

Molecular Dynamics simulations of lysozyme in water

MSc in Bioinformatics and Medical Informatics

Fall Semester 2015-2016

Paraskevi Gkeka and Zoe Cournia

Objective : The main objective of this practical is to provide an overview of classical Molecular Dynamics (MD) simulations and Normal Mode Analysis (NMA) by examining the protein Lysozyme within the framework of the NAMD program.

1 Introduction

One of the principal tools in the theoretical study of biomolecules is the method of MD. It is a computational method which calculates the time dependent behaviour of a molecular system. MD methods are used to describe a complex molecular system in terms of a realistic atomic model, with an aim to understand and predict macroscopic properties based on detailed knowledge on an atomic scale.

MD simulations solve Newton's equations of motion for a system of N interacting atoms:

$$m_i \frac{\partial^2 r_i}{\partial t^2} = F_i, \quad i = 1, \dots, N. \quad (1)$$

The forces are the negative derivatives of the potential function $V(r_1, r_2, r_3, \dots, r_N)$:

$$F_i = -\frac{\partial V}{\partial r_i}. \quad (2)$$

The equations are solved simultaneously in small time steps. The system is followed for sometime, taking care that the temperature and pressure remain at the required values and the coordinates are written to an output file at regular intervals. The coordinates as a function of time represent a trajectory of the system. After initial changes, the system will usually

reach an equilibrium state. By averaging over an equilibrium trajectory many macroscopic properties can be extracted from an output file.

2 Potential energy function

Theoretical studies of biological molecules permit the study of the relationships between structure, function and dynamics at the atomic level. Since biological systems involve many atoms, quantum mechanical treatment of these atoms is not feasible. The usual way to solve them is to use empirical potential energy functions which are computationally less expensive, but involve numerous approximations leading to certain limitations. Current generation force fields provide a reasonable good compromise between accuracy and computational efficiency. They are often calibrated to experimental results and quantum mechanical calculations of small model compounds. Among the most commonly used potential energy functions are the AMBER, CHARMM, GROMOS and OPLS force fields. One of the most important limitation of the empirical force fields is that no drastic changes in the electronic structure are allowed. i.e. no events like bond making or breaking can be modeled.

3 Setting up and running molecular dynamics simulations

To begin a molecular dynamics simulation, you must choose an initial configuration of the system, a starting point, or $t = 0$. Most often, in simulations of biomolecules, an X-ray crystal structure or an NMR structure is obtained from the Protein data bank (<http://www.rcsb.org>) and used as the initial structure. It is also possible to use a theoretical structure developed by homology modelling. The choice of the initial configuration must be done carefully as this can influence the quality of the simulation.

MD simulations involve 4 main steps and these will be discussed below:

3.1 Energy minimization

The potential energy function of a biomolecular system is a very complex and multi-dimensional landscape. It has one deepest point, the global minimum and a very large number of local minima. The goal of the energy minimization is to find a local minimum. The energy at this local minimum may be much higher than the energy of the global minimum. Performing an energy minimization will guarantee the removal of any unfavorable van der Waals interactions that may exist, which might otherwise lead to local structural distortion and result in an unstable simulation.

3.2 Heating

The initial velocities at low temperature are assigned to each atom of the system and Newton's equations of motion are integrated to propagate the system in time. During the heating phase, initial velocities are assigned at a low temperature and the simulation is started with periodically assigning new velocities at a slightly higher temperature and letting the simulation continue. This step is repeated until the desired temperature is reached.

3.3 Equilibration

Once the heating process is over and the desired temperature is reached, the simulation is continued and during this phase, properties such as structure, pressure, temperature and the energy are monitored. The point of the equilibration phase is to run the simulation until these properties become stable with respect to time. If in the process, the temperature increases or decreases significantly, the velocities are scaled such that the temperature returns to its desired value.

3.4 Production and Analysis

The final step of the simulation is the production phase, wherein the system is simulated for the time length required from several hundred ps to ns or more. During this process coordinates of the system at different times are stored in the form of trajectories. These are then used for calculations of mean energy, root mean square (RMS) fluctuations between structures etc. From MD simulations, time dependent properties such as correlation functions can also be calculated and these in turn can be related to spectroscopic measurements.

4 Normal mode analysis

Normal Mode Analysis (NMA) is a classical technique for studying the vibrational and thermal properties of various molecular structures at the atomic level. Although this technique is widely used for molecular systems consisting of a small number of atoms, performing NMA on large-scale systems is computational challenging. Mathematically, the motion of the molecule is often described by a second order ordinary differential equation

$$\ddot{q} + F_q = 0, \quad (3)$$

where the matrix F_q is a force constant matrix derived from the second derivative of the potential with respect to the Cartesian coordinates. The standard procedure for solving this equation is to diagonalize the matrix F_q by computing its eigenvalues and eigenvectors. Each eigenvector is often referred to as a normal mode with certain vibrational frequency. The frequency is determined by the eigenvalue. The overall dynamics of the molecular system can be described by a superposition of a number of linearly independent normal modes.

5 Force field example: CHARMM

The force field description of the interatomic forces is split into two categories: the bonded terms and the non-bonded terms. The bonded terms regroup simple covalent binding as well as the more complex hybridization and π -orbital effects, these are the bonds, angles, dihedrals and improper-dihedrals terms. These terms are schematically drawn in Fig. 1. The non-bonded terms describe the van der Waals forces and the electrostatic interactions between the atoms. The different terms will now be presented in more detail.

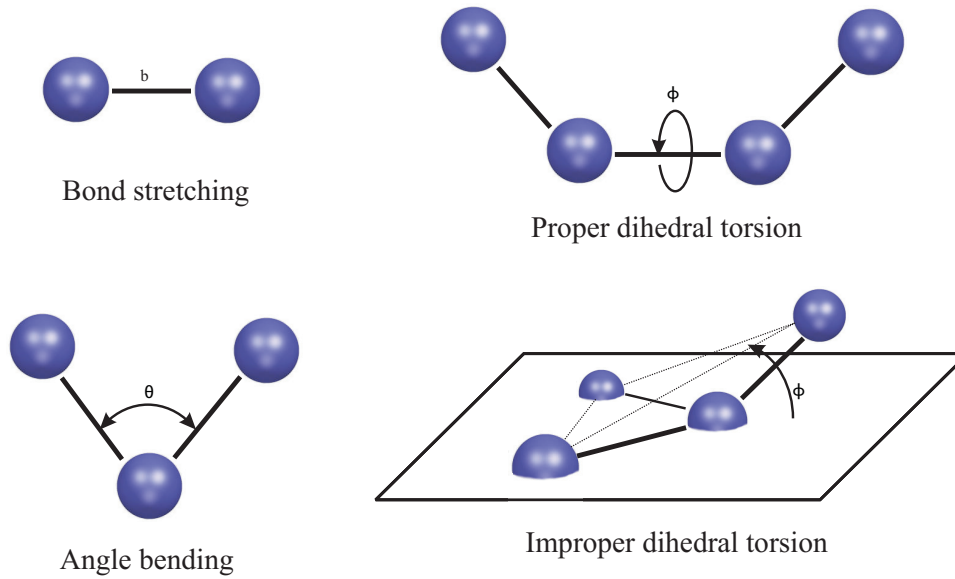


Figure 1: Schematic representation of the bonded interaction terms contributing to the force field: bond stretching, angle bending, proper and improper dihedrals.

Bond stretching

The bond stretching term describes the forces acting between two covalently bonded atoms. The potential is assumed to be harmonic:

$$V_b = k_b(b - b_0)^2 \quad (4)$$

where b is the distance between the two atoms. Two parameters characterize each bonded interaction: b_0 the average distance between them and a force constant k_b .

Angle bending

The angle bending terms describes the force originating from the deformation of the valence angles between three covalently bonded atoms. The angle bending term is described using a harmonic potential:

$$V_\theta = k_\theta(\theta - \theta_0)^2 \quad (5)$$

where θ is the angle between three atoms. There again two parameters characterize each angle in the system: the reference angle θ_0 and a force constant k_θ .

Torsional terms

The torsional terms are weaker than the bond stretching and angle bending terms. They describe the barriers to rotations existing between four bonded atoms. There are two type of torsional terms: proper and improper dihedrals. Proper torsional potentials are described by a cosine function:

$$V_\phi = k_\phi[1 + \cos(n\phi - \delta)], \quad n = 1, 2, 3, 4, 6 \quad (6)$$

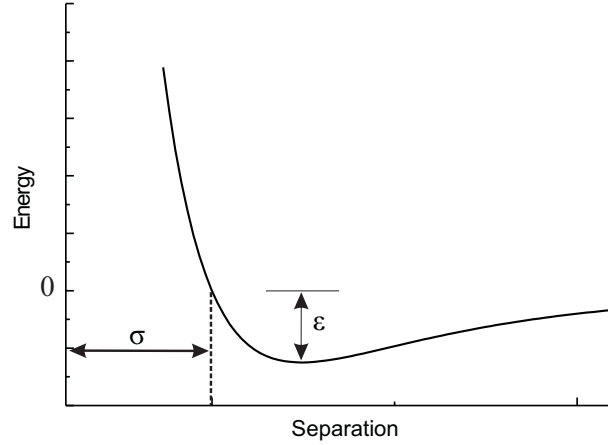


Figure 2: The Lennard-Jones potential. The collision parameter, σ , is shown along with the well depth, ϵ .

where ϕ is the angle between the planes formed by the first and the last three of the four atoms. Three parameters characterize this interaction: δ sets the minimum energy angle, k_ϕ is a force constant and n is the periodicity.

The improper dihedral term is designed both to maintain chirality about a tetrahedral heavy atom and to maintain planarity about certain atoms. The potential is described by a harmonic function:

$$V_\omega = k_\omega (\omega - \omega_0)^2 \quad (7)$$

where ω is the angle between the plane formed by the central atom and two peripheral atoms and the plane formed by the peripheral atoms (see Fig 1).

van der Waals interactions

Van der Waals interactions and electrostatic interactions are non-bonded interactions, i.e., they act between atoms which are not covalently bonded together.

The van der Waals force acts on atoms in close proximity. It is strongly repulsive at short range and weakly attractive at medium range. The interaction is described by a Lennard-Jones potential:

$$V_{vdW} = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right] \quad (8)$$

where r is the distance between two atoms. It is parameterized by σ : the collision parameter (the separation for which the energy is zero) and ϵ the depth of the potential well. The Lennard-Jones potential is represented in Fig. 22.

Electrostatic interactions

Finally, the long distance electrostatic interaction between two atoms is described by Coulomb's law:

$$V_{Elec} = \frac{q_1 q_2}{4\pi\epsilon_0 r_{12}} \quad (9)$$

where q_1 and q_2 are the charges of both atoms and r_{12} the distance between them. ϵ_0 is the electric susceptibility of vacuum.

So finally, the equation for the potential energy describing the force field can be written:

$$\begin{aligned}
 V = & \sum_{bonds} k_b(b - b_0)^2 + \sum_{angles} k_\theta(\theta - \theta_0)^2 \\
 & + \sum_{\substack{proper \\ dihedrals}} k_\phi[1 + \cos(n\phi - \delta)] + \sum_{\substack{improper \\ dihedrals}} k_\omega(\omega - \omega_0)^2 \\
 & + \sum_{\substack{i,j \\ i < j}} 4\epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^6 \right] + \sum_{\substack{i,j \\ i < j}} \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}}
 \end{aligned} \tag{10}$$

5.1 Data Structures

The data structures include information about the molecule, its chemical composition, its chemical connectivity, certain atomic properties, internal coordinates for the energy function etc. This information for a particular class of molecule, e.g. protein and nucleic acids, is contained in the topology and parameter files.

5.1.1 Residue topology file

This file (`top.chm`) contains the information of the residues which are used to build large molecules. It contains the atom type, mass, hydrogen bond donors and acceptors as well as atoms partial charges in particular residue for each and every atom in the system.

5.1.2 Parameter file

This file (`par.chm`) is associated with the RTF file as it contains all the necessary parameters for calculating the energy of the molecule. These include equilibrium bond distances, angles for bond stretching, angle bending and dihedral angle terms in the potential energy function as well as the force constants and the Lennard Jones parameters.

5.1.3 Protein structure file (PSF)

It is the most fundamental data structure for NAMD. It is generated for a specific molecule and it is the concatenation of the information contained in the RTF file. It provides detailed information on the composition and the connectivity of the atoms in the molecule of interest. It gives the total number of bonds and provides information as to which atoms connect to form a particular bond. For all the bonds, angles, the torsion and the dihedral angles are listed. The PSF file must be specified before any calculation is performed. It contains the molecular topology, but not any information regarding bond lengths, angles etc. These informations are provided by the above topology and parameter files.

5.1.4 Coordinate file (PDB)

The coordinate file contains the Cartesian coordinates of all atoms in the system. These are mostly obtained by the X-ray or the NMR experiments. Missing coordinates can be built

within CHARMM using the internal coordinate facility. In addition, the hydrogens which are not present in the X-ray crystal structure can also be built in by CHARMM by using the module HBUILD.

6 NAMD

NAMD is a parallel, Molecular Dynamics (MD) simulation program used to model biomolecular systems using high performance computing (HPC) clusters.

VMD is a molecular viewer program that can be used to help set up NAMD simulation and to help analyze and visualize NAMD output. VMD can be freely obtained from:

<http://www.ks.uiuc.edu/Research/vmd/> <http://www.ks.uiuc.edu/Training/Tutorials/>

This tutorial provides a first introduction to NAMD and assumes that NAMD VERSION or later has been installed on your computer. For the accompanying VMD tutorial go to

<http://www.ks.uiuc.edu/Training/Tutorials/>

The tutorial is subdivided in two sections. The first one covers the basic steps of a MD simulation, i.e., preparation, minimization, and equilibration of your system. The second section introduces typical simulation techniques and the analysis of equilibrium properties.

For a detailed description of NAMD the reader is referred to the NAMD User's guide located at

<http://www.ks.uiuc.edu/Research/namd/current/ug/>

The examples in the tutorial will focus on the study of lysozyme, a small protein with interesting properties.

In the next section, you will learn about typical NAMD input and output files and how to set up basic MD simulations. You will be generating output data by performing simulations and using other features of NAMD. These files are needed for the next sections. If you are not able to produce the output, correct versions have been provided for each section and may be found in the example-output folder inside the directory corresponding to the given simulation or exercise.

6.1 What is Needed

In order to run any MD simulation, NAMD requires (at least) the following:

- a Protein Data Bank (pdb) file which stores atomic coordinates and/or velocities for the system.¹

¹Pdb files may be generated by hand, but they are also available via the Internet for many proteins at <http://www.pdb.org>.

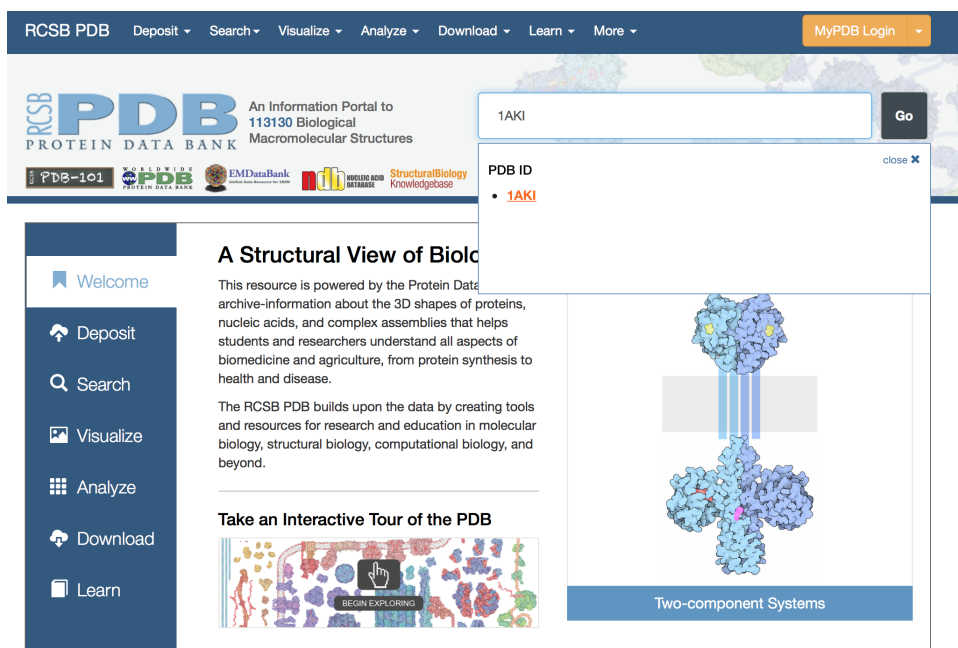


Figure 3: Downloading lysozyme pdb file from RCSB website.

- a Protein Structure File (psf) which stores structural information of the protein, such as various types of bonding interactions.
- a force field parameter file. A force field is a mathematical expression of the potential which atoms in the system experience. The parameter file defines bond strengths, equilibrium lengths, etc.
- a configuration file, in which the user specifies all the options that NAMD should adopt in running a simulation. The configuration file tells NAMD how the simulation is to be run.

6.2 Lysozyme

As mentioned in the previous subsection, the first file needed to run an MD simulation with NAMD (in fact with any simulation package) is a pdb file. To do this, one needs to follow the next steps:

1. We first must download the protein structure file with which we will be working. For this tutorial, we will utilize hen egg white lysozyme (PDB code 1AKI). Go to the RCSB website and download the PDB text for the crystal structure (Figure 3).
2. Once you have downloaded the structure, you can visualize the structure using a viewing program such as VMD (Figure 4). Once you've had a look at the molecule, you are going to want to strip out the crystal waters. Note that such a procedure is not

universally appropriate (i.e., the case of a bound active site water molecule). For our intentions here, we do not need crystal water. There are (at least two ways) to do this:

- (a) Use a plain text editor like vi, emacs, nedit (Linux/Mac), or Notepad (Windows). **Do not use word processing software!** Delete the lines corresponding to these molecules (residue "HOH" in the PDB file). Save the new .pdb with a new file-name, e.g., 1AKI_noW.pdb.
- (b) Alternatively, waters can be removed using VMD.

- i. Launch VMD. Open VMD graphically by double-clicking on the VMD icon. Then, navigate to the NAMD-Tutorial/1.Build_system directory by using the cd command in the VMD command terminal:

cd ~/NAMD-Tutorial/1.Build_system

- ii. Load 1AKI.pdb by clicking File → New Molecule menu item in the VMD Main window. In the Molecule File Browser use the Browse button to find the file 1AKI.pdb. Load it by pressing the Load button. Note that the X-ray structure from the Protein Data Bank does not contain the hydrogen atoms of lysozyme. This is because X-ray crystallography usually cannot resolve hydrogen atoms.² The pdb file you will generate with psfgen along with the psf will contain guessed coordinates for hydrogen atoms of the structure. Later, energy minimization of the protein will ensure their positions are reasonable.
- iii. Choose the Extensions → Tk Console menu item and in the VMD TkCon window. Be sure you are in the ~/NAMD-Tutorial/1.Build_system directory. If you are not, navigate there using the ls command to list files and directories and the cd command to change directories. Then, type the following commands:
set protein [atomselect top protein]
\$protein writepdb 1AKI_noW.pdb
(Hit the Return key after each command.)
In the previous step you have created the file AKI_noW.pdb, which contains the coordinates of the lysozyme alone without crystal waters, in the ~/NAMD-Tutorial/1.Build_system directory.
- iv. Quit VMD by choosing File → Quit.

3. Always check your .pdb file for entries listed under the comment MISSING, as these entries indicate either atoms or whole residues that are not present in the crystal structure. Terminal regions may be absent, and may not present a problem for dynamics. These missing atoms/residues must be modeled in using other software packages.

²X-ray crystallography methods utilize the optical rule that electromagnetic radiation will interact most strongly with matter the dimensions of which are close to the wave-length of that radiation. X-rays are diffracted by the electron clouds in molecules, since both the wavelength of the X-rays and the diameter of the cloud are on the order of Angstroms. The diffraction patterns formed when a group of molecules is arranged in a regular, crystalline array, may be used to reconstruct a 3-D image of the molecule. Hydrogen atoms, however, are not typically detected by X-ray crystallography since their sizes are too small to interact with the radiation and since they contain only a single electron. The best X-ray crystallography resolutions currently available are around 0.9 Å.

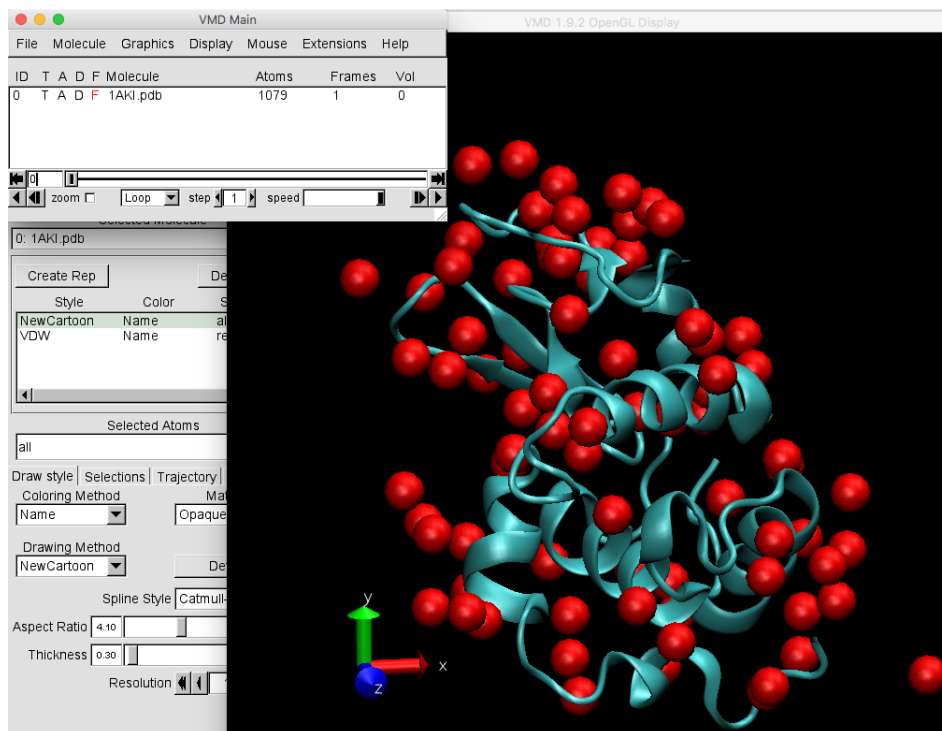


Figure 4: Crystal waters, shown as red spheres, in lysozyme crystal structure should be removed.

6.3 Generating a Protein Structure File (PSF)

To prepare the psf file, you should follow the next steps:

1. Now, you will create the psf file of lysozyme. Note that VMD offers an automatic psf file builder via the VMD Main menu by clicking Extensions → Modeling → Automatic PSF Builder. We will create the psf file manually to teach you exactly how it is done. In the end of this section you can also read how the psf can be generated using the Autopsf Builder module from VMD. The psfgen package of VMD is very useful in this regard. In order to create a psf, you will first make a pgn file, which will be the target of psfgen. In a Terminal window type nedit to open the text editor, type in the following commands and save as a script. Alternatively, you can type the commands directly in the Tk Console (VMD → Extensions → Tk Console):

```

package require psfgen
topology top_all27_prot_lipid.inp
pdbalias residue HIS HSE
pdbalias atom ILE CD1 CD
segment A {pdb 1AKI_noW.pdb}
coordpdb 1AKI_noW.pdb A
guesscoord
writpdb lysozyme.pdb
writpsf lysozyme.psf

```

2. After typing this - if you have opened a text editor - save the file by clicking File → Save. Be sure that you are in the correct directory and enter the file name as 1AKI_noW.pgn. Quit the text editor by clicking File → Exit.

The file you just created contains the necessary commands to create the psf file of lysozyme with hydrogen atoms and without water. Each command of the pgn file is explained:

- Line 1: You will be running psfgen within VMD. This line requires that the psfgen package is available to be called by VMD.
- Line 2: Load the topology file top all27 prot lipid.inp
- Line 3: Change the residue name of histidine to the proper name found in the topology file. HSE is one of three names for histidine, based on the protonation state of its side group.³
- Line 4: The atom named "CD1" (δ carbon) in isoleucine residues is renamed as "CD", its proper name from the topology file. Since isoleucine contains only one δ carbon atom, the psf file does not use the number label after "CD".
- Line 5: A segment called A is created, containing all atoms from 1AKI_noW.pdb. The segment command also adds hydrogen atoms.
- Line 6: Coordinates are read from 1AKI_noW.pdb and residue and atom names are matched. Old segment labels will be overridden with the new segment label ?A?.
- Line 7: Coordinates of missing atoms (like hydrogens) are guessed based on residue definitions from the topology file.
- Line 8: A new pdb file with the complete coordinates of all atoms, including hydrogens, is written.
- Line 9: A psf file with the complete structural information of the protein is written.

3. In a Terminal window (again, be sure that you are in the ~/NAMD-Tutorial/1.Build_system directory), type the following command:

vmd -dispdev text -e 1AKI_noW.pgn

This will run the package psfgen on the file 1AKI_noW.pgn and generate the psf and the pdb file of lysozyme with hydrogens.

In your screen you will see different messages. Warnings are related to the ends of your molecule and are normal.

³Histidine Residues. Of the 20 amino acids, histidine is the only one which ionizes within the physiological pH range (~7.4). This effect is characterized by the pKa of the amino acid side chain. For histidine, the value is 6.04. This leads to the possibility of different protonation states for histidine residues in a protein, and makes the consideration of the proper state important in MD simulations. The viable states are one in which the δ nitrogen of histidine is protonated (listed with residue name "HSD" in the topology file), one in which the ϵ nitrogen of histidine is protonated (?HSE?), and one in which both nitrogens are protonated ("HSP").

4. Type `exit` in the Terminal to exit VMD.
Two new files will appear in your working directory: `lysozyme.pdb` and `lysozyme.psf`. Check this by typing `ls` in a Terminal. You have created the `psf`!
5. Using the AutoPSF Builder from VMD: Extensions → Modeling → Automatic PSF Builder.
Input and Output files:
Step 0. Load protein in VMD
Step 1. Insert topology file: "Topology files → Add". Click on "Load input files".
Step 2. Selections to include in PSF/PDB. Select "Protein" (.pdb file). Click on "Guess and split chains using current selections".
Step 3. Select identified segments from the white box. Click on "Create Chains".
Step 4. Click on "Apply patches and finish PSF/PDB".

Your PSF file is ready!

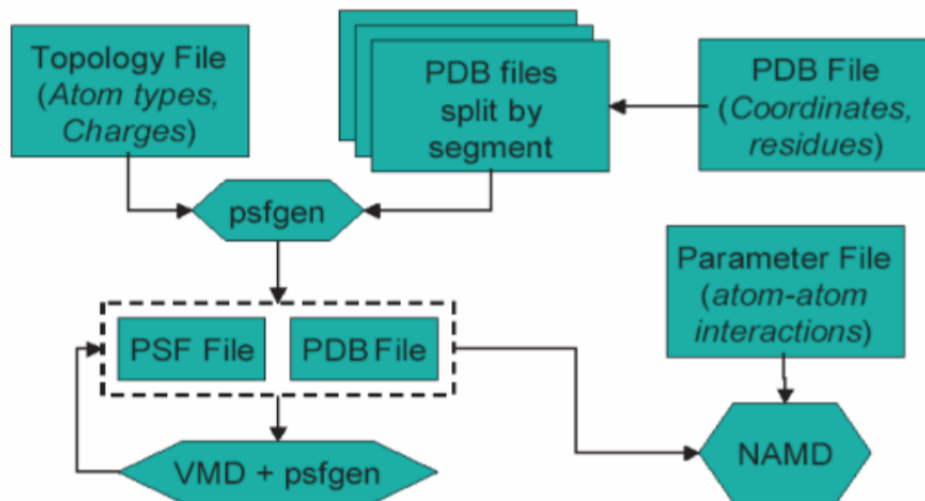


Figure 5: Flowchart indicating the role of files as used by VMD, NAMD, and psfgen.

6.4 Solvating the Protein

In this section we will learn how to solvate a protein, i.e., put inside water, to more closely resemble the cellular environment. You will do so by placing lysozyme in a water box, in preparation for minimization and equilibration with periodic boundary conditions.

1. In the VMD Main window, click Extensions → Tk Console. In the VMD TkCon window of your VMD session type:

```

package require solvate
solvate lysozyme.psf lysozyme.pdb -t 5 -o lysozyme_wb
  
```

The **package require solvate** command loads the *solvate package*, so that VMD will be able to call it.

The *solvate package* will put your protein in a box of water. The -t option creates the water box dimensions such that there is a layer of water 5 Å in each direction from the atom with the largest coordinate in that direction. The -o option creates the output files lysozyme_wb.pdb and lysozyme_wb.psf for lysozyme with the water box.

Size of Water Box. Ideally, one should work with a box which is large enough that the protein does not interact with its image in the next cell if periodic boundary conditions are used.

Ions. Ions should be placed in the water to represent a more typical biological environment. They are especially necessary if the protein being studied carries an excess charge. In that case, the number of ions should be chosen to make the system neutral. The ions present will shield the regions of the protein which carry the charge, and make the entire system more stable. They should be placed in regions of potential minima, since they will be forced to those regions during the simulation anyway. The psf file contains the charge of each atom and may be used to determine the charge of the total system or parts of it.

2. In the VMD Main window, click File → New Molecule. In the Molecule File Browser use the Browse button to find the file lysozyme_wb.psf. Load it by pressing the Load button. This will load the structural information into VMD.
3. In the same Molecule File Browser, browse for lysozyme_wb.pdb and load it. In the display, check if the water box surrounds lysozyme, and demonstrate to yourself that the hydrogen atoms now exist.
4. In the VMD TkCon window type:

```
set everyone [atomselect top all]  
measure minmax $everyone
```

This analyzes all atoms in the system and gives you the minimum and maximum values of x, y and z coordinates of the entire protein-water system. You will get something that looks like the following

```
{4.311999797821045 -1.3930000066757202 -21.77199935913086}  
{52.625 50.944000244140625 22.663999557495117}
```

5. These values are defined relative to the origin of the coordinate system, set by the initial pdb file, 1AKI.pdb. The center of the water box may be determined by calculating

the midpoint of each of the three sides in the coordinate system. Alternatively, one can use tcl scripting in VMD to find the center of the box by typing

```
measure center $everyone
```

in the VMD TkCon window. Determine the coordinates of the center of your water box and record these numbers. In our case:

```
28.69862174987793 24.836271286010742 0.4592978060245514
```

6. Close VMD by choosing File → Quit in the VMD Main window.

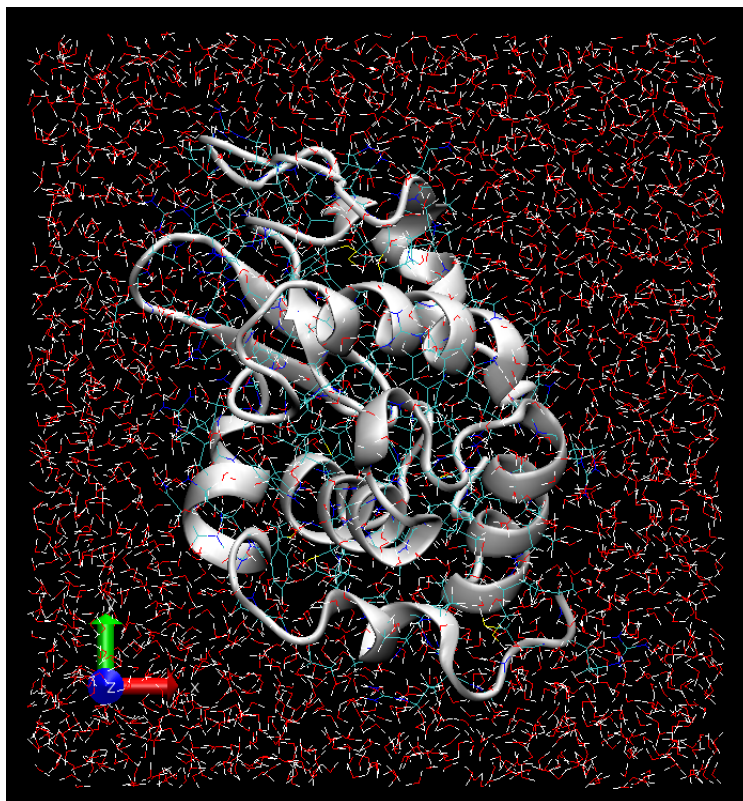


Figure 6: Lysozyme in a water box.

Now that you have the pdb and psf files for solvated lysozyme, along with the parameter file for proteins in general, you only need a NAMD configuration file to be able to minimize and equilibrate your lysozyme-water system.

6.5 Analysis of the Structure and Parameter Files

This section of the practical uses files in your ~/Topology-Parameter directory and the files that you have generated just now (.pdb and .psf). The .psf file is the protein structure file. It contains information needed by NAMD to run the simulation. Each atom is associated with its mass, partial charge, atom reference number (in the topology file top.inp), and residue. Please ask your supervisor to explain these files, and the top.inp and par.inp files needed by NAMD in more detail.

1. How many amino acids and atoms are in lysozyme, how many atoms are hydrogen?
2. Which amino acids are in lysozyme, are they hydrophilic or hydrophobic?
3. How many CYS amino acids are present, what is their residue number and what is their respective distance (C_{α} - C_{α} in Å)?
4. How are the amino acids connected to each other? Name properties of this bond and explain their origin!
5. How many and which parameters are needed to describe the protein peptide bond (see par.inp)?

Di-Sulfide Bonds: Inspect the top.inp and par.inp files

6. What is the partial charge of the hydrogen (HS/HG1) and the sulfur (S/SG) atom in CYS?
7. What are the parameters for the sulfur-hydrogen (S-HS) and the sulfur-sulfur (SM-SM) bond? Which of these two bonds is stronger?

6.6 Minimization

In this section, you will examine the minimization of lysozyme in a water box with periodic boundary conditions.

Periodic Boundary Conditions. The use of periodic boundary conditions involves surrounding the system under study with identical virtual unit cells. The atoms in the surrounding virtual systems interact with atoms in the real system. These modeling conditions are effective in eliminating surface interaction of the water molecules and creating a more faithful representation of the in vivo environment than a water sphere surrounded by vacuum provides.

Minimization and Equilibration. Minimization and equilibration differ from each other by the nature in which they implement the molecular dynamics force field. Energy

minimization involves searching the energy landscape of the molecule for a local minimum, i.e., place in which the molecule is relaxed, by systematically varying the positions of atoms and calculating the energy. Energy minimization should remove any unwanted high-energy interactions that may have been present in the crystal structure. Equilibration involves molecular dynamics, whereby Newton Second Law is solved for each atom in the system to dictate its trajectory. Achievement of equilibrium is judged by how well velocities, pressure, etc. are distributed in the system over a given amount of time.

6.6.1 Configuration File

Go to your minimization directory by typing **cd 2.Minimization**. You will find a configuration file for the minimization of lysozyme in a water box. All output files for the minimization of your lysozyme in a water box system should be placed in this directory.

Open the configuration file, lysozyme_wb_mini.conf by typing
ncedit lysozyme_wb_mini.conf

- **Periodic Boundary Conditions**

- Three periodic cell basis vectors are to be specified to give the periodic cell its shape and size. They are **cellBasisVector1**, **cellBasisVector2**, and **cellBasisVector3**.
- **cellOrigin**: specifies the coordinates of the center of the periodic cell in Å. This is why you need to calculate the center of your water box after you solvate!
- **wrapWater**: This command may be used with periodic boundary conditions. If a water molecule crosses the periodic boundary, leaving the cell, setting this command to on will translate its coordinates to the mirror point on the opposite side of the cell. Nothing can escape. The command may be set to on or off.
- **wrapAll**: same as wrapWater, except this applies to all molecules.

- **PME (for full-system periodic electrostatics)**

Particle Mesh Ewald or PME is a useful method for dealing with electrostatic interactions in the system when periodic boundary conditions are present. The Ewald sum is an efficient way of calculating long range forces in the periodic system. The particle mesh is a 3-D grid created in the system over which the system charge is distributed. From this charge, potentials and forces on atoms in the system are determined. As a result, your grid size should be chosen such that it is fine enough to accurately represent the configuration of your system.

- **PME**: indicates whether or not the simulation uses the Particle Mesh Ewald Sum method; uses values yes and no.
- **PMEGridSpacing**: sets the minimum ratio between the number of PME grid points along each cellBasisVector and the physical dimensions. Since the grid will replicate the charge distribution in your system, PMEGridSpacing should be chosen

to be large enough so that the grid spacing accurately reproduces your charge distribution. However, it should not be so large that it will slow down your simulation to provide unnecessary precision. Typically, a grid density of slightly more than 1 Å is a good choice to reproduce charge distribution in biological systems, where the closest atoms have a bond separation on the order of 1 Å. This corresponds to a `PMEGridSpacing` of 1.0. NAMD will then automatically set the PME grid sizes (see below) such that there is always less than 1.0 Å between grid points and the sizes contain only small prime factors (e.g. 2, 3, and 5). Alternatively, one can define the PME grid sizes manually, using `PMEGridSizeX`, `PMEGridSizeY`, and `PMEGridSizeZ`. These set the size of the PME grid along `cellBasisVector1`, 2 and 3, respectively (not the x, y, and z directions as implied). For speed in computing Fast Fourier Transforms, `PMEGridSizeX` should be chosen so that it can be factorized by 2, 3, or 5. Note that since `cellBasisVector` is defined with slightly different values in each direction, the size of the mesh spacing (in length) will be different in each direction. Also, note also that when using the PME method, the command cutoff dictates the separation between long and short range forces for the method; it does not simply turn off interactions.

- **Output**

- **xstFreq**: The extended system trajectory file contains a record of the periodic cell parameters, essentially recording the trajectory of the cell boundaries over the run time. This command specifies how often, in time steps, the configuration will be recorded. If this command is set, three xst files will be output: 1 final and 2 restarts.

6.6.2 Run the minimization

Run your simulation by typing in a Terminal window:

```
namd2 lysozyme_wb_mini.conf > lysozyme_wb_mini.log &
```

After minimization, energy are presented in the log file. The first lines confirm that NAMD is writing the proper information at the proper time step, which you have specified in the configuration file. Energies at that time step are then fully reported. In this case, they are output every 100 time steps, as specified in the configuration file.

Follow the output as NAMD runs by typing `tail -f lysozyme_wb_mini.log`. At the end of the run, type:

```
grep 'ENERGY:' lysozyme_wb_mini.log > lysozyme_wb_mini.dat
```

to look at how the energy has changed during the simulation. This can be plotted using `XMGRACE`. The second column will be the number of time steps (dimensionless) and the

eleventh the total energy (in kcal.mol⁻¹).

1. Plot the total energy vs. minimization step! Has the minimization converged?
2. Plot the different energy terms, e.g. VdW, electrostatic. (Hint: the sixth column is electrostatics energy and the seventh the van der Waals energy.)

If the energy hasn't converged, create a new input file and edit it to increase the number of minimization steps. Run this new script with a similar command as above; output file will be `lysozyme_wb_mini2.log` and the result of the `grep` command will be `lysozyme_wb_mini2.dat`.

3. Examine the `.pdb` files before and after energy minimization using VMD. Notice any differences? (Hint 1: calculate the RMSD using VMD, Hint 2: first load the `.psf` and then the `.pdb` file into VMD.)
4. What is the greatest gradient during any of the minimizations? When does it appear and why?
5. Try to imagine a few structures for which the CHARMM force-field would give an energy that tends to ∞ .

In the following you will perform a molecular dynamics (MD) simulation of lysozyme in water, in the isobaric-isothermic (NPT) ensemble. The simulation will be analysed and the dynamics of the protein explored by plotting time series as a function of simulation time, for different structural parts of the protein. The simulation will consist of four parts:

1. heating up the system from the energy-minimized structure at 0 K (Kelvin) to the desired temperature for collecting results (e.g. the physiological temperature, 310 K).
2. equilibration phase, where the system stabilizes in an equilibrium position before the production run. This is done to relax the system before collecting data.
3. production run, which is normally a longer run to collect data.

6.7 Heating

You will start the dynamics runs using the energy minimized structure of lysozyme, so you must also copy the `psf lysozyme_wb.psf`, `pdb lysozyme_wb.pdb` and minimized coordinates in binary format `lysozyme_wb_mini.coor` across to the directory `3.Heating`. Also don't forget to copy your parameter file. The binary coordinate file contains the minimized position coordinate data. This binary coordinate was created as output from NAMD by activating the `binaryrestart` or `binaryoutput` options (set to `yes` by default) in the minimization step. Note that, in the current implementation at least, the `bincoordinates` option must be used (`.coor` file) in addition to the `coordinates` option, but the positions specified by `coordinates` will then be ignored (`.pdb` file).

Open the configuration file (`lysozyme_wb_heat.conf`) to inspect it.

- **Constant Pressure Control (variable volume)**

- **useGroupPressure:** NAMD calculates system pressure based on the forces between atoms and their kinetic energies. This command specifies whether interactions involving hydrogen should be counted for all hydrogen atoms or simply between groups of hydrogen atoms; uses values yes and no and must be set to yes if rigidBonds are set.
- **useFlexibleCell:** specifies whether or not you want to allow the three dimensions of the periodic cell to vary independently; uses values yes or no.
- **useConstantArea:** NAMD allows you to keep the x ? y cross sectional area constant while varying the z dimension; uses values yes and no.
- **langevinPiston:** indicates whether or not the simulation uses a Langevin piston to control the system pressure; uses values on and off.
- **langevinPistonTarget:** specifies, in units of bar, the pressure which the Langevin piston tries to maintain. (1 atm = 1.013 bar)

More information on all the keywords can be found in the NAMD manual

<http://www.ks.uiuc.edu/Research/namd/current/ug/>

You should be heating for 500 steps (1ps) every 10 K for a total of 32 ps, at a time step of 0.002 ps. During the heating, the velocities of all of the atoms are scaled to the desired temperature.

6.7.1 Run the heating

Run your simulation by typing in a Terminal window:

```
namd2 lysozyme_wb_heat.conf > lysozyme_wb_heat.log &
```

as before, tail the output by typing: `tail -f lysozyme_wb_heat.log`. At the end of the run type:

```
grep 'ENERGY >' lysozyme_wb_heat.log > lysozyme_wb_heat.dat
```

The file `heat.dat` will contain the number of time steps (dimensionless), bond, angle, dihedral, improper torsion, electrostatic, van der Waals energies, boundary and misc (two columns set to zero), kinetic and total energy (all in kcal.mol⁻¹), temperature (in K), potential energy, total3, temperature average, pressure (in atm), GP Prssure, Volume, Pressure average and GP Pressure avarage and temperature (in K) in the first to fifteenth columns, respectively.

Have a look at the protein using VMD. Compare the structures of the hot protein (i.e. heated structure) with the cold protein (minimized or crystal structure).

1. What limits the simulation time step to 1 fs. How could this be improved?
2. Why is the protein slowly heated in short intervals during the heating phase and not at once at the beginning. Why is there only one heating step every 500 simulation time steps.
3. Examine and plot using XMGRACE the total energy, potential and kinetic energy, and temperature versus simulation time. Notice anything unusual in your plots?
4. Can you relate the kinetic energy and the temperature to each other? (Hint: Inspect the CHARMM output-file and look at the parameters just after the dynamics command.)

6.8 Equilibration

After heating the protein so quickly the structure may be a little unstable, and in further simulations, the system may become unstable. To avoid this, you will simulate further in an equilibration period, by using as input the heated structure. Go to the directory 4.Equilibration. The input file for this is `lysozyme_wb_eq.conf`. Check that the final temperature now is set to 310 K. Now you will have to copy the coordinate file that was produced after heating. This file contains the binary coordinates for the heated protein structure. Also, the psf and the pdb files are also needed for reference. Some more parameters appear in the input file.

- **Constant Temperature Control (variable volume)**

The following are specifications for the Langevin piston which NAMD allows you to specify.

- **langevinPistonPeriod**: sets the oscillation time constant in fs for the Langevin piston.
- **langevinPistonDecay**: sets the damping time constant in fs for the Langevin piston.
- **langevinPistonTemp**: sets the noise temperature in K for the Langevin piston; should be set equal to the target temperature for the temperature control method (here, set by `langevinTemp`).

You will be running this simulation for a total of 10000 steps with a 2 fs timestep, i.e. 20 ps.

Run the simulation by typing:

```
namd2 lysozyme_wb_eq.conf > lysozyme_wb_eq.log &
```

and examine the output as before.

1. Are the temperature and energy stable? Did the system reach thermodynamic equilibrium?
2. Print out the same plots as in the first equilibration; total energy, potential energy, and temperature.

6.9 Production Dynamics

If the system has equilibrated, you are ready to start the analysis run ! Now you will simulate for 50 ps, with no heating or temperature scaling of course, and write out the trajectory file `lysozyme-prod.dcd`. The production run file has not been generated, so it is your task to produce your own input file.

1. Analyse the output from the final run as before, i.e. plot energies and temperature versus time.
2. Was the system stable during the entire run ?
3. Generate histogram plots for the total energy and temperature using XMGRACE (`data> transformation> histograms`). What is their functional form? (Hint: This is discussed in any textbook about Thermodynamics.)
4. Is the total energy conserved? Why?

7 Trajectory analysis using tcl scripting

This section presents a series of examples of how to analyze data produced from NAMD simulations. For these examples you will not run the simulations themselves; instead you will analyze already produced data. Nevertheless, the instructions to set up and run the simulations are included, so you can actually produce the data yourself if time permits.

7.1 Calculating properties in equilibrium

This section contains exercises that will let you analyze properties of a system in equilibrium. The exercises mostly use pre-equilibrated trajectories from different ensembles to calculate properties of the lysozyme system.

7.1.1 Root Mean Square Deviation (RMSD)

1. Go to the directory `6.Analysis/a.rmsd/`. You will use a script within VMD that will allow you to compute the average RMSD of each residue in your protein, and assign this value to the User field for each residue.
2. Load the structure file. First, open the Molecule File Browser from the File → New Molecule menu item. Browse for and Load the file `lysozyme_wb.psf`.
3. Browse for the file `lysozyme_wb_eq.dcd` located in the directory `box/` and click on Load again. You can close the Molecule File Browser window.
4. Go to the TkCon window in VMD, and navigate to the `3.Analysis/a.rmsd/` directory.
5. The script you will use is called `residue rmsd.tcl`. In the TkCon window, type:

source residue_rmsd.tcl

This command itself does not actually perform any calculations. It reads the script `residue_rmsd.tcl` which contains a procedure called `rmsd` residue over time. Calling this procedure will calculate the average RMSD for each residue you select over all the frames in a trajectory. The procedure is called as:

rmsd_residue_over_time <mol> <sel_resid>

where <mol> is the molecule in VMD that you select (if not specified, defaults to the top molecule), and <sel resid> is a list of the residue numbers in that selection.

6. For this example, you will select all the residues in the protein. The list of residue numbers can be obtained by typing in the TkCon window:

set sel_resid [[atomselect top "protein and alpha"] get resid]

The above command will get the residue numbers of all the α -carbons in the protein. The command will create a list of residue numbers in the variable `$sel_resid`.

7. Call the procedure to calculate the RMSD values of all atoms in the newly created selection:

rmsd_residue_over_time top \$sel_resid

You should see the molecule wiggle as each frame gets aligned to the initial structure. At the end of the calculation, you will get a list of the average RMSD per residue. This data is also printed to the file `residue_rmsd.dat`. The procedure also sets the value of the User field of all the atoms of the residues in the selection to the computed RMSD value. You will now color the protein according to this value. You will be able to recognise which residues are free to move more and which ones move less during the equilibration. In order to use these new values to learn about the mobility of the residues, you will create a representation of the protein colored by User value.

8. Open the Representations window using the Graphics → Representations menu item.
9. In the Atom Selection window, type `protein`. Choose `NewCartoon` as Drawing Method, and in the Coloring Method drop-down menu, choose `Trajectory > User`. Now, in the Color Scale Data Range, type `0.40` and `1.00`. Click the Set button. You should now see the protein colored according to average RMSD values. The residues displayed in blue are more mobile while the ones in red move less. This is somewhat counterintuitive, so you will change the color scale.
10. Choose the Graphics → Colors menu item. Choose the Color Scale tab. In the Method pull-down menu, choose for example `GWB` (Figure 7).

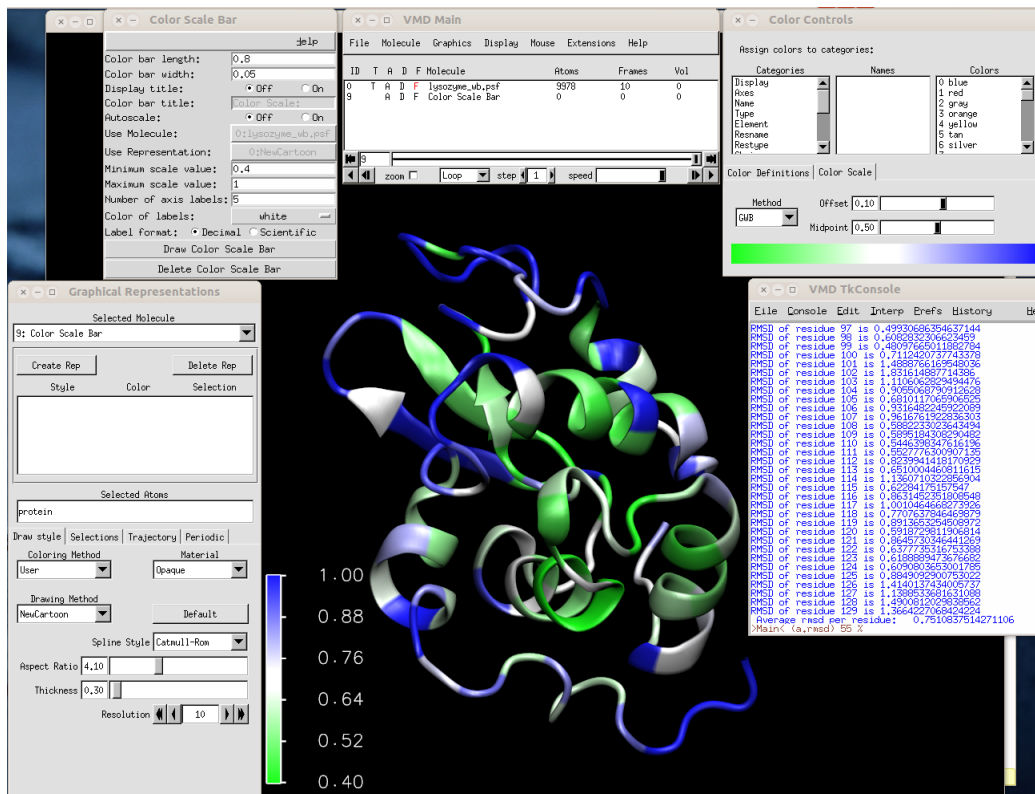


Figure 7: RMSD calculation with VMD.

11. Plot the RMSD value per residue by typing in a Terminal window:

xmgrace residue_rmsd.dat

Take a look at the RMSD distribution (Figure 8). You can see regions where a set of residues shows less mobility. Compare with the structural feature of your protein. You will find a correlation between the location of these regions and secondary structure features like α helices or β sheets. When you are finished investigating the plot, close xmgrace, saving your plot if you wish.

1. Plot the RMSD as a function of time. Is it stable?
2. Visualize the trajectory of your simulation. Which parts of the protein are more flexible ?

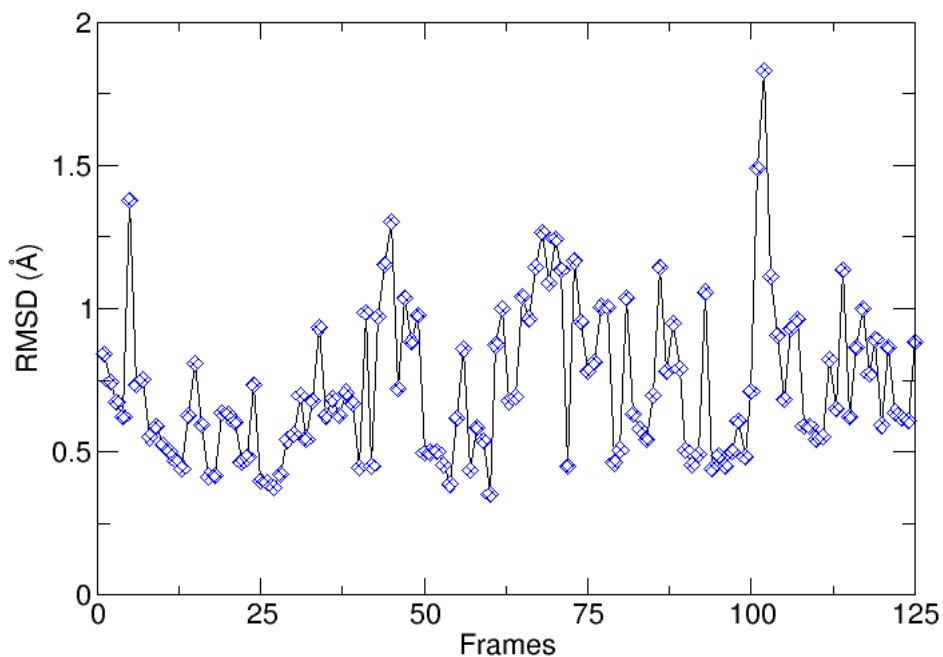


Figure 8: Plotting RMSD in xmgrace.

8 Trajectory analysis using VMD

Trajectory analysis can be performed directly from the VMD interface and several properties of the system can be computed.

Load the psf file of lysozyme and the production trajectory.

8.1 Time series of the hydrogen bonds

- Identify one hydrogen bond in the interior of the protein and select it (hit `?2?` on your keyboard to select two atoms) so that the two heavy atoms are selected (hydrogen bond donor and acceptor heavy atoms).
- Go to the Graphics menu -> Labels -> Bonds
- Select the bond you would like to observe (it should be highlighted in yellow).
- Click on the Graph tab.
- Click on Graph.
- The time series of the selected hydrogen bond is displayed.

- You can save the data and plot in Excel by clicking on the ?Save? button.
- Is the hydrogen bond distance criterion satisfied throughout the trajectory for the bond you have chosen?
- What is the average value and standard deviation of the hydrogen bond distance?
- Distance is one of the hydrogen bond criteria. The other one is that the angle between the electronegative atom, the hydrogen and the acceptor atom has to be 0 (or 180). Plot the time series of the hydrogen bonds angles for the Hbonds that satisfy the distance criterion. Are these angles linear?

8.2 Time series of the bonds, angles, dihedrals

- This exercise is optional.
- One can choose any bond, angle, dihedral or distance that they are interested in and plot the time series. For example, one can monitor the stability of the disulfide bonds by plotting the time series of the S-S distance, the C-S-S angle, and the C-S-S-C dihedral angle ?
- What is the average value and standard deviation for these properties?

8.3 RMSD time series calculation

- In the ?Extensions? menu, choose ?Analysis? and ?RMSD Trajectory Tool?.
- Type ?protein and name CA? in the white box. That will select all the alpha carbons in the protein (backbone).
- Click on ?Align?. (Note that the ?trajectory frame ref: is set to 0).
- Click on ?Plot?.
- Click on ?RMSD?. What is being plotted?
- What conclusions do you draw for the stability of the protein?

8.4 Other VMD trajectory analysis tools

- The ?Extensions? menu provides a lot of other possibilities for Analyses.
- You can try at home finding all the salt bridges in the protein, plotting the Ramachandran plot for a residue, calculating radial distribution functions, $g(r)$, etc.

8.5 Maxwell-Boltzmann Energy Distribution

Objective: Confirm that the kinetic energy distribution of the atoms in a system corresponds to the Maxwell distribution for a given temperature.

1. In a Terminal window, type **cd ../b.maxwell**. For this exercise, you need a file that contains the velocities of every atom for one instant of the simulation. You will use a file corresponding to the last frame of an equilibration of lysozyme. As discussed previously, one should use rigidBonds for water in any production simulation since water molecules have been parametrised as rigid molecules. But for the illustration of Maxwell-Boltzmann energy distribution, the effect of force field can be negligible. To be simple, we are not using the option rigidBonds. Here, we provide you with the file lysozyme_wb_eq_restart.vel. Alternatively, you can use any velocity file, as long as constraints were not used in the simulation.
2. You will now load the structure file. First, open the Molecule File Browser from the File → New Molecule menu item. Browse for and Load the file lysozyme_wb.psf.
3. Now, browse for the file lysozyme_wb_eq_restart.vel in the directory: b.maxwell/Short_MD_without_rigidBonds.

This time you need to select the type of file. In the Determine file type: pull-down menu, choose NAMD Binary Coordinates, and click on Load again. Close the Molecule File Browser. The molecule you loaded looks terrible! That is because VMD is reading the velocities as if they were coordinates. It does not matter how it looks, because what we want is to make VMD write a useful file: one that contains the masses and velocities for every atom.

4. In the VMD TkCon window, navigate to the b.maxwell/ directory by typing **cd b.maxwell**.
5. First, you will create a selection with all the atoms in the system. Do this by typing in the TkCon window:

```
set all [atomselect top all]
```

6. Now, you will open a file energy.dat for writing:

```
set fil [open energy.dat w]
```

7. For each atom in the system, calculate the kinetic energy $\epsilon_k = \frac{1}{2}mv^2$, and write it to a file

```
foreach m [$all get mass] v [$all get {x y z}] { puts $fil [expr 0.5 * $m * [vecdot $v $v] ] }
```

8. Close the file:

close \$fil

9. Outside VMD, take a look at your new energy.dat file. In a Terminal window, make sure you are in the 2-2-maxwell/ directory and type: nedit energy.dat It should look something like the list below (although the values will not necessarily be the same):

```
1.26764386127
0.189726622306
0.268275447408
...
```

10. Quit nedit and VMD.
11. If you chose not to go through the steps above, there is a VMD script that reproduces them and will create the file energy.dat. In a Terminal Window, type:

vmd -dispdev text -e get energy.tcl

This will run VMD in text mode, source the script get energy.tcl, and produce the energy.dat file. When it is finished, type exit to exit VMD. You now have a file of raw data that you can use to fit the obtained energy distribution to the Maxwell-Boltzmann distribution. You can obtain the temperature of the distribution from that fit. In the rest of this section, we will show how to do this with the 2-D graphics package xmgrace. If you are more familiar with another tool, you may use it instead.

12. Quit VMD.
13. In a Terminal window, type xmgrace.
14. Once you see the xmgrace window, choose the Data → Import → ASCII menu item. In the Files box, select the file energy.dat. Click on the OK button. You should see a black trace. This is the raw data. You can now close the Read sets window.
15. To look at the distribution of points, you will make a histogram of this data. Choose the Data → Transformations → Histograms menu item. In the Source → Set window, click on the first line, in order to make a histogram of the data you just loaded.
16. Click on the Normalize option.
17. Fill in Start: 0, Stop at: 10, and # of bins: 100. Click Apply.
18. You have created a plot, but you cannot see it yet. Use the right button of your mouse to click on the first set (in the Source → Set window). Click on Hide. Now, go to the main Grace window and click on the button labeled AS, that will resize the plot to fit the existing values. This is your distribution of energies.

Now, you will fit a Maxwell-Boltzmann distribution to this plot and find out the temperature that the distribution corresponds to. (Hopefully, the temperature you wanted to have in your simulation!)

19. Choose the Data → Transformations → Non-linear curve fitting menu item. In the Source → Set window, click on the last line, which corresponds to the histogram you created.

20. In the Main tab, type in the Formula window:

$$y = (2/\sqrt{\pi}) * (A0^3)^{-1} * \sqrt{x} * \exp(-x/A0)$$

This will fit the curve and get a fit for the parameter A0, which corresponds to $k_B T$ in units of kcal/mol. (These are the energy units that NAMD uses)

21. In the Parameters drop-down menu, choose 1. Fill-in forms will appear below. You can give an initial value to A0, so that the iterations will look for a value in the vicinity of your initial guess. In the A0 window, type 0.6, which in kcal/mol corresponds to a temperature of ~300K. The result are shown in Figure 9

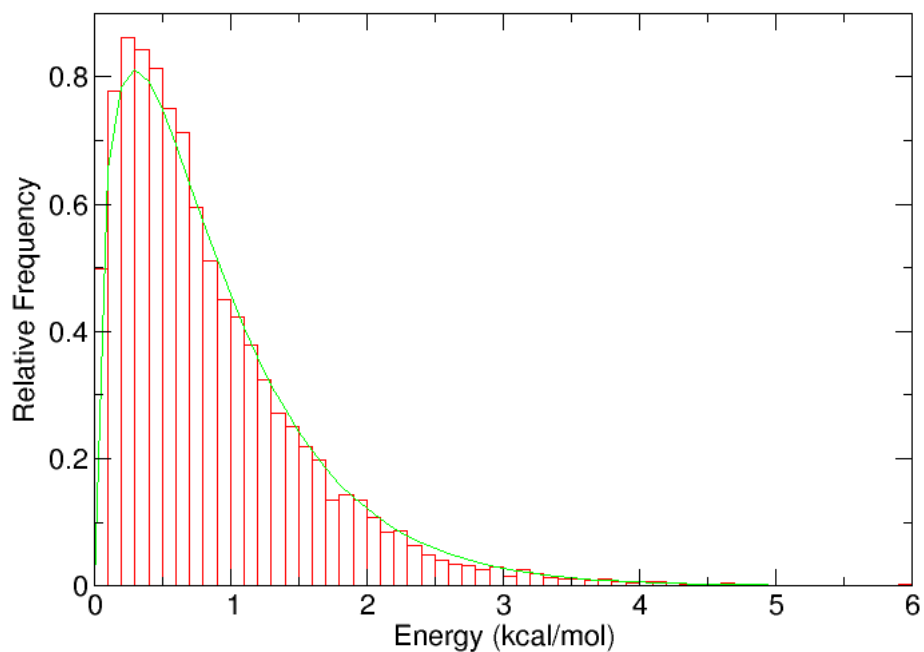


Figure 9: Maxwell-Boltzmann distribution for kinetic energies.

9 Normal Mode Analysis

In this section you will be examining the vibrations of parts of lysozyme using normal mode analysis.

Some facts about Vibrational Modes

http://chemwiki.ucdavis.edu/Physical_Chemistry/Spectroscopy/Vibrational_Spectroscopy/Vibrational_Modes

- A normal mode is a molecular vibration where some or all atoms vibrate together with the same frequency in a defined manner.
- Normal modes are basic vibrations in terms of which any other vibration is derived by superposing suitable modes in the required proportion.
- On the other hand, no normal mode is expressible in terms of any other normal mode. Each one is pure and has no component of any other normal mode (i.e. they are orthogonal to each other).
- The required number of normal modes is equal to the vibrational degree of freedom available so the number of modes for a nonlinear molecule is $3N-6$ and that for a linear molecule is $3N-5$.
- Each mode has a definite frequency of vibration. Sometimes 2 or 3 modes may have the same frequency but that does not change the fact that they are distinct modes; these modes are called degenerate.

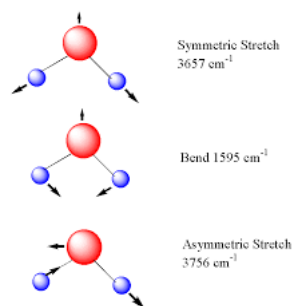


Figure 10: Normal modes of water

In the analysis of protein dynamics, an important goal is the description of slow large-amplitude motions. These motions are inherent in each protein and characterize its function. They describe conformational changes, which are essential for the functioning of proteins. Only global collective motions can significantly change the exposed surface of the protein and hence influence interactions with its environment. Such structural rearrangements in the protein can occur on a local level within a single domain or can involve large movements of protein domains in a multidomain protein. Protein dynamics thus cover a broad timescale: 10^{-14} - 10 sec. However, many large-amplitude conformational changes are

not on a timescale accessible by most time-dependent theoretical methods, such as phase space sampling techniques (e.g., molecular dynamics). Therefore, in order to gain insight into the mechanism of slow, large-amplitude motions, one must resort to the use of a *time-independent approach*, such as **Normal Mode Analysis (NMA)**.

NMA is a fast and simple method to calculate vibrational modes and protein flexibility. In NMA, which is usually restrained to Ca atoms only, the atoms are modeled as point masses connected by springs, which represent the interatomic force fields.

It has also been shown that the low-frequency modes describing the large-scale real-world motions of a protein can be related to fundamental biological characteristics (Brooks and Karplus 1985; Thomas et al. 1999). For example, Bahar and Jernigan (1998) successfully analyzed the vibrational dynamics of transfer RNAs, both free and complexed with the cognate synthetase, using the elastic network model. They examined the global mode of motion of tRNAGln complexed with glutamyl-tRNA synthetase, and established that certain residues that cluster near the ATP binding site form a hinge-bending region controlling the cooperative motion and thereby the catalytic function of the enzyme. Normal modes have been successfully used to display concerted motions of proteins (Noguti and Go 1982; Brooks and Karplus 1983; Go et al. 1983; Levy et al. 1984; Levitt et al. 1985; Henry et al. 1986), including slow motions between protein domains as in the hinge-bending motion of lysozyme (Brooks and Karplus 1985; Gibrat and Go 1990). It was recently shown that the first step of the gating mechanism in the mechanosensitive channel (MscL) can be described with only the three lowest-frequency modes (Valadie et al. 2003). Those results clearly indicate that the movement associated with these modes is an iris-like movement involving both tilts and twists. Several other works showed that low-frequency modes overlap with real conformational changes (Thomas et al. 1999; Tama and Sanejouand 2001). There is also evidence to suggest that proper, symmetric normal mode vibration of binding pockets is crucial to correct biological activity in some proteins (Thomas et al. 1996a,b; Hinsen 1998).

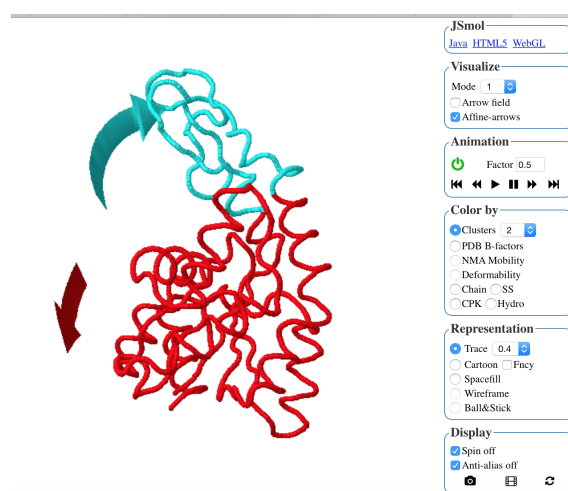


Figure 11: First normal mode of adenosyl kinase as calculated by <http://imods.chaconlab.org>

Using the following site <http://imods.chaconlab.org> and the structure of our protein (1AKI.pdb) we can obtain normal mode analysis results.

Visualize also the examples of Adenosyl kinase, HIV-protease, calcium ATP-ase, nucleosome, GroEL, or any other protein you would like to download from the PDB.

1. Why does one need to perform energy minimization prior to an NMA?
2. Visualize the first 3 modes of lysozyme. What motions do you observe? Could these motions be correlated to the protein function?
3. Visualize mode No. 100. What is the difference from mode No. 1?
4. Why are the low-frequency modes more interesting? Which parts of the protein are involved in these modes?
5. Which parts of the protein move in the high frequency modes?
6. What is the difference between a MD and a NM trajectory?

How to use XmGrace

You will be using XMGRACE on your PC to produce plots of your data. This is a short guide outlining the basics of XMGRACE. First of all, to start XMGRACE, type `xmgrace` in the directory where you want to plot the data.

Reading in data

1. Using the mouse, at the top click on Data, then under file click on Import and follow the arrow with your mouse to ASCII. You should now see a Read sets pop-up window.
2. Inside this window, scroll through the Files section on the right hand side using your mouse. This lists all of the files in the directory you are running XMGRACE in. Highlight the file that contains the data you want to plot.
3. Click on Load as block data,
4. Hit OK at the bottom of the Read sets pop-up window. A new pop-up window, Edit block data, should appear.
5. Inside this window, you will need to select which column you want the x-axis of the plot to be by clicking on X from column: index and changing the value of index to your column number. You also need to select the y-axis in the same way.
6. Hit Accept at the bottom of the Edit block data window. Hit Cancel at the bottom of the Read sets window.
7. Your data should be inside xmgrace ready for plotting !

Simple plotting of data

1. To see your data quickly (if it isn't already plotted), hit the A_5 button, located on the left hand side of xmgrace, below the Draw button. Your data should now be plotted.

Adding axis labels

1. Click at the top on Plot and pull down to Axis properties. An Axes pop-up window should appear.
2. Inside the window, type your x-axis label inside the Axis label: box. If you want to write a superscript, e.g. kcal.mol^{-1} , type `kcal mol\^S-1\N`. If you want to write a subscript, e.g. E_i , type `E\^s i\N`. To get an angstrom symbol, type `\cE\C`.
3. Hit Apply at the bottom of the window.
4. At the top of this window, click on Edit: X axis to change from x-axis to y-axis labeling.
5. Type your y-axis label inside the Axis label: box.
6. Hit Apply, then Close at the bottom of the window.

Adding a graph label

1. Double-click at the top of the graph, but still inside the white part of the xmgrace screen. A Graph Appearance window should appear.
2. Inside the window, type your title into the box marked Title:. You can also add a smaller subtitle in the Subtitle box below.
3. Hit Apply, then Close at the bottom of the window.

Checking on data statistics

1. At the top, click on Edit, then pull down the menu to Data sets. A Data set properties pop-up window should appear. The set are written at the top, you normally should only have one set. Click on the set to highlight it black. The data statistics of this set will be written below. The number of data points is written in Length:. The numbers in min are your lowest-value x-axis and y-axis data points. The points which have these values are next to min in at. The numbers in max are your highest-value x-axis and y-axis data points. Again at to the right of this indicates the points which have these values.
2. To close this window, click on Close in the bottom-left-hand corner.
3. If you want to keep this window open but it's in the way, you can move it around by clicking and holding in the title bar at the top of the window.

To delete a graph without leaving XmGrace

1. Click on Plot, then pull down to Set appearance. A Set Appearance pop-up window should appear, select the data set you want to delete. This is described above.
2. To kill completely: in the Data sets: box, click and hold with the right mouse button and slide down to Kill data.
3. To hide: follow the steps above except release the mouse on Hide.
4. To show: follow the steps above except release the mouse on Show.

Printing graphs

1. At the top of the window pull down the File menu to Print setup. A Device setup window should appear.
2. Type in the box for Print command, lpr for black and white printing. For colour, type in lpr -Pdcolor.
3. Click on Size: and change the paper size from Letter to A4.
4. Hit Accept.
5. Again pull down the File menu to Print. The file will now be printed !

Manipulating data sets

For example: use this if you want to convert your energy data from kcal.mol^{-1} to kJ.mol^{-1} , or convert trajectory time steps into simulation time.

1. Click on the Data menu and drag the menu down to Transformations. Then drag horizontally from Transformations to Evaluate expression. A pop-up window, Evaluate expression should appear.
2. The boxes Set: contains all sets that are available. Click on the sets you want to alter, if they aren't already highlighted black.
3. In the Formula: box, type in the equation for transforming the data.
For example: if you want to convert from kcal.mol^{-1} to kJ.mol^{-1} type: $y=y*4.184$. Your energy data should be in the y-axis, and you are multiplying the y-axis data by the conversion factor of 4.184.
For example: if you want to convert from trajectory time steps to simulation time, type: $x=x*0.1$. Your time step data should be in the x-axis, and you are multiplying the x-axis data by the trajectory time step in ps. This will give the simulation time in ps.
4. Hit Accept at the bottom of the window when you are finished.
5. You may need to hit autoscale A_S or rescale your axes manually to see your transformed data.

And finally...

To exit XMGRACE, click on Exit, then OK in the following pop-up window !