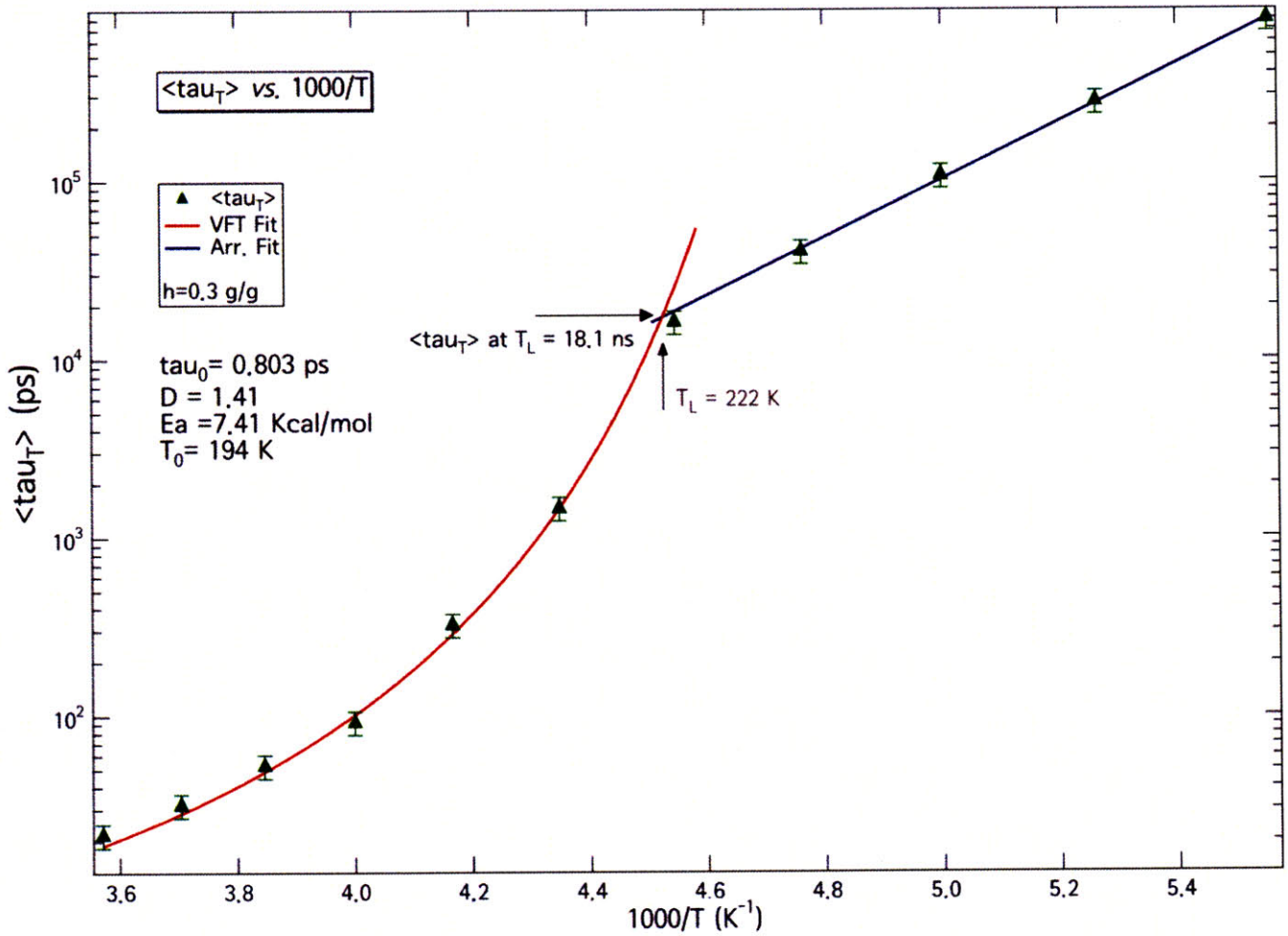


Figure 17. The extracted $< \tau_T >$ from fitting of MD simulation ISF spectra by RCM plotted in the log scale *vs.* $1/T$ with hydration level h = 0.3. The crossover temperature given here is 222 K, which is within 225 K ± experimental error 10 K.

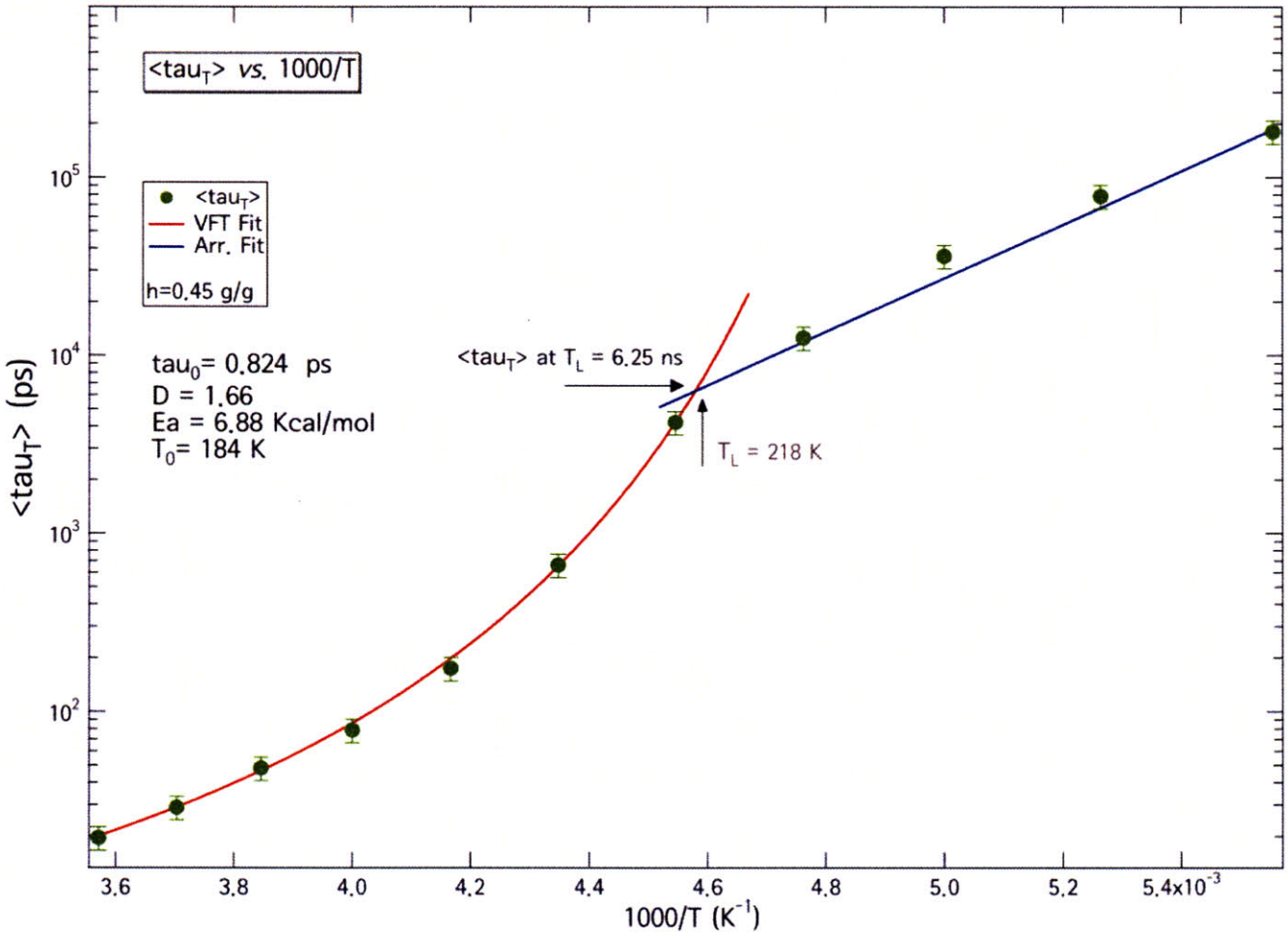


Figure 18. The extracted $< \tau_T >$ from fitting of MD simulation ISF spectra by RCM plotted in the log scale *vs.* $1/T$ with hydration level h = 0.45. The crossover temperature given here is 222 K, which is within 218 K ± experimental error 10 K.

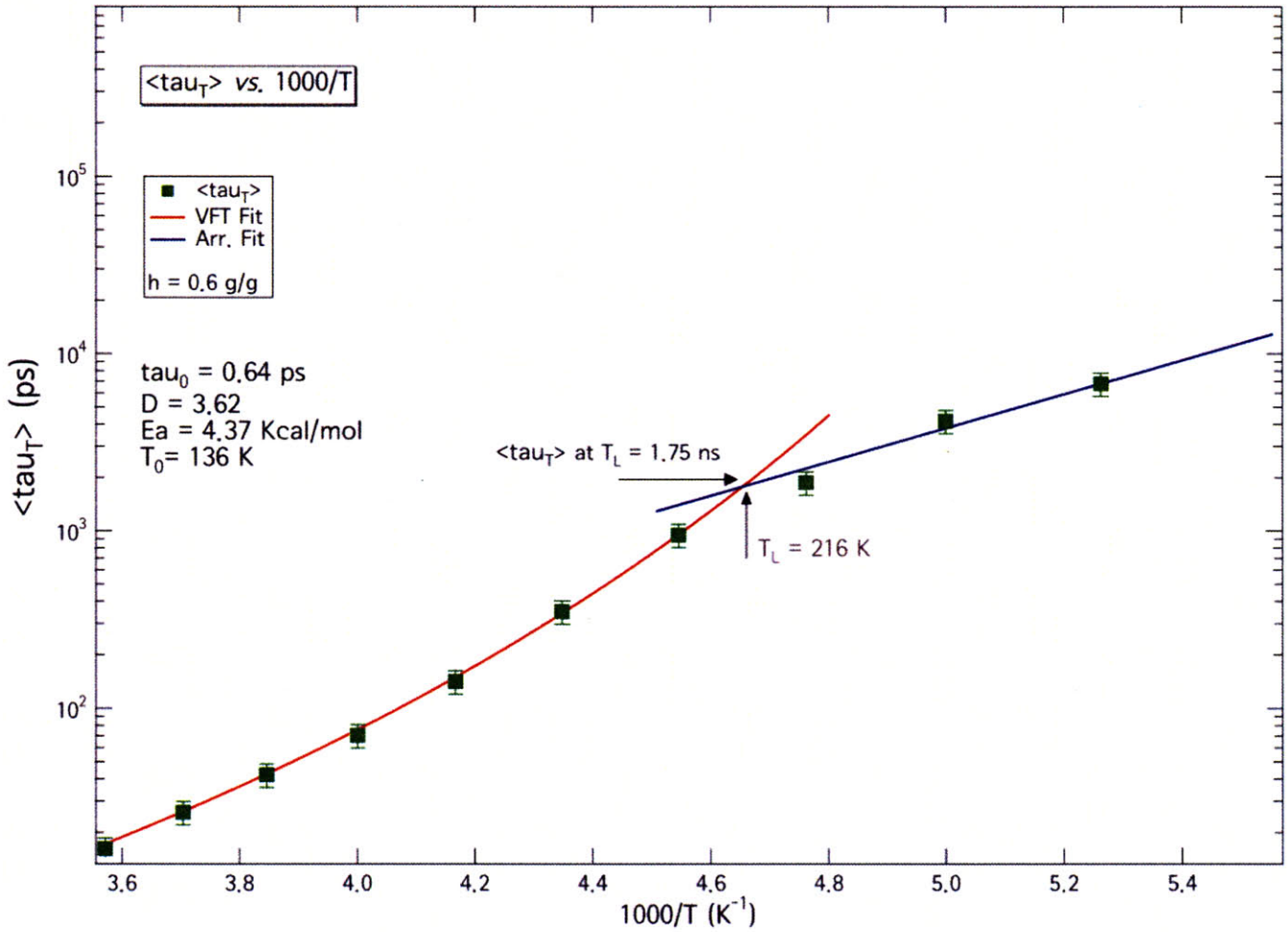


Figure 19. The extracted $< \tau_T >$ from fitting of MD simulation ISF spectra by RCM plotted in the log scale *vs.* $1/T$ with hydration level h = 0.6. The crossover temperature given here is 222 K, which is within 216 K ± experimental error 10 K.

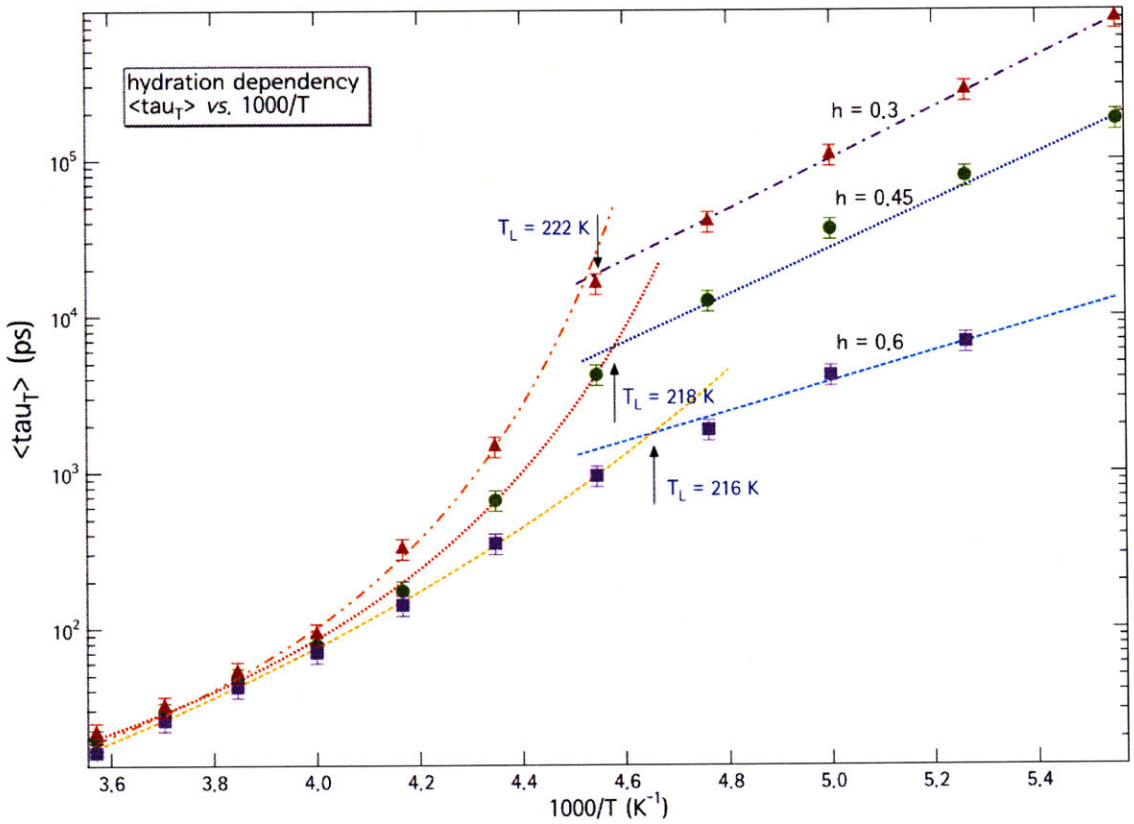


Figure 20. This figure compares all of the three difference cases of MD simulation. As one can see in the graph, there are three hydration level shown: h = 0.3, 0.45 and 0.6. The average translational relaxation time $< \tau_T >$s are extracted from RCM fitting to the ISF curves of the each temperature and hydration level. Those are plotted in the log scale *vs.* $1/T$. As already shown before, all of them show the crossover temperature around 225 K ± 10 K, so I can say that the crossover in water strongly exists and it occurs around 225 K. In VFT region, which is high temperature ones, the exponent of the three cases is increasing (becomes increasing faster) when we lower the hydration level. This could be reasonable, because it is strongly related to the number of hydrogen atoms. While the Arrhenius fits look almost parallel among hydration levels. As a result, the intersecting point of those two fit functions make the crossover temperature have a sort of tendency that it increases as lowering the hydration level of the hydration Lysozyme water sample.

4.2. Mean Square Displacement (MSD) and diffusion constant, D

The realistic powder model, which is used in this research with varying the hydration level by controlling the number of concentration of, can actually reproduce experimental data within the statistical error bars.

In particular, we show the striking agreements for a rough crossover temperature, where the inclines are changing in the MSD of hydrogen atoms in water. For all three MD simulation configurations, it agrees with each other. The significance of these comparisons is that hydration level actually has not affected to the existence of the crossover phenomena: in other words the crossover exists regardless the hydration levels.

What we can conclude from this section is that the crossover phenomenon exists and it changes depending on the hydration level.

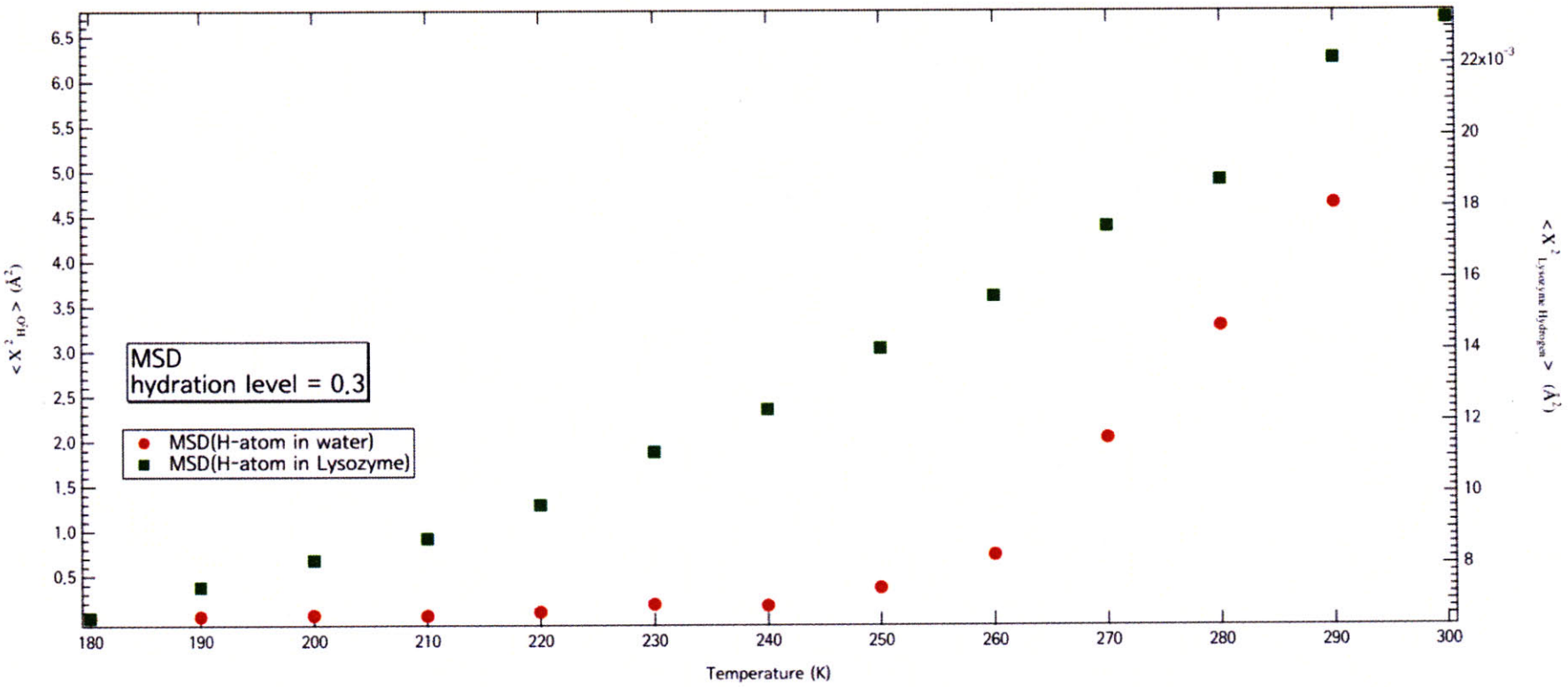


Figure 21. MSD of two cases: hydrogen atoms in Lysozyme and water of hydration level h = 0.3 condition. Please be careful on that the left axis refers to the water case, right for Lysozyme case.

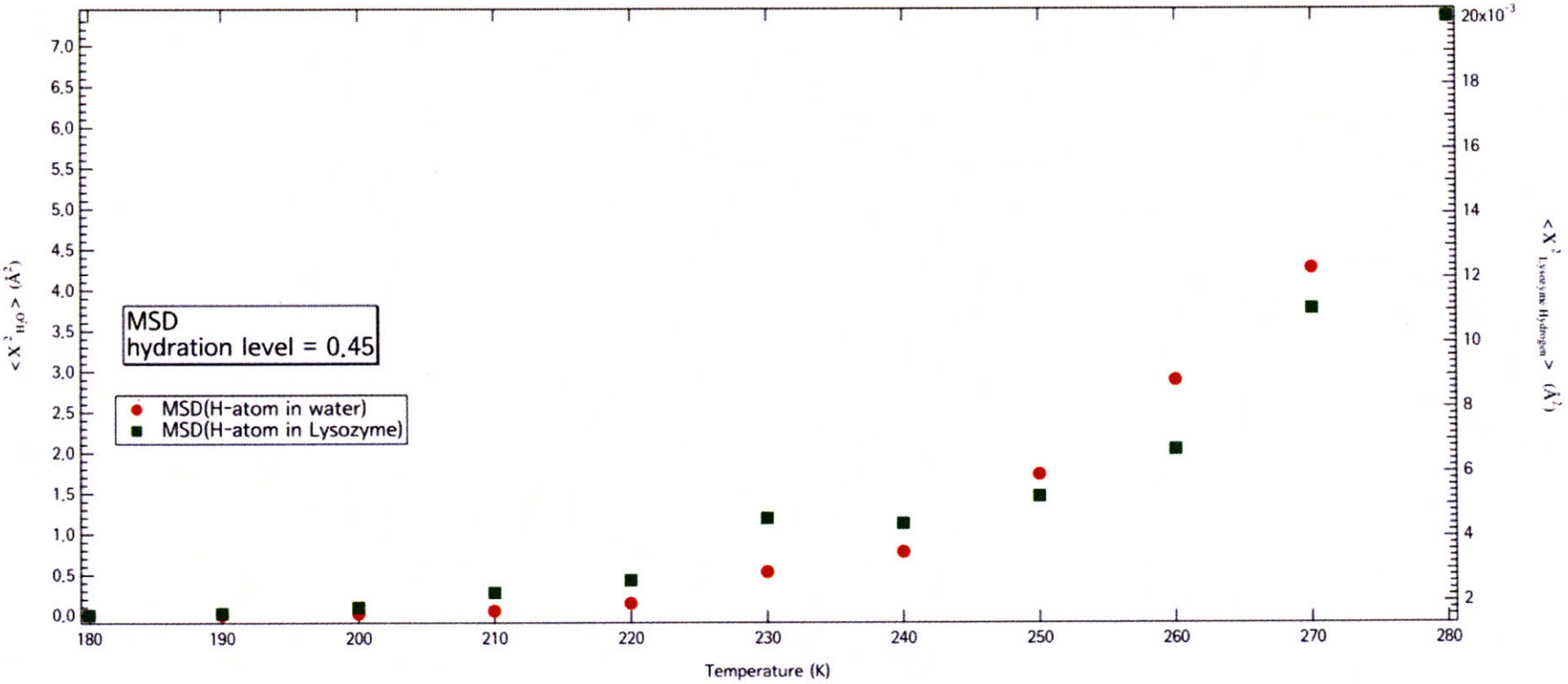


Figure 22. MSD of two cases: hydrogen atoms in Lysozyme and water of hydration level h = 0.45 condition. Left axis is for the water case, right for Lysozyme case.

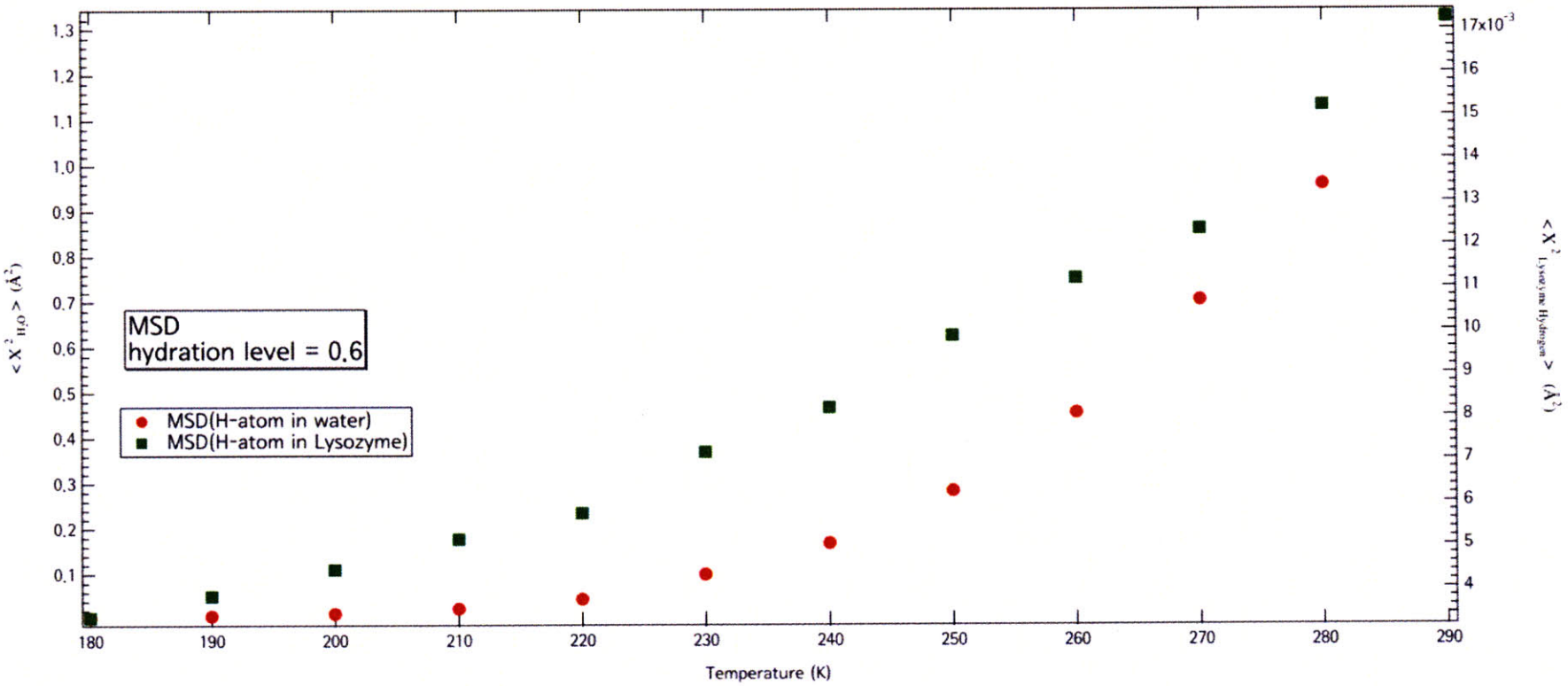


Figure 23. MSD of two cases: hydrogen atoms in Lysozyme and water of hydration level h = 0.6 condition. Left axis is for the water case, right for Lysozyme case.

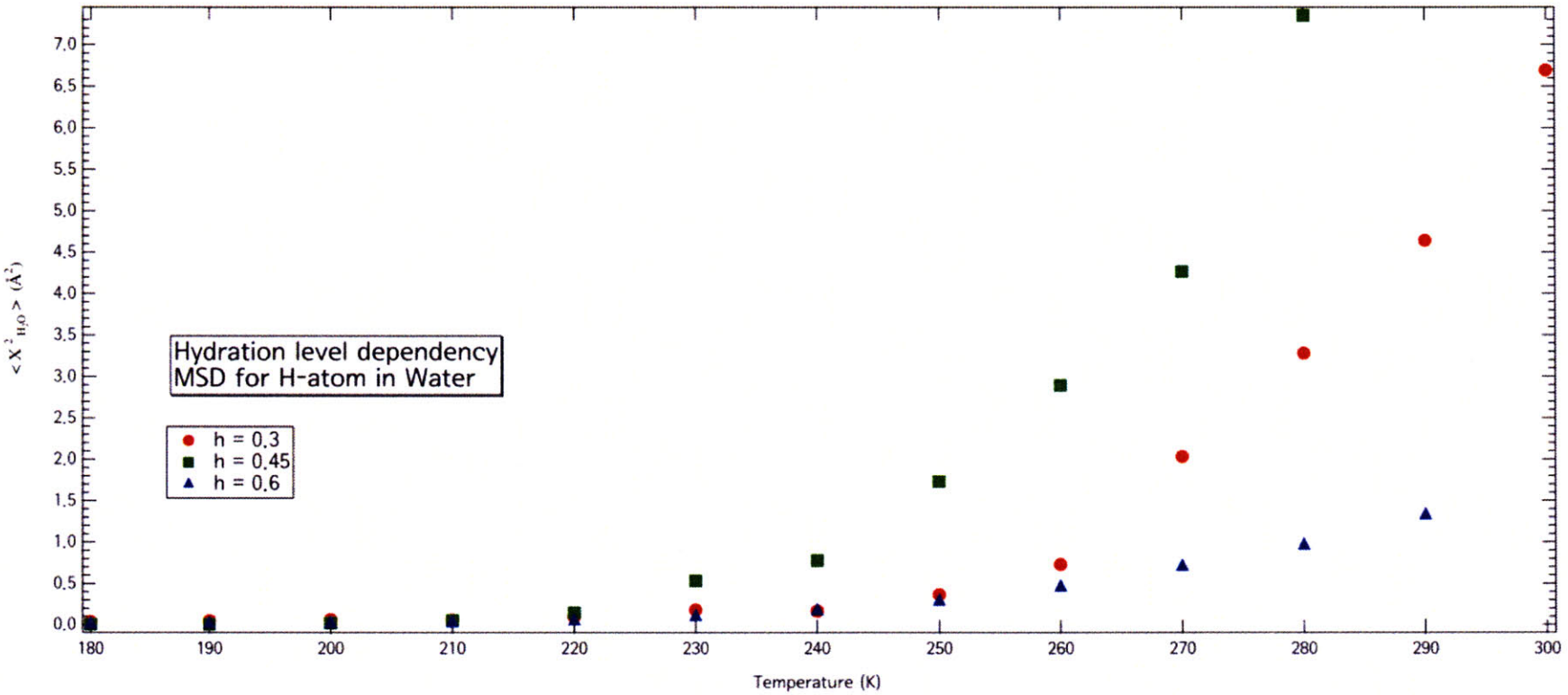


Figure 24. Comparison among three hydration levels in terms of MSD of hydrogen atoms in water. As we have seen in the QENS experiments [54], all cases show a point where the inclines are changing.

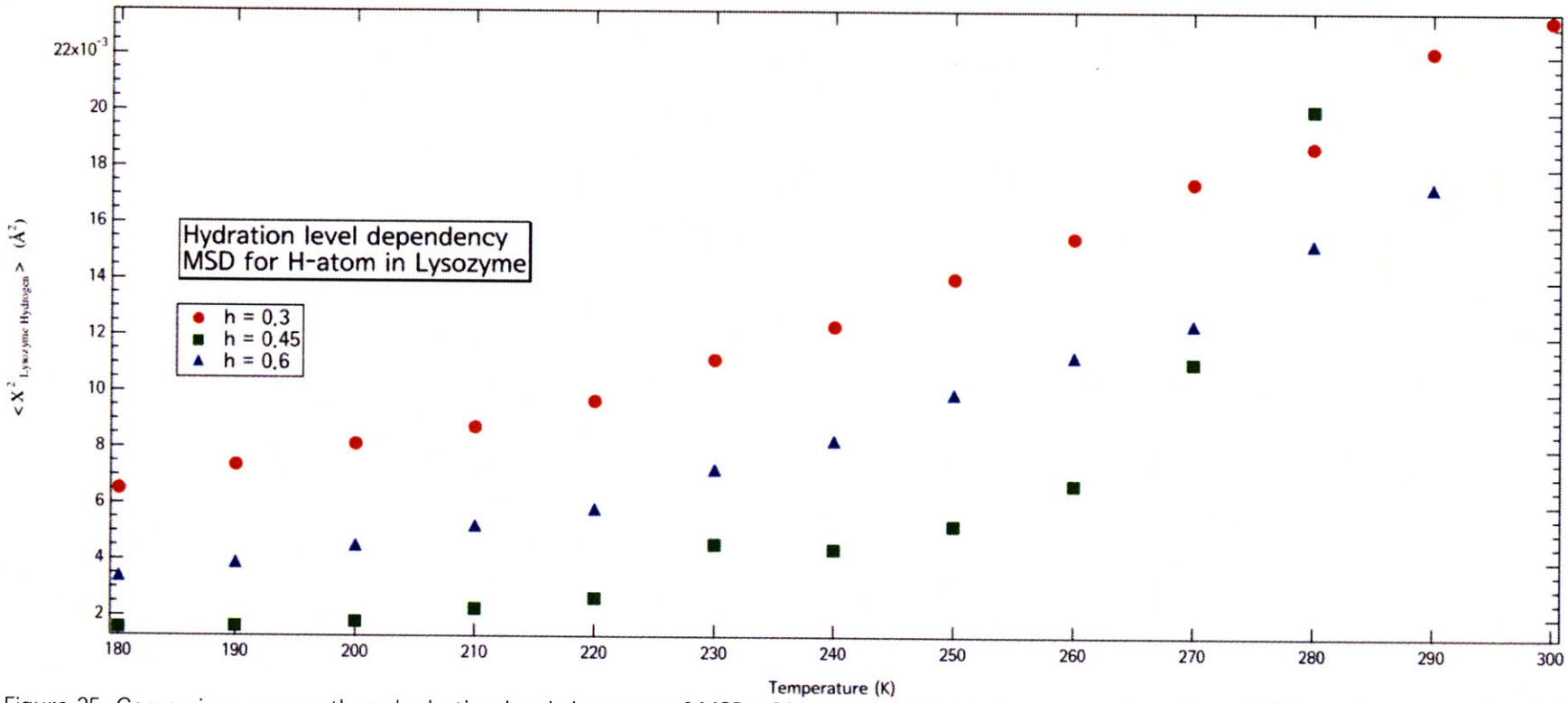


Figure 25. Comparison among three hydration levels in terms of MSD of hydrogen atoms in Lysozyme. Even though it is weak, one can check that there is the crossover phenomenon by seeing the incline changes of two linear lines.

4.3. Number of hydrogen bonds

From the comparison result in figure 29 (with figure 27 and 28 also), the crossover is shown only in the hydrogen bonds between water and water case. However one cannot see any crossover or changing points in the hydrogen bonds between water and Lysozyme. This fact leads us an important subsequent idea: hydrogen bonds between water and water are more important and can possibly trigger biopolymer's behavioral change.

I could suggest an explanation why any kind of tendency does not appear in the lowest hydration level case (h=0.3). It might be that the number of hydrogen atoms is too low (484 atoms here) to show kinds of tendency or significant change in the viewpoint of "the number of hydrogen bonds". Following this explanation, it is quite reasonable to treat the h = 0.3 case as a relatively weak result. Therefore, it can be strongly believed that the crossover also exists in lower hydration level, even though it is hard to observe the crossover phenomenon here. The reason is that the average translational relaxation time graph shows the striking and strong cusp-like behaviors.

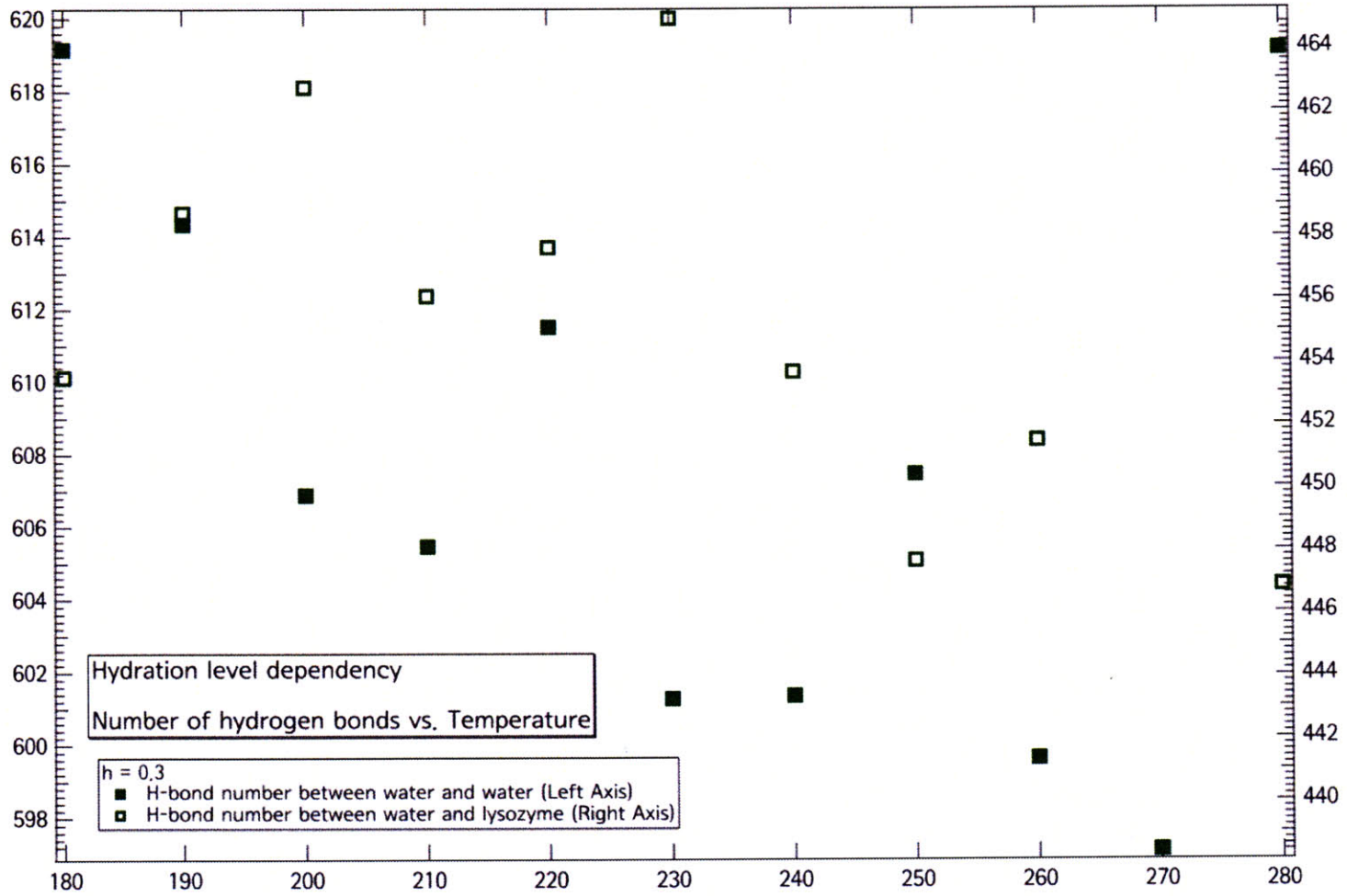


Figure 26. Number of hydrogen bonds (1) between water and water and (2) between water and Lysozyme. This refers the case of hydration level h = 0.3. We cannot find any crossover or tendency in this case.

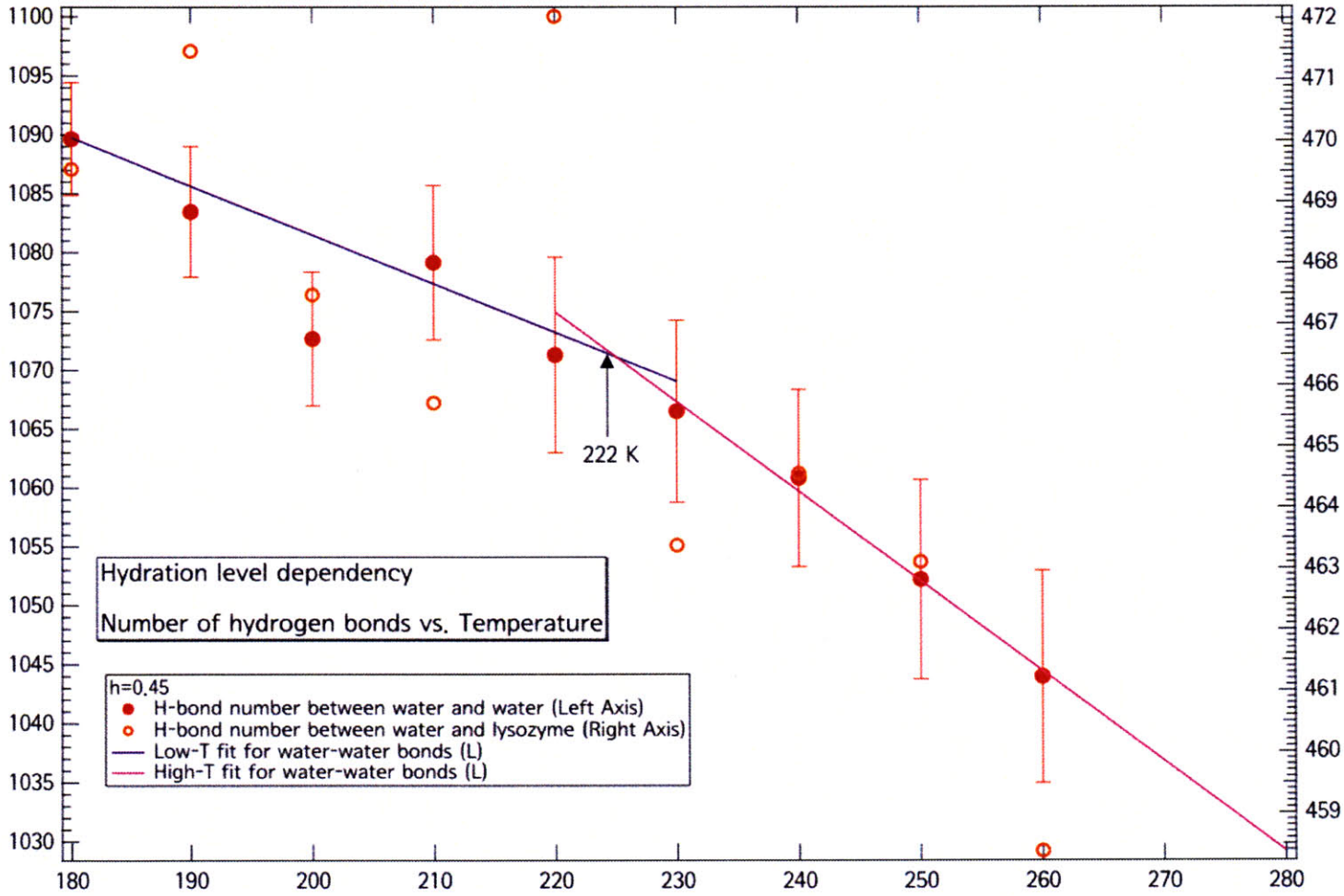


Figure 27. Number of hydrogen bonds (1) between water and water and (2) between water and Lysozyme. This refers the case of hydration level h = 0.45. In the case of bonds between water and water, one can check there is a sort of crossover temperature at 222 K.

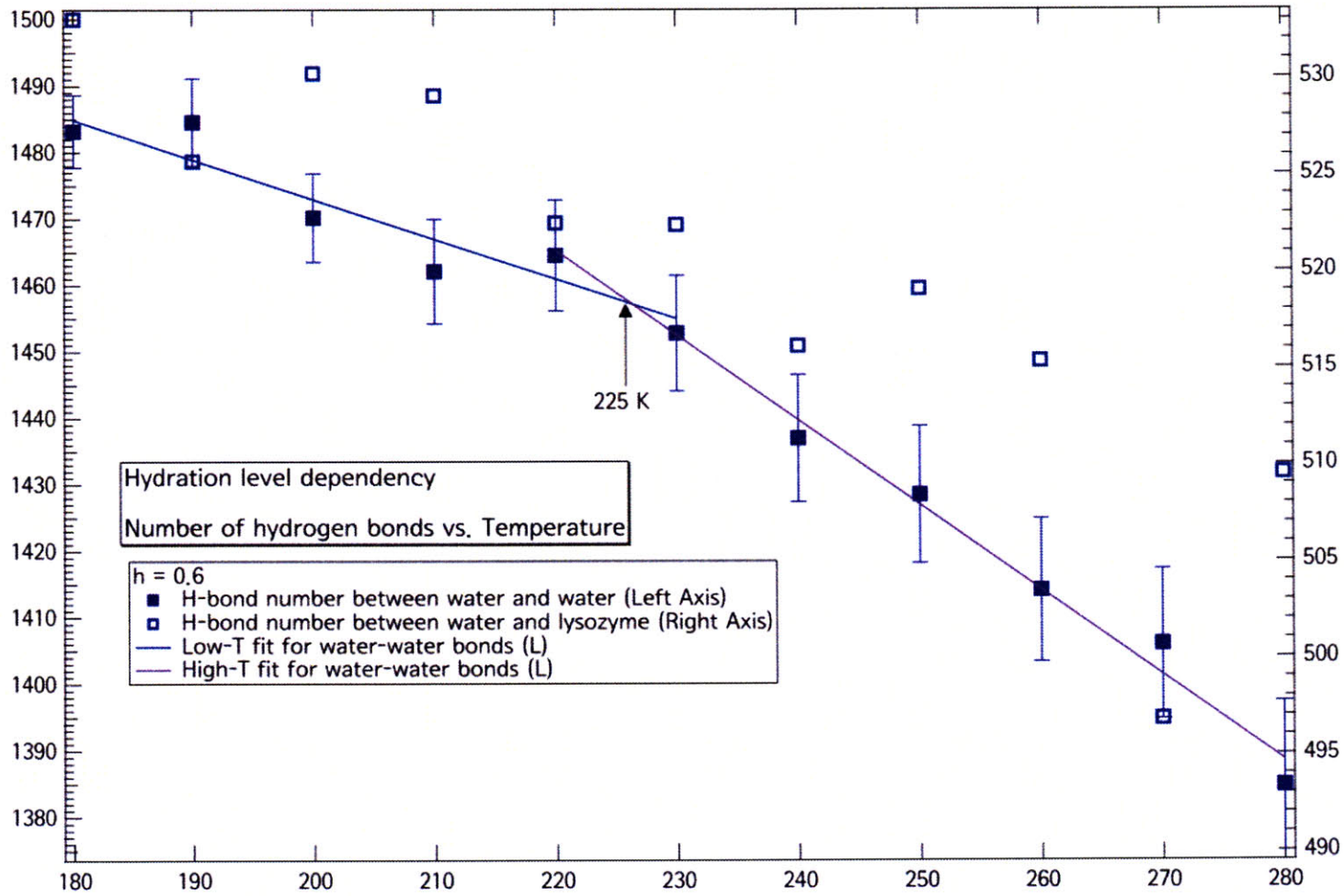


Figure 28. Number of hydrogen bonds (1) between water and water and (2) between water and Lysozyme. This refers the case of hydration level h = 0.45. In the case of bonds between water and water, one can check there is a sort of crossover temperature at 225 K.

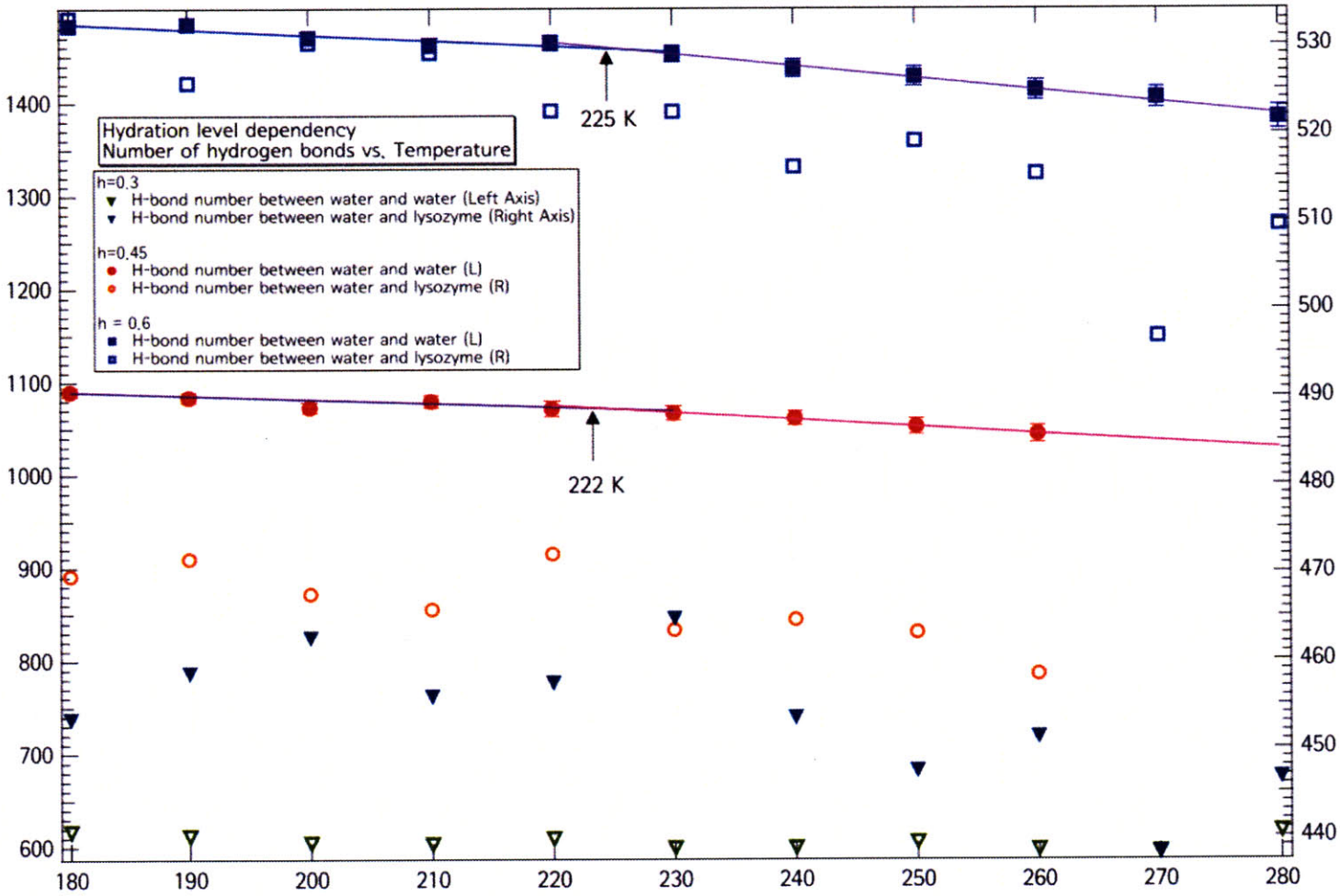


Figure 29. This graph compares all the results of three different hydration levels. We can see that two higher hydration levels (h=0.45 and h=0.6) show crossover in terms of the number of hydrogen bonds.

4.4. Structure factor, $S(Q)$

As one can see in the comparison graphs figure 33~35, the hydration level changes do not affect on the structure factor. Even though there exist some noise in the structure, all of them oscillate in almost the same way. In other words, the peak positions are independent on the hydration levels.

We observe that it rather depends on temperature changes at small Q region, while large Q parts are almost the same. If the temperature is low, the structure factor has the larger value. One can see that this is reasonable result.

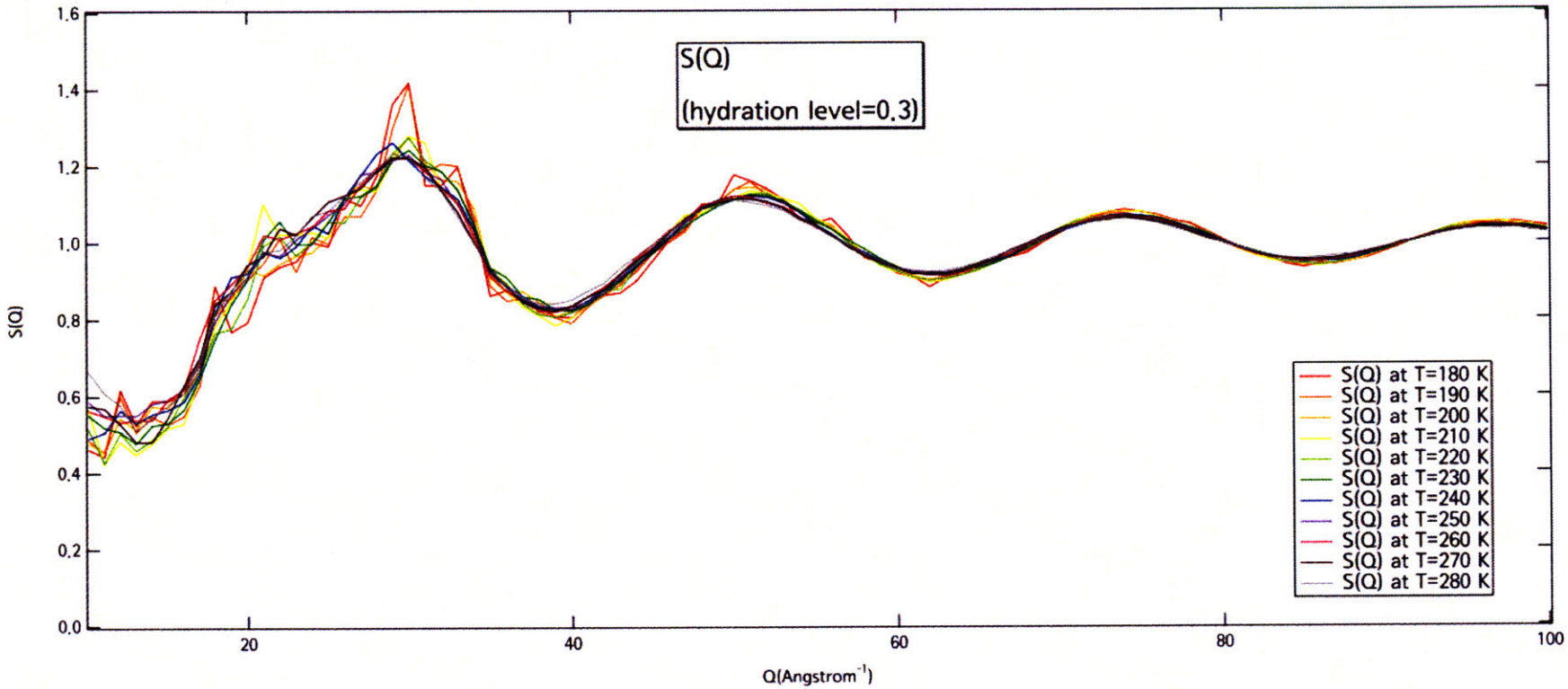


Figure 30. The structure factor of hydration h = 0.3 case for all simulated temperatures.

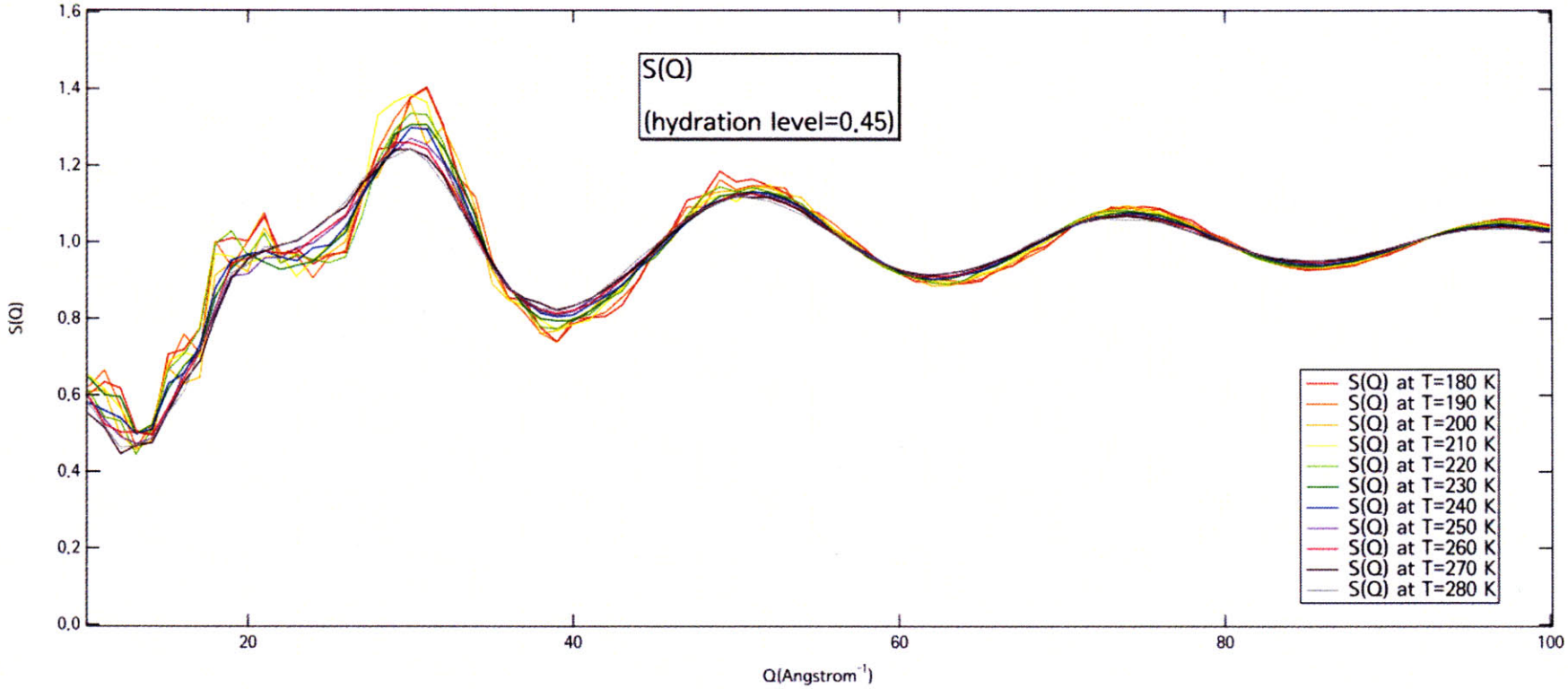


Figure 31. The structure factor of hydration h = 0.45 case for all simulated temperatures.

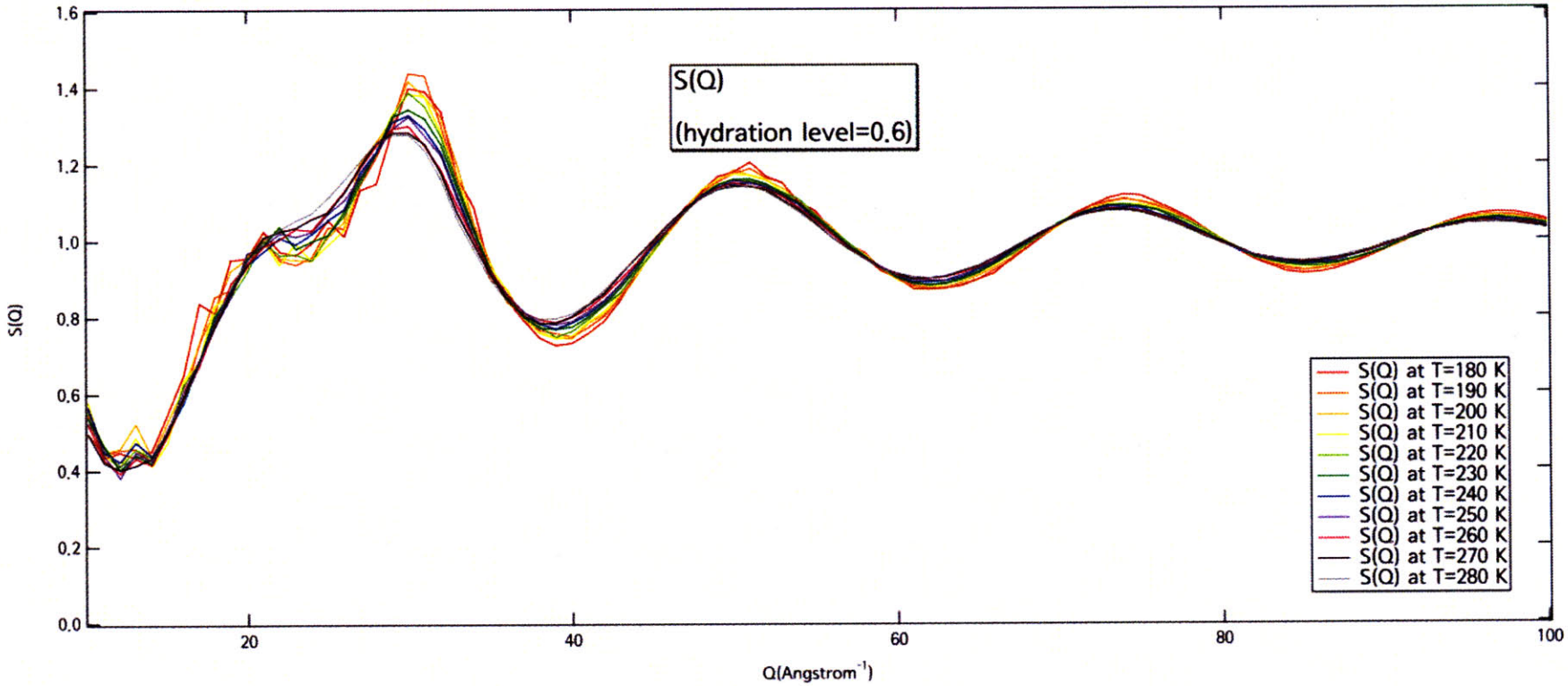


Figure 32. The structure factor of hydration h = 0.6 case for all simulated temperatures.

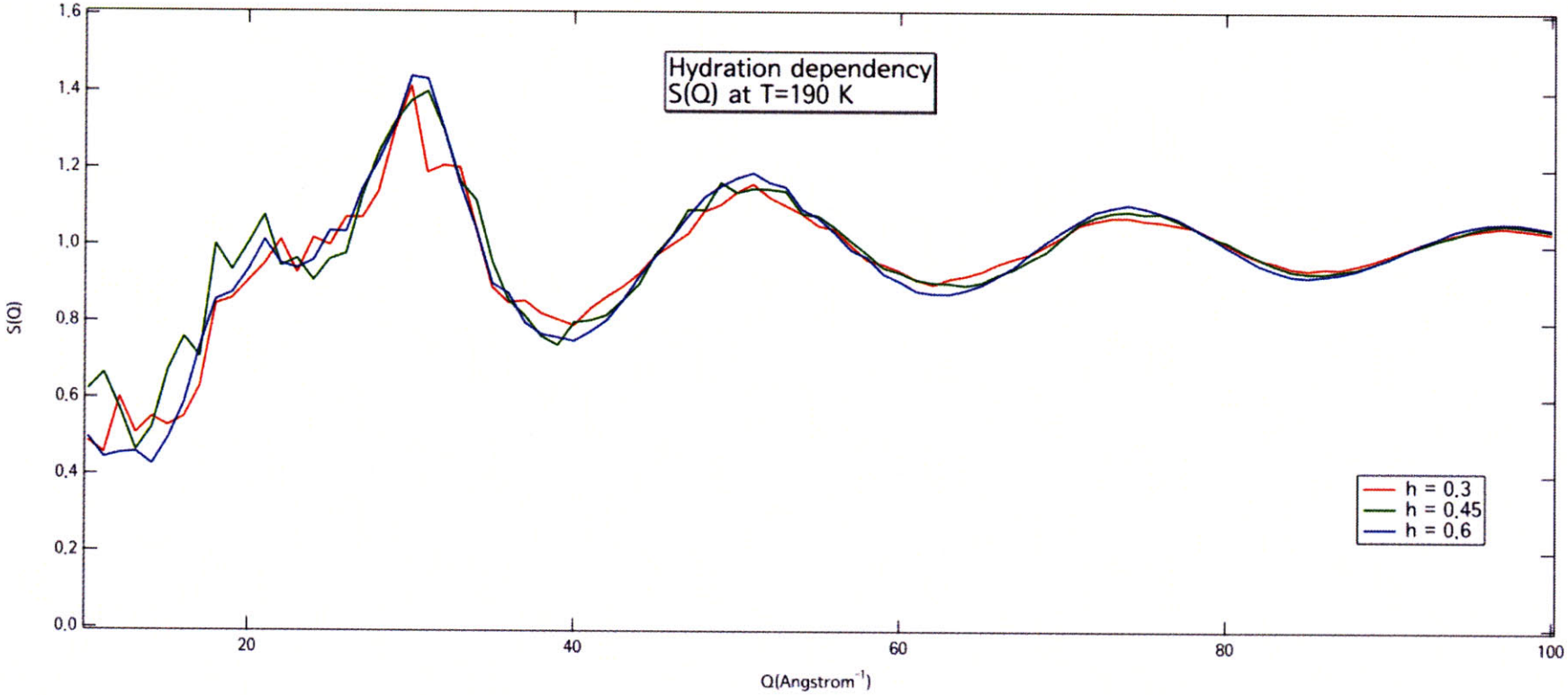


Figure 33. A comparison among three different hydration levels at the temperature T = 190 K

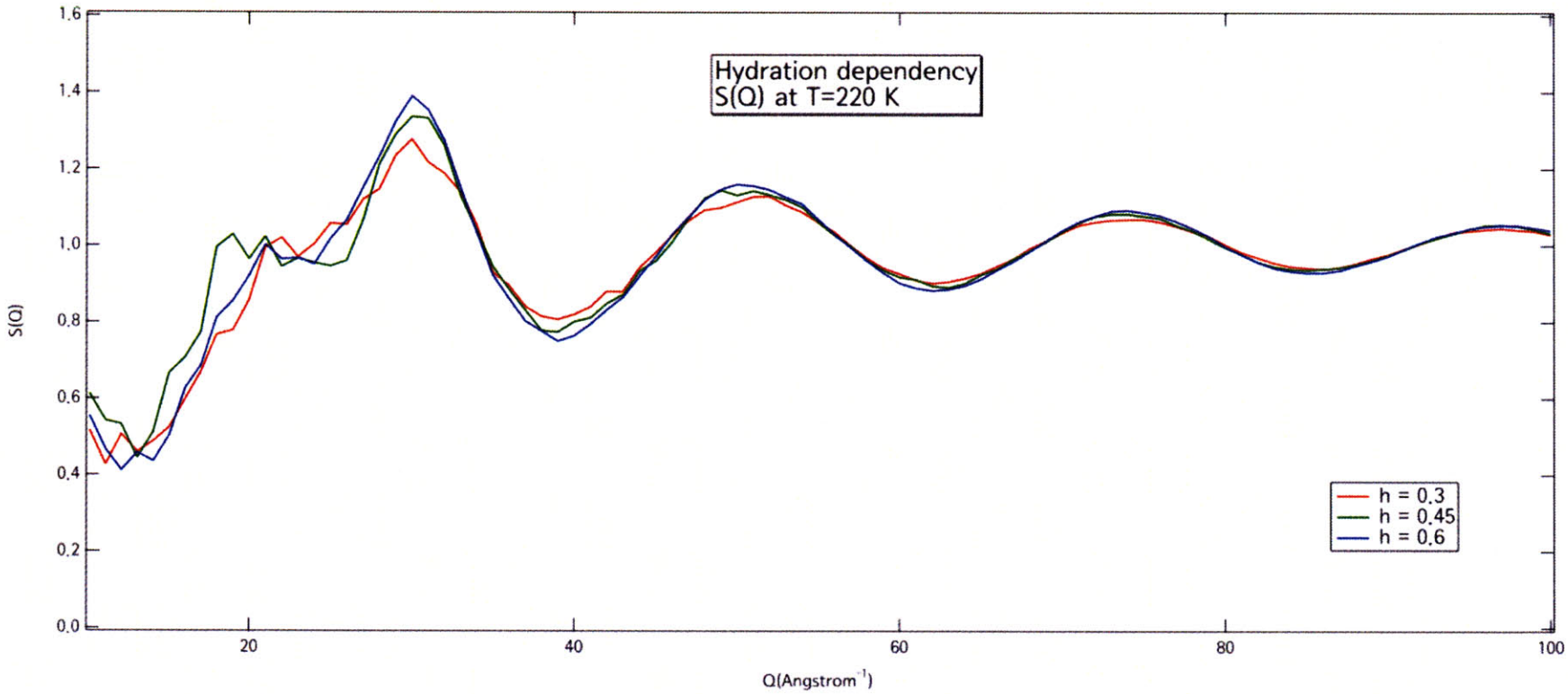


Figure 34. A comparison among three different hydration levels at the temperature T = 220 K

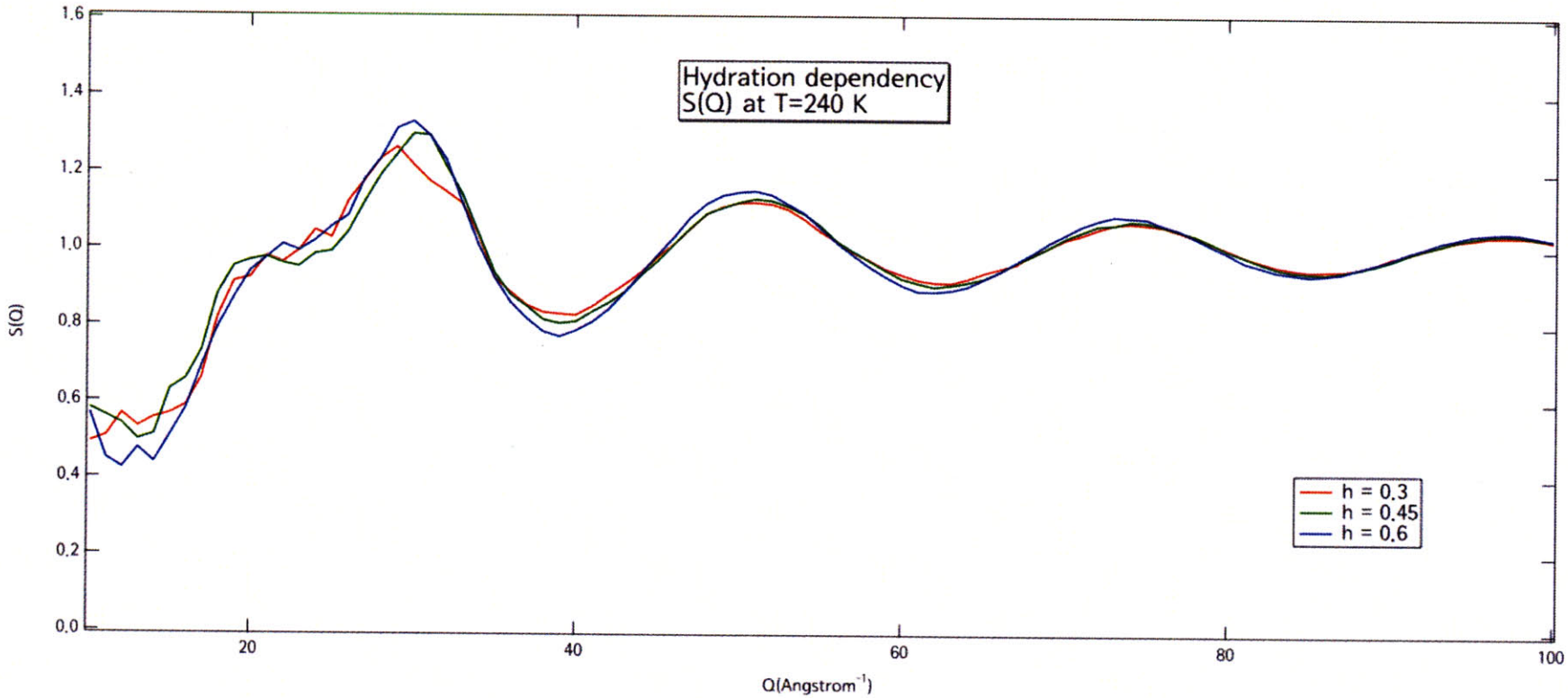


Figure 35. A comparison among three different hydration levels at the temperature T = 240 K

4.5. Radial Distribution function, $g(r)$

As one can see in the comparison graphs, the hydration level changes do not affect on the structure factor. Even though there exist some noise in the structure, all of them oscillate in almost the same way.

By the following three comparison figures 39~41, we can confirm that it could not be an accident that the function at h=0.45 has the smallest value. The main observation from comparing all the radial distribution functions at the different hydration level is that all of the hydration cases show the same peak positions. According to the meaning of the radial distribution function, the main observation provides us the locations of hydrogen atoms are not varying with hydration levels.

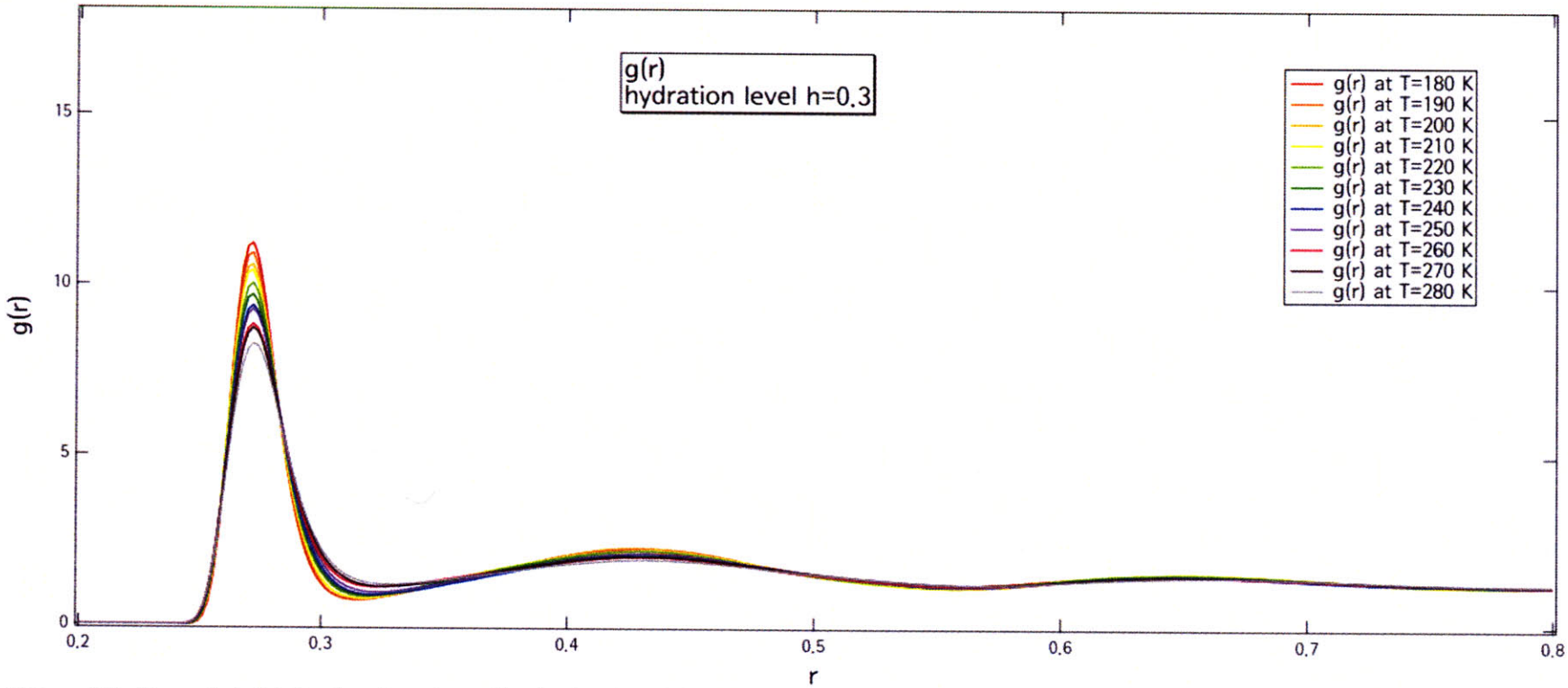


Figure 36. The radial distribution function of hydration h = 0.3 case. We can see that the lowest temperature has the highest peak value as seen in the structure factor.

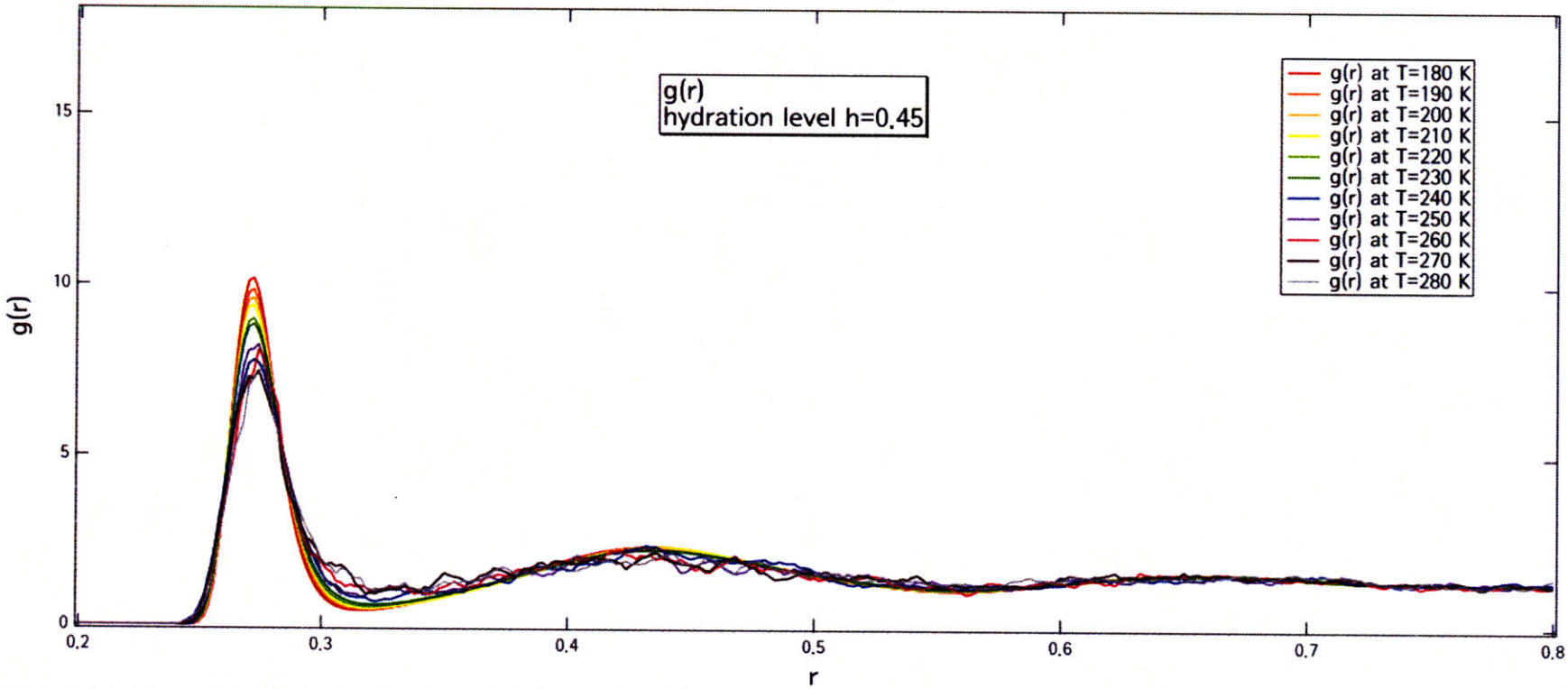


Figure 37. The radial distribution function of hydration h = 0.45 case. The radial distribution function has the highest values at the lowest temperature as seen in the 0.3 hydration level.

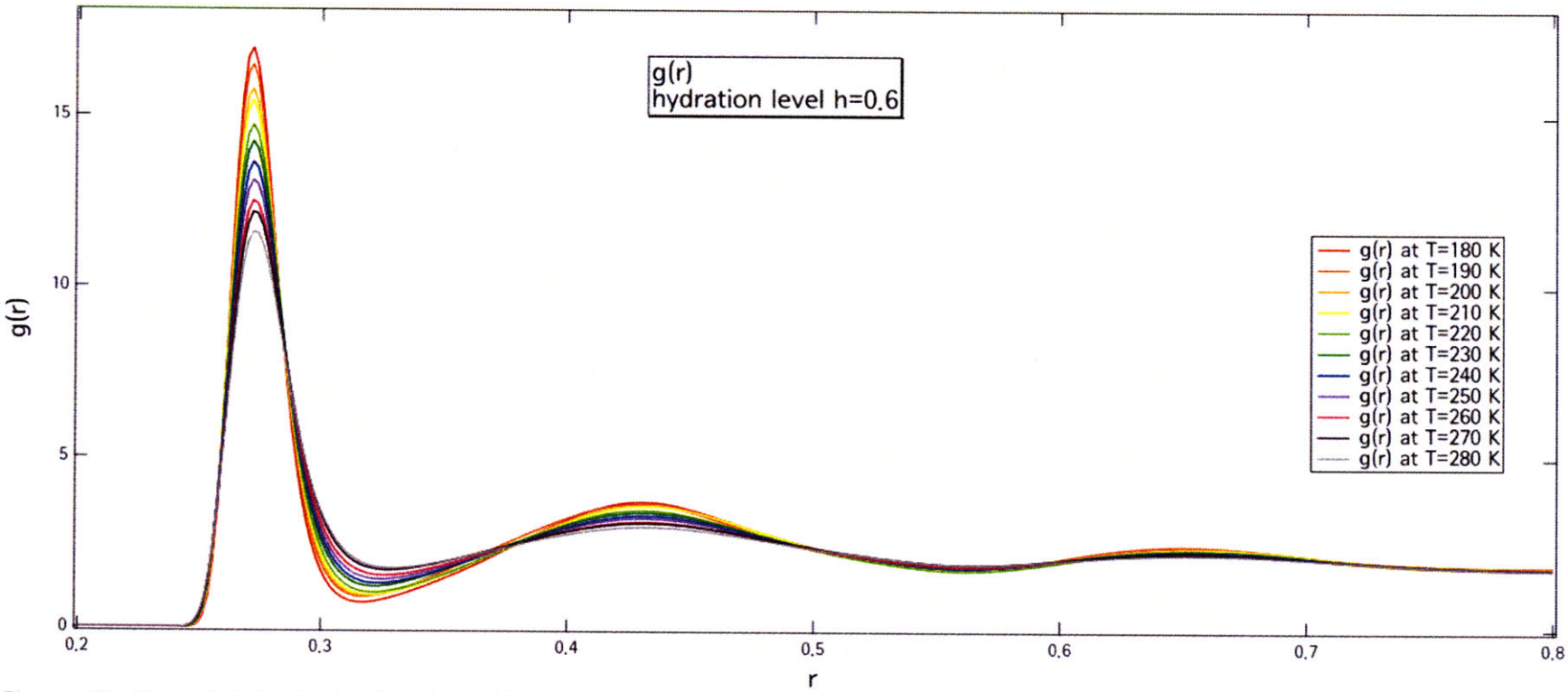


Figure 38. The radial distribution function of hydration h = 0.6 case.

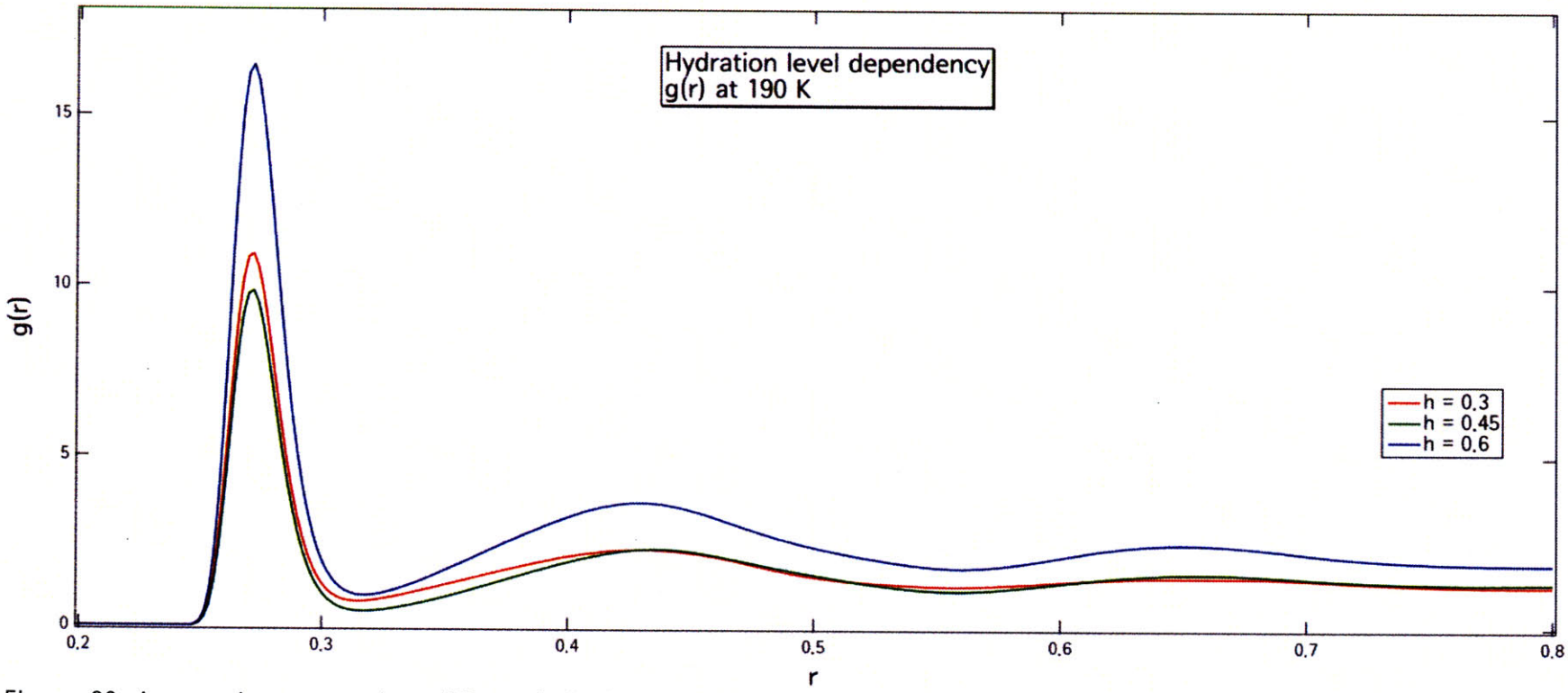


Figure 39. A comparison among three different hydration levels at the temperature T = 190 K. One can check that the function at h=0.45 has smaller values than h=0.3 case. The highest hydration level case shows the largest values. However, important fact is that all of the hydration cases show the same peak positions.

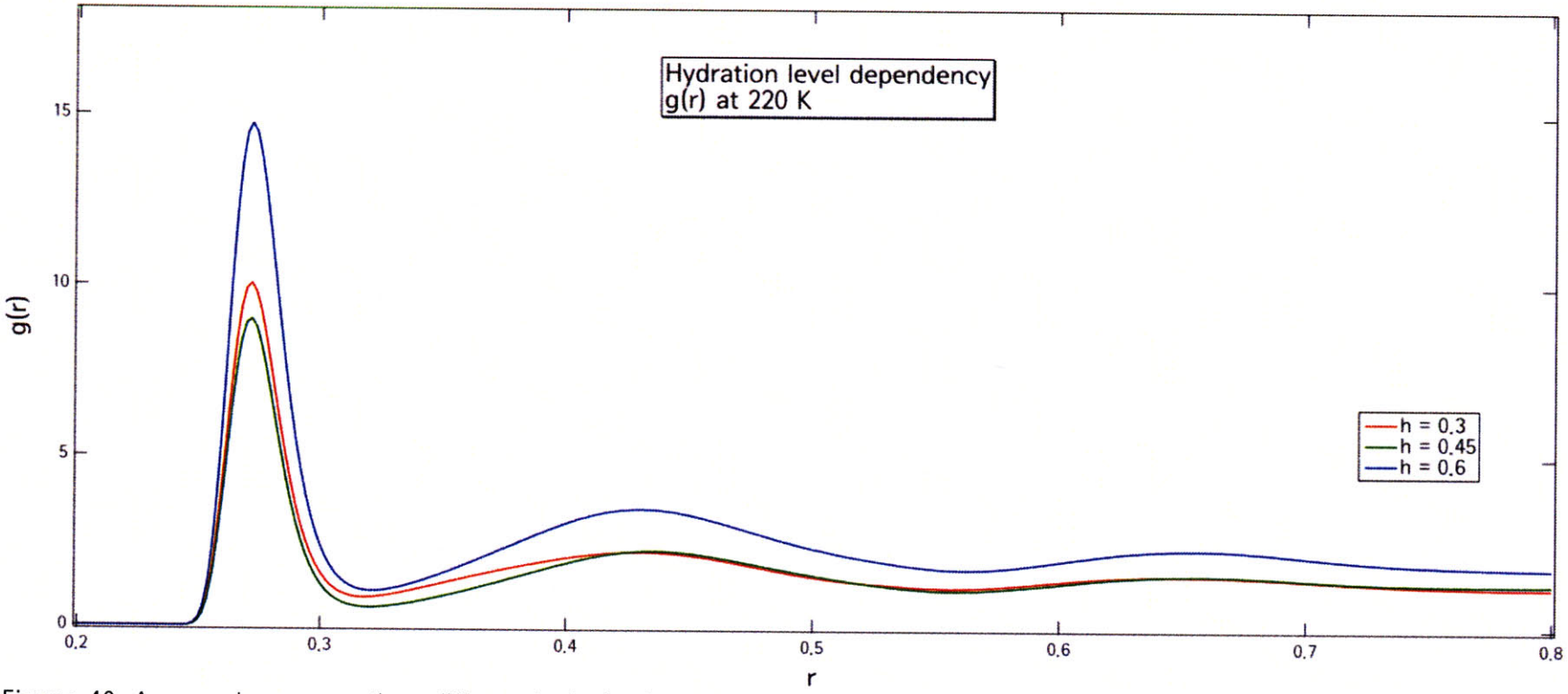


Figure 40. A comparison among three different hydration levels at the temperature T = 220 K. Here, the function at h=0.45 has the smallest value, too. As the same as before, the highest hydration level case shows the largest values. It can be also observed all of the hydration cases show the same peak positions.

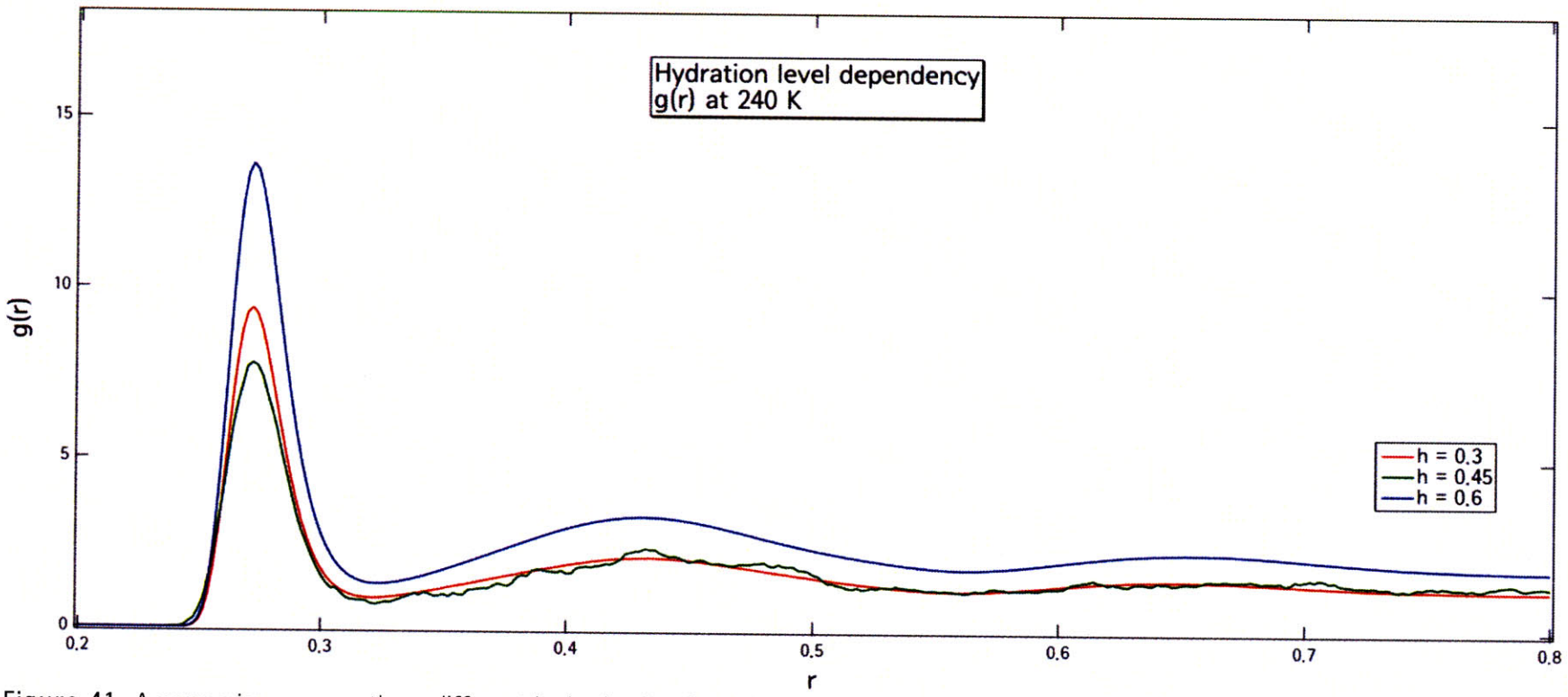


Figure 41. A comparison among three different hydration levels at the temperature T = 240 K. By these three comparison graphs, we can confirm that it could not be an accident that the function at h=0.45 has the smallest value. The highest hydration level case shows the largest values. Peak positions are still the same as for every hydration level case.

4.6. Autocorrelation functions and the hydrogen bond relaxation time

In this section, I am describing the autocorrelation functions of hydrogen bonds

(1) between water and water, and

(2) between water and Lysozyme.

From the figure 42~44, one can check that only higher temperature cases can be used to obtain the relaxation time. What we get from temperatures higher than 230 K is not useful, because the crossover is expected to occur around 220 K and 230 K. Therefore, the hydrogen bonds relaxation time does not provide meaningful information here and not shown here.

From that we can actually find that the shape (behavior) of the autocorrelation function is changed between 220 K and 230 K. Even though we are not able to extract the hydrogen bonds relaxation time, it is pointed out that there is a possibility to find the transition temperature by looking at the shape changes of all the graphs according to the temperature change.

The autocorrelation functions computed from the hydrogen bonds between water and water, and water and Lysozyme can be compared at different hydration levels at a specific temperature. We found that the smallest hydration level usually has the largest values. From this fact, we may say that hydrogen bonds changes a bit slowly due to the small number of hydrogen bonds. However, its tendency is almost the same.

From those temperature comparison and hydration level comparison, one can say that there could possibly exist a crossover or transition in between 220 K and 230 K (from the temperature dependent graph) and it could depends on hydration level changes even though its existence is confirmed (from the hydration level comparison).

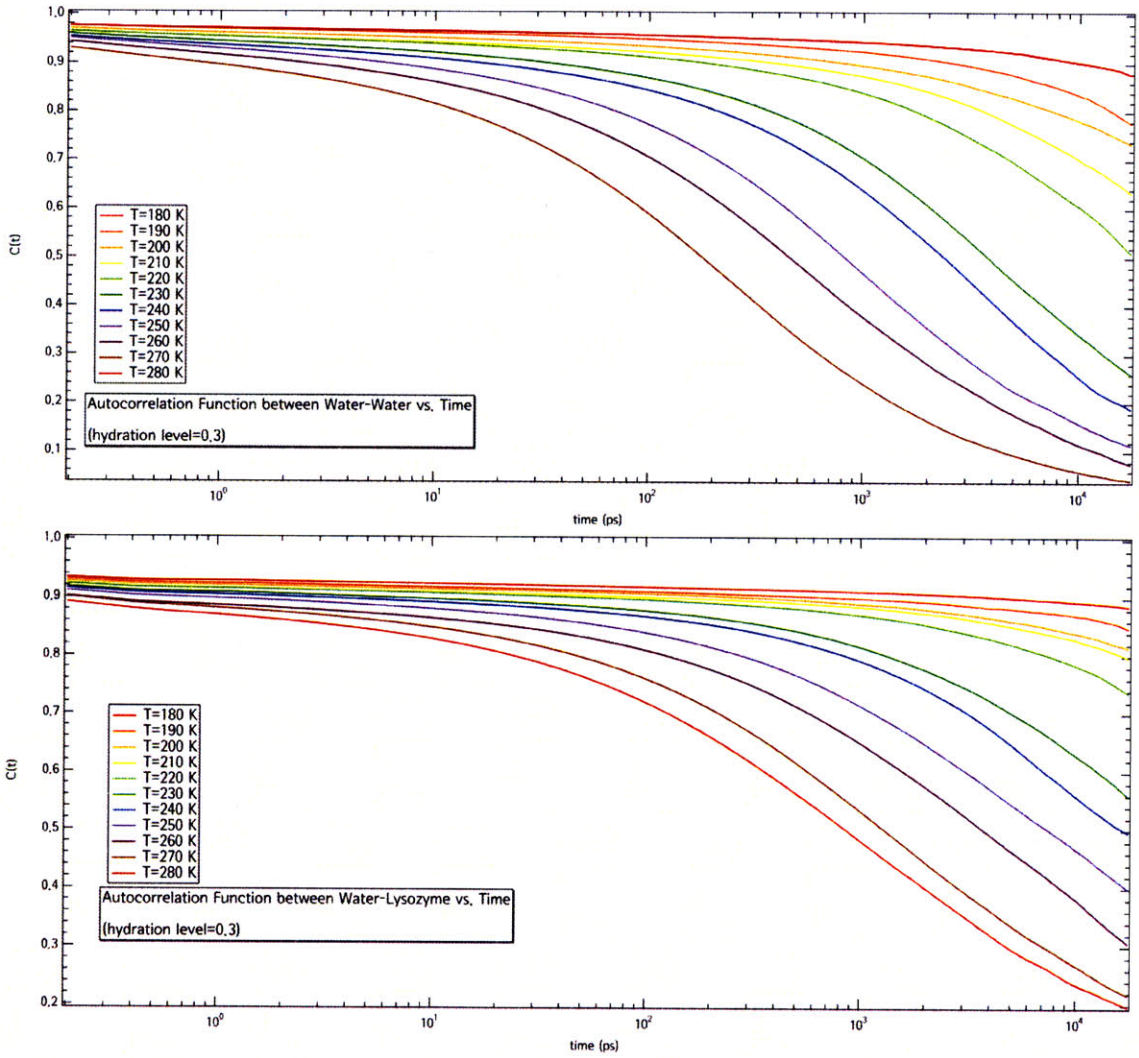


Figure 42. Autocorrelation functions of hydrogen bonds (1) between water and water and (2) between water and Lysozyme. This is the case of hydration level h = 0.3. Since both of them cannot have the relaxation time higher than 230 K, it is truly hard to say about the crossover, which is expected to appear around 220 K and 230 K.

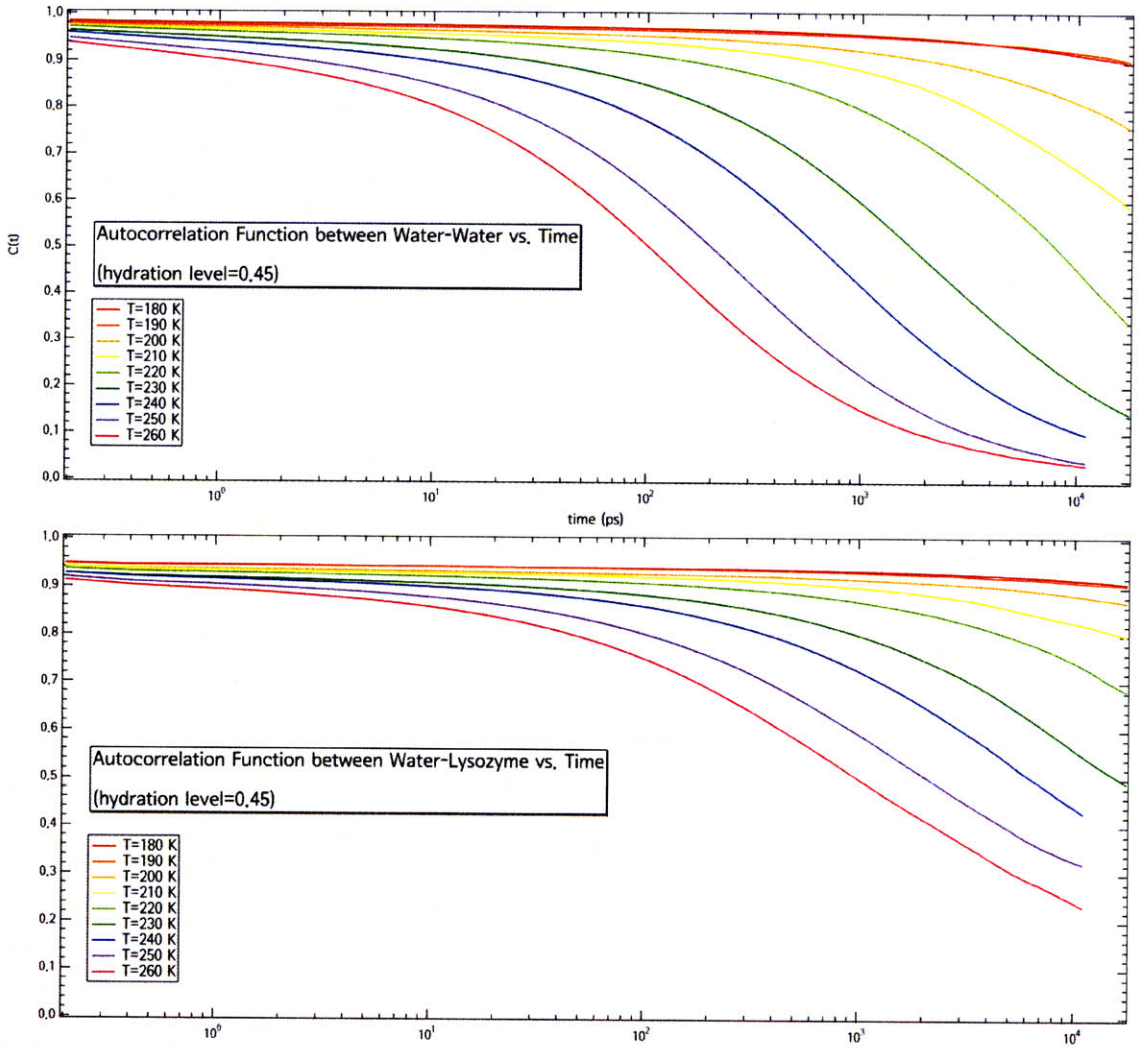


Figure 43. Autocorrelation functions of hydrogen bonds (1) between water and water and (2) between water and Lysozyme. This is the case of hydration level h = 0.45. Since both of them cannot have the relaxation time higher than 230 K, it is truly hard to say about the crossover, which is expected to appear around 220 K and 230 K.

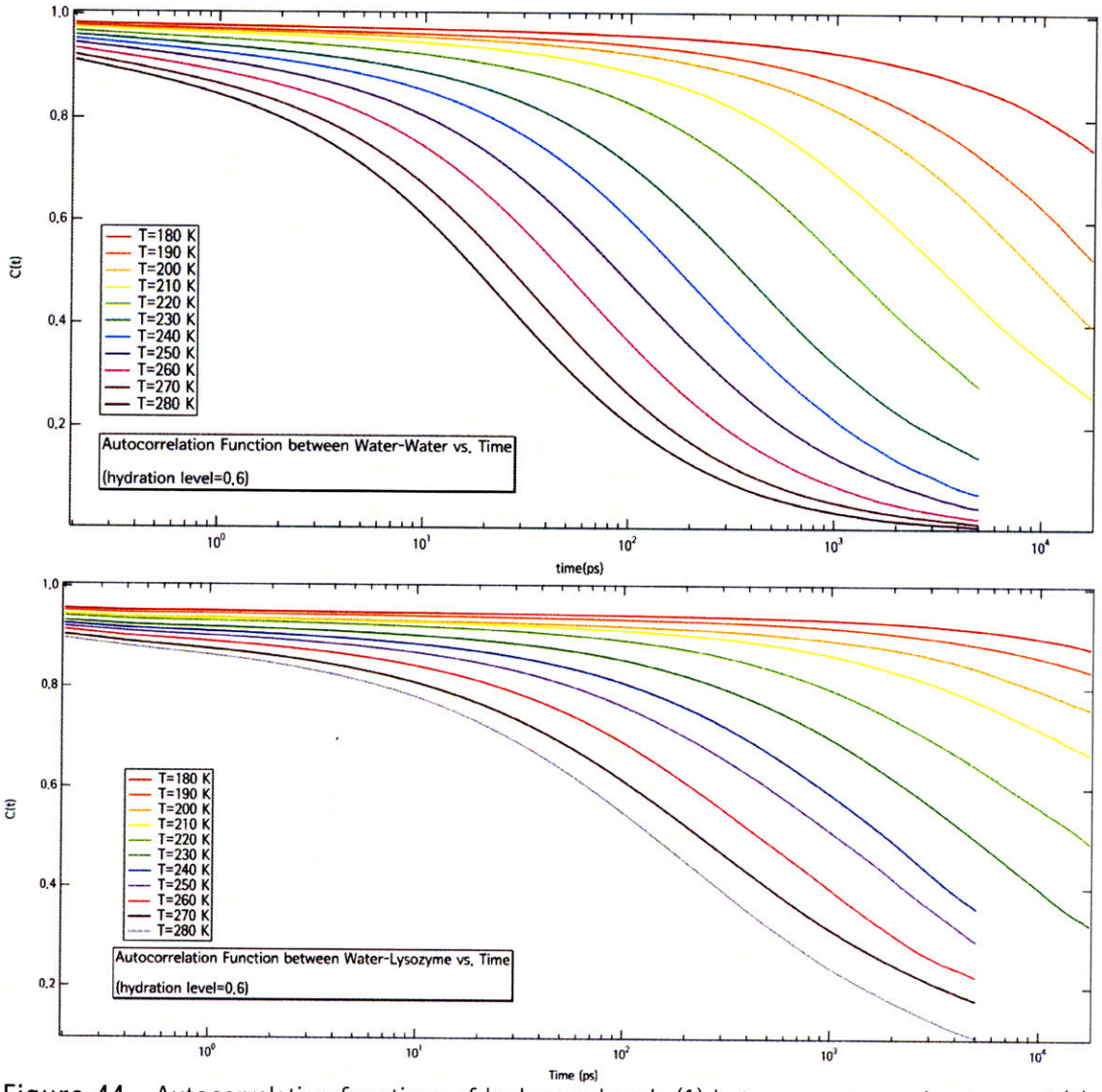


Figure 44. Autocorrelation functions of hydrogen bonds (1) between water and water and (2) between water and Lysozyme. This is the case of hydration level h = 0.6. Since both of them cannot have the relaxation time higher than 230 K, it is truly hard to say about the crossover, which is expected to appear around 220 K and 230 K.

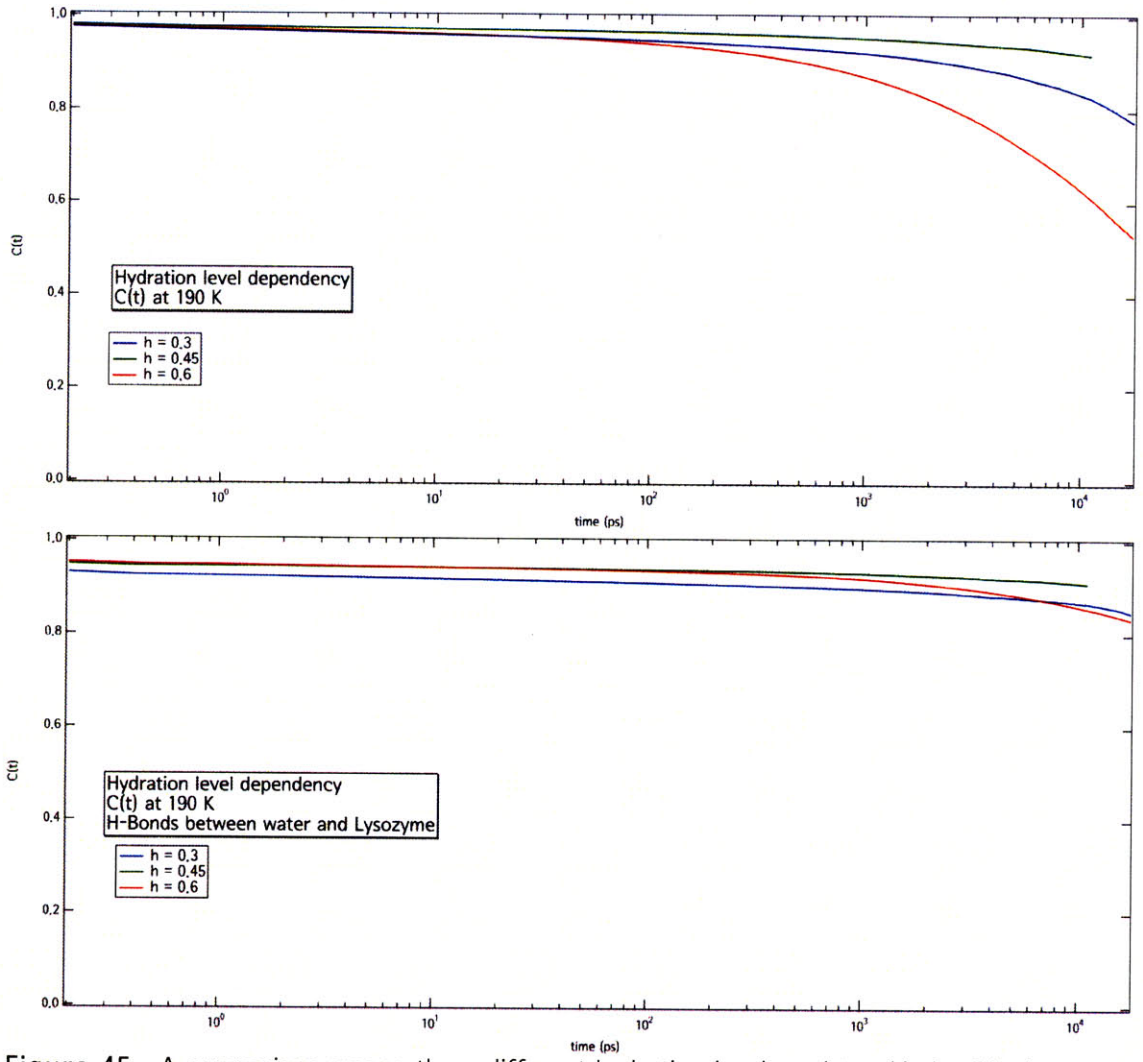


Figure 45. A comparison among three different hydration levels and two kinds of hydrogen bonds (water-water and water-Lysozyme) at the temperature T = 190 K. Upper panel is for the autocorrelation function for hydrogen bonds between water and water. Lower is for the bonds between water and Lysoyme.

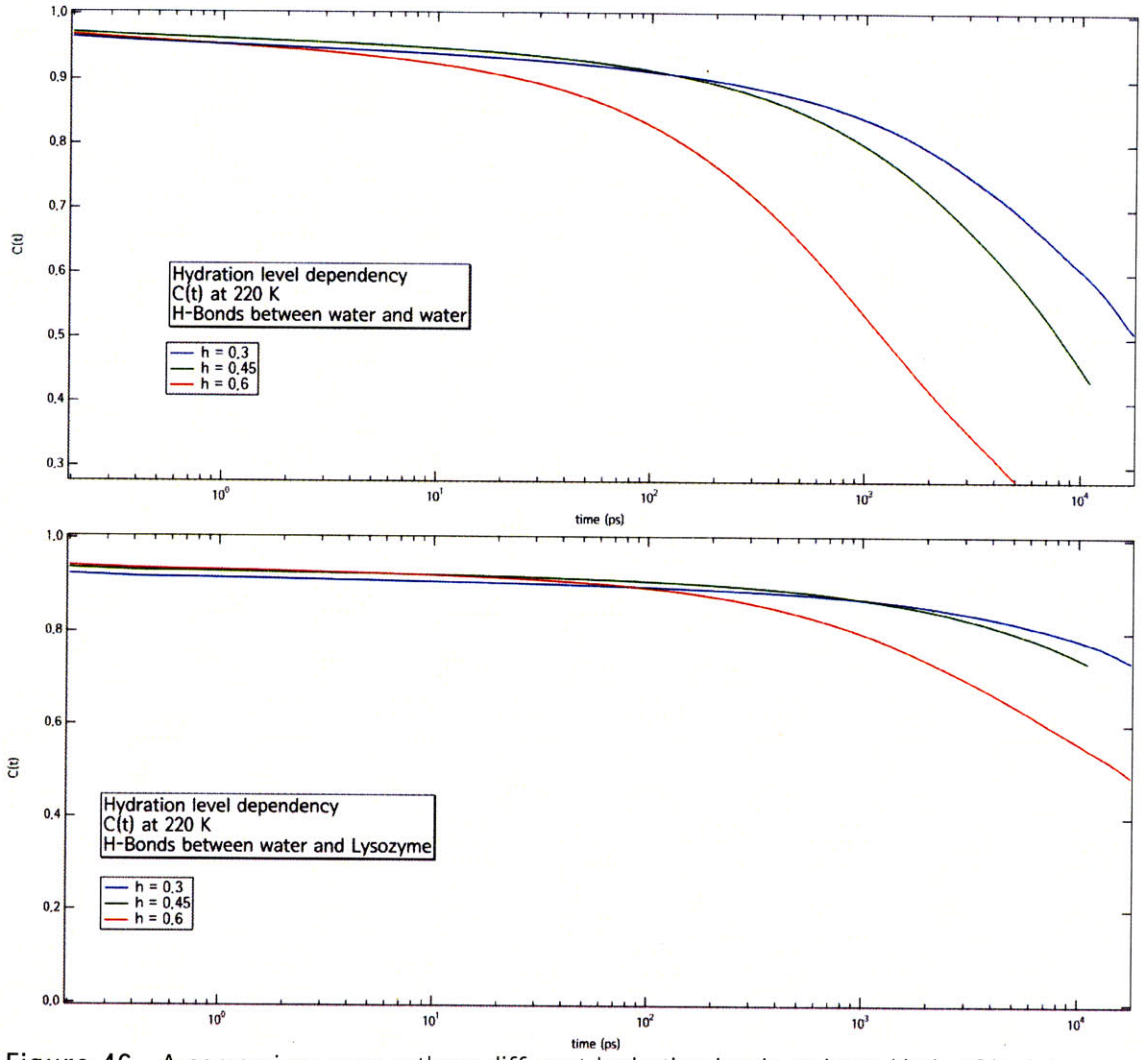


Figure 46. A comparison among three different hydration levels and two kinds of hydrogen bonds (water-water and water-Lysozyme) at the temperature T = 220 K. Upper panel is for the autocorrelation function for hydrogen bonds between water and water. Lower is for the bonds between water and Lysoyme. It is easy to find that the fastest decaying one is the case of the smallest hydration level (h = 0.3)

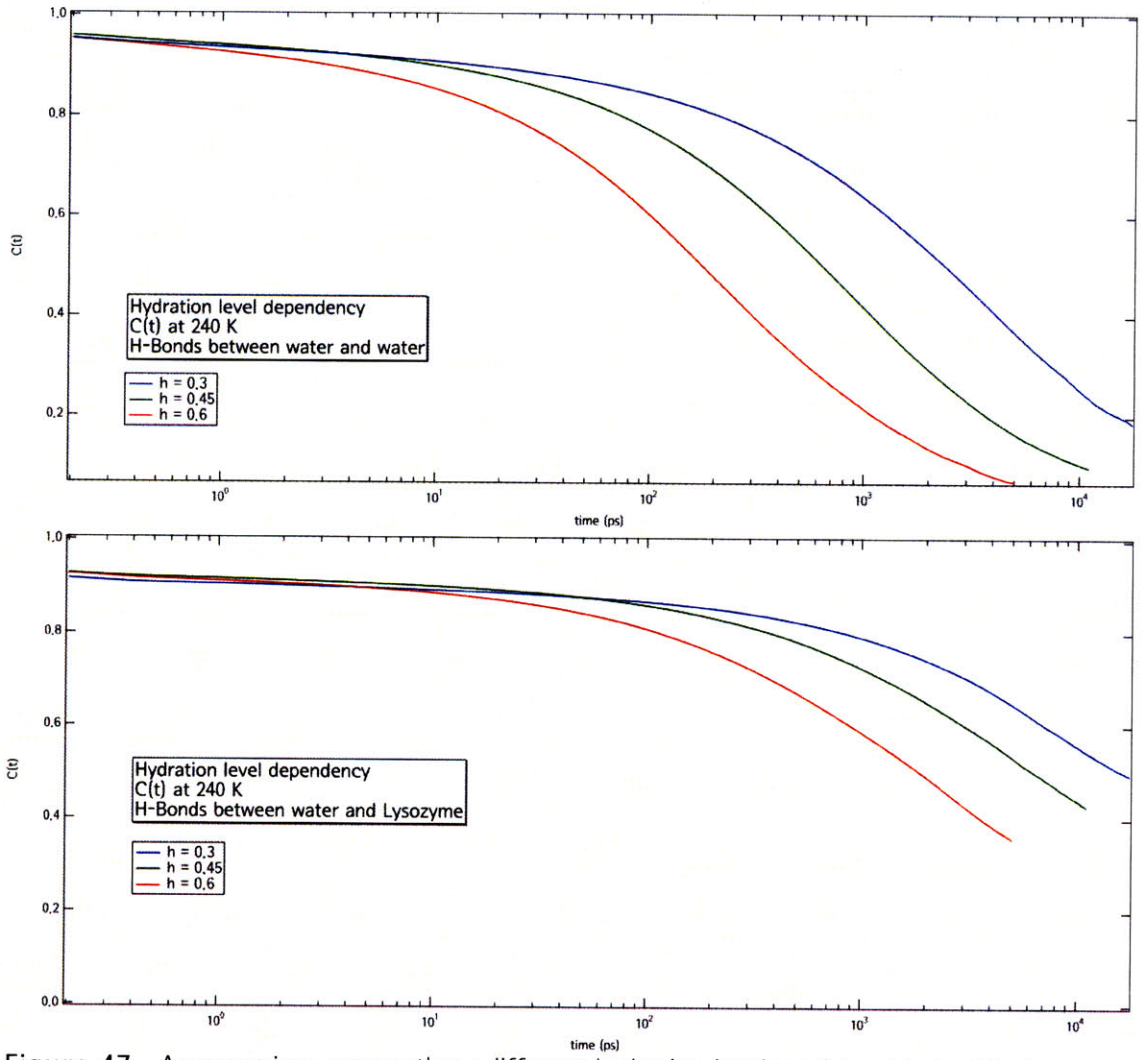


Figure 47. A comparison among three different hydration levels and two kinds of hydrogen bonds (water-water and water-Lysozyme) at the temperature T = 240 K. Upper panel is for the autocorrelation function for hydrogen bonds between water and water. Lower is for the bonds between water and Lysoyme. We can see the smallest hydration level one (h = 0.3) has the largest values.

Chapter 5. CONCLUSION AND FUTURE WORK

In the previous Chapter, MD simulation results show many quantities that possibly allow us the crossover phenomenon. All of those factors actually allow us to confirm the existence of the crossover phenomenon, even though all of them cannot show clearly with the exact crossover temperatures. Moreover, the comparisons among three hydration level results provide that the crossover temperature is dependent on the hydration level.

The average translational relaxation time computed from ISF, the number of hydrogen bonds and the MSD result show clearly the existence of the crossover phenomenon. Other factors such as the structure factor, radial distribution function and the autocorrelation function support its conclusion with showing hydration dependencies.

In the average translational relaxation results, experiments and MD show a slight difference in $\langle \tau_T \rangle$ value. That difference is in the error range, so the FSC temperature for both case are prediceted as 225 ± 10 K ($T_L(experiment)$ = 226 K and $T_L(simulations)$ = 222 K). By the MD simulation results, this temperature is depending on the hydration level of the protein (Lysozyme) – water system. The crossover temperature increases as lowering the hydration level of the hydration Lysozyme water sample.

In particular, we show the striking agreements for a rough crossover temperature, where the inclines are changing in the MSD of hydrogen atoms in water. For all three MD simulation configurations, it agrees with each other. From MSD, we can grasp that the hydration level does not have an effect on the existence of the crossover phenomena.

From the comparison result of the hydrogen bonds between water and water case, the crossover is shown. However one cannot see any crossover or changing points in the hydrogen bonds between water and Lysozyme. This fact allow us to conclude that hydrogen bonds between water and water are more important and can possibly trigger biopolymer's transitions.

From the autocorrelation functions of the hydrogen bonds, we could point out that there is a higher possibility that there is a crossover or transition in between 220 K and 230 K (from the temperature dependent graph) and it could depends on hydration level changes even though its existence is confirmed (from the hydration level comparison).

The first main observation of this research: the crossover temperature depends on the hydration level, but the crossover phenomenon occurs at any hydration level.

The second is provided by the analysis of number of two cases bonds: (1) hydrogen bonds between water and water and (2) those between water and Lysozyme. The second one that I have found is that water possibly triggers the biomolecules' functionality.

As the third one, the usefulness that I want to stress it that I have used one of the MD packages, GROMACS to investigate mainly behaviors of water molecules than those of biopolymers, which has been originally the simulation target of GROMACS package.

As the future work, I want to continue to the research in this area by

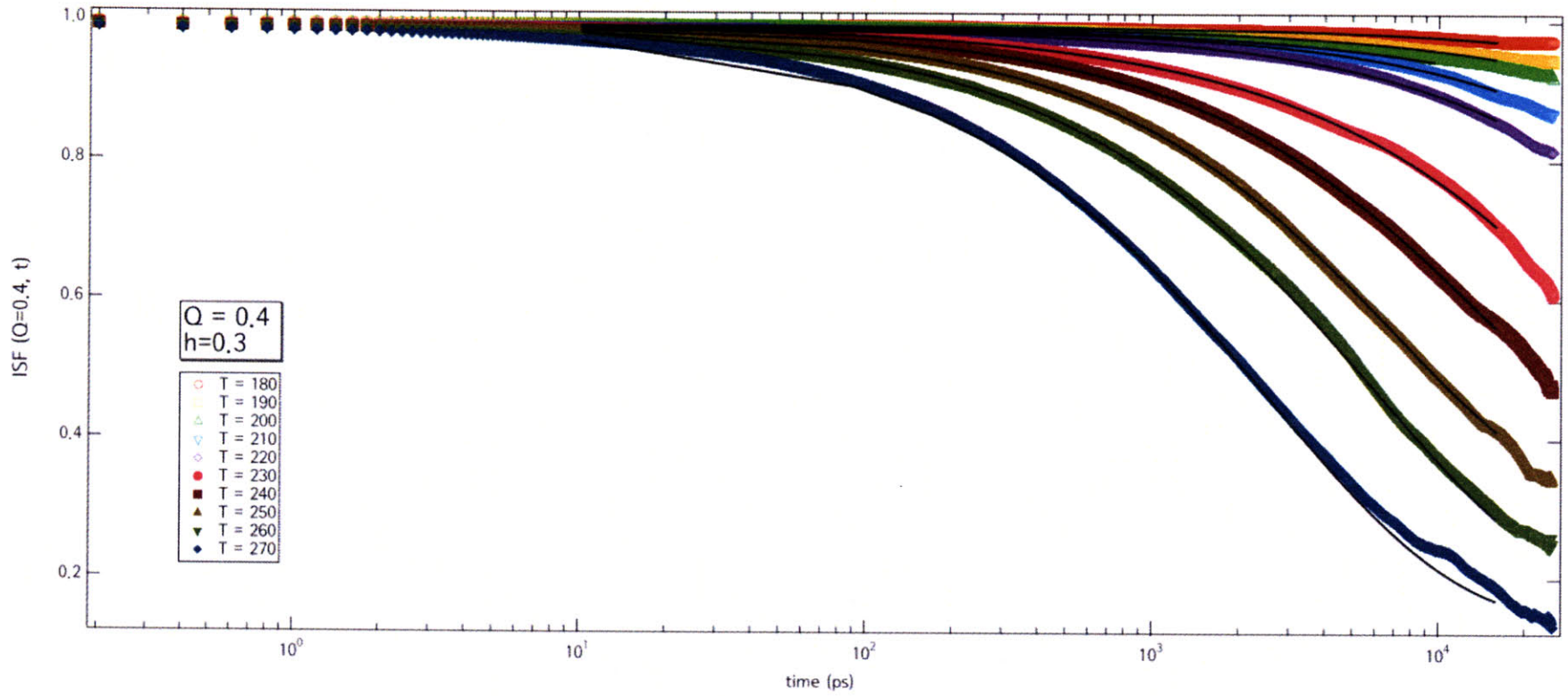
(1) having more hydration levels of the MD simulation configuration,

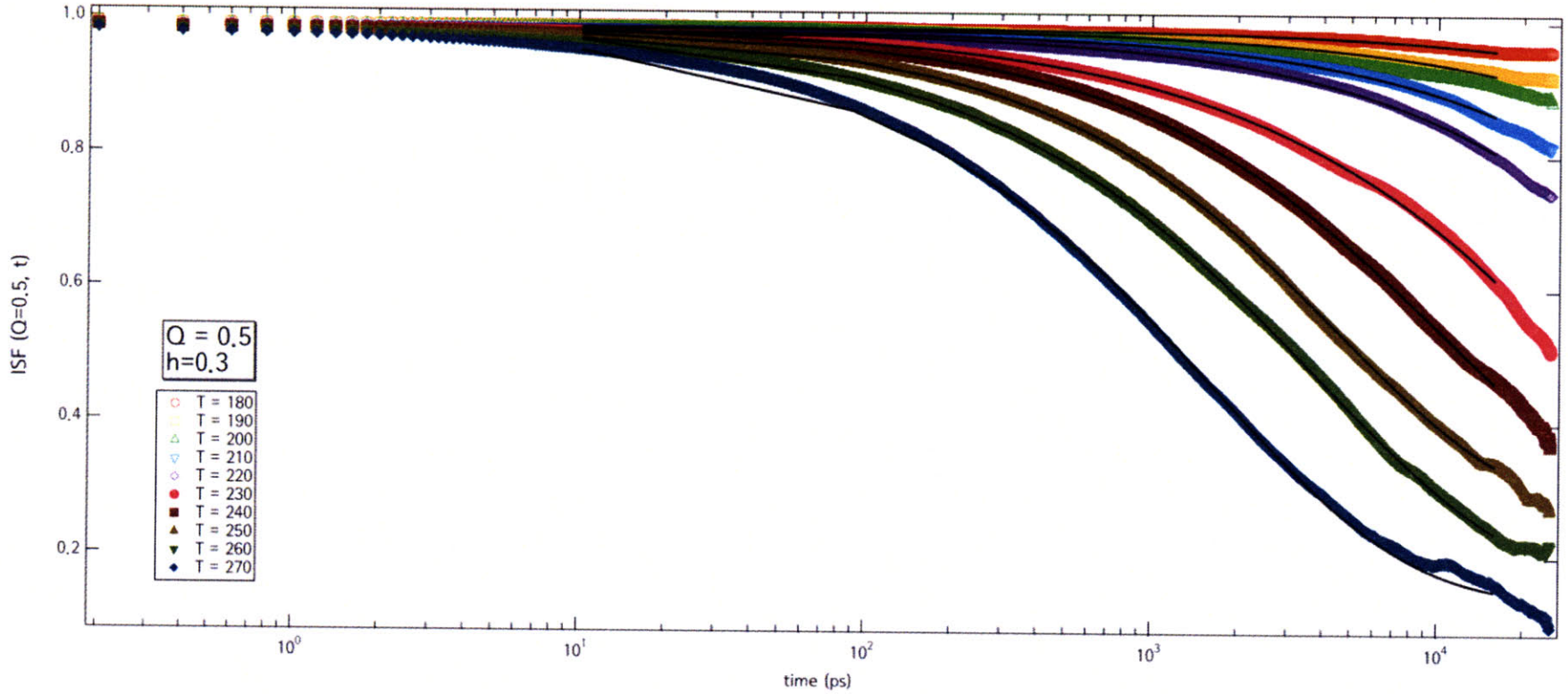
(2) comparing with experimental hydration water sample with higher hydration level such as h = 0.45 and 0.6.

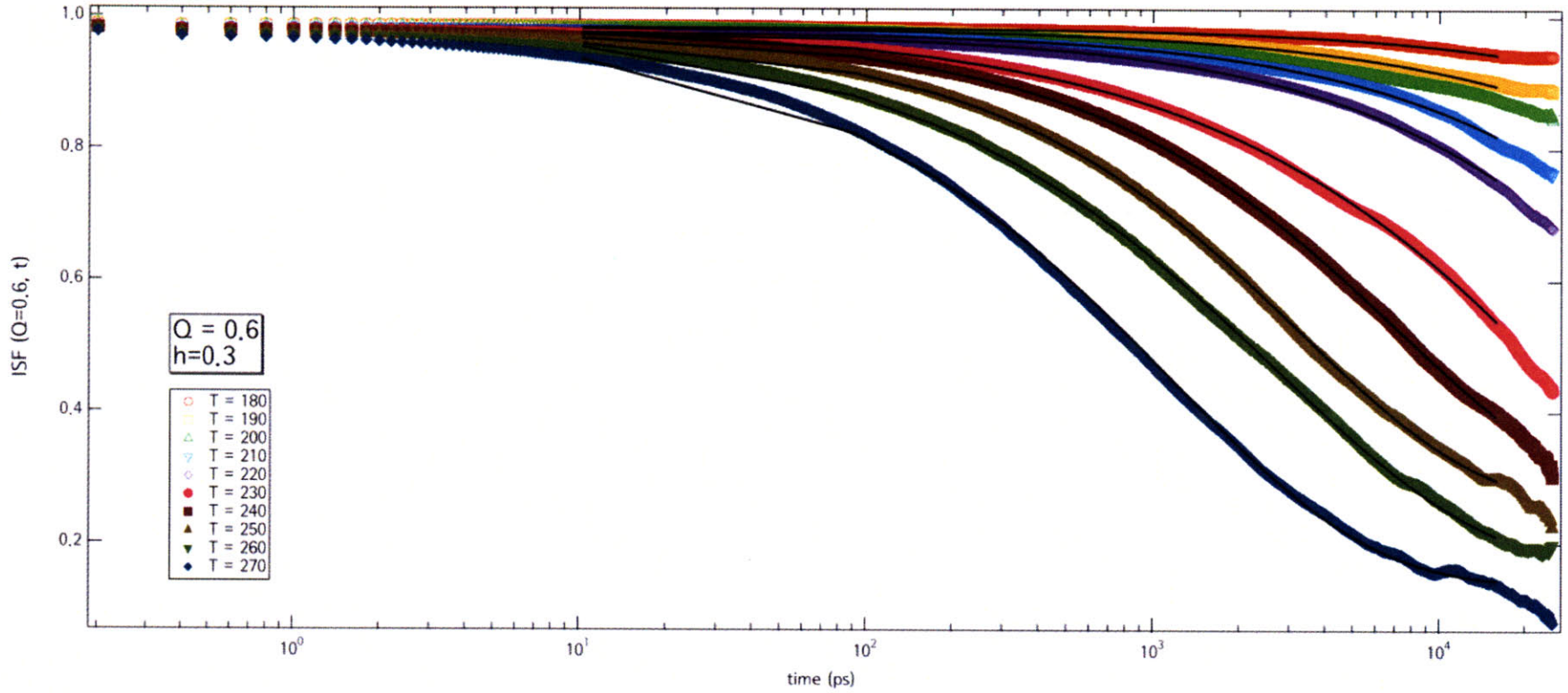
(3) getting longer time results of the autocorrelation function for lower temperature cases to extract the relaxation times.

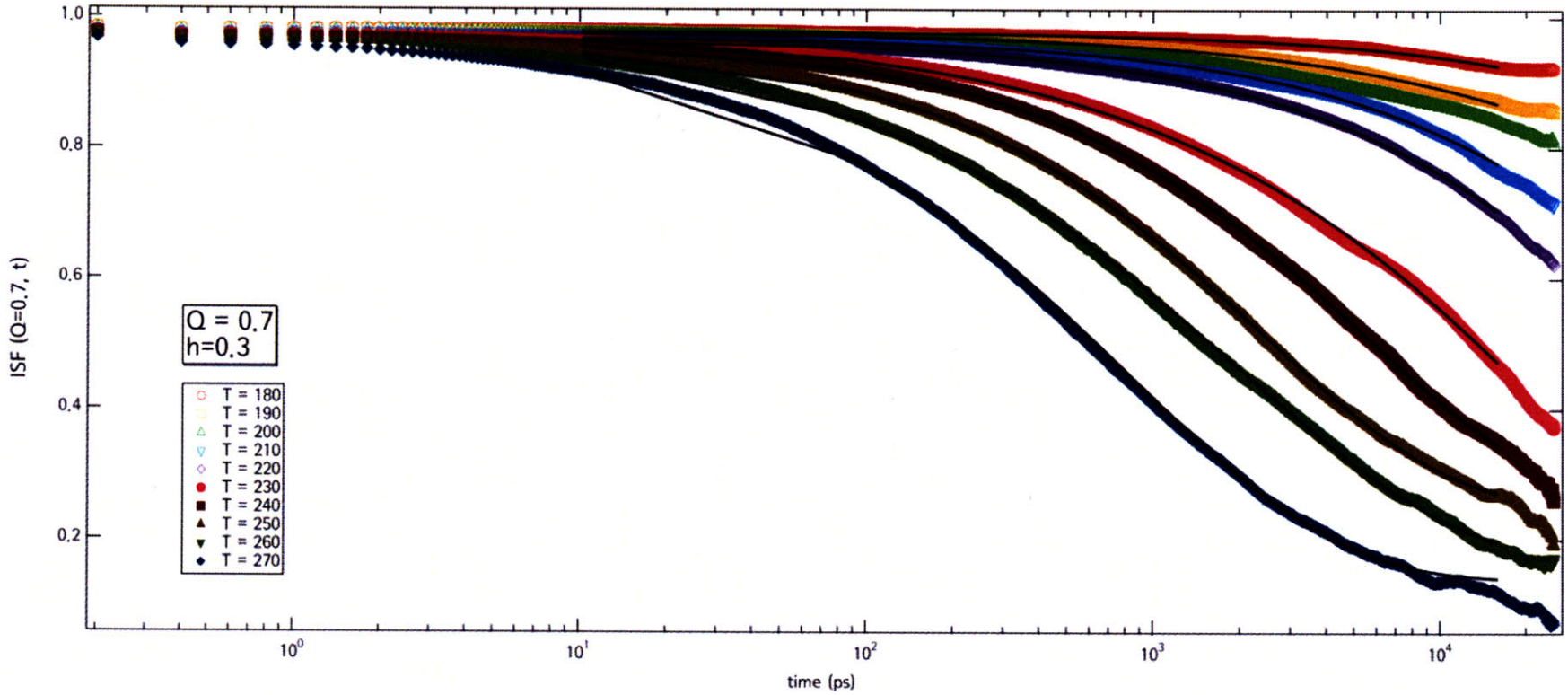
Chapter 6. APPENDIX

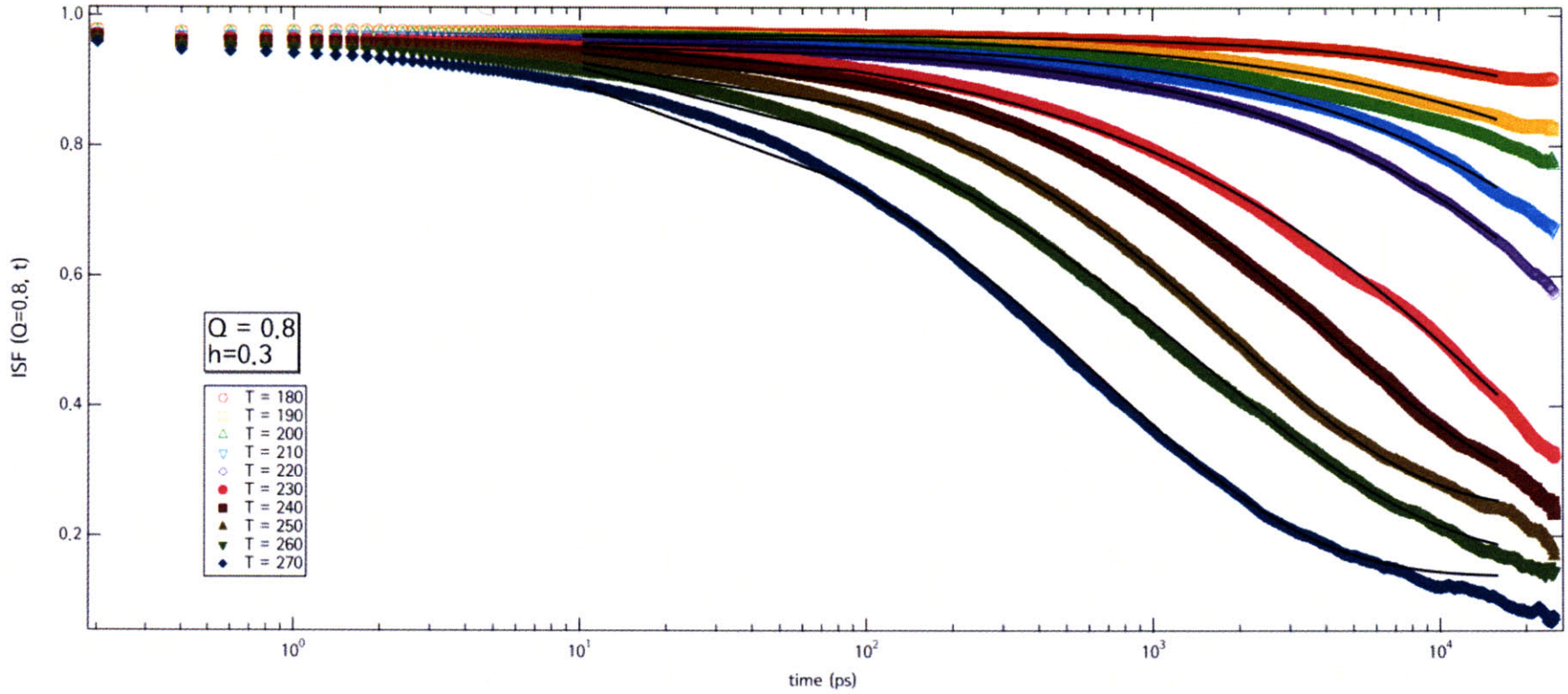
6.1. All the results of ISF in MD simulation

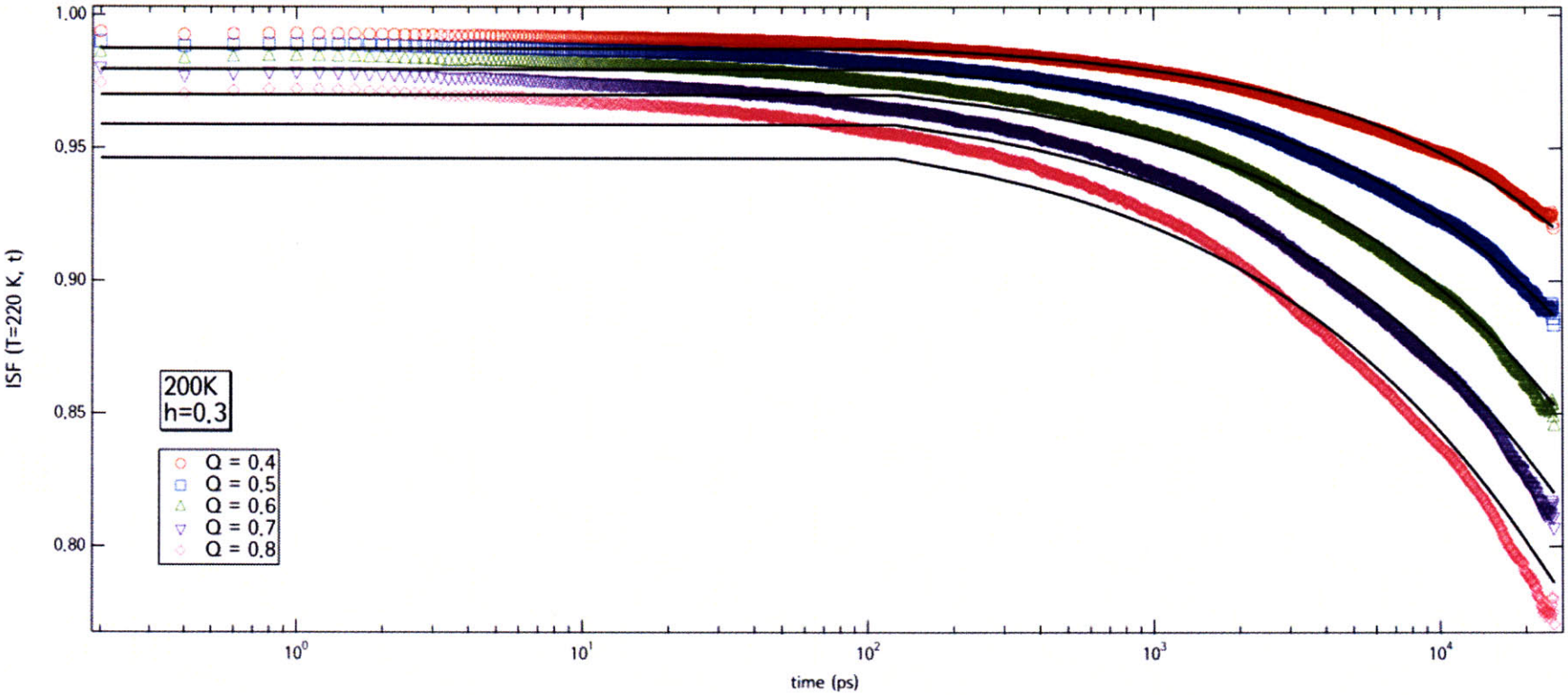

ISF (Q=0.4, t)
time (ps)
Q = 0.4
h=0.3
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250
T = 260
T = 270

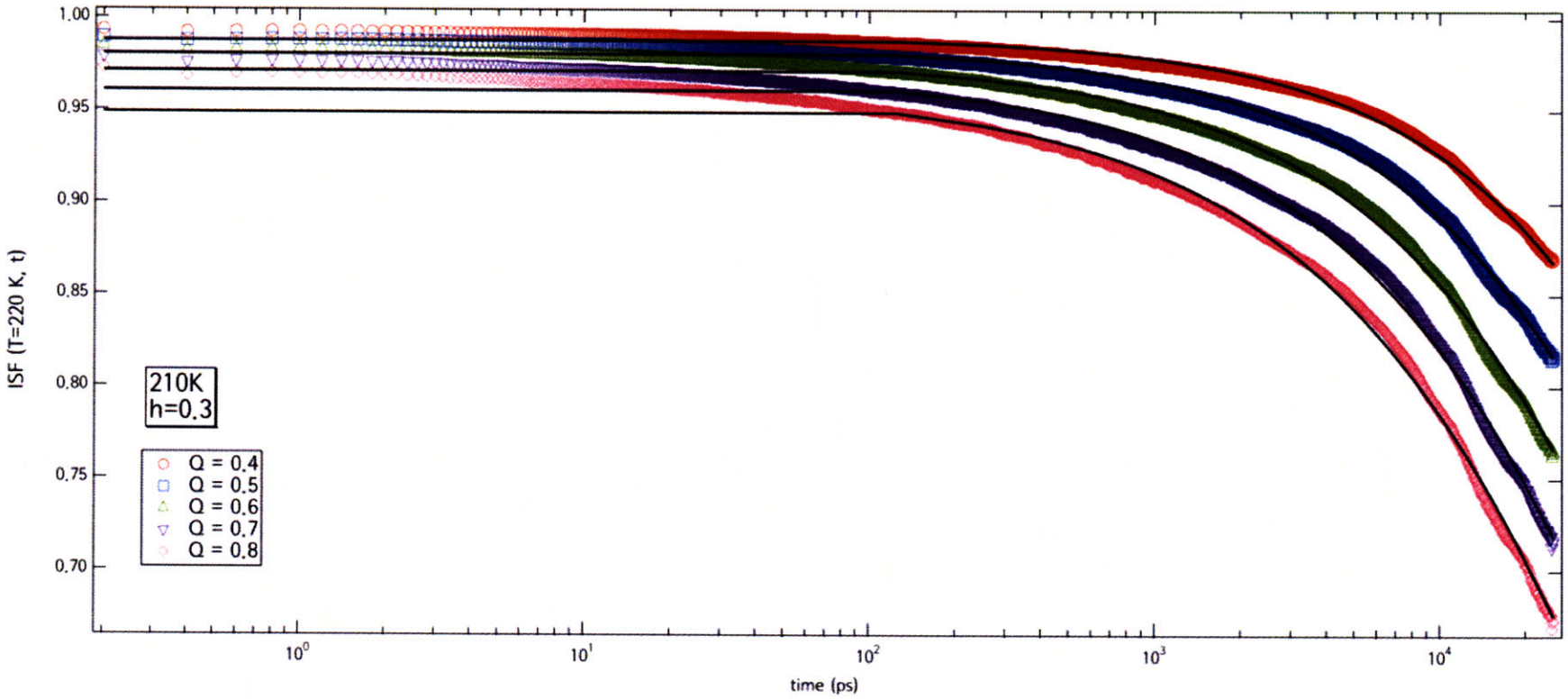

ISF (Q=0.5, t)
time (ps)
Q = 0.5
h=0.3
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250
T = 260
T = 270

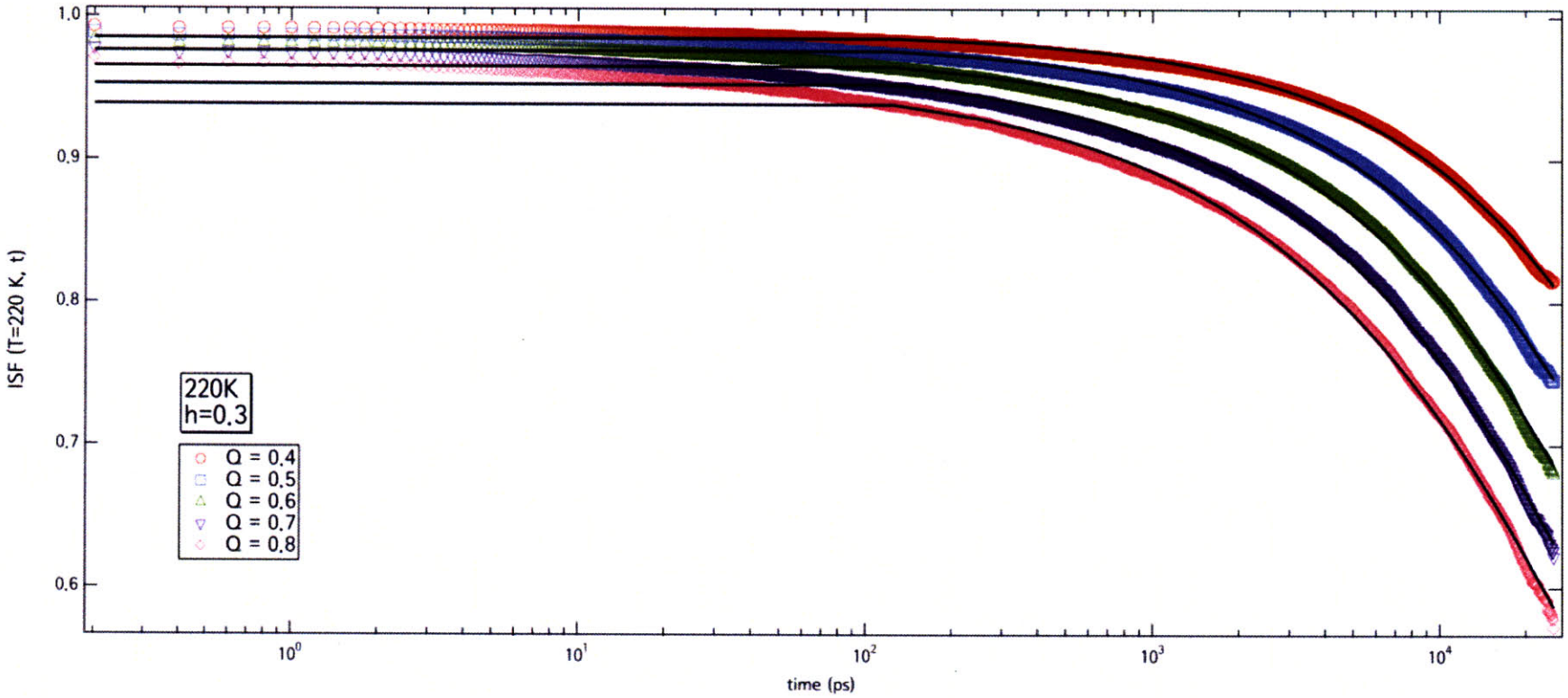

ISF (Q=0.6, t)
time (ps)
Q = 0.6
h=0.3
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250
T = 260
T = 270

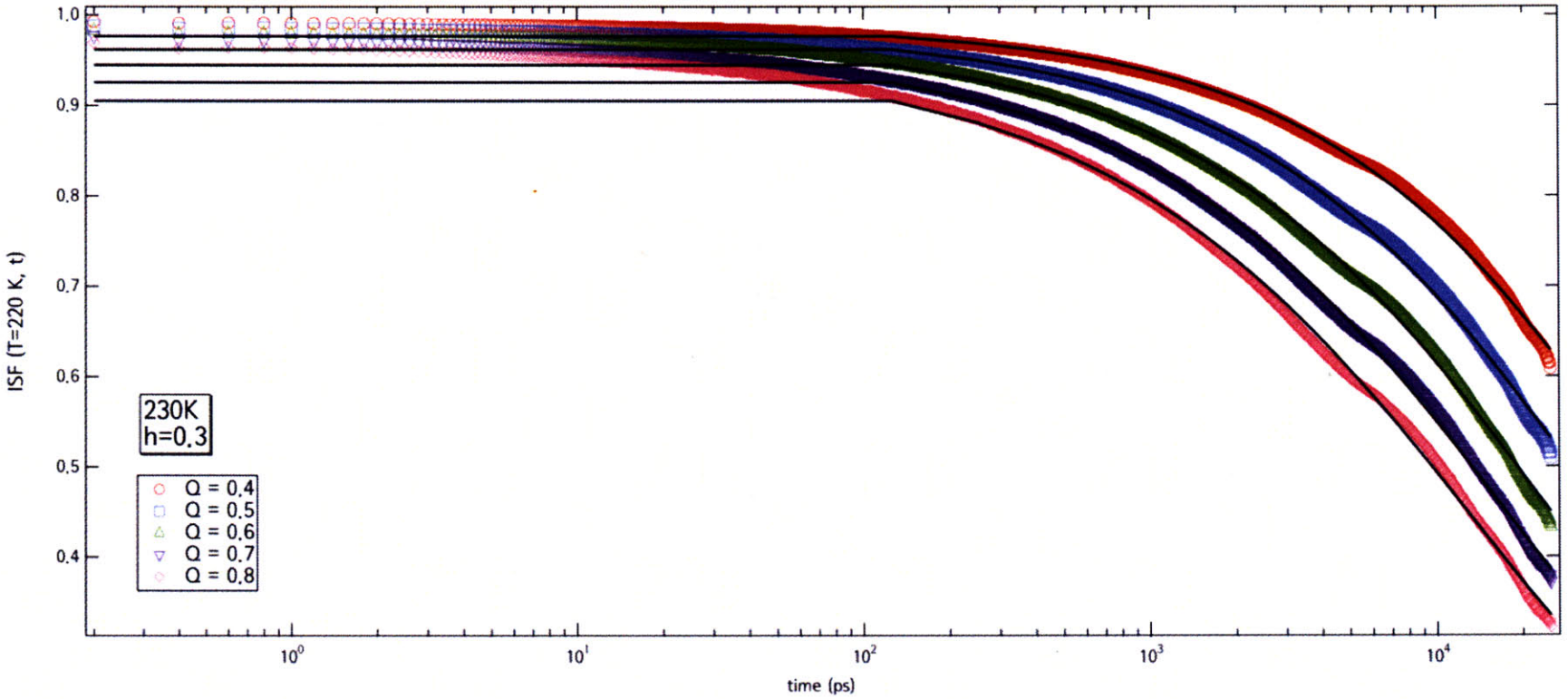

ISF (Q=0.7, t)
time (ps)
Q = 0.7
h=0.3
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250
T = 260
T = 270

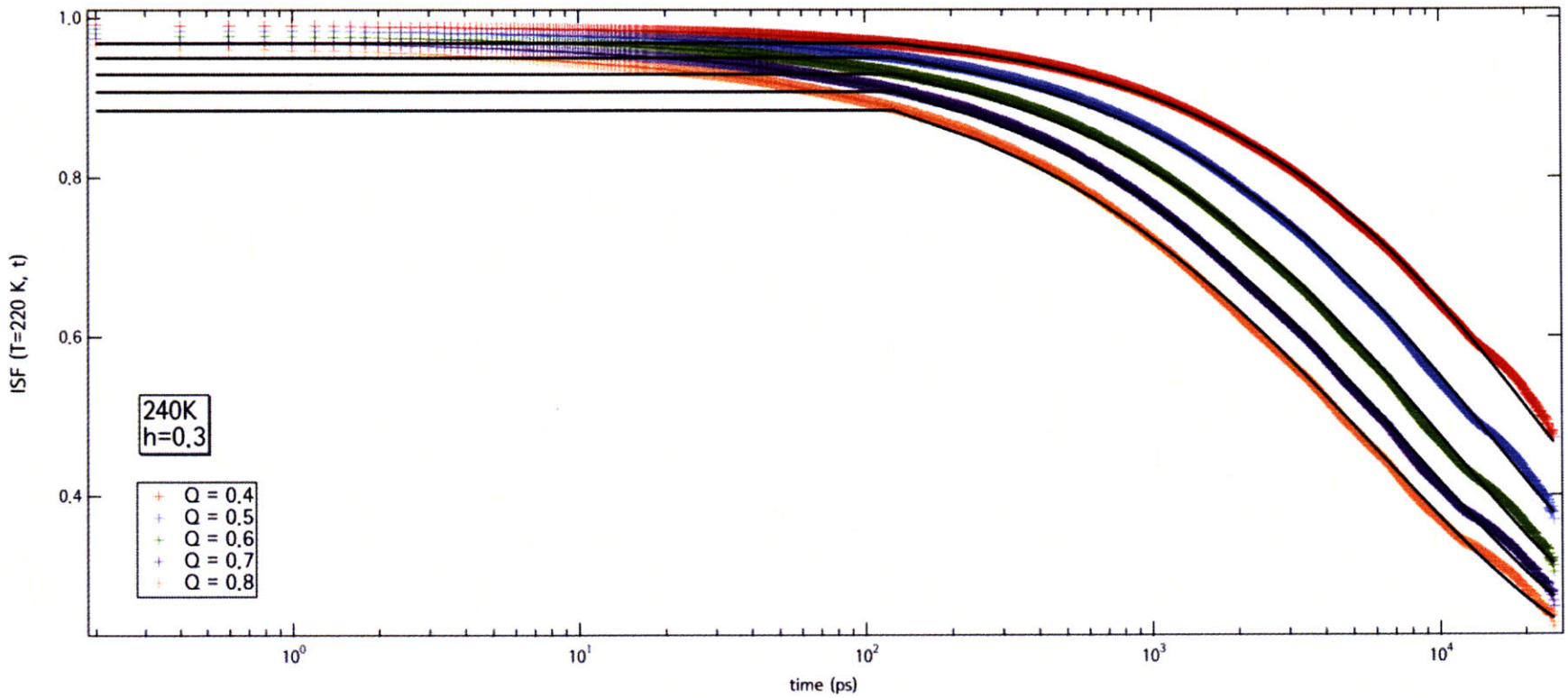

ISF (Q=0.8, t)
time (ps)
Q = 0.8
h=0.3
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250
T = 260
T = 270

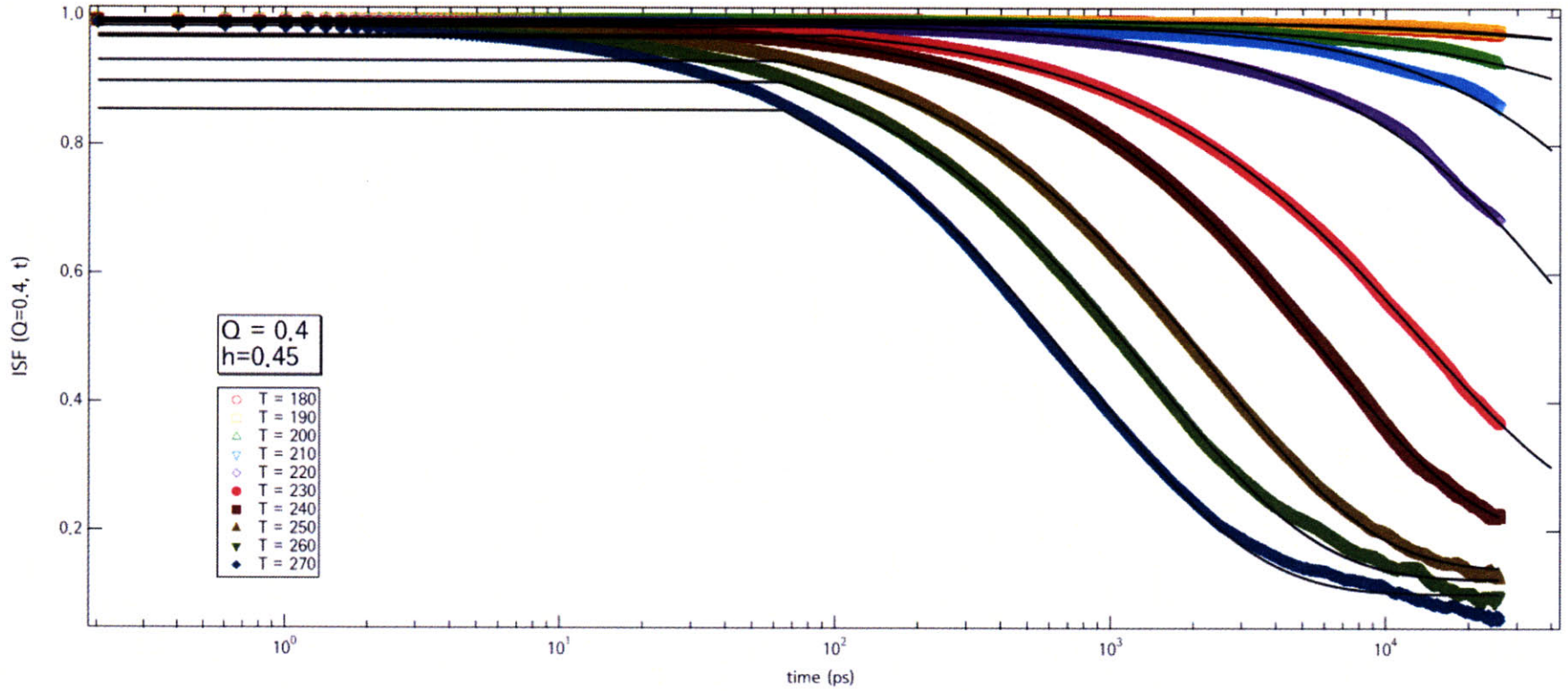

1.00
0.95
0.90
0.85
0.80
ISF (T=220 K, t)
200K
h=0.3
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8
10^0
10^1
10^2
10^3
10^4
time (ps)

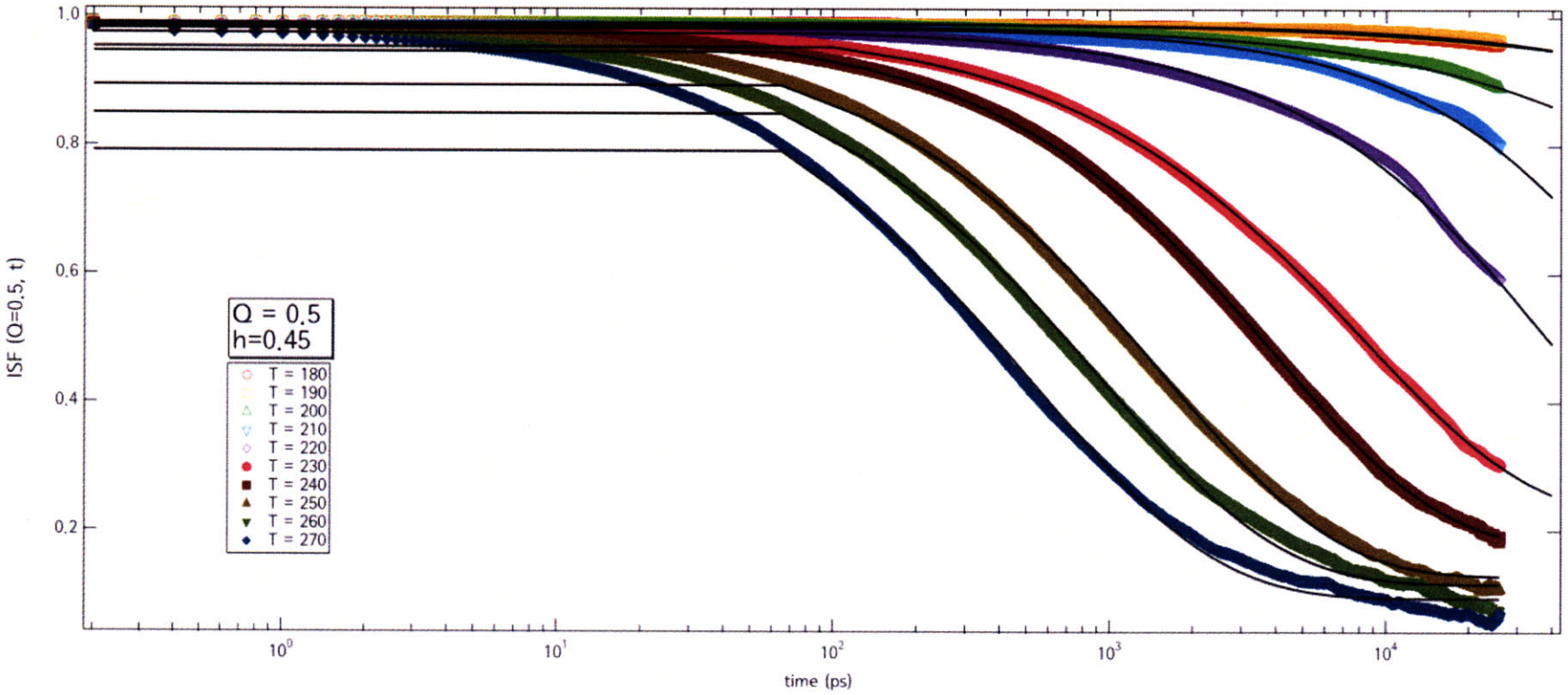

1.00
0.95
0.90
0.85
0.80
0.75
0.70
ISF (T=220 K, t)
210K
h=0.3
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8
10^0
10^1
10^2
10^3
10^4
time (ps)

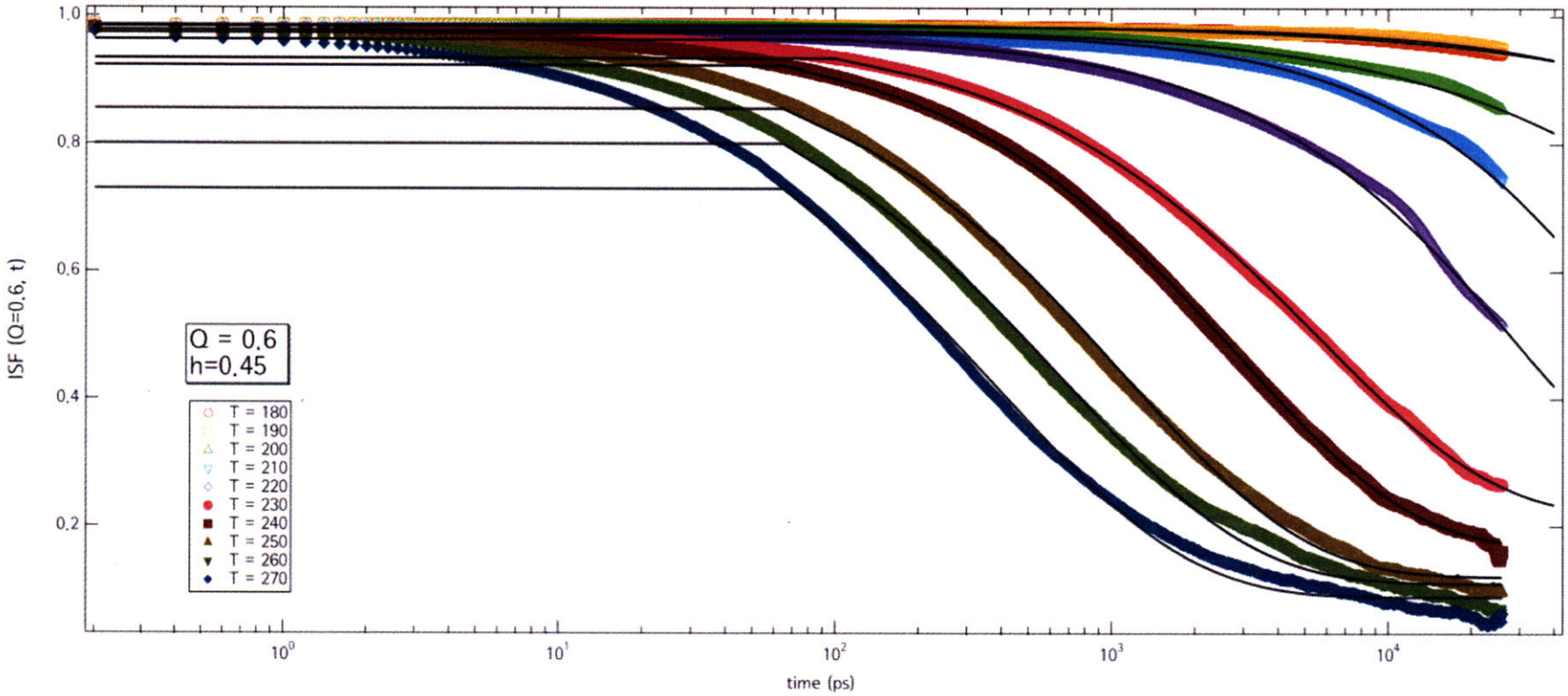

ISF (T=220 K, t)
time (ps)
220K
h=0.3
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8

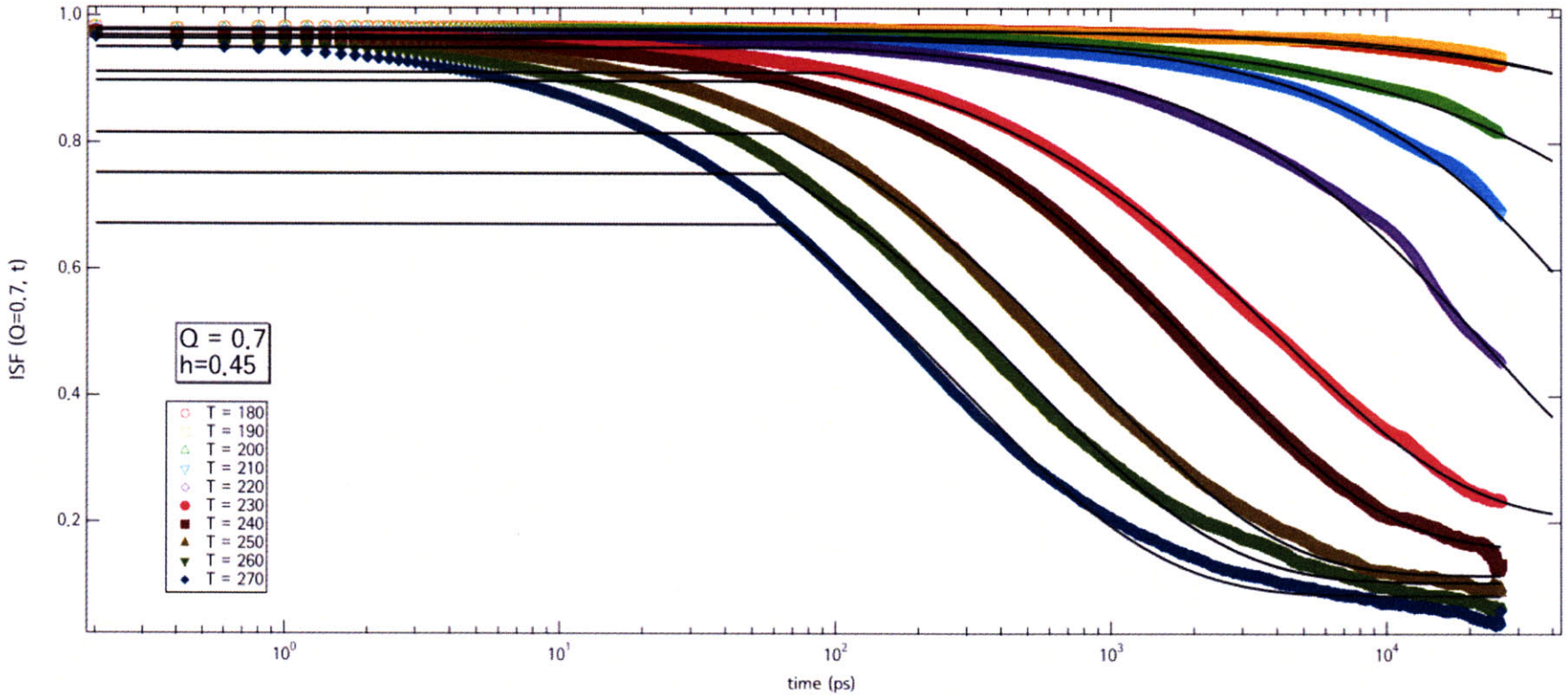

ISF (T=220 K, t)
time (ps)
230K
h=0.3
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8

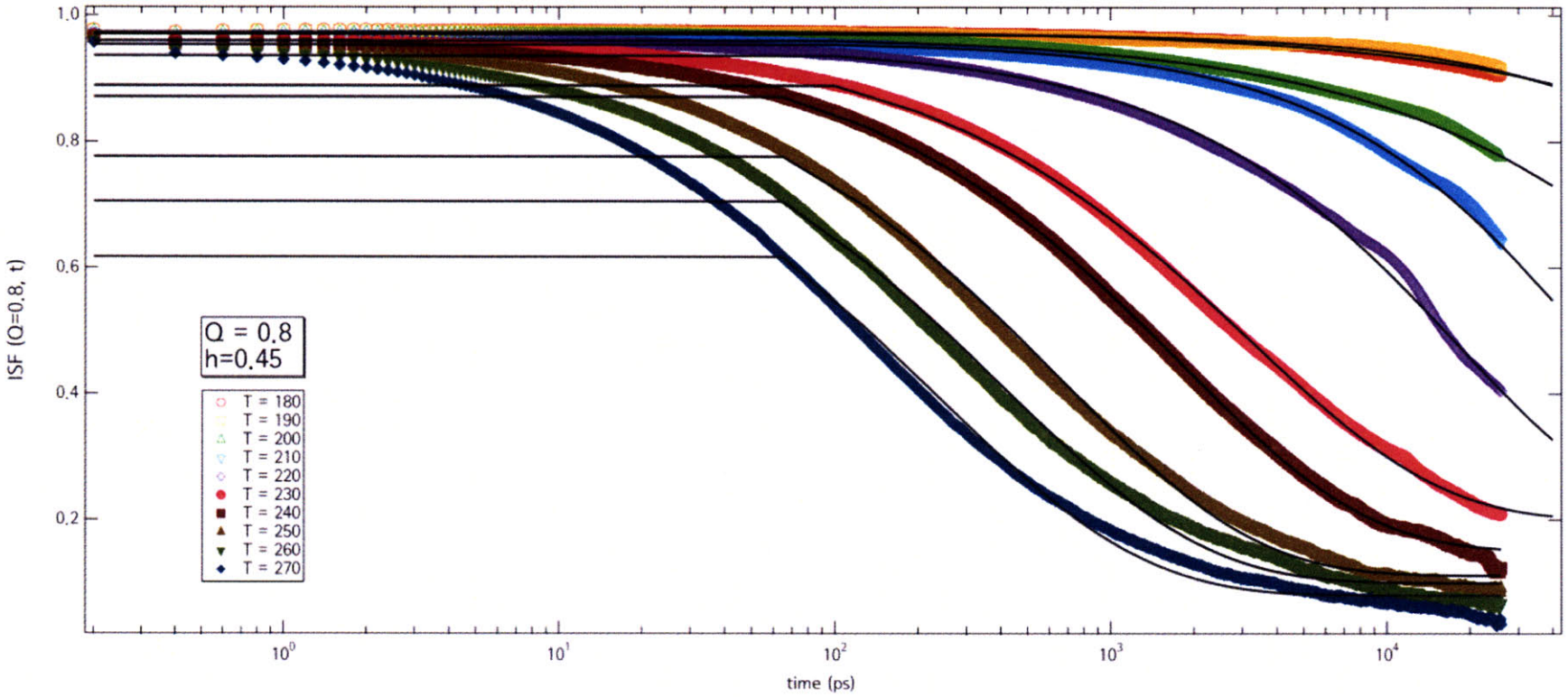

ISF (T=220 K, t)
time (ps)
240K
h=0.3
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8

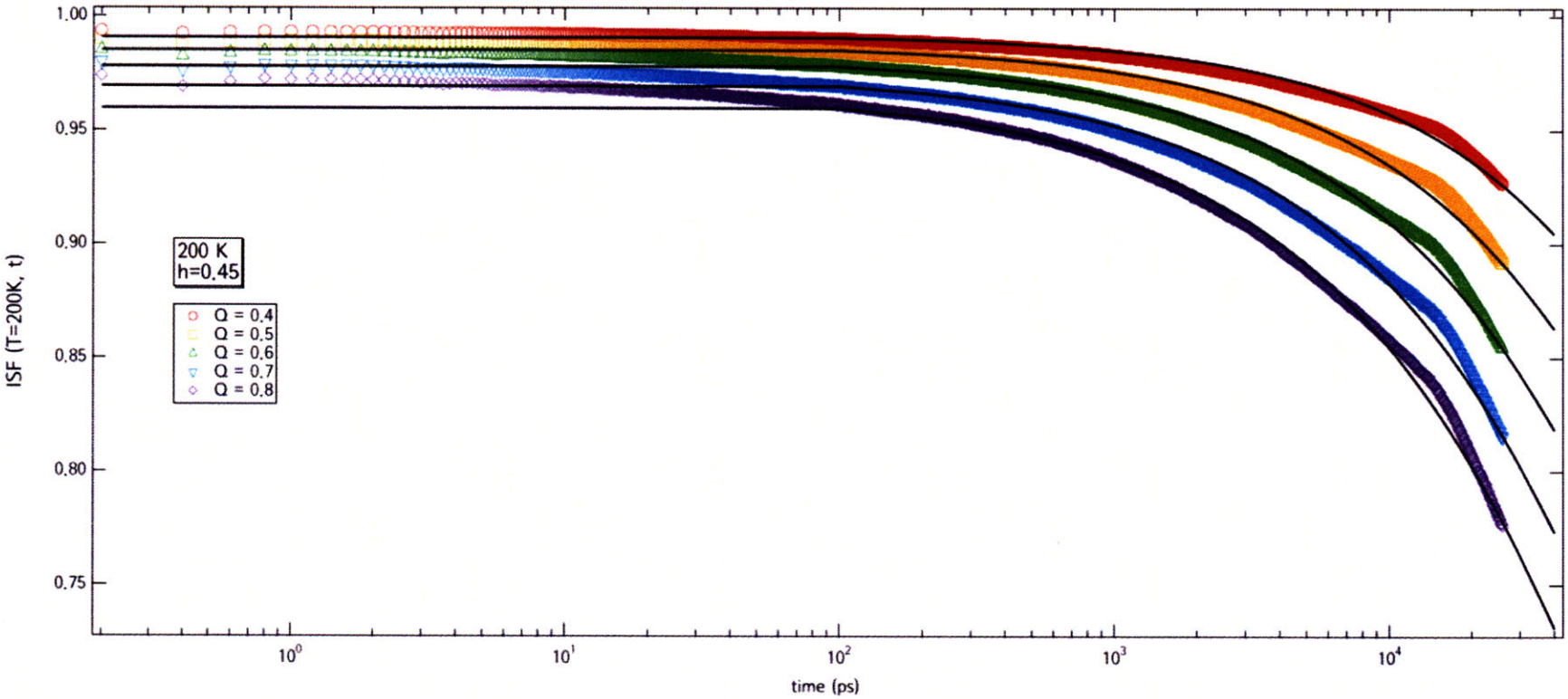

ISF (Q=0.4, t)
time (ps)
Q = 0.4
h=0.45
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250
T = 260
T = 270

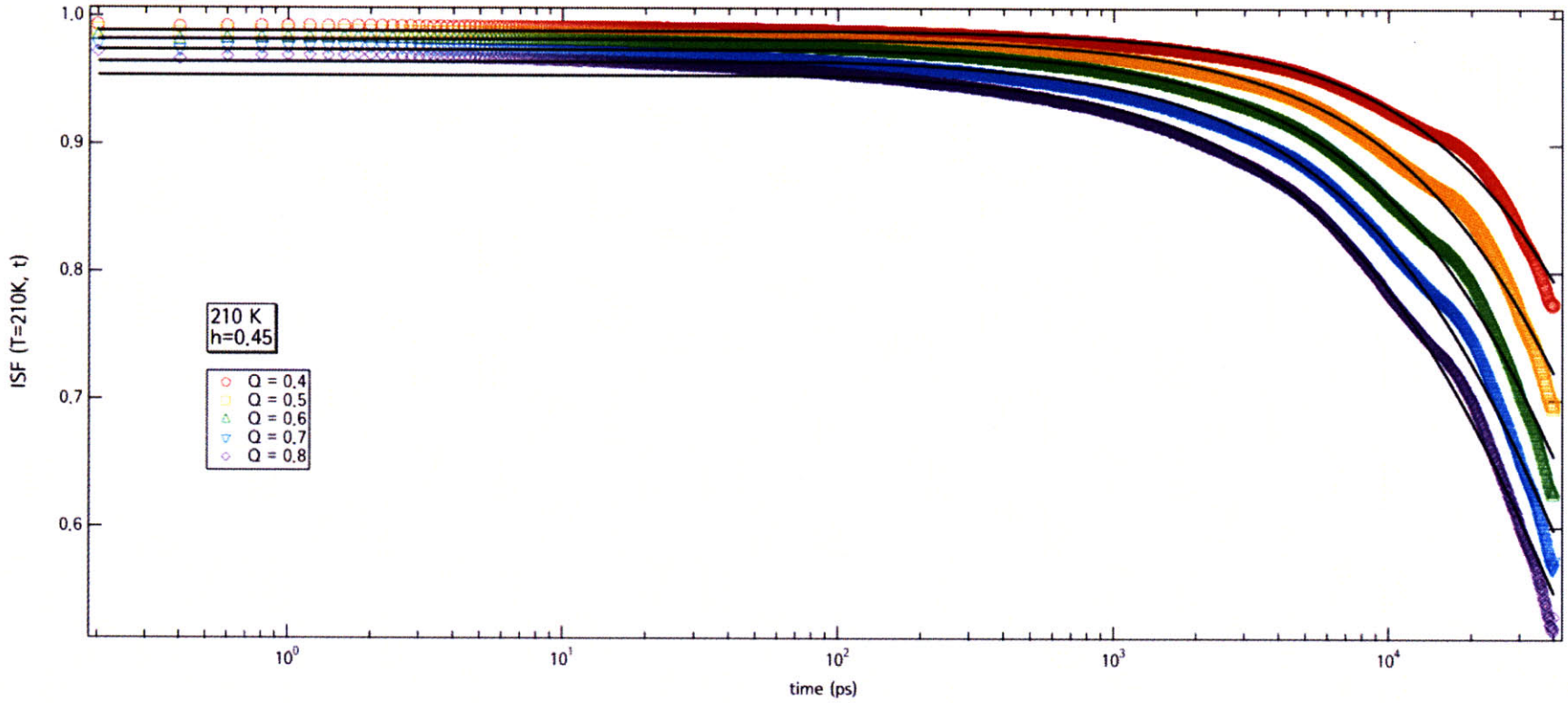

ISF (Q=0.5, t)
time (ps)
Q = 0.5
h=0.45
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250
T = 260
T = 270

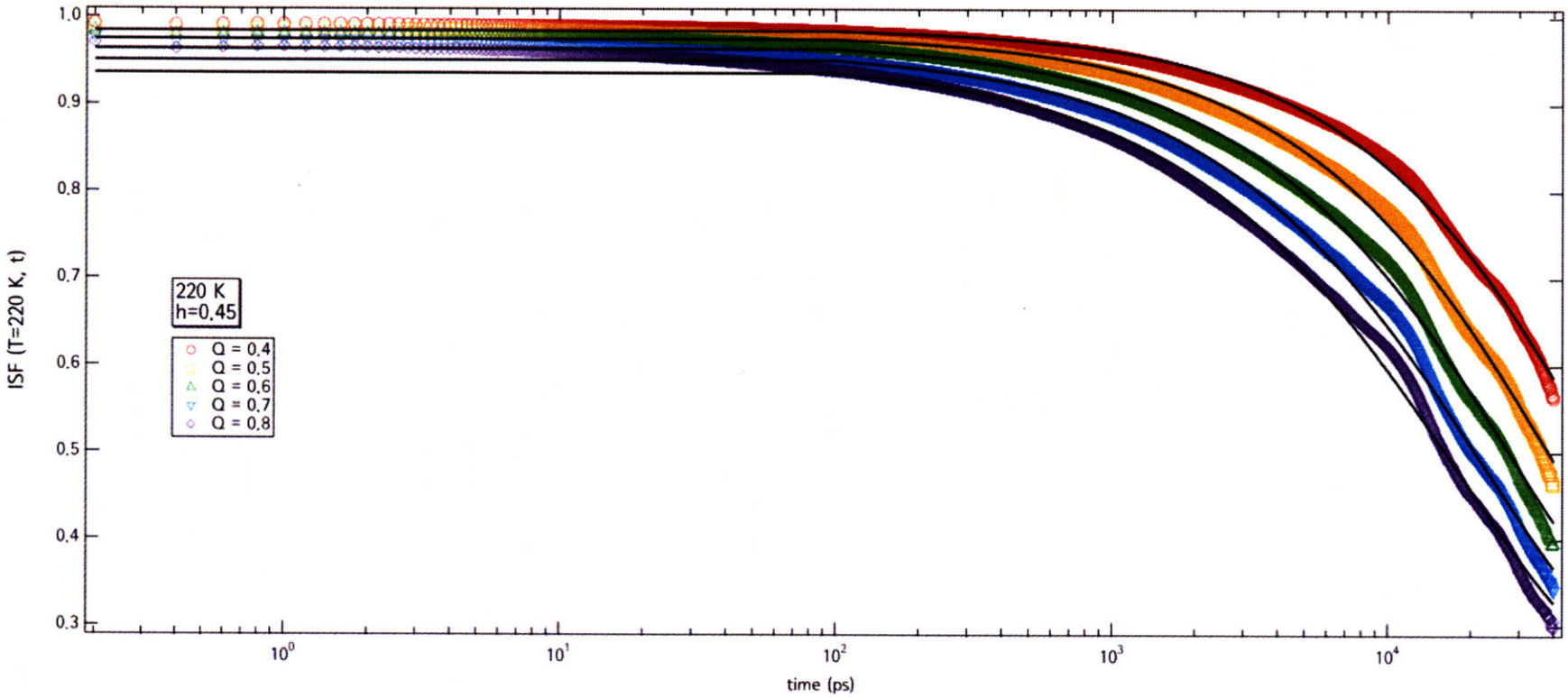

ISF (Q=0.6, t)
time (ps)
Q = 0.6
h=0.45
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250
T = 260
T = 270

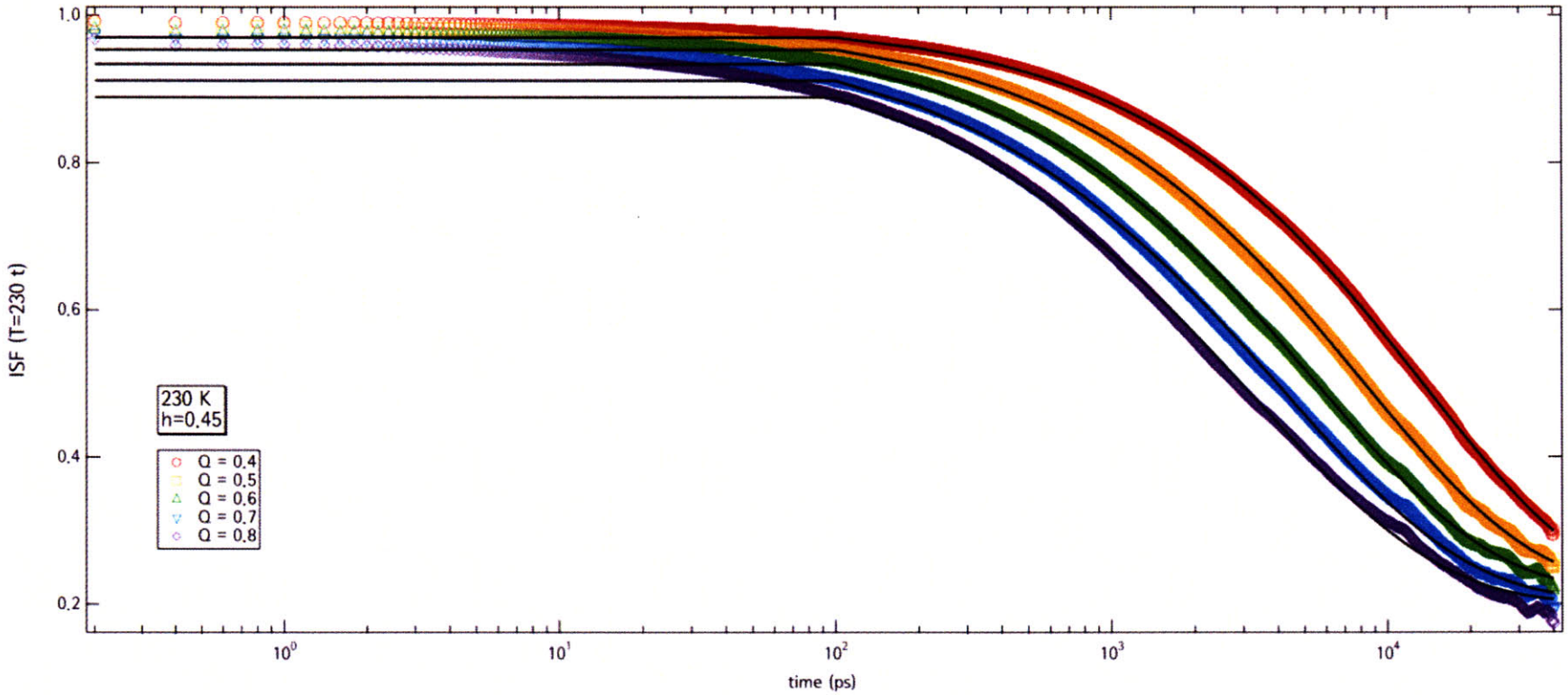

ISF (Q=0.7, t)
time (ps)
Q = 0.7
h=0.45
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250
T = 260
T = 270

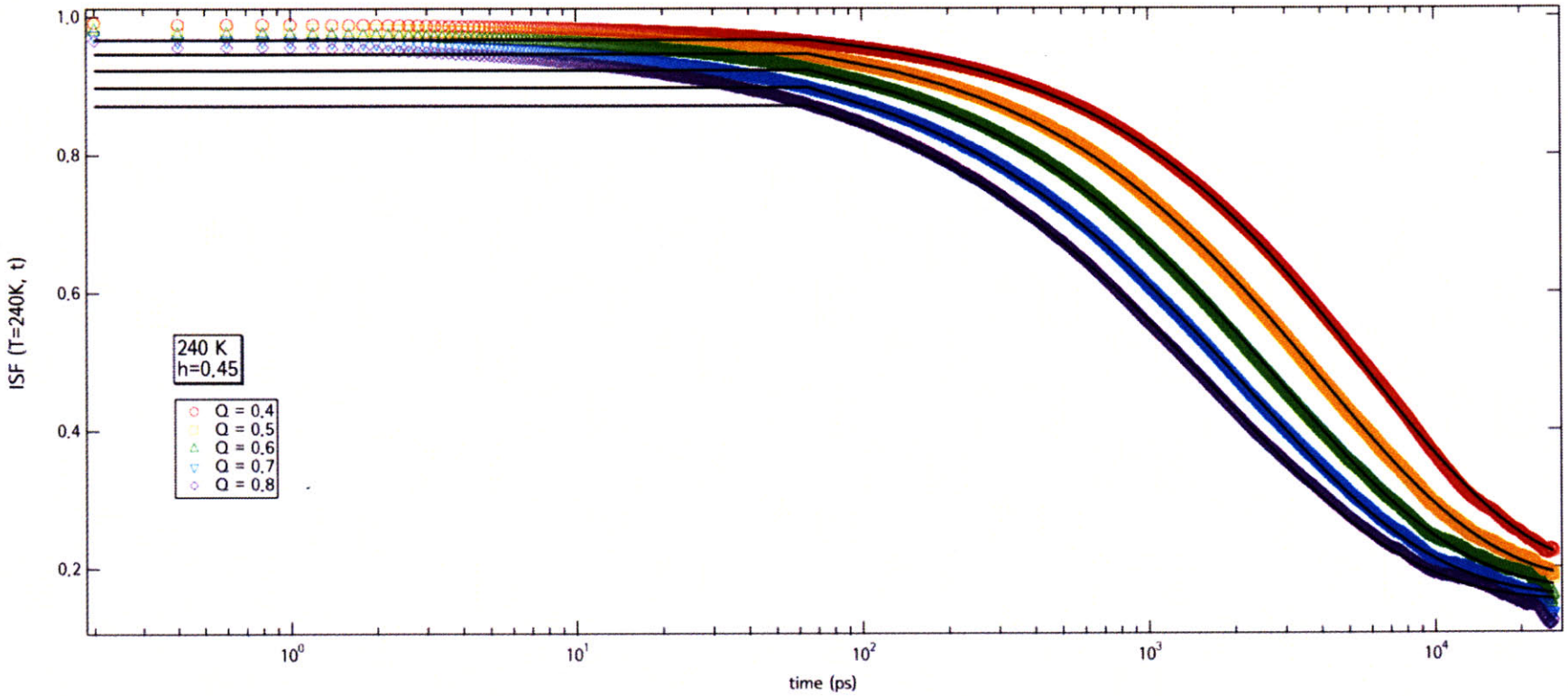

ISF (Q=0.8, t)
time (ps)
Q = 0.8
h=0.45
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250
T = 260
T = 270

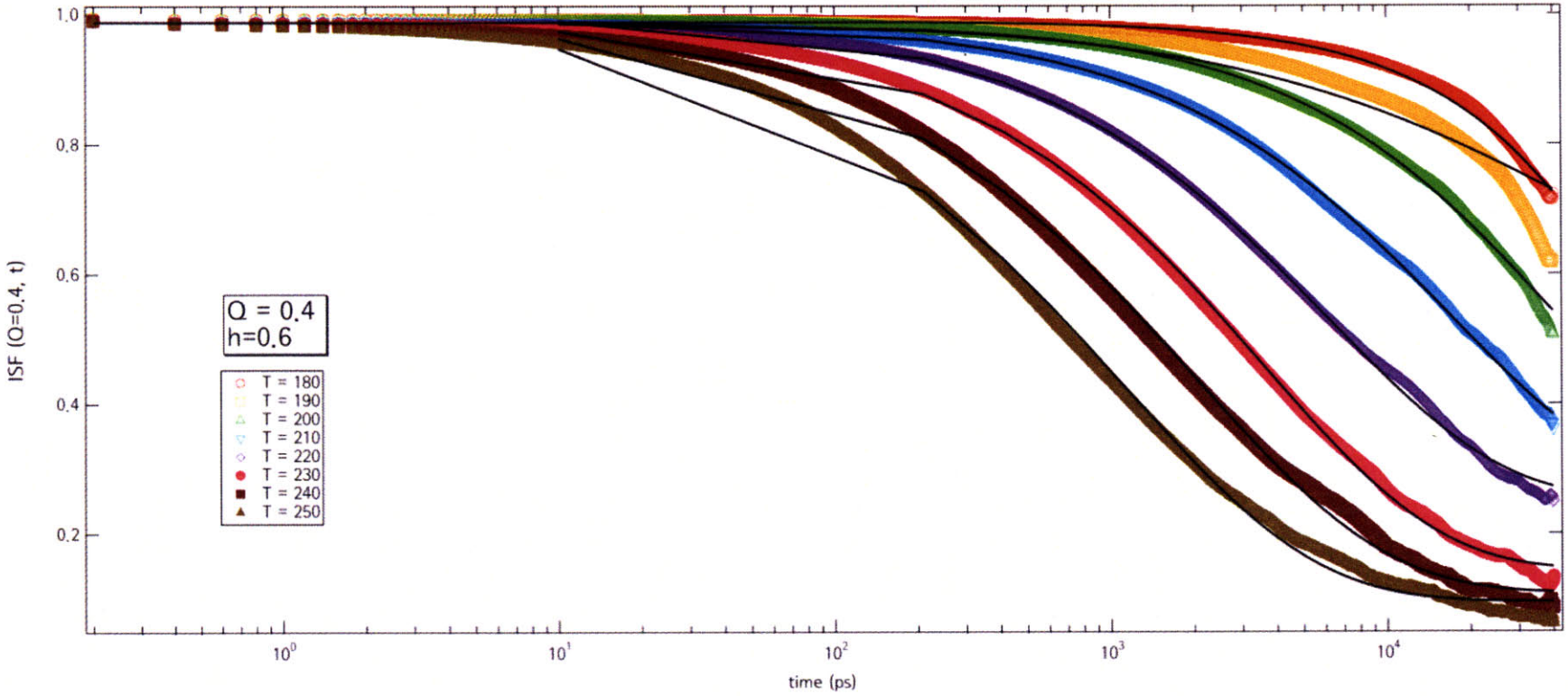

1.00
0.95
0.90
0.85
0.80
0.75
ISF (T=200K, t)
200 K
h=0.45
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8
10^0
10^1
10^2
10^3
10^4
time (ps)

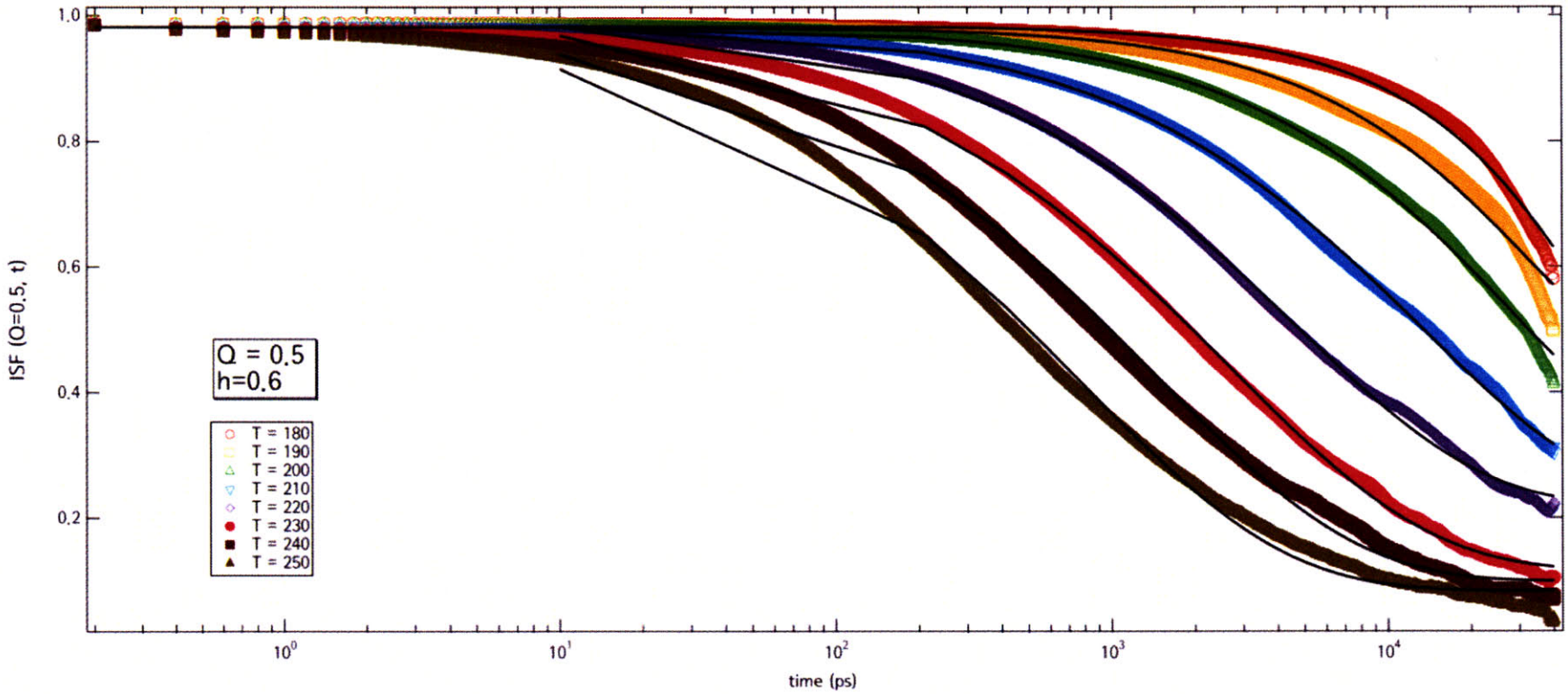

ISF (T=210K, t)
time (ps)
210 K
h=0.45
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8
1.0
0.9
0.8
0.7
0.6
10^0
10^1
10^2
10^3
10^4

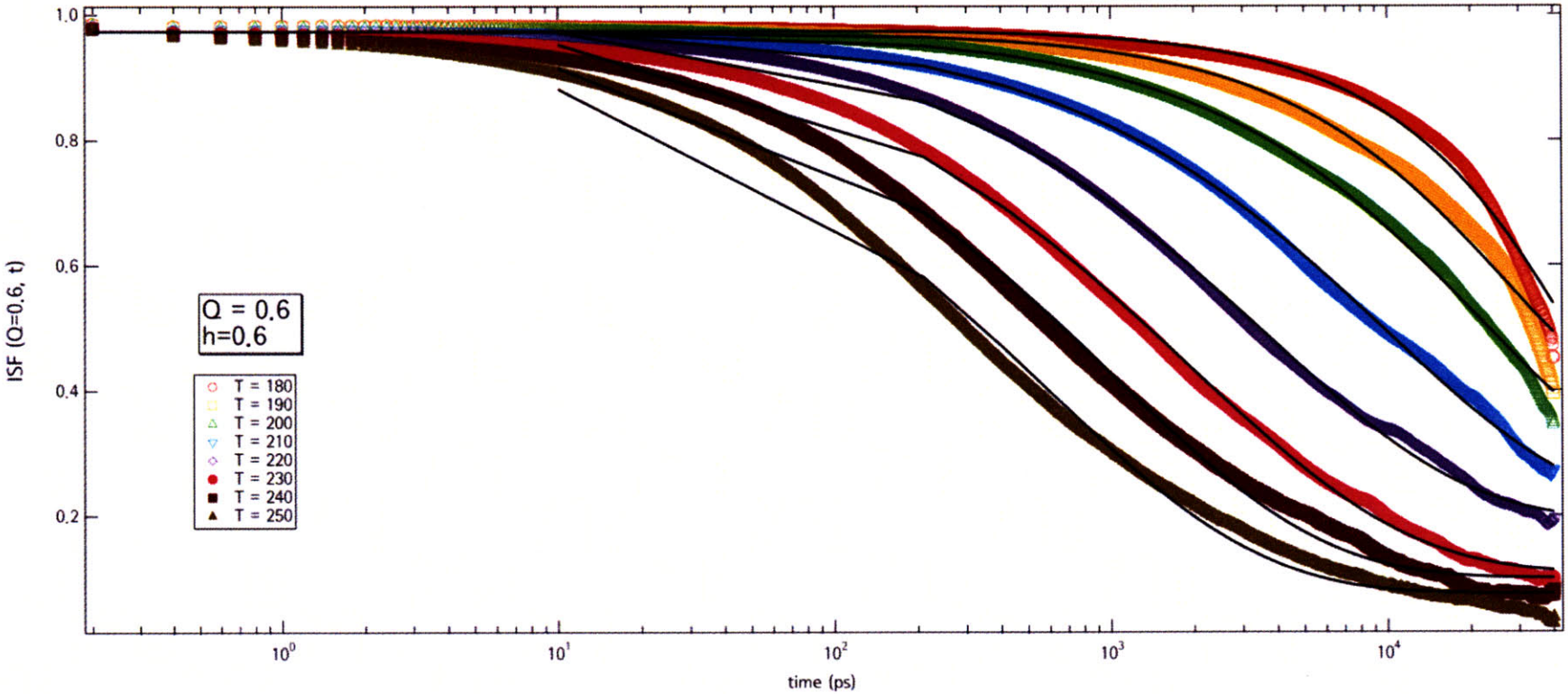

ISF (T=220 K, t)
time (ps)
220 K
h=0.45
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
10^0
10^1
10^2
10^3
10^4

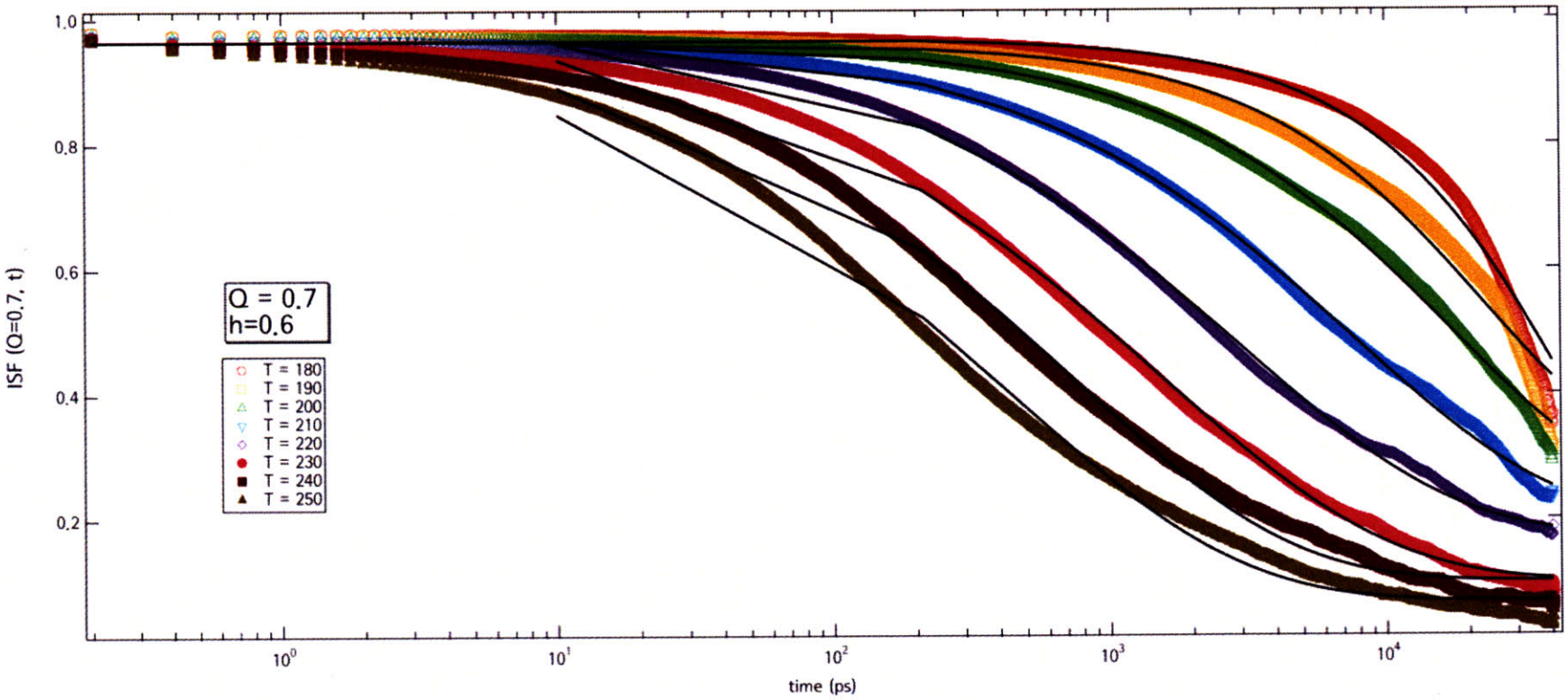

ISF (T=230 t)
time (ps)
230 K
h=0.45
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8

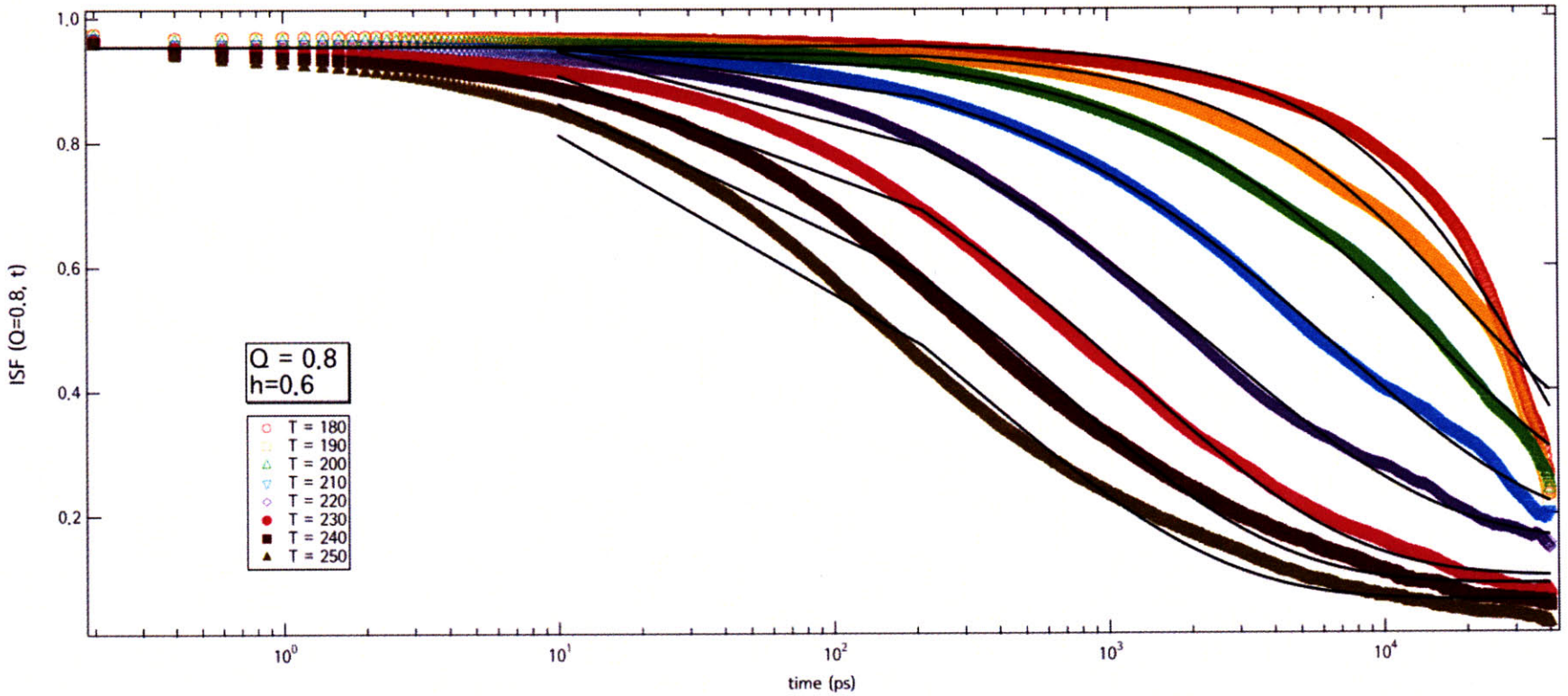

ISF (T=240K, t)
time (ps)
240 K
h=0.45
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8
1.0
0.8
0.6
0.4
0.2
10^0
10^1
10^2
10^3
10^4

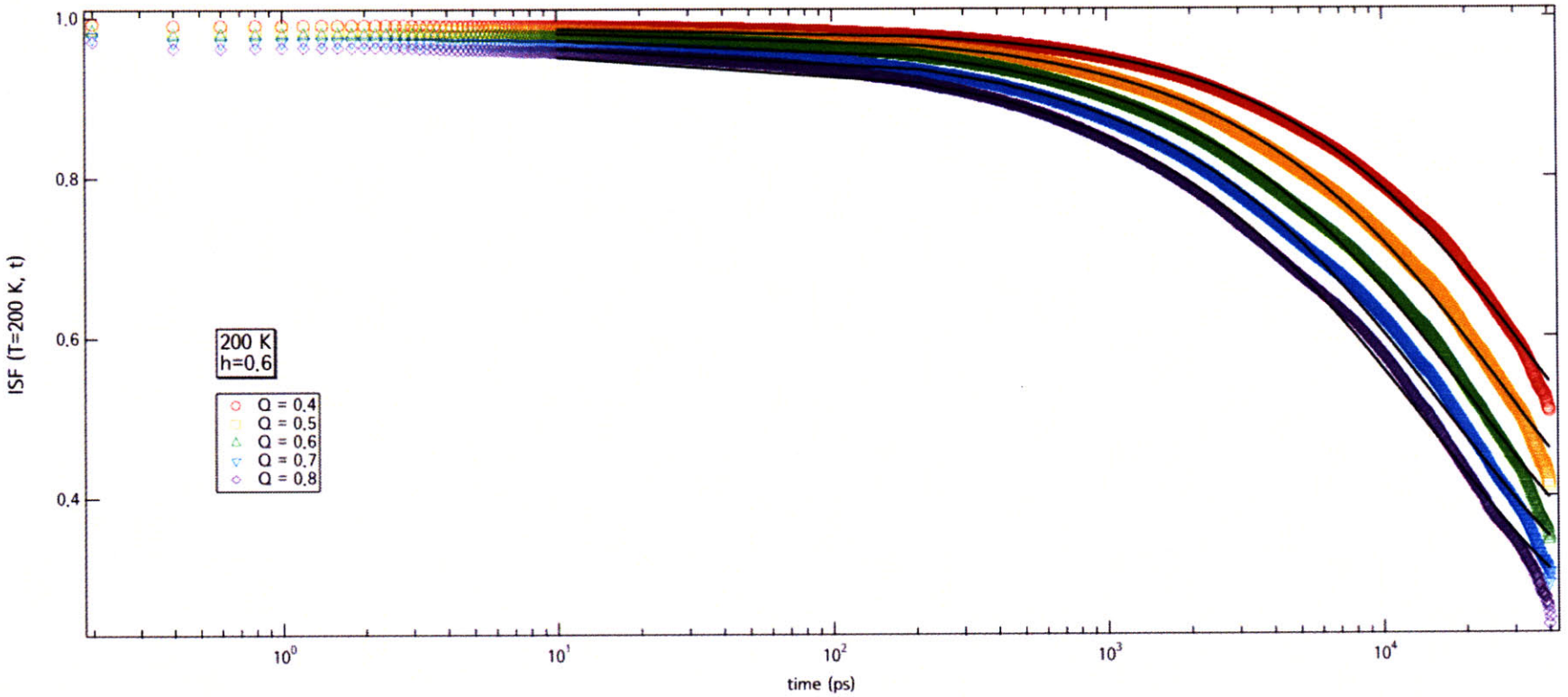

ISF (Q=0.4, t)
time (ps)
Q = 0.4
h=0.6
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250

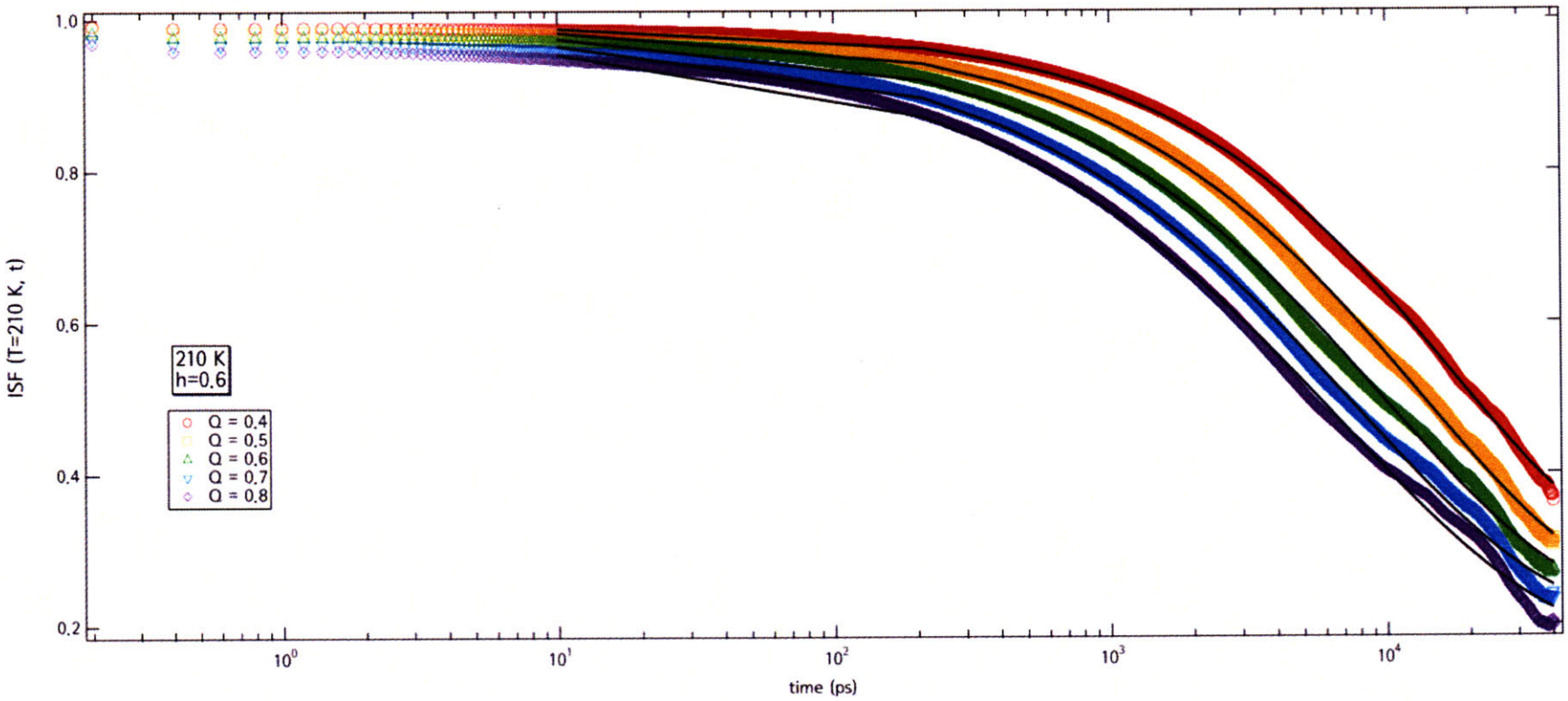

ISF (Q=0.5, t)
time (ps)
Q = 0.5
h=0.6
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250

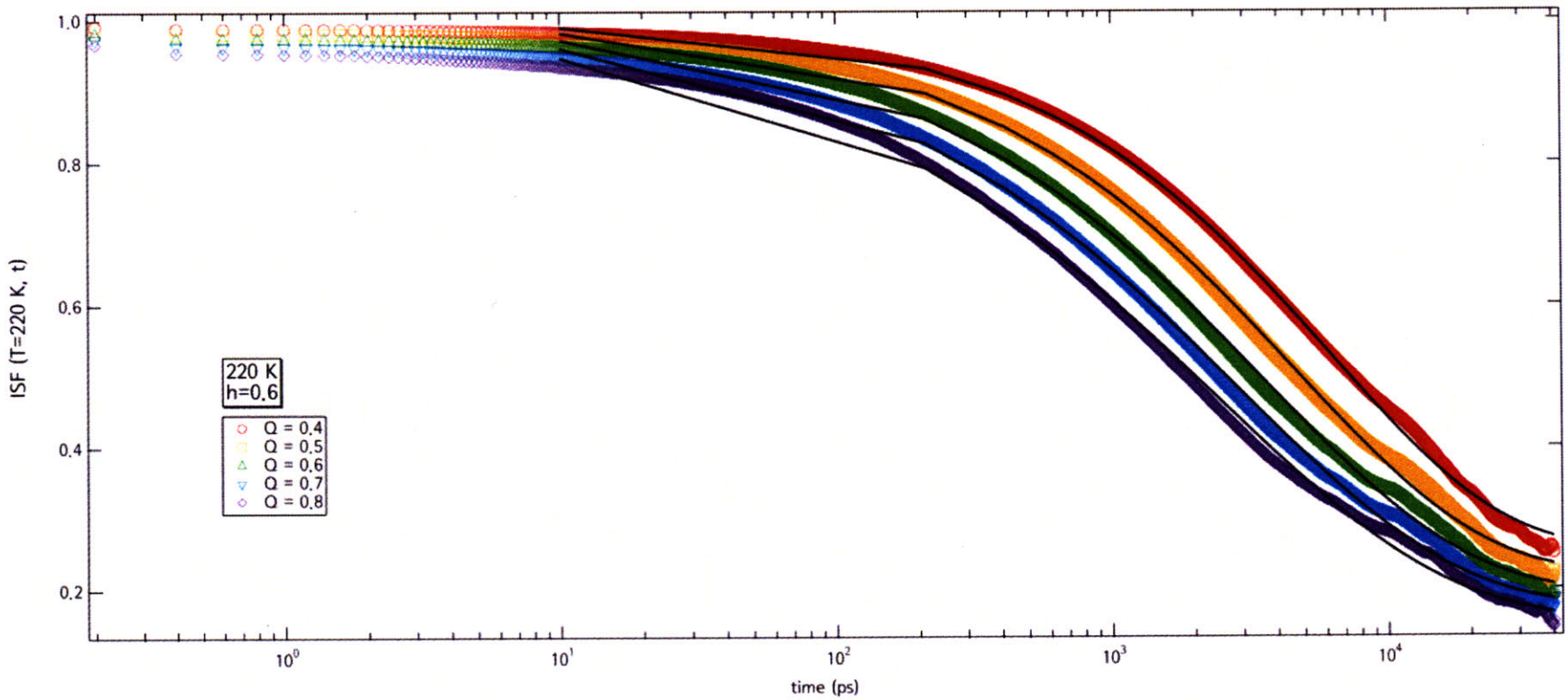

ISF (Q=0.6, t)
time (ps)
Q = 0.6
h=0.6
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250

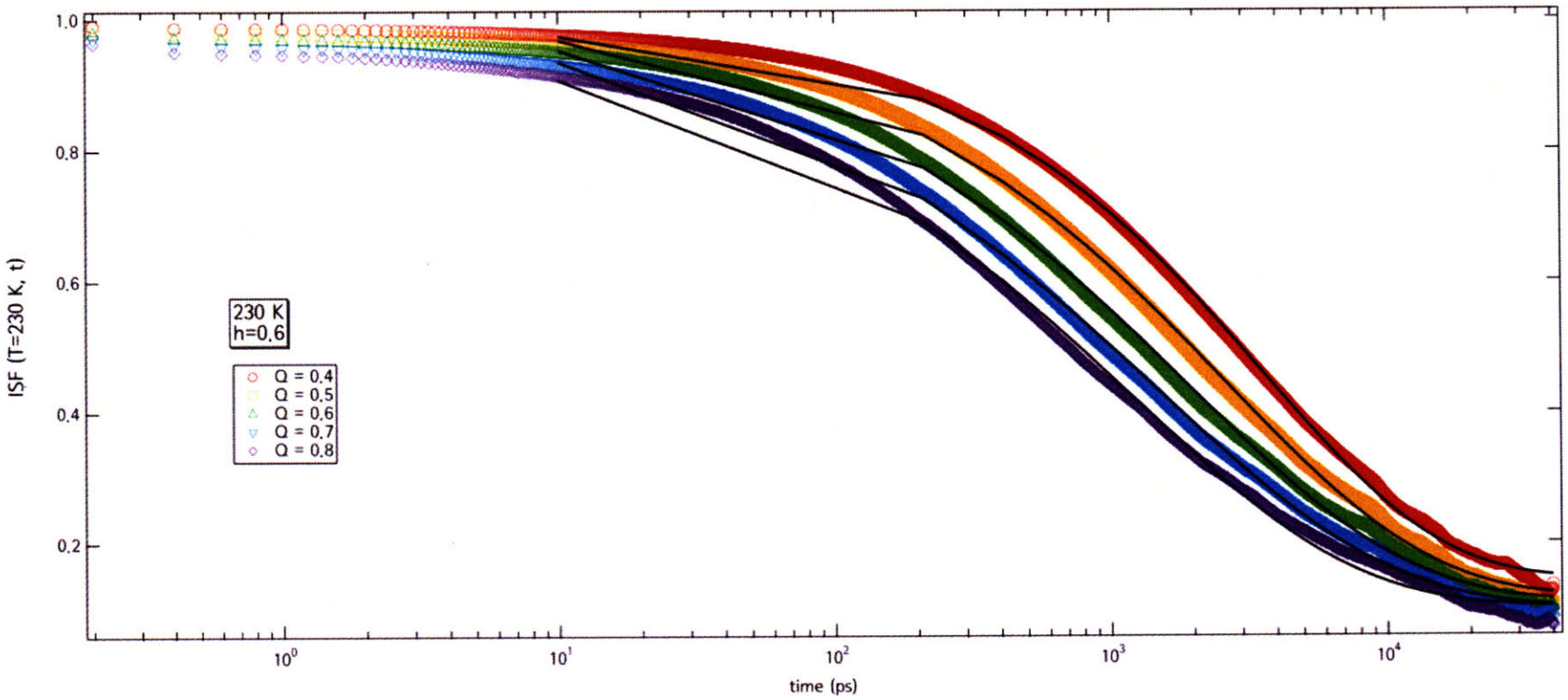

ISF (Q=0.7, t)
time (ps)
Q = 0.7
h=0.6
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250

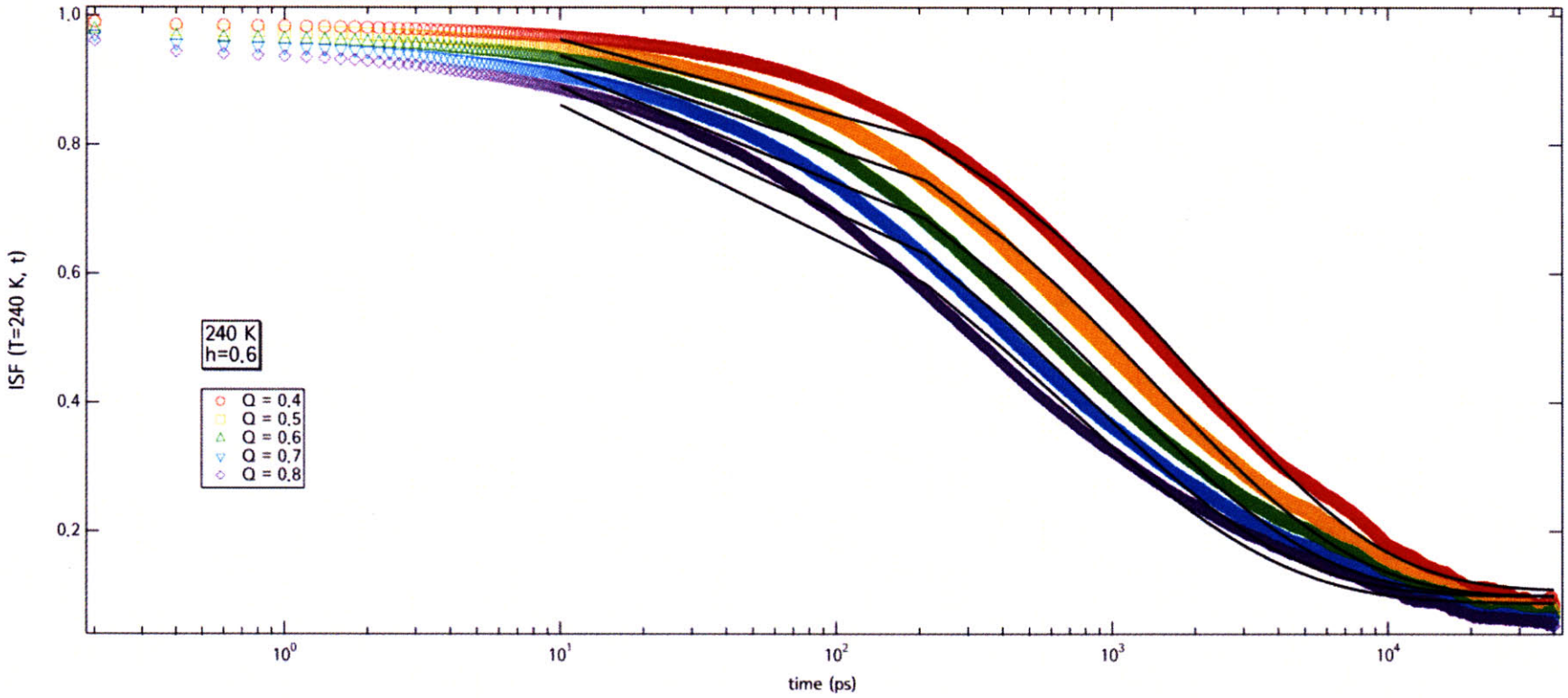

ISF (Q=0.8, t)
time (ps)
Q = 0.8
h=0.6
T = 180
T = 190
T = 200
T = 210
T = 220
T = 230
T = 240
T = 250


1.0
0.8
0.6
0.4
ISF (T=200 K, t)
200 K
h=0.6
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8
10^0
10^1
10^2
10^3
10^4
time (ps)


ISF (T=210 K, t)
time (ps)
210 K
h=0.6
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8
1.0
0.8
0.6
0.4
0.2
10^0
10^1
10^2
10^3
10^4


ISF (T=220 K, t)
time (ps)
220 K
h=0.6
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8
1.0
0.8
0.6
0.4
0.2
10^0
10^1
10^2
10^3
10^4


ISF (T=230 K, t)
time (ps)
230 K
h=0.6
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8
1.0
0.8
0.6
0.4
0.2
10^0
10^1
10^2
10^3
10^4


ISF (T=240 K, t)
time (ps)
240 K
h=0.6
Q = 0.4
Q = 0.5
Q = 0.6
Q = 0.7
Q = 0.8
1.0
0.8
0.6
0.4
0.2
10^0
10^1
10^2
10^3
10^4

6.2. Publications

"Pressure dependence of the dynamic crossover temperatures in protein and its hydration water" X.Q. Chu, A. Faraone, C. Kim, et al., submitted to *Phys. Rev. Lett.* (2008).

"Clustering dynamics in water/methanol mixtures: a Nuclear Magnetic Resonance study at 205 K < T < 295 K", C. Corsaro, J. Spooren, C. Branca, N. Leone, Nancy, M. Broccio, C. Kim, S.-H. Chen, E. Stanley, F. Mallamace, *J. Phys. Chem. B.*, **112** 10449–10454, (2008).

"The Low-Temperature Dynamic Crossover Phenomenon in Protein Hydration Water: Simulations vs Experiments", M. Lagi, X. Chu, C. Kim, F. Mallamace, P. Baglioni, S. H. Chen, *J. Phys. Chem. B*, 112, 1571 (2008).

"Dynamic Crossover Phenomenon in Confined Supercooled Water and its Relation to the Existence of a Liquid-Liquid Critical Point in Water", S.H. Chen, F. Mallamace, L. Liu, D. Z. Liu, X. Chu, Y. Zhang, C. Kim, A. Faraone, C. Y. Mou, E. Fratini, P. Baglioni, A.I. Kolesnikov, V. Garcia-Sakai, the 5th Int'l Workshop on Complex Systems.

6.3. Matlab® Code for QENS ISF fitting by Gaussian Functions

Matlab code is available for fitting QENS ISF functions using the new time domain fitting shown in Chapter 3 and Chapter 4. Please contact Professor Sow-Hsin Chen (sowhsin@mit.edu) or Chansoo Kim (chance@mit.edu or water@alum.mit.edu) to obtain the code.

Chapter 7. BIBLIOGRAPHY

[1] P. G. Debenedetti and H. E. Stanley, *Phys. Today* **56**, 40-46 (2003).

[2] P. G. Debenedetti, *J. Phys.: Condes. Matter* **15**, R1669-R1726 (2003).

[3] R. J. Speedy and C. A. Angell, *J. Chem. Phys* **65**, 851-858 (1976).

[4] D. H. Rasmussen and A. P. MacKenzie, "Interactions in the water-polyvinylpyrrolidone system at low temperatures" in *Water Structure at the Water-Polymer Interface* (Eds. H.H. Jellinek) Plenum Press, NY, 1972, 126.

[5] P. H. Poole, F. Sciortino, U. Essmann, and H. E. Stanley, *Nature (London)* **360**, 324-328 (1992).

[6] K. Skosld and D. L. Price, *Methods of Experimental Physics Vol. A-C*, London: Academic Press, 1986.

[7] Proceeding of the 2nd International Workshop on Dynamics in Confiement, organized by B. Frick, M. Koza, R. Zorn, ILL, France, *European Phys. J. E* **12**, no. 1 (2003).

[8] M. -C. Bellissent-Funnel, S. Longeville, J. -M. Zanotti, and S. -H. Chen, *Phys. Rev. Lett.* **85**, 3644-3647 (2000).

[9] J. -M. Zanotti, M. -C. Bellissent-Funnel, and S. -H. Chen, *Phys. Rev. E.* **59**, 3084-3093 (1999).

[10] P. Gallo, F. Sciortino, P. Tartaglia, and S. -H. Chen, *Phys. Rev. Lett* **76**, 2730-2733 (1996).

[11] S. -H. Chen, C. Liao, F. Sciortino, P. Gallo, and P. Tartaglia, *Phys. Rev. E* **59**, 6708-6714 (1999).

[12] A. Faraone, L. Liu, C. -Y. Mou, C. -W. Yen, and S. -H. Chen, *J. Chem. Phys.* **121**, 10843-10846 (2004).

[13] L. Liu, S. -H. Chen, A. Faraone, C. -W. Yen, and C. -Y. Mou, *Phys. Rev. Lett.* **95**, 117802-117802 (2005).

[14] L. Liu, S. -H. Chen, A. Faraone, C.-W. Yen, C.-Y. Mou, A. I. Kolesnikov, E. Mamontov, and J. Leao, *J. Phys.: Cond. Matter* **18**, S2261-S2284 (2006).

[15] B. F. Rasmussen, A. M. Stock, D. Ringe, and G. A. Petsko, *Nature* **357**, 423-424 (1992).

[16] 45. R. M. Daniel, J. C. Smith, M. Ferrand, S. Hery, R. Dunn, and J. L. Finney, *Biophys. J.* **75**, 2504-2507 (1998).

[17] W. Doster, S. Cusack, and W. Petry, *Nature* **337**, 754-756 (1989).

[18] M. Ferrand, A. J. Dianoux, W. Petry, and G. Zaccai, *PNAS USA* **90**, 9668-9672 (1993).

[19] M. Tarek and D. J. Tobias, *Phys. Rev. Lett.* **88**, 138101-138104 (2002).

[20] A. L. Tournier, J. Xu, and J. C. Smith, *Biophys. J.* **85**, 1871-1875 (2003).

[21] P. Kumar, Z. Yan, L. Xu, M. G. Mazza, S. V. Buldyrev, S.-H. Chen, S. Sastry, and H. E. Stanley, *Phys. Rev. Lett.* **97**, 177802-177806 (2006).

[22] A. Paciaroni, A. R. Bizzarri, and S. Cannistraro, *Phys. Rev. E* **60**, R2476-R2479 (1999).

[23] G. Caliskan, A. Kisliuk, and A. P. Sokolov, *J. Non-Crys. Sol.* **307-310**, 868-873 (2002).

[24] P. W. Fenimore, H. Frauenfelder, B. H. McMahon, and F. G. Parak, *PNAS USA* **99**, 16047-16051 (2002).

[25] Bernal, J.D. "The Bakerian lecture, 1962: The structure of liquids" in *Proc. R. Soc.* **280**, 299–322 (1964).

[26] Alder, B. J., T. E. Wainwright, *J. Chem. Phys.* **31**, 459 (1959).

[27] Stauffer D., *Physica A: Statistical and Theoretical Physics* **336**, 1-5 (2004).

[28] Schlick, T., "Pursuing Laplace's Vision on Modern Computers" in *Mathematical Applications to Biomolecular Structure and Dynamics* (Eds. J. P. Mesirov, K. Schulten and D. W. Sumners) IMA Volumes in Mathematics and Its Applications **82**, New York (1996).

[29] *Spectroscopy in Biology and Chemistry: Neutron, X-ray, Laser*, edited by S. -H. Chen and S. Yip, London: Academic Press (1974).

[30] S. -H. Chen, "Quasi-Elastic and Inelastic Neutron Scattering and Molecular Dynamics of Water at Supercooled Temperature" in *Hydrogen Bonded Liquids*, edited J. C. Dore and J. Teixeira, Kluwer Academic Publishers, pp. 289-332, (1991).

[31] E. Fratini, S. -H. Chen, P. Baglioni, and M. -C. Bellissent-Funel, *Phys. Rev. E* **64**, 020201-020204 (R) (2001).

[32] E. Fratini, S. -H. Chen, P. Baglioni, and M. -C. Bellissent-Funel, *J. Phys.Chem* **106**, 158-166 (2002).

[33] A. Faraone, S. -H. Chen, E. Fratini, P. Baglioni, L. Liu, and C. Brown, *Phys. Rev. E* **65**, 040501-040503 (2002).

[34] A. Faraone, L. Liu, C. -Y. Mou, P. -C. Shih, J. R. D. Copley, and S. -H. Chen, *J. Chem. Phys* **119**, 3963-3971 (2003).

[35] L. Liu, A. Faraone, C. -Y. Mou, C. -W. Yen, and S. -H. Chen, *J. Phys: Cond. Matter* **16**, S5403-S5436 (2004).

[36] M. -C. Bellissent-Funel, S. -H. Chen, and J. M. Zanotti, *Phys. Rev. E* **51**, 4558-4569 (1995).

[37] P. Gallo, F. Sciortino, P. Tartaglia, and S. -H. Chen, *Phys. Rev. Lett* **76**, 2730-2733 (1996).

[38] GROMACS User Manual obtained from http://www.gromacs.org/

[39] http://en.wikipedia.org/wiki/GROMACS

[40] http://www.ks.uiuc.edu/Research/vmd/

[41] http://www.fftw.org/

[42] http://www.ks.uiuc.edu/Research/vmd/

[43] M. Lagi et. al, *J. Phys. Chem. B*, **112** 1571 (2008)

[44] Bée, M., "Quasielastic Neutron Scattering" Adam Hilger, Philadelphia, PA (1959).

[45] Tarek, M. and Tobias, D. J. *Biophys. J.*, **79** 3244-3257 (2000).

[46] Tarek, M. and Tobias, D. J. *Phys. Rev. Lett.*, **88** 138101 (2002).

[47] Tarek, M. and Tobias, D. J. *Phys. Rev. Lett.*, **89** 275501 (2002).

[48] Horn, H. W. et. al., *J. Chem. Phys.*, **120** 9665-9678 (2004).

[49] Jorgensen, W. L., et. al., *J. Am. Chem. Soc.*, **110** 1657-1666 (1988).

[50] Udier-Blagovic, M. et.al., *J.Am.Chem. Soc.*, **125** 6016-6017 (2003).

[51] Li Liu et. al., *J. Phys.: Condens. Matter,* **18** S2261–S2284 (2006).

[52] Swenson, J. *Phys Rev. Lett.*, **97** 189801 (2006).

[53] Chen, S.-H. et. al.,. *Phys Rev. Lett.*, **97** 189803 (2007).

[54] Chen, S.-H. et. al, *PNAS USA*, **103** 9012-9016 (2006).

[55] A. M. Tsai, D. A. Neumann, and L. N. Bell, *Biophys. J.* **79**, 2728-2732 (2000).