CHAPTER 3

COMPUTATIONAL METHODS

COMPUTATIONAL METHODS

3.1 Molecular dynamics (MD) simulations

MD simulations is linked between theory and experiments. MD simulation exposes the invisible microscopic details including atomic positions, bonded and non-bonded interactions, velocities etc. And the microscopic information can be converted into macroscopic observables such as pressure, energy, heat capacities, etc., and these details are essential to study the biological systems. In MD simulations, the physical motions of atoms in the protein molecule present in the actual environment is mimicked wherein atoms are allowed to interact for a certain period of time and that molecular interactions are generally studied in detail. And the information about individual motion of atoms as a function of time can be obtained in detail by simulation [116]. To determine the internal motion of proteins, the role of solvent is very important in simulation [117] at different temperatures, particularly below the glass transition temperature, as it may be sometimes experimentally difficult to capture the dynamics of the internal motion of proteins [118]. MD simulations provide connection between structure and dynamics by aiding the study of the conformational energy landscape accessible to protein molecules [119]. In recent years MD simulation packages such as NAMD [120], GROMACS [121], and AMBER [122], have been significantly improved their algorithmic sophistication and parallel performance, and able to perform up to ~10-100 ns/day/workstation/cluster [123].

MD simulation delivers an alternate approach in order to study and understand the protein dynamics at NMR relaxation time scales to calculate order parameter [124] and residual dipolar coupling [125] of proteins. Residual dipolar coupling gives information about the relative orientation of the protein's portions that are present far away in the structure. NMR spectroscopy aids the measurement of ordered parameters that provides an atomistic depiction of fluctuations in protein structure over pico and nanoseconds [126]. Contrast between NMR spectroscopy and MD simulations can be helpful to understand the experimental results [127] as well as to improve the quality of force fields related to simulation and integration methods

[128]. In the recent past, great number of ingenious alternative approaches to classical MD simulations have been developed such as Monte-Carlo sampling of conformational space [129], steered MD [130], hybrid Quantum Mechanics/ Molecular Mechanics (QM/MM) 131,132], coarse-grained dynamics [133], Brownian dynamics [134, 135], normal vibration modes analysis [136,137], and molecular docking simulations [138], all are important to the spectacular applications and developments in biomolecular simulation.

Additionally, in MD simulation, one can explore the macroscopic properties of a system through microscopic simulations, for example, to calculate changes in the binding free energy of a drug candidate, or to examine the energetics and mechanisms of conformational change. MD simulations provide the means to solve the equation of motion of the particles and evaluate these mathematical formulas. With MD simulations, one can study both thermodynamic properties and/or time dependent (kinetic) phenomenon [139-141].

3.1.1. Historical Background

The MD method was first presented by Alder and Wainwright in the late 1950's [142,143] to study the interactions of hard spheres. From their studies many important information regarding the behavior of simple liquids emerged. In 1964, Rahman carried out the first simulation using a realistic potential for liquid argon [144]. In 1974, Stillinger and Rahman, for the first time carried MD simulation of a realistic system of liquid water [145]. In 1977, Mc Cammon, et al, did the first protein simulations for bovine pancreatic trypsin inhibitor (BPTI) [146]. Due to the innovatory advances in computer technology and algorithmic improvements, today we can see the number of simulation techniques has greatly expanded, to study the solvated proteins [147], protein-DNA complexes as well as lipid systems addressing a variety of issues including the thermodynamics of ligand binding and the folding of small proteins. Molecular dynamics simulation techniques are also widely used in experimental procedures such as X-ray crystallography and NMR structure determination [148].

3.1.2 Theory of molecular dynamics simulation

The principle of MD simulation lies on integrating Newton's law of motion for a system of interacting particles with mass 'm' and initial positions and velocities with an accurate description of the potential energy as a function of the atomic coordinates. It generates the positions and velocities of the particles in the system that varies with time in phase space and specified as trajectories. These trajectories provide the average values of physical and chemical properties of the particle which describes how positions and velocities of the atoms change with time. This is a deterministic method. By solving the differential equation of Newton's second law, the trajectory is attained

$$\vec{F} = ma \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 3.1$$

$$F = -\frac{d}{dr}\mu \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 3.2$$

The forces F is acting on the particles with mass of the particles = m and acceleration of the particle = a. And these are derived from the potential energy $\mu(r^{N})$, where $r^{N} = (r1, r2 \ldots r\,N)$ represents the complete set of $3N$ atomic coordinates.

The purpose of the numerical integration of Newton's equation of motion is to find an expression that defines position r_i (t+Δt) at time t+Δt in terms of the already known positions at time t. In MD simulation, to calculate the trajectories of particles, Verlet algorithm is frequently used because of its simplicity, time-reversibility and numerical stability. The basic formula of this algorithm use Taylor series expansions of the positions and dynamic properties.

A variation on the Verlet algorithm is the leap-frog algorithm [149] where velocities can be calculated from the positions or propagated explicitly.

The leap-frog algorithm use velocities at half time step:

$$\dot{r}_i\left(t+\frac{\Delta t}{2}\right) = \dot{r}_i\left(t-\frac{\Delta t}{2}\right) + \ddot{r}_i(t)\Delta t \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ 3.3$$

The velocities at time t can be also computed from:

$$\dot{r_i}(t) = \frac{\dot{r}_\iota\left(t+\frac{\Delta t}{2}\right)+\dot{r}_\iota(t-\frac{\Delta t}{2})}{2} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ 3.4$$

This is useful when the kinetic energy is needed at time *t*, as for example in the case where velocity rescaling must be carried out. The atomic positions are then obtained from:

$$r_i\ (\mathrm{t}+\Delta\mathrm{t}) = r_i\ (\mathrm{t}) + \dot{r_i}(t+\frac{\Delta t}{2})\ \Delta\mathrm{t} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ 3.5$$

The leap frog algorithm is computationally less expensive and requires less storage. This could be an important advantage for large scale calculations. Moreover, the conservation of energy is respected, even at large time steps. Therefore, the computation time could be greatly decreased when this algorithm is used. However, when more accurate velocities and positions are needed, another algorithm should be implemented, like the Predictor-Corrector algorithm.

The molecular trajectory theoretically imitates the motion of the real system. If the potential energy function is a good approximation of the real interactions between the particles, this can give a detailed description of both the dynamics as well as equilibrium properties of the system under consideration. The functional form of the potential energy function together with the set of interaction parameters used is called a *force field*.

3.1.3 Force field

Force fields provide information about the potential energy of a system of particles. From experimental and quantum mechanical studies of small molecules, force field parameters are obtained, and it is suggested that such parameters may be transferred to desired larger molecules. Force field function includes bonded and non-bonded interaction terms. Bonded interactions consists of harmonic oscillator energy of bond lengths, bond angles, and sometimes improper dihedrals (hard terms) and torsional dihedral angles (soft terms, sometimes including improper dihedrals). Non-bonded interactions contribute van der Waals interactions and electrostatic interactions. van der Waals interactions are described by a Lennard-Jones [150-153] potentials, that includes only dispersion or London interactions between transient dipoles. Electrostatic interactions are described by Coulomb potentials. Numerous different force fields have been developed by different research groups those are AMBER03 [154], AMBER94 [155], AMBER96 [156], CHARMM27 [157], OPLS-AA [158], GROMOS87 [159], GROMOS96 [160], General Amber force field (GAFF) [161]. The typical functional form of a force field is:

$$V(r^N) = \sum_{bonds} \frac{k_i}{2}\left(l_i - l_{i,o}\right)^2 + \sum_{angles} \frac{k_i}{2}\left(\theta_i - \theta_{i,o}\right)^2 + \sum_{torsions} \frac{V_n}{2}\left(1+\cos(n\emptyset - \emptyset_o)\right) + \sum_{i=1}^{N}\sum_{j=i+1}^{N}\left(4\varepsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right] + \frac{q_i q_j}{4\pi\epsilon_o\varepsilon_r r_{ij}}\right) \qquad 3.6$$

Where,

V (r^N) : potential energy as a function of the positions (r) of N atoms;

k_i : force constant;

l ,l_0 : current and reference bond lengths;

θ, θ_0 : current and reference valence angle:

V_n : barrier height of rotation;

Ø : torsion angle;

n : multiplicity that determines the number of energy minima during a full rotation;

σ_{ij} : collision diameter for the interaction between two atoms *i* and *j*;

ε_{ij} : well depth of the Lennard-Jones potential for the *i*-*j* interaction;

q_i, *qj* : partial atomic charges on the atoms *i* and *j*;

r_{ij} : current distance between the atoms *i* and *j*;

ε_0, ε_r : permittivity of the vacuum and relative permittivity of the environment respectively;

$Ø_0$: phase factor that determines where the torsion angle passes through its energy minima.

The types of interactions that is the representative of potential energy functions are schematically presented in **Figure 3.1.**

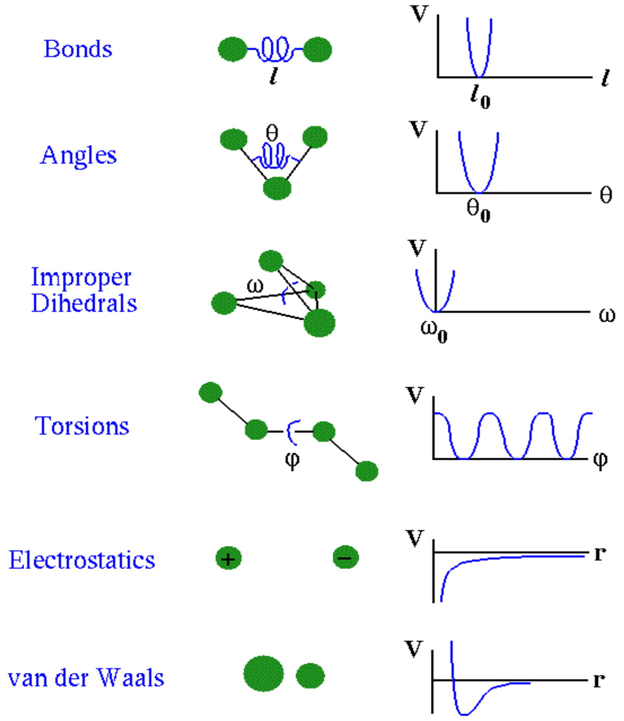


Figure 3.1. *Schematic representation of the interactions that contribute to the potential energy function for MD simulation. Taken from* [162]

The first term in the equation represents the bond stretching between pairs of covalently bonded atoms. The second term describes the contribution of each angle. Angle bending due to vibrational motions requires less energy to distort an angle from its equilibrium value. The third term models the torsion angle. It shows how energy is changed due to the rotation around a bond. The fourth term of the equation models the contribution of non-bonded interactions using a Lennard-Jones potential for van der Waals interaction and a Coulomb potential for electrostatic interactions.

3.1.4. Periodic boundary conditions

The size of the model systems consists of a small number of particles compared to real macroscopic systems. Many atoms experience a large boundary surface to a vacuum environment while simulating which is irrelevant to study phenomena taking place in bulk. Periodic boundary conditions make it possible to small particles to experience forces if they are in a bulk solution [163]. The atoms are placed in a simulation box that is surrounded by translated copies of the coordinates of the atom as shown in **Figure 3.2**. A periodic 3-dimensional array surrounds the inner cell. If an atom crosses the boundary it is replaced by an image atom that enters from the opposite side with unchanged velocity. Thus, the number of particles within the central box remains constant. A non-bonded cutoff is used to deal with the non-bonded interactions such that each atom interacts with only one image of every other atom in the system.

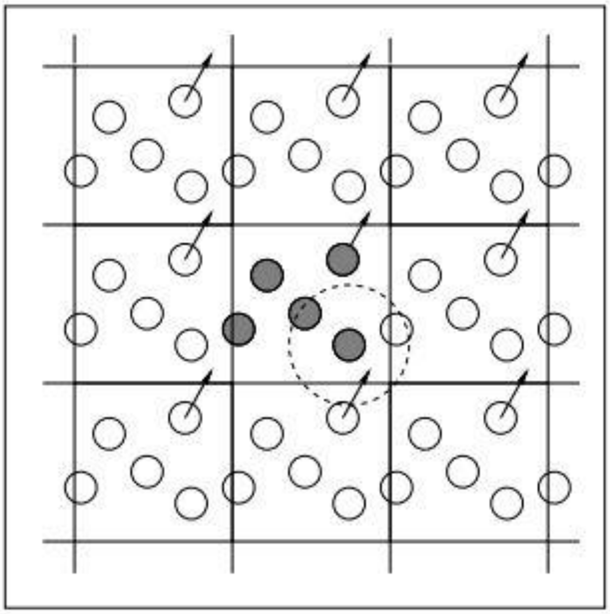


Figure 3.2. *Periodic boundary conditions in two dimensions. The simulation cell (solid) is surrounded by translated copies of itself (dashed).*

3.1.5. Long range interactions Ewald sum

Ewald summation [164] is one of the most commonly used techniques to treat long range interaction in periodic system and it is the most correct way yet devised to accurately include all the effects of long-range forces in a computer simulation. In this method, a particle interacts with all the other particles in the simulation box and with all of their images in an infinite array of periodic cells. Main idea of Ewald sum is to consider a charge distribution of opposite sign charge site; this charge distribution screens the interaction between neighboring atoms.

This method is efficiently used to calculate the infinite range Coulomb interaction under periodic boundary condition (PBC). And Particle Mesh Ewald (PME) is a modification to accelerate the Ewald reciprocal sum to near linear scaling, using the three dimensional fast Fourier transform (3DFFT). Because the coulombic interaction has infinite range, under PBC particle i within the unit cell interact electrostatically with all other particle j within the cell, as well as with all the periodic image of j, it also interacts with all of its own periodic images. The total Coulomb energy of a system of N particles in a cubic box of size L and their infinite replicas in PBC is given by

$$U = \frac{1}{2}\sum_{n}^{N}\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{q_i\, q_j}{r_{ij,n}} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 3.7$$

Ewald recast the potential energy of Eq. (3.7), a single slowly and conditionally convergent series, into the sum of two rapidly converging series plus a constant term,

$$U_{\text{Ewald}} = U^{r} + U^{m} + U^{o} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 3.8$$

The Ewald sum is therefore written as the sum of these three parts, namely, the real (direct) space sum (U^{r}), the reciprocal (imaginary, or Fourier) sum (U^{m}), and the constant term *(*U°*)*, known as the self-term.

Ewald sum has been extensively used in simulations which involves highly charged system (such as ionic melts and in studies of processes in and on solids) and is increasingly being applied to other systems where electrostatic effects are essential, such as lipid bilayers, proteins and DNA.

3.1.6. SHAKE algorithm

In a molecular system, the choice of time step is limited due to the various time scales associated with vibrational degrees of freedom such as bond vibration, angle stretching or torsional mode. Generally, the bonds involving hydrogen atoms have the fastest vibrational mode and they limit the time step of integration to 1 fs. In order to use a larger time step one can restrain these fast degrees of freedoms while solving the un-constrained degrees of freedom. Bonds involving hydrogen have highest frequency hence they can be constrained during dynamics using the SHAKE algorithm which was introduced by Ryckaert *et al* [165].

Basic idea of SHAKE is to use Lagrange multiplier formalism to enforce bonds distances constant. Suppose we have *N*c such constrained given by

$\propto_k = r^2{}_{k1\,k2} - R^2{}_{k1\,k2} = 0,$ where *k* = 1, 2, 3……*N*c..................... 3.9

R_{k1k2} being constrained distant between atoms k_1 and k_2 atoms. This leads to modified constrained equation of motion

$m_i \frac{d^2 r_i\ (t)}{dt^2} = -\frac{\partial}{\partial_{ri}}\left[V\ (r_1\ \dots\dots\ r_N) + \sum_{k=1}^{N_c} \tau_k\ (t)\alpha_k(r_1 \dots .\, r_N)\right]$.............3.10

Where m_i is mass of i^{th} particle and τ_k is the Lagrange multiplier (unknown) for k^{th} constraint. This equation can be solved for unknown multiplier by solving N_c quadratic coupled equations. And we get the following equation of motion:

$r_{k1}(t + \Delta t) = r_{k1}^{uc}(t + \Delta t) - 2(\Delta t)^2 m_{k1}^{-1} \tau_k\ (t) r_{k1k2}(t)$............................. 3.11

Where r_{uc} is position updates with unconstrained force only. This procedure is repeated till defined tolerance is given.

3.1.7. Temperature and pressure computation and control

The initial temperature of the system is computed by coupling to a Berendsen thermal bath [166]. The bath supply or remove heat from the system as appropriate, thereby acts as a source of thermal energy. The system temperature T (t) that deviates from the bath temperature T_0 is corrected giving to:

$\frac{dT(t)}{dt} = \frac{1}{\tau}\{To - T(t)\}$.. 3.12

Where τ (time constant) defines the strength of the coupling between the bath and the system. By scaling the atom velocities at each step the temperature of the system is corrected by a factor χ, given by:

$\chi = [1 + \frac{\Delta \mathrm{t}}{\tau_T}\ (\frac{To}{T(t)} - 1)]$..3.13

By changing the time constant τ the strength of the coupling can be varied.

The method used for pressure control is similar to that of temperature control. The system can be coupled to a barostat and the pressure can be maintained at a constant value by periodic scaling of the simulation cell size and atomic positions with a factor μ:

$$\mu = 1 - \omega \frac{\Delta t}{\tau_p} (P-P_0) \quad \text{3.14}$$

where ω represents the isothermal compressibility, τ_p represents the relaxation constant, P_0 is the pressure of the barostat, P, the momentary pressure at time t and Δt is the time step. The standard simulation package AMBER12 is used in the present work [122]. *Pmemd*, one of the AMBER modules carries out the molecular dynamics simulation.

3.1.8. Water molecule models

In MD simulation many molecular water models TIP3P, TIP4P [167], TIP5P [168], simple point charge (SPC/E) [169] model have been proposed for describing water. These models can be categorized according to the number of sites, the structure (rigid or flexible), and the polarization effects. The 3-site models are the most popular one to be used in MD simulations because of the simplicity, reasonable structural and thermodynamic descriptions and computational efficiency. These kinds of models have three interaction sites which correspond to the three atoms of the water molecule. Each atom gets assigned a point charge. Only the oxygen atom has Lennard Jones parameters for interaction. Some of the popular 3-site models include transferable intermolecular potential three-point (TIP3P) model, simple point charge (SPC) model, extended simple point charge (SPC/E) model, etc. [170]. Most of these models use a rigid geometry matching the known geometry of the water molecule. The simulations in this thesis are carried out using TIP3P water model. The TIP3P water model used here is specified with the O-H bond length (*rOH*) and H-O-H bond angle (*θHOH*) to be 0.9572 Å and 104.52° respectively which are equal to experimental gas-phase values. The simple model for TIP3P water is shown in **Figure 3.3**

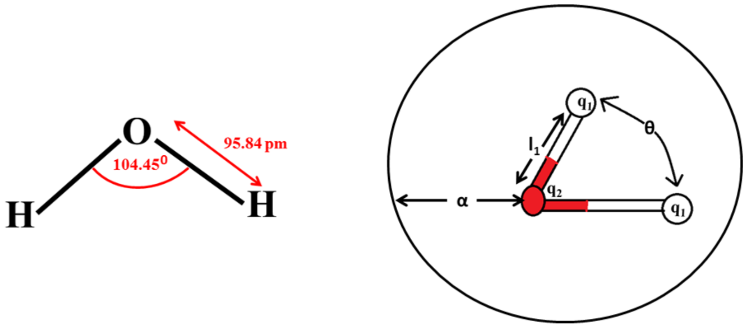


Figure 3.3. *Schematic representation of TIP3P water model.*

3.2. Simulation Methodology in AMBER

The multiple steps involved in setting up and running MD simulation are shown in **Figure 3.4.** MD simulation starts with the knowledge of the potential energy of the system with respect to its position coordinates. The first derivative of the potential function to the position coordinates helps in computing the force acting on individual atoms of the system. The important steps involved in the MD simulations of proteins are as follows.

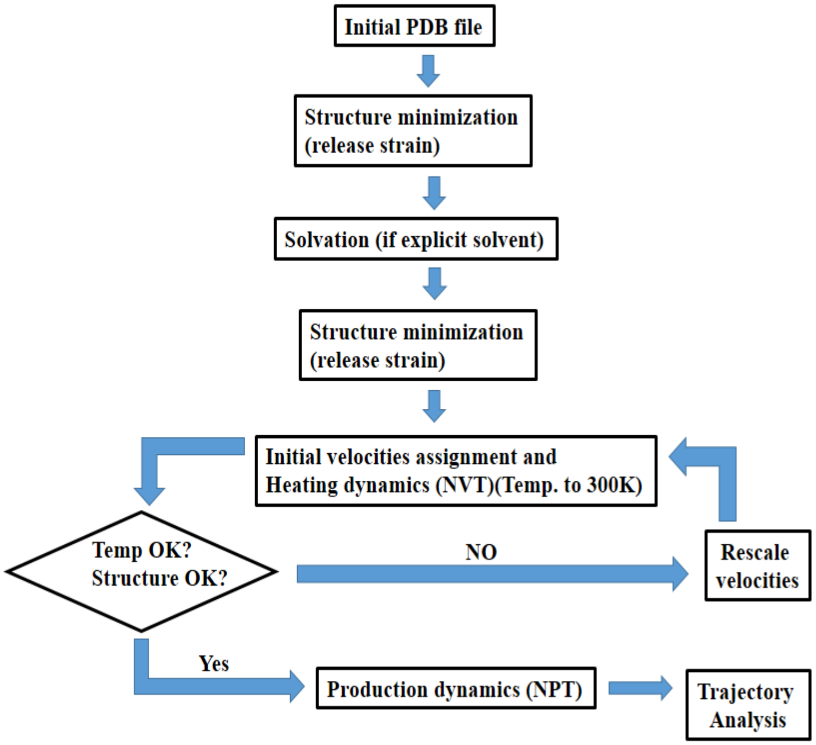

Initial PDB file
Structure minimization (release strain)
Solvation (if explicit solvent)
Structure minimization (release strain)
Initial velocities assignment and Heating dynamics (NVT)(Temp. to 300K)
Temp OK? Structure OK?
NO
Rescale velocities
Yes
Production dynamics (NPT)
Trajectory Analysis

Figure 3.4. *Flowchart showing the steps involved in MD Simulation.*

3.2.1. Simulation environment

To mimic the experimental conditions protein simulation is done, therefore numerous parameters for the different physical conditions are considered (such as pressure, temperature). Generally the protein simulation is done in canonical ensemble (NVT**)** [171], particularly up to the initial equilibration steps, after equilibration, production dynamics is generally carried out in isothermal-isobaric (NPT) [172] ensemble. The canonical ensemble (NVT) is the collection of all systems whose thermodynamic state is characterized by a fixed number of atoms, *N*, fixed volume, *V*, and fixed temperature, *T*. The isobaric-isothermal ensemble (NPT): An ensemble with a fixed number of atoms, *N*, fixed pressure, *P*, and fixed temperature, *T*.

In order to run MD simulation, the protein molecules should be kept in the unit cell and solvated with explicit solvent. We used TIP3P water model in our simulation. Water models are essential to mimic the specific nature and complexity of hydration of molecule, including orientation of solvent dipoles and effective electrostatic shielding, subtle hydrogen bond network rearrangements, and accompanying changes in entropy. Unfortunately, due to limitation in the time resolution of MD simulations and the complicated quantum nature of hydrogen bonds, it is difficult for simulation environments to treat them explicitly, thus SHAKE algorithm is used for solvent hydrogen repositioning. Conversely, when we use implicit solvent models it search to approximate the solute potential of the mean force, which governs the statistical weight of solute conformations, and is obtained by averaging over the solvent degrees of freedom [173]. Prior to MD simulation the total charge of the system should be kept zero by adding positive or negative ions accordingly to avoid polarization of the simulation ensemble. In addition to avoid interaction problems at system boundary, constrained spherical boundary models for solute and solvent may be considered or the highly popular approach of cubic or rectangular PBC can be applied. The Ewald summation has been directly applied in standard solvated periodic boundary simulations of biomolecular systems to compute the electrostatic interaction in the system [174].

3.2.2 Energy minimization

Energy minimization implicates finding the global minimum energy with respect to the position of side chains atoms that represents the geometry of particular arrangements of atoms in which the net attractive force on each atom reaches a maximum. It is a numerical procedure in which using the initial structure at higher energy, minimum is traced out on the potential energy surface [175], for instance labeled "1" as illustrated in **Figure 3.5**. During energy minimization, the geometry is changed in a stepwise fashion so that the energy of the molecule is reduced, from steps 2 to 3 to 4 as shown in **Figure 3.5**. After a number of steps, a local or global minimum on the potential energy surface is reached.

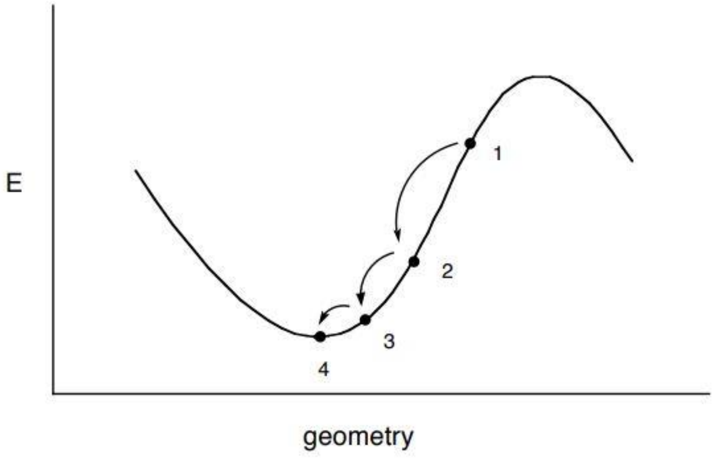


Figure 3.5. *The process of energy minimization changes the geometry of the molecule in a step-wise fashion until a minimum is reached. Taken from* [175]

It is necessary to perform energy minimization of the structure in order to remove the bad contacts, which may otherwise lead to structural distortion. There are many methods to compute the minimum energy but most commonly used methods are steepest descent and conjugate gradient.

***(i) The Steepest Descents Method*:**

Steepest descent method [176] is one of several first-order iterative descent methods and utilizes the gradient of the potential energy surface. It directly relates to the forces in the Molecular mechanical description of molecular systems, to guide a search path toward the nearest energy minimum. It moves in the direction parallel to the net force. For 3N Cartesian coordinates this direction is most conveniently represented by a 3N dimensional unit vector, $\boldsymbol{s_k}$ Thus:

$$\boldsymbol{s}_k = -\boldsymbol{g}_k/|\boldsymbol{g}_k| \text{ .... 3.15}$$

Having defined the direction along which to move it is then necessary to decide how far to move along the gradient. The gradient direction from the starting point is along the line indicated if we imagine a cross-section through the surface along the line; the function will pass through a minimum and then increase, as shown in the **Figure 3.6**. We can choose to locate the minimum point by performing a line search or we can take a step of arbitrary size along the direction of the force [176].

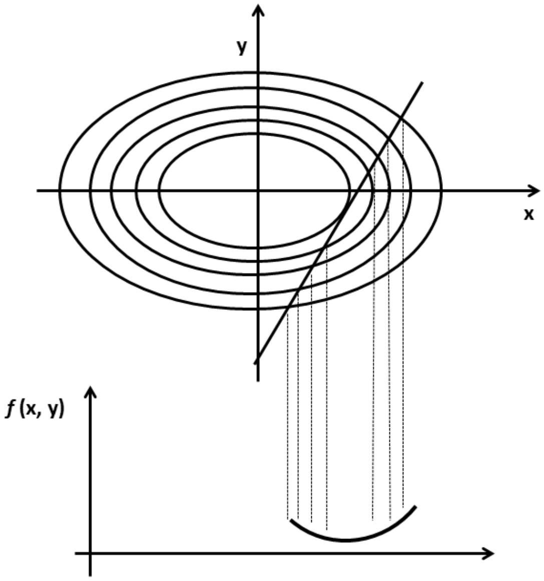


Figure 3.6. *A line search is used to locate the minimum in the function in the direction of the gradient.*

(ii) Conjugate Gradients Minimization:

The conjugate method produces a set of directions which does not show the oscillatory behavior of the steepest descents method in narrow valleys. In conjugate gradients, the gradients at each point are orthogonal but the directions are conjugate. A set of conjugate directions has the property that for a quadratic function of M variables, the minimum will be reached in M steps. The conjugate gradients method moves in a direction $\boldsymbol{v}_k$ from point $\boldsymbol{x}_k$ where $\boldsymbol{v}_k$ is computed from the gradient at the point and the previous direction vector v_{k-1} [177].

$$v_k = -g_k + \gamma_k v_{k-1}$$

(iii) Newton-Raphson Method:

The Newton-Raphson method [175] uses the second derivatives as well as the first derivatives. In addition to using the gradient information, it uses the curvature to predict where along the gradient of the function will change direction. It is the most computationally expensive method utilized to perform energy minimization. Since the complete second-derivative matrix defines the curvature in each gradient direction, we can multiply the inverse of the second-derivative matrix by the gradient to obtain a vector that will translate us directly to the nearest minimum. This is expressed mathematically as:

$$r_{min} = r_o - A_o{}^{-1.\nabla V(r_o)} \text{ .... } 3.16$$

where r_{min} is the predicted minimum, r_o is an arbitrary starting point, A_o is the matrix of second partial derivatives of the energy with respect to the coordinates at r_o (also known as the Hessian matrix), and $\nabla V(r_o)$ is the gradient of the potential energy at r_o.

Prior to minimization, water molecules are added to solvate the system if required. A suitable large box of water that has already been equilibrated is used for solvation purpose. The system is entirely covered by the water box and those water molecules that overlap the proteins are removed. At this point energy minimization should be done with the protein fixed in its energy minimized position. This allows the water molecules to readjust to the protein molecule.

3.2.3 Heating the system

During heating phase, initial velocities (at 0 K) are allocated to each atom of the system and Newton's equations of motion are numerically integrated that represent the time evolution of system. At short predefined intervals, new velocities are allocated corresponding to a slightly higher temperature and the simulation is allowed to continue until desired temperature is achieved (that is 300 K). Force constrains on different subdomains of the simulation system are gradually removed as structural tensions dissipate by heating. Heating dynamics is usually carried out at constant volume (NVT).

3.2.4 Equilibration

Equilibration phase is used to equilibrate kinetic and potential energies, that is, to distribute the kinetic energy "pumped" into the system during heating among all degrees of freedom. This usually infers that the kinetic energy must be transferred to potential energy. As soon as potential energy levels off, the equilibration stage is completed. Generally the system is equilibrated on a timescale much shorter than 300 ps. During explicit solvent simulation, protein positions are fixed and waters moves accordingly. Once the solvent is equilibrated, the restraints on the protein can be removed and the whole system (protein + solvent) can evolve in time.

3.2.5 Production phase

Production phase is the last step of MD simulation. It is performed for desired time scale to generate trajectory of protein molecule in compliance with particular equilibrium conditions (NPT). In production phase of MD simulation thermodynamic parameters can be calculated. The timescale can be varied from several hundred picoseconds to microseconds or more.

3.2.6 Analysis

In this step, stored coordinates and velocities of the system are used for further analysis. MD trajectory files are required for analysis. MD simulations can help to visualize and understand conformational changes at an atomic level when combined with visualization software (e.g VMD) which can display the structural parameters of interest in a time dependent way. Using cpptraj or ptraj module of AMBER12, quantities like time average structure, Root Mean Square Deviation (RMSD) difference between two structures, Root Mean Square Fluctuation (RMSF), Radius of Gyration (Rg), Secondary

Structure Analysis: The secondary structure using DSSP algorithm, Total energy of system can be calculated

(i) Time average structure: This particular structure is obtained by considering coordinate frames which have been averaged over sliding time windows of a certain size.

(ii) RMSD: The deviation of a structure with respect to a particular conformation is measured by RMSD. It is defined as:

$$\mathrm{RMSD} = \left(\frac{\sum_N \left(R_i - {R_i}^0\right)^2}{N}\right)^{1/2} \text{..} \; 3.17$$

where N is the total number of atoms/residues considered in the calculation, and Ri is for the vector position of particle i (target atom) in the snapshot, R_i^0 is the coordinate vector for reference atom i. RMSD was computed based on backbone atoms and taking the first frame of the simulation as the reference.

(ii) RMSF: It is useful for characterizing local changes along the protein chain. It is calculated as:

$$\mathrm{RMSF} = \left(\frac{1}{T}\sum_{t=1}^{T}\left(r_i(t) - {r_i}^{ref}\right)^2\right)^{1/2} \text{......................................} \; 3.18$$

T is the trajectory time over which the average is taken, r_i (t) is the position of the atoms in residue i and r_i^{ref} is the reference position of particle i.

(iii) Rg: It calculates the distribution of the components of an object around the axis. It gives the compactness of a protein. It is calculated as:

$$\mathrm{Rg} = \left(\frac{1}{N}\sum_i \left(r_i - r_{cm}\right)^2\right)^{1/2} \text{...} 3.19$$

where r_i - r_{cm} is the distance between atom i and the center of mass of the molecule.

(iv) Secondary Structure Analysis: The secondary structure content of each protein was calculated using DSSP algorithm which recognizes cooperative secondary structures as repeats of the elementary hydrogen-bonding patterns “turn” and “bridge.” Repeating turns are “helices,” repeating bridges are “ladders,” connected ladders are “sheets. We consider that a secondary structure element is stable at a given position of the protein if it is the predominant in more than 50% of the collected snapshots [178].

3.3. 3-D structure visualization tools

3.3.1 Visual molecular dynamics (VMD): VMD is a molecular modelling and visualization computer program [179]. VMD is mainly a tool to view and analyze the results of MD simulations. It also includes tools for working with volumetric data, sequence data, and arbitrary graphics objects.

3.3.2 UCSF Chimera: UCSF Chimera is a highly extensible program for interactive visualization and analysis of molecular structures and related data, including density maps, supramolecular assemblies, sequence alignments, docking results, and conformational ensembles [180]. Chimera is developed by the Resource for Biocomputing, Visualization, and Informatics (RBVI), supported in part by the National Institutes of Health.

3.4. Potential of mean force

The potential of mean force (PMF) [181] is a concept of the free energy changes as a function of some inter or intramolecular coordinates of molecular systems which may be the distance between two atoms, or the torsion angle of a bond within a molecule. The PMF incorporates solvent effects along with the intrinsic interaction between the two particles when the system is in a solvent. The transition state for the process is related to the point of highest energy on the free energy profile, from which rate constant can be derived. There exist various methods to calculate the PMF. The simplest type of PMF is the free energy change when the separation (r) between the particles is changed [182]. PMF can be calculated from the radial distribution function using the expression for the Helmholtz free energy:

$$\mathrm{A}\ (\mathrm{r}) = -k_B\mathrm{T}\ \ln \mathrm{g}\ (r) + \text{constant} \qquad \text{.................................. } 3.20$$

The PMF may vary by several multiples of k_BT over the relevant range of the parameter r. The logarithmic relationship between the PMF and radial distribution function means that a relatively small change in the free energy may correspond to g(r) changing by an order of magnitude from its most likely value. Unfortunately, MD simulation method does not sufficiently sample regions where the radial distribution function differs drastically from the most likely value, leading to inaccurate values from the PMF. To avoid this problem one of the most widely used sampling techniques is the *umbrella sampling (US)*.

3.4.1 Umbrella sampling

To calculate potentials of mean force (at least for simple distance, angle, or torsion variables) umbrella sampling [183] is used to overcome the sampling problem by restraining a system to a specific region of its conformational space by modifying the potential function so that the unfavorable states are sampled appropriately. The modification of the potential function can be written as:

$$\vartheta' \left(r^N\right) = \vartheta \left(r^N\right) + W \left(r^N\right) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \; 3.21$$

Where W (r^N) is a weighting function, which takes a quadratic form:

$$W \left(r^N\right) = k_W \left(r^N - {r_0}^N\right)^2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \; 3.22$$

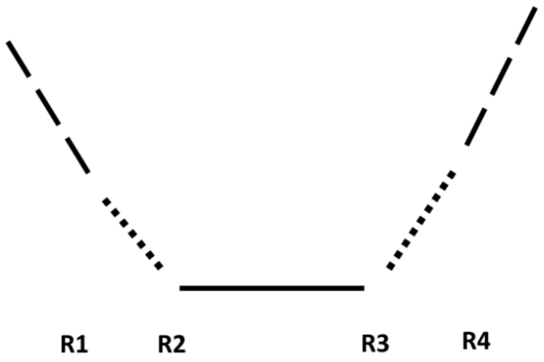
For configurations that are far from equilibrium state $r_0^N$the weighting function will be large and so a simulation using the modified energy function $\vartheta'(r^N)$ will be biased along some relevant 'reaction coordinate' (RC) away from the configuration r_0^N. The resulting distribution will, of course, be non-Boltzmann. The corresponding Boltzmann averages can be extracted from the non-Boltzmann distribution using a method introduced by Torrie and Valleau [183]. The result is:

$$< A > = \frac{< A \left(r^N\right)\exp[\, +W\frac{r^N}{k_B\,T}]>_W}{<\exp[+\frac{W(r^N)}{k_B\,T}]>_W} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 3.23$$

The subscript W indicates that the average is based on the probability $P_W(r^N)$, which in turn is determined by the modified energy function $\vartheta'(r^N)$. It is usual to perform an umbrella sampling calculation in a series of stages, each of which is characterized by a particular value of the coordinate and an appropriate value of the forcing potential W (r^N). However, if the forcing potential is too large, the denominator in eqn 3.23 is dominated by contributions from only a few configurations with especially large values of exp [W (r^N)] and the average takes too long to converge.

3.4.2 Running the umbrella sampling calculations

Using a relaxed starting structure MD can be run on the individual umbrella windows. The main point to remember when selecting the number of windows is that the end points must overlap, that is, window 1 must sample some of window 2 and so on. The force constant similarly has to be big enough to ensure that the subset of phase space are sampled but not too strong that the windows become too narrow and can't overlap.



"\" = lower bound linear response region

"/" = lower bound linear response region

"…" = parabola

"_" = flat region

Normally, one can vary the size of the windows and the constraints as a function of position along the pathway. The amount of simulation we do in each window needs to be such that we can converge our sampling. To specify the harmonic restraint a reference file is employed where R1, R2, R3, R4 define a flat-welled parabola which becomes linear beyond a specified distance. Essentially between r1 and r2 will be harmonic with force constant rk2, between r2 and r3 it will be flat and between r3 and r4 it will be harmonic with force constant rk3.

3.4.3 The Weighted Histogram Analysis Method (WHAM) for free-energy calculations

The WHAM method [184, 185] is an extension of the US method but it has a number of advantages over the conventional US method. The WHAM method, in addition is used to optimize the links between simulations, also it allows multiple overlaps of probability distributions to obtain better estimates of the free-energy differences. The older method of obtaining a single distribution function by requiring that the probability distributions agree at some point in the overlap region will fail to yield unique free-energies if three or more distributions are involved in the overlap region. This algorithm provides a built-in estimate of errors that give investigators objective estimates of the optimal location and length of additional simulations to improve the accuracy of their results. The WHAM method takes into account all the simulations that produce overlapping distributions. The WHAM method links the different simulations through the overlapping histograms in an optimal manner. The WHAM equations can also be readily used to generate PMFs and free energies as a function of the coupling parameter(s) *hi* and/or the temperature. This is useful as simulations can be carried out at a range of temperatures to improve conformational sampling and the results extrapolated (or interpolated) to the desired temperature [184].

3.5 Binding free energy calculation using Molecular Mechanics energies combined with the Poisson-Boltzmann or Generalized Born and Surface Area continuum solvation method (MM-PBSA/GBSA)

3.5.1 Free energy calculation using Perl Script (mm_pbsa.pl)

The MM-PBSA and MM-GBSA [186-188] methods are popular approaches to estimate the free energy of the binding of small ligands to receptor protein or protein-protein complex. They are typically based on MD simulations of the protein–ligand complex and can often be reproduced with good accuracy.

In MM-PBSA or MM-GBSA, the binding free energy (ΔG_{bind}) between a ligand and a receptor to form a protein-ligand complex is calculated as

$\Delta G_{bind} = \Delta G_{complex,solv} - (\Delta G_{protein,solv} + \Delta G_{ligand,solv})$ ………………3.24

where $\Delta G_{complex,solv}$, $\Delta G_{protein,solv}$, and $\Delta G_{ligand,solv}$ are the free energy differences for the complex, the protein, and the ligand with or without solvent, respectively. Herein, a subscript "solv" in Eq. (3.24) represents the aqueous solution. The solvation free energies are calculated as follows;

$\Delta G_{comp,\ solv} = E_{MM} + \Delta G_{solvation} - TS_{solute}$ ………………………..3.25

$E_{MM} = E_{intra} + E_{elec} + E_{vdW}$ …………………………… 3.26

$E_{internal} = E_{bond} + E_{angle} - E_{torsion}$ …………………………. ...3.27

$\Delta G_{solvation} = \Delta G_{PB/GB\ solvation-elec} + \Delta G_{SASA,nonpolar}$ ……………….....3.28

Where 'comp' in Eqn. 3.25 represents the complex (protein +ligand). E_{MM} is the molecular mechanics (MM) energy from the force field without the solvent. E intra consists of three intramolecular contributions, i.e. E_{bond}, E_{angle}, and $E_{torsion}$. E_{elec} and E_{vdW} are the intermolecular electrostatic and van der Waals interaction energies, respectively. $\Delta G_{solvation}$ is the solvation free energy, and $\Delta G_{solvation-elec}$ is estimated from the Poisson–Boltzmann method. ΔG nonpolar is estimated from the solvent-accessible surface area (SASA). T and S_{solute} are the temperature and the entropy of a solute. We show the relationship for each energy in **Figure 3.7.**

Using PB and GB method the electrostatic solvation energy can be determined. The dielectric constants used for the interior (solute) and exterior (water) were set to 1 and

80 respectively. Atomic radii and charges are same as used in the MD simulations. From solvent accessible surface-area, the non-polar contribution (ΔG_{SASA}) to the solvation free energy was calculated using eqn. 3.29.

$$\Delta G_{SASA} = \gamma \times SASA + b \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ 3.29$$

Here, SASA is the solvent-accessible surface-area and γ is surface tension parameter. 'γ' is set as 0.005 kcal ($mol^{-1}Å^{-2}$) for PB and 0.0072 kcal ($mol^{-1}Å^{-2}$) for GB. '*b'* is a parameterized value set as 0.92 kcal mol^{-1} for PB and 0 kcal mol^{-1} for the GB method. The probe radius of the solvent is set to 1.4 Å. The entropy calculation was neglected in the above calculation as we are interested in calculating only relative binding energy contribution to the formation of protein-ligand complex.

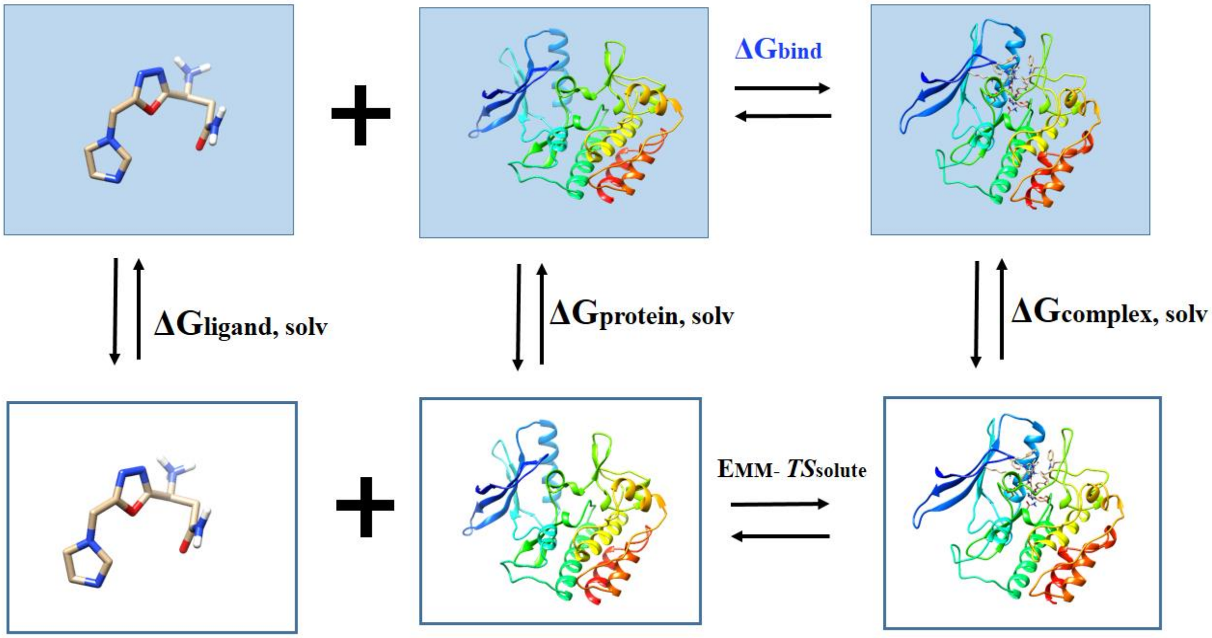


Figure 3.7. *Computational schemes of the binding free energies based on MM-PBSA/GBSA. The free energies colored in black are directly calculated, while the free energy of interest colored in blue is indirectly did using the thermodynamic cycle of other free energies. Modified from* [189].

3.5.2 Free energy decomposition using Python Script MMPBSA.py

Using either the GB or PB models, Amber12 provides several schemes to decompose calculated free energies into specific residue contributions [190], following the work of

Gohlke et al [191, 190]. For each residue, interactions can be decomposed by including only those interactions in which one of the residue's atoms is involved which is called *per-residue* energy decomposition. On the other hand, interactions can be decomposed by specific residue pairs by including only those interactions in which one atom from each of the analyzed residues is participating which is called as *pairwise* decomposition. These decomposition schemes can provide useful insights into important interactions in free energy calculations [191, 190]**.** However, it is important to note that solvation free energies using GB and PB are not strictly pairwise decomposable, since the dielectric boundary defined between the protein and the bulk solvent is inherently nonlocal and depends on the arrangement of all atoms in space. Thus, care must be taken when interpreting free energy decomposition results.

We can calculate partial binding free energy contribution to the amino acid residue Y, (ΔG^{Y} bind) by using Per-residue decomposition method in Python Script MMPBSA.py [192]. Per-residue based decomposition can determine the contribution of individual residue to the total binding free energy [190, 193-196]. To obtain ΔG^{Y} bind we first divide terms in Eqn. (3.25) into its atomic contribution. The contribution of each atom *a* to the total electrostatic interaction energy is obtained by

$$E_{elec}^{a} = \frac{1}{2}\sum_{b\neq a}\frac{q_a q_b}{r_{ab}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots..3.30$$

where q_a and q_b are atomic partial charge of the atom *a* and *b*, r_{ab} is the distance between them. Similarly, one half of the pairwise energy for van der Waals interaction energy between protein and ligand, E^{a}_{vdW}, to avoid double counting. Using the SASA of each atom *a*, a non-polar part of solvent effects on binding free energy is represented as

$$\Delta G^{a}_{nonpolar,solv} = \gamma\{(SASA^{a,complex} - (SASA^{a,protein} + SASA^{a,ligand})\} \ldots\ldots\ldots\ldots3.31$$

Where, $SASA^{a,\ protein}$ and $SASA^{a,ligand}$ is equal to zero depending on which component the atom belong to. γ is set to 0.0072 kcal mol^{-1} $Å^{-2}$ in AMBER 12. To calculate the contribution of atom *a,* to the electrostatic part of solvent effects, the generalized Born (GB/PB) approach is used. The contribution of atom *a* is given by;

$$\Delta \mathrm{G}^{\mathrm{a}}_{\mathrm{elec,sol}} = -\frac{1}{2}\sum_{a}\left(1-\frac{e^{-k\int_{ab}^{GB}}}{\varepsilon_{\omega}}\right)\frac{q_a\ q_b}{\int_{ab}^{GB}(r_{ab})}+\frac{1}{2}\sum_{b\neq a}\frac{q_a\, q_b}{r_{ab}} \quad \text{…………..} \ 3.32$$

$$\int_{ab}^{GB} = [\ r_{ab}^{2} + \alpha_a \alpha_b \exp\left(\frac{-r_{ab}^{2}}{4\ \alpha_a\ \alpha_b}\right)]^{1/2} \quad \text{……………………......}3.33$$

where κ is the Debye-Huckel screening parameter. ε_{ω} is a dielectric constant for the solvent set as 80. α_a and α_b are the effective Born radii of atoms a and b, respectively. Using these contributions to each atom, the partial binding free energy contribution to the amino acid residue Y is evaluated as

$$\Delta G_{bind}^{Y} = \sum_{a\in Y}\left(E_{elec}^{a} + \ E_{vdw}^{a} + \Delta G_{nonpolar,solv}^{a} + \Delta G_{elec,solv}^{a}\right) \quad \text{………}3.34$$

Here the entropic and intra-molecular contributions appearing in eqn. (3.25) and (3.26) are neglected in this analysis.

Another way of decomposing free energies is to introduce specific mutations in the protein sequence and analyze how binding free energies or stabilities are affected [197]. Alanine scanning mutagenesis, is a technique in which an amino acid in the system is mutated to alanine, which can highlight the importance of the electrostatic and steric nature of the original side chain [198]. Assuming that the mutation will have a negligible effect on protein conformation, we can incorporate the mutation directly into each member of the original ensemble. This avoids the need to perform an additional MD simulation to generate an ensemble for the mutant.

3.6 Molecular docking

Using molecular docking, the interaction between a protein and small molecule or between two proteins can be modelled at the atomistic level, which allows to know the behavior of small molecules at the binding site of target protein or we can get the interacting interface residues in protein-protein interaction which may reveal fundamental biochemical processes [199]. The docking process involves two steps. In the first step the pose of ligand at the binding site. In the second step rank is allotted to the conformers of ligands using scoring function that is based on the binding affinity. Initially the sampling algorithms reproduce the experimental binding mode and then scoring function should be able to rank it highest among all conformations that is generated.

3.6.1 Docking methodologies

3.6.1.1. Rigid ligand and rigid receptor docking

In rigid docking the search space is very limited because both the receptor and ligand are considered as rigid bodies, and also the degrees of freedom is restricted to only the three translational and three rotational. In this circumstances, the ligand flexibility is addressed using a pre-computed set of ligand conformations. This protocol has been followed in previous version of DOCK [200-204].

3.6.1.2. Flexible ligand and rigid receptor docking

In the systems wherein, it follows the induced fit model [205, 206], both the ligand and receptor are consider to be flexible, so that a perfect-fit complex with minimum energy can be obtained. If both receptor and ligand are considered to be flexible, then the cost becomes very high and also time consuming. But cost can be minimized if ligand is kept flexible while the receptor is kept rigid during docking. This particular methodology is followed in AutoDock [207] and FlexX [208]. In AutoDock 3.0 Monte Carlo simulated annealing, evolutionary, genetic and Lamarckian genetic algorithm methods [209] are used to model the flexible ligand and keeping the receptor rigid. The scoring function is based on the AMBER force field, which includes, van der Waals interactions, hydrogen bonding, electrostatic interactions, conformational entropy and desolvation terms. Each term is weighted using an empirical scaling factor obtained from experimental data. In AutoDock 4.0 the flexible receptor is modelled by allowing the side-chains to move. In addition this version of Autodock supports the interaction of protein-protein docking [210]. In this thesis we have used the recent version of AutoDock that is AutoDock 4.2 [211] to dock small molecules with LMTK3 domain.

3.6.1.3. Steps performed in AutoDock 4.2

<u>Step 1. Coordinate File Preparation</u>: Initially, AutoDock 4.2 [211] prepare protein and ligand which add polar hydrogen atoms to the protein, but not hydrogen atoms bonded

to carbon atoms. Then, PDBQT file is generated which is used for coordinate files that includes atomic partial charges and atom types. The current AutoDock force field uses several atom types for the most common atoms, including distinct types for aliphatic and aromatic carbon atoms, and separate types for polar atoms that form hydrogen bonds and those that do not. PDBQT files also include information on the torsional degrees of freedom.

Step2. AutoGrid Calculation: In the AutoGrid process the protein is embedded in a 3-D grid box and a probe atom is placed at each grid point **(Figure 3.8**). The energy of interaction of single atom with the protein is assigned to the grid point. AutoGrid affinity grids are calculated for each type of atom in the ligand, usually carbon, oxygen, nitrogen and hydrogen, along with the grids of electrostatic and desolvation potentials. When AutoDock calculation is performed, the energetics of a particular ligand conformation is evaluated using the values from the grids.

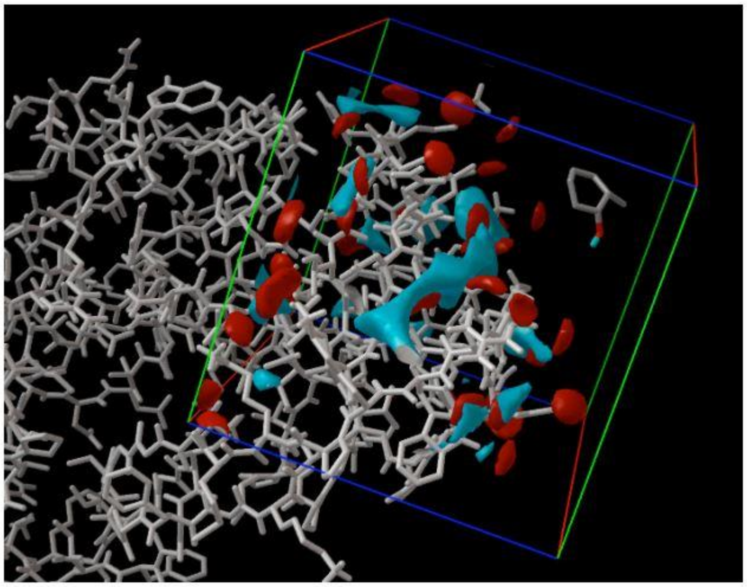


Figure 3.8. *Viewing Grids in AutoDockTools. The protein is shown on the left in white bonds, and the grid box is shown on the right side. The blue contours surround areas in the box that are most favorable for binding of carbon atoms, and the red contour areas that favor oxygen atoms. A ligand is shown inside the box at upper right. Taken from Autodock user guide* [212].

Step 3. Docking using AutoDock: Docking is carried out using a number of search methods. The Lamarckian genetic algorithm (LGA) is the most efficient method, but traditional genetic algorithms and simulated annealing are also available. AutoDock is run for a typical systems for several times and provide quite a few docked conformations. Analysis of the docked conformations with predicted energy and the consistency of results is combined to identify the best solution with high affinity of binding.

Step 4. Analysis using AutoDockTools: For analyzing the results of docking simulations, AutoDockTools includes various methods, such as tools for clustering results by conformational similarity, visualizing conformations, visualizing interactions between ligands and proteins, and visualizing the affinity potentials created by AutoGrid.

3.7. *In silico* prediction of protein-protein interaction

Protein-protein interaction (PPIs) is an essential driving mechanism in many physiological processes in the cell, which is also involved in the pathogenesis of numerous diseases [213-215]. Due to the diversity of protein–protein interactions there is a need for careful examination of the nature of the protein interface. The determinant of the specificity and stability of protein–protein interaction is important. The size of the protein interface decides whether the complex will be transient or obligatory. The interface between two proteins typically has an area of 1500-3000 Å^2 with approximately 750-1500 Å^2 of surface area buried in each protein [216-218]. The protein-protein interaction sites are formed by proteins with good shape complementarity [219-221], driven by hydrophobic effects [222], which occur between the nonpolar regions of protein residues through van der Waals contacts. Electrostatic complementarity of the interacting protein surfaces between two proteins promotes the formation and lifetime of the complex. For some interface, hydrogen bonding and electrostatic interaction play a major role in directing one protein to dock with the binding site of the second protein. Prediction of protein-protein interaction is crucial in drug discovery. Many physiological and pathological cellular processes depends on the protein-protein interaction which can be disturbed by external compounds. The modern drug discovery process involves two main steps: identification of prospective drug target, investigating its properties and designing of a corresponding ligand [223]. Therefore, the knowledge of protein-protein interaction can be useful in designing modulators that can target the protein complex.

Computerized prediction of protein-protein interaction and protein-small molecule interaction is one of the most challenging task in structural biology. Many biological studies, in academic world as well in industry, may benefit from reliable high-accuracy interaction prediction. In the protein-protein docking the problem is to find the accurate association of two interacting molecules. The accurate prediction is based on the residues contacts involved in the target interaction. Many docking algorithms [224-228] have been developed. However only a few algorithms are currently available as a free web service. The algorithms mostly differ in the method for searching the six-dimensional transformation space that they apply, and in their evaluation of the resolved complexes. In this thesis we used PatchDock [229] and ClusPro [230] server for protein-protein docking.

3.7.1. PatchDock web server

PatchDock is a very efficient algorithm for protein-protein and protein-small ligand docking which performs rigid docking. It is a geometry-based molecular docking algorithm [229]. The algorithm was verified on the enzyme-inhibitor and antibody-antigen complexes from benchmark 0.0 [231], where it successfully found near-native solution for most of the cases. The algorithm was also successfully tested in [232-234] of the Critical Assessment of Prediction of Interaction (CAPRI) [235].

PatchDock is based on local shape feature matching algorithm established by Kuntz [236]. This algorithms employ shape complementarity constraints, when searching for the correct association of molecules. At first, it detects those molecular surface areas which have a high probability to belong to the binding site. This reduces the number of potential docking solutions and retaining the correct conformation. It identified docking transformations that yield good molecular shape complementarity. The algorithm functions through three major stages:

(i) Molecular Shape Representation - Molecular surface of the molecule is compute in this step. Next, a segmentation algorithm is applied for the detection of geometric patches (concave, convex and flat surface pieces). The patches are filtered, in order to retain those patches which contain 'hot spot' residues [237].

(ii) Surface Patch Matching - A hybrid of the Geometric Hashing [238] and Pose-Clustering [239] matching techniques are applied to match the patches those are detected in the previous step. Here concave patches are matched with convex and flat patches are matched with any type of patches (**Figure 3.9)**

(iii) Filtering and Scoring – In this step, the candidate complexes from the previous step are examined. RMSD clustering is applied to the candidate solutions to discard the redundant solutions or unacceptable penetrations of the atoms of the receptor to the atoms of the ligand. Finally, the remaining candidate transformations are evaluated by scoring function that considered both geometric fit (geometric shape complementarity score) and atomic desolvation energy [240].

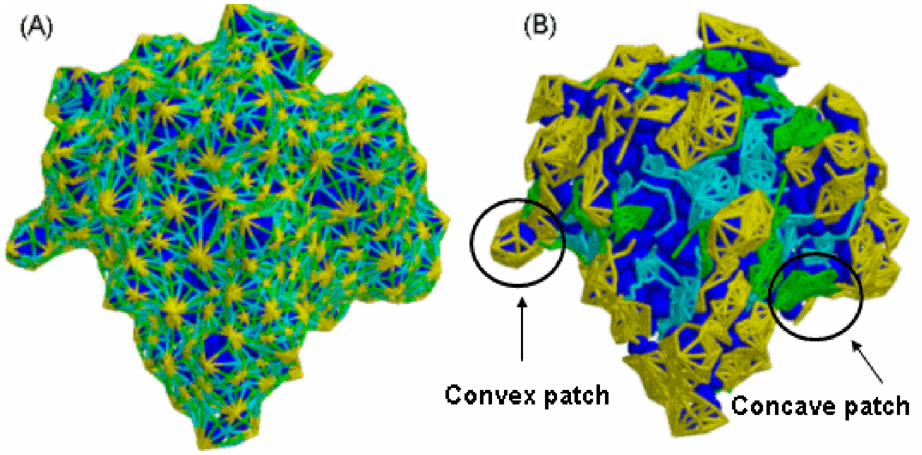
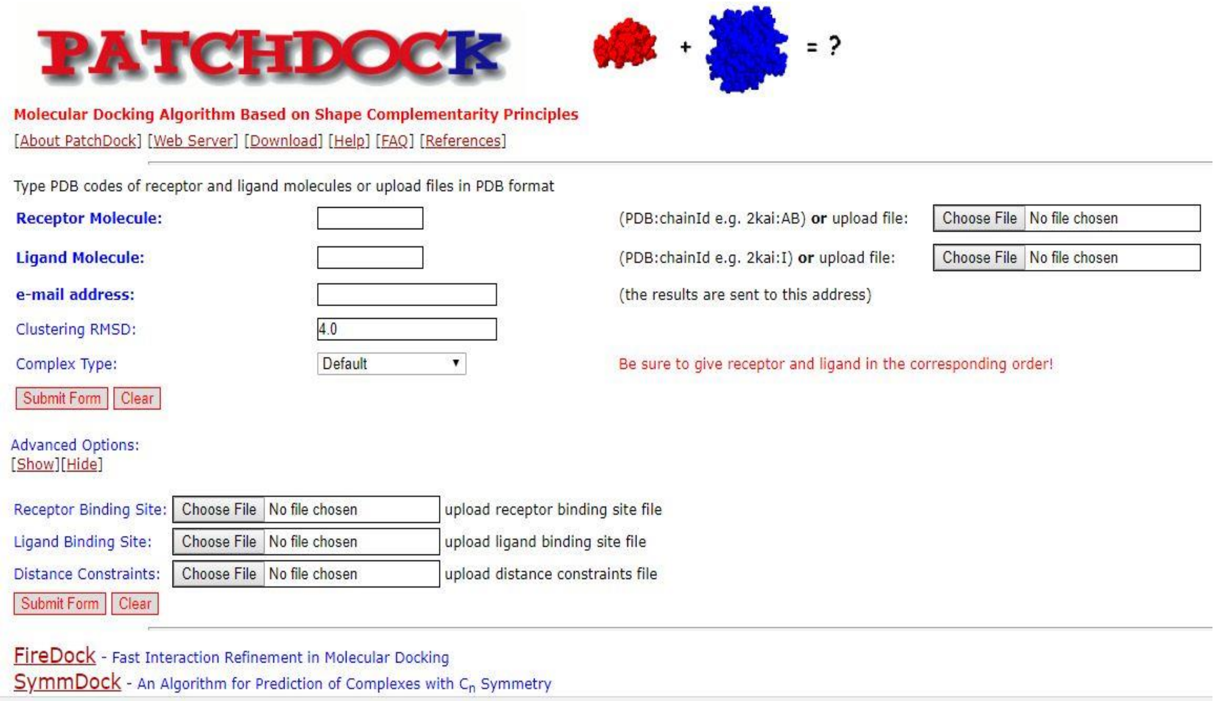


Figure 3.9. *(A) Surface topology graphs for trypsin inhibitor (PDB code 1ba7). The caps, belts and pits are connected with edges. (B) Geometric patches: the patches are in light colors and the protein is dark. Taken from* [229].

3.7.1.1 PatchDock web server: Input, output and user interface

PatchDock algorithm is available at https://bioinfo3d.cs.tau.ac.il/PatchDock/. When the docking request is submitted by the user, the prediction process starts by PatchDock algorithm. The user is notified by an email when the results are ready, it contains a link to the web page where the prediction results are presented. On this page user can view specific predictions as well as download the compressed file of the top scoring solutions (**Figure 3.10**)



Molecular Docking Algorithm Based on Shape Complementarity Principles

[About PatchDock] [Web Server] [Download] [Help] [FAQ] [References]

Type PDB codes of receptor and ligand molecules or upload files in PDB format

Receptor Molecule: (PDB:chainId e.g. 2kai:AB) **or** upload file: Choose File No file chosen

Ligand Molecule: (PDB:chainId e.g. 2kai:I) **or** upload file: Choose File No file chosen

e-mail address: (the results are sent to this address)

Clustering RMSD: 4.0

Complex Type: Default ▾ Be sure to give receptor and ligand in the corresponding order!

Submit Form | Clear

Advanced Options:
[Show][Hide]

Receptor Binding Site: Choose File No file chosen upload receptor binding site file

Ligand Binding Site: Choose File No file chosen upload ligand binding site file

Distance Constraints: Choose File No file chosen upload distance constraints file

Submit Form | Clear

FireDock - Fast Interaction Refinement in Molecular Docking

SymmDock - An Algorithm for Prediction of Complexes with C_n Symmetry

Figure 3.10. *The PatchDock user interface: The receptor molecule and the ligand molecule are given either by the PDB code of the molecule (chain IDs are optional) or by uploading a file in PDB format. Taken from* [241]

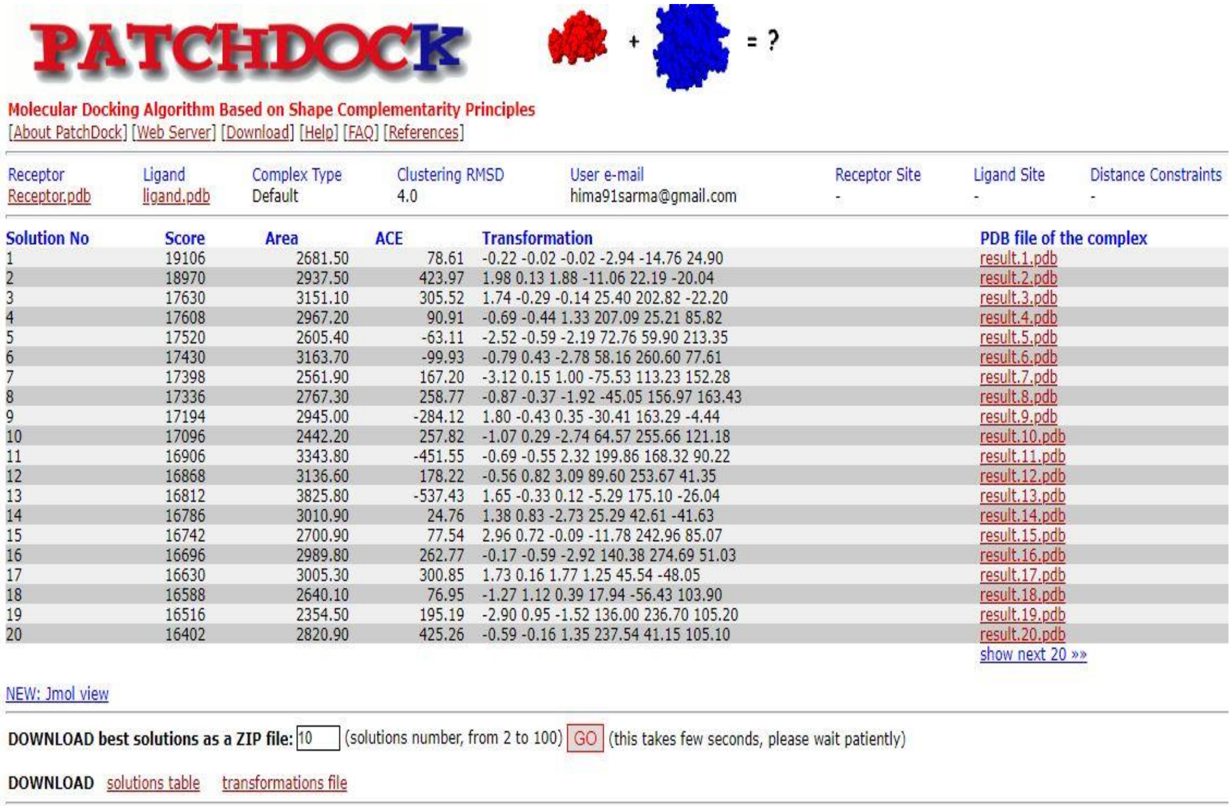
Input

For protein-protein docking, the input is two protein molecules in PDB format. The molecules are either uploaded to the server or it can be retrieved from the Protein Data Bank. In the second case the user can enter the PDB code. In order to dock two protein, the user should specify those two proteins in a desired chain IDs. For result notification the user has to provide email ID.

Output

The user receives an email message with the URL of the web link, wherein, top 20 solutions (docked complexes) will be automatically generated. In a table, in each row the solutions are presented. **Figure 3.11** depicts the geometric score, desolvation energy [242], interface area size and the actual rigid transformation of the solution. A link to a

PDB file that presents the docking solution is also available in each line. The user may view or download it. There is also an option to view additional, lower ranking solutions by pressing the 'next 20 solutions' button at the lower right corner of the table (**Figure 3.11**). In the solutions page an option to download the top scoring solutions is available. The solutions are downloaded as a compressed ZIP file format. This compressed file contains the PDB files of the top scoring solutions.



Molecular Docking Algorithm Based on Shape Complementarity Principles
[About PatchDock] [Web Server] [Download] [Help] [FAQ] [References]

Receptor	Ligand	Complex Type	Clustering RMSD	User e-mail	Receptor Site	Ligand Site	Distance Constraints
Receptor.pdb	ligand.pdb	Default	4.0	hima91sarma@gmail.com	-	-	-

Solution No	Score	Area	ACE	Transformation	PDB file of the complex
1	19106	2681.50	78.61	-0.22 -0.02 -0.02 -2.94 -14.76 24.90	result.1.pdb
2	18970	2937.50	423.97	1.98 0.13 1.88 -11.06 22.19 -20.04	result.2.pdb
3	17630	3151.10	305.52	1.74 -0.29 -0.14 25.40 202.82 -22.20	result.3.pdb
4	17608	2967.20	90.91	-0.69 -0.44 1.33 207.09 25.21 85.82	result.4.pdb
5	17520	2605.40	-63.11	-2.52 -0.59 -2.19 72.76 59.90 213.35	result.5.pdb
6	17430	3163.70	-99.93	-0.79 0.43 -2.78 58.16 260.60 77.61	result.6.pdb
7	17398	2561.90	167.20	-3.12 0.15 1.00 -75.53 113.23 152.28	result.7.pdb
8	17336	2767.30	258.77	-0.87 -0.37 -1.92 -45.05 156.97 163.43	result.8.pdb
9	17194	2945.00	-284.12	1.80 -0.43 0.35 -30.41 163.29 -4.44	result.9.pdb
10	17096	2442.20	257.82	-1.07 0.29 -2.74 64.57 255.66 121.18	result.10.pdb
11	16906	3343.80	-451.55	-0.69 -0.55 2.32 199.86 168.32 90.22	result.11.pdb
12	16868	3136.60	178.22	-0.56 0.82 3.09 89.60 253.67 41.35	result.12.pdb
13	16812	3825.80	-537.43	1.65 -0.33 0.12 -5.29 175.10 -26.04	result.13.pdb
14	16786	3010.90	24.76	1.38 0.83 -2.73 25.29 42.61 -41.63	result.14.pdb
15	16742	2700.90	77.54	2.96 0.72 -0.09 -11.78 242.96 85.07	result.15.pdb
16	16696	2989.80	262.77	-0.17 -0.59 -2.92 140.38 274.69 51.03	result.16.pdb
17	16630	3005.30	300.85	1.73 0.16 1.77 1.25 45.54 -48.05	result.17.pdb
18	16588	2640.10	76.95	-1.27 1.12 0.39 17.94 -56.43 103.90	result.18.pdb
19	16516	2354.50	195.19	-2.90 0.95 -1.52 136.00 236.70 105.20	result.19.pdb
20	16402	2820.90	425.26	-0.59 -0.16 1.35 237.54 41.15 105.10	result.20.pdb

show next 20 »»

NEW: Jmol view

DOWNLOAD best solutions as a ZIP file: 10 (solutions number, from 2 to 100) GO (this takes few seconds, please wait patiently)

DOWNLOAD solutions table transformations file

Figure 3.11. *The solutions page presents the geometric score, interface area size and desolvation energy of the 20 top scoring solutions. Modified from* [241]

3.8. ClusPro web server

ClusPro a web based server (https://cluspro.org) was introduced in 2004 [243,244] and has been substantially modified and expanded later on [245-247]. ClusPro, is used for direct docking of two interacting proteins [248]. For docking the server needs two protein files in PDB format. During docking the server performs three computational steps:

(i) Sampling billions of conformations, using rigid body docking.

(ii) Clustering of 1000 lowest energy structures based on root-mean-square deviation (RMSD), in order to find the largest clusters that represent the most likely models of the complex.

(iii) Refinement of the selected structures using energy minimization (**Figure 3.12**). The rigid body docking step uses PIPER, [249] a docking program based on the Fast Fourier Transform (FFT) correlation approach.

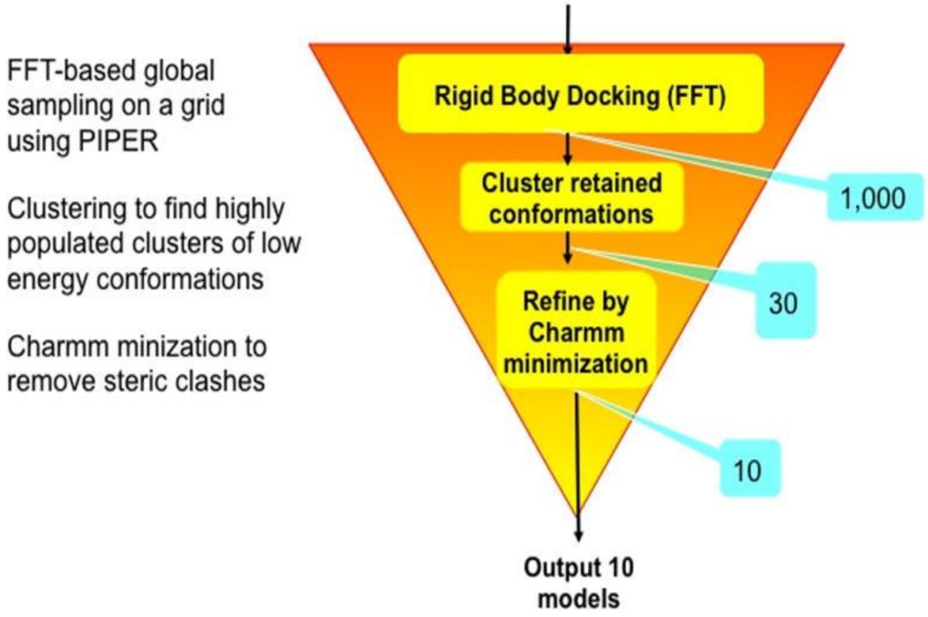

FFT-based global sampling on a grid using PIPER
Clustering to find highly populated clusters of low energy conformations
Charmm minization to remove steric clashes
Rigid Body Docking (FFT)
Cluster retained conformations
Refine by Charmm minimization
1,000
30
10
Output 10 models

Figure 3.12. *Representation of the ClusPro algorithm, the number of structures retained after each step is shown in a blue box. Taken from* [248]

3.9. PDBsum web server

PDBsum (http://www.ebi.ac.uk/pdbsum) [250] is a web-based database provides pictorial summary of the important information on each macromolecular structure deposited at the Protein Data Bank (PDB). It includes images of the structure, annotated plots of each protein chain's secondary structure, detailed structural analyses, summary PROCHECK results and schematic diagrams of protein–protein, protein-small molecule and protein–DNA interactions. RasMol scripts highlight the important features of the structure, such as the protein's domains, PROSITE patterns and protein–protein/ligand interactions. PDBsum is updated whenever any new structures are released by the PDB and is freely accessible via http://www.biochem.ucl.ac.uk/bsm/pdbsum.

3.9.1. Wiring diagram

For each unique protein chain of a structural model, PDBsum provides a ‘protein page’ that includes a schematic diagram of the protein's secondary structure that is ‘wiring diagrams’ (**Figure 3.13**)

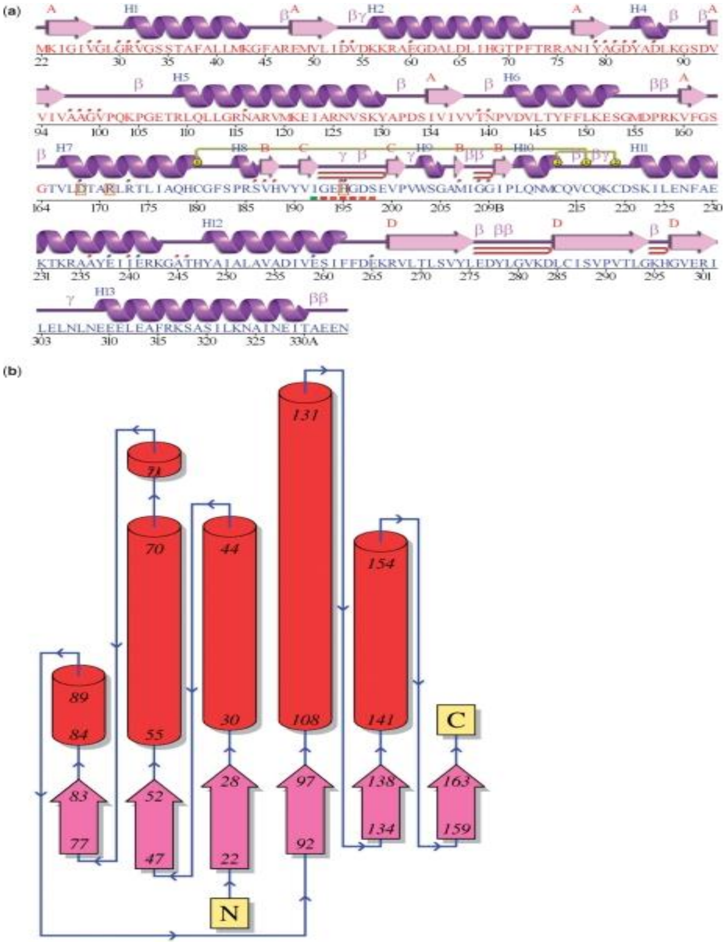


Figure 3.13. *Schematic diagrams from the PDBsum for entry 1a5z: (A) The ‘wiring diagram’ shows the protein's secondary structure elements (α-helices and β-sheets) together with β- and γ-turns, and β-hairpins. The yellow linking bars labelled 1 and 2 represent disulphide bonds. The single-letter amino acid codes showing the protein's sequence are coloured red or blue depending on whether they belong to CATH* [251] *structural domain. Red dots above the single-letter codes signify residues that interact with any bound ligand(s) while coloured lines underneath represent residues belonging to a PROSITE pattern, the redder the colour the more highly conserved the residue in the pattern. (B) Topology diagram illustrates the β-strands by the large arrows joined side-by-side (pink colour), forming central β-sheet. The α-helices represented by the red*

cylinders. The small arrows indicate the directionality of the protein chain, from the N-to the C-terminus. Taken from [252]

3.9.2. Topology diagram

The protein page also includes a topology diagram that display the arrangement and connectivity of the protein's helices and strands (**Figure 3.13 B**). Where the protein chain consists of more than one domain, a separate diagram is generated for each and is colour-coded according to the domain coloring on the wiring diagram. The topology diagrams are generated from the hydrogen bonding plots of Gail Hutchinson's HERA program [253].

3.9.3. Protein-protein interfaces

In PDBsum another new feature that demonstrates the interactions across protein–protein interfaces. When the protein-protein complex contains more than one protein chain (e.g. **Figure 3.14 A**), the interfaces between the chains are depicted by three types of plot: first summarizes an overview of which chain interact with which (**Figure 3.14 B**), the second summarizes the interactions across any selected interface (**Figure 3.14 C**), and the third shows which residues actually interacting across that interface in detail (**Figure 3.14 D**).

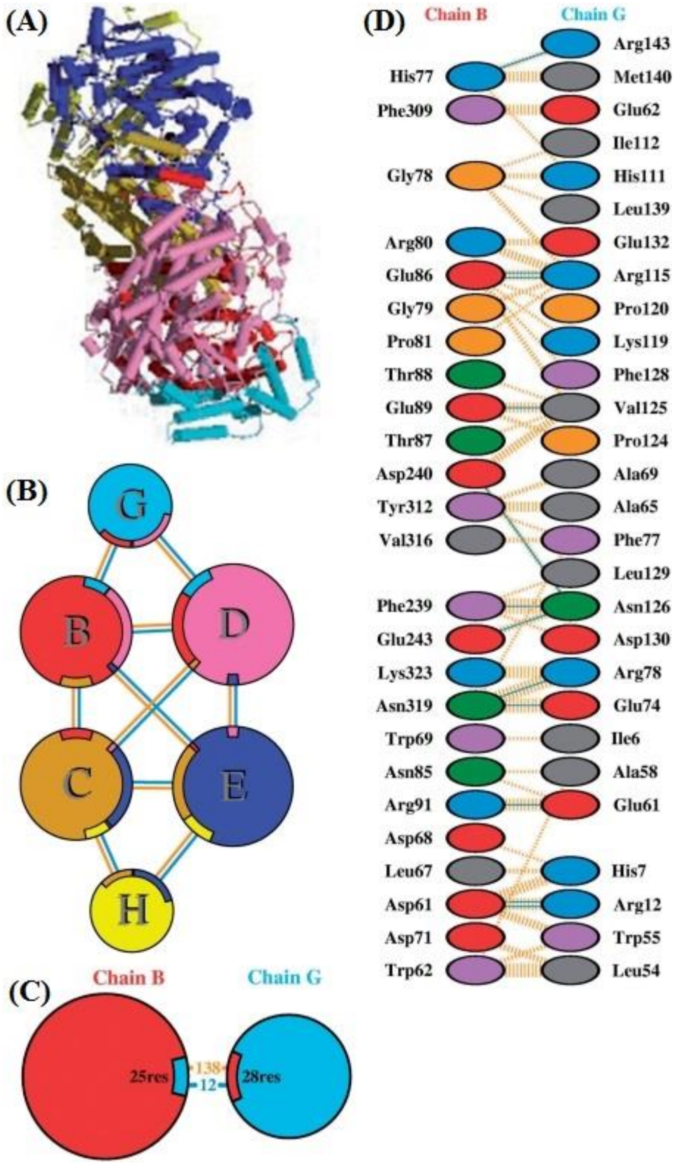


Figure 3.14 *Protein–protein interaction diagrams in PDBsum for PDB entry 1mmo: (A) Thumbnail image of the 3D structural model which contains six protein chains (B) Schematic diagram showing the interactions between the chains. The area of each circle is proportional to the surface area of the corresponding protein chain. The joining lines are coloured light blue for hydrogen bonds and orange for non-bonded contacts. (C) A schematic diagram showing the numbers of interactions across one of the interfaces, namely the B–G protein interface, and the numbers of residues involved. (D) Detail of the individual residue–residue interactions across this interface. Hydrogen bonds (blue lines), non-bonded contacts (orange tick-marks), and salt bridges (red lines) between residues on either side of the protein-protein interface. Taken from* [252]

From the input of protein-ligand complex in PDB format, LIGPLOT program [254] generates a 2-D schematic depiction of the hydrogen bonds and non-bonded interactions between ligand and the residues of the protein with which it interacts (**Figure 3.15**). The output is a color, or black-and-white, PostScript file gives a simple and informative representation of the intermolecular interactions and their strengths, including hydrogen bonds, hydrophobic interactions and atom accessibilities. The program is completely general for any ligand and can also be used to show other types of interaction in proteins and nucleic acids**.**

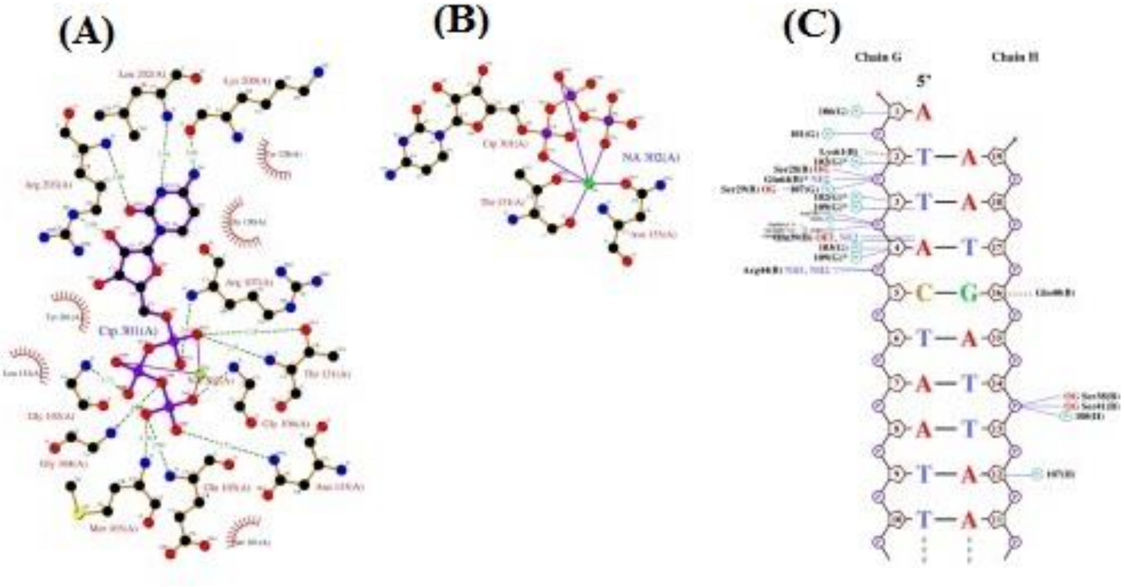


Figure 3.15. *PDBsum interaction plot for PDB entry 5trd in LIGPLOT (A) LIGPLOT diagram showing the protein residues that interact with the CTP (cytidine-5'-triphosphate) ligand, with hydrogen bonds shown by the green dashed lines and non-bonded contacts by the brown rays, and; (B) as in (A), but for residues interacting with the bound sodium ion; (C) Diagram of protein-DNA interactions, with H-bonds as blue dashed lines and non-bonded contacts as brown dashed lines. Taken from* [250]

3.10. Hot spot residue prediction

The residues on the protein-protein interface do not contribute equally to the protein-protein interactions (PPIs). A small subset of residues contributes to the majority of the binding free energy, they are called as hot spots [255]. A hot spot is defined as a residue, when substitution by an alanine leads to the significant increase in the binding free energy of at least 2.0 kcal mol^{-1} [256]. Conversely, Null-spots (NS) corresponds to residues with (change in binding free energy) ΔΔG binding are lower than 2.0 kcalmol^{-1} when mutated to alanine and null-spots exist in the surrounding regions of the hots pots and protect them from solvent exposure [257]. Hot spot residues exist in clusters and are well conserved and more buried in comparison to other interface residues in the protein-protein complex [258-261]. Tyr, Arg and Trp amino acids have a greater tendency in being a ho tspot [262], while Leu, Thr, Ser and Val are less likely to act as a hot spot. Similarly, Asp and Asn have been observed as hot spots more frequently than Glu and Gln [263,262].

Identifying these hot spot residues within the protein-protein interfaces can help us to understand protein-protein interactions and may also help us to modulate protein-protein binding [264]. We have identified hot spots at protein-protein interface by using different computational methods which are freely available online servers including KFC (Knowledge-based FADE and Contacts) [265], PredHS [266], Robetta [267, 268] and DrugScorePPI server [269].

KFC server is a machine learning based tool that utilizes in silico alanine scanning mutagenesis, considering hydrogen bonds, atomic contacts and residue sizes for hot spot identification [265].

PredHS server is a structure-based hot spot prediction method which predicts hot spot residues using algorithms based on structural neighborhoods, and then selects optimal features using random forest and sequential backward elimination algorithms [266].

Robetta [267,268] and DrugScorePPI [269] server predicted hot spot residues computational alanine scanning mutagenesis.

3.11. Virtual Screening

To access novel drug like compounds, virtual screening has become an important tool. To carry out the biological screening of billions of compounds, the experimental efforts generally requires more time and cost, therefore, computer-aided drug design approaches have become attractive alternatives. In the recent years, virtual screening has reached a status of a dynamic and beneficial technology in searching for drug-like novel compounds in the pharmaceutical industry [270]. Virtual screening methods are of two types.

3.11.1. Ligand-based virtual screening

Ligand based virtual screening can rank novel ligands by 3-D similarity searching or by pharmacophore pattern matching. It involves different sequential computational phases, including database or library creation, and ranking of compounds accordingly for testing. These computational or theoretical methods can be employed to predict the putative binding affinities between small molecules and biological receptors of pharmaceutical interest [271].

3.11.2. Structure based virtual screening

When the target protein structure is available, rapid docking algorithms are used to dock the available candidate compounds within the active site of the target protein of interest and then the activity of compounds is ranked based on the steric and electrostatic components. Structure based virtual screening involves automated and fast docking of a large number of chemical compounds against a protein-binding or active site [272-274].

3.11.2.1. Structure based virtual screening using DOCK Blaster server

In this thesis DOCK Blaster server [275] is used to screen potential inhibitors against LMTK3 domain. Dock Blaster is an online virtual screening server that picks and scores thousands of small molecules when user uploads a target protein structure. DOCK Blaster utilizes DOCK 3.6 for docking and ZINC database [276] for ligands. And use Pocket Pickker (CLIPPERS) [277] to identify the binding pockets in the target protein. The details of virtual screening has been described in Chapter 8 section: 8.3.1.2

3.12. Energy optimized (E) Pharmacophore model generation

A pharmacophore is a description of molecular features that are necessary for molecular recognition of a ligand by a biological macromolecule. It can also be defined as an ensemble of steric and electronic features that is necessary to ensure the optimal supramolecular interactions with a specific biological target to trigger (or block) its biological response [278]. Typical pharmacophore features include hydrophobic centroids, aromatic rings, hydrogen bond acceptor or donor, cations and anions. These pharmacophoric features may be located on the ligand itself or may be located on the receptor. In this thesis, Schrodinger drug discovery suite is used for e-pharmacophore modelling of LMTK3 inhibitors. Here, at first protein is prepared in Maestro 9.0.111 (Maestro v 9.0.111 Schrodinger LLC, New York, NY) [279] wherein protein is minimized with the restraints using the OPLS 2005 (optimized potential for liquid simulations 2005) force field [280]. The Asinex database is used to screen the lead compounds and the compounds are then prepared using the Ligprep program (LigPrep, version 2.5, Schrodinger, LLC, New York, NY, 2011). This program generates the ligands based on the variations in their ionization states, wide pH ranges 5-9, combinations of stereoisomers, and tautomers [281]. The processed compounds are subjected to high throughput virtual screening protocol using the Glide docking algorithm implemented in Maestro 9.2V. The Glide scoring is done based on the Chem Score function of Eldridge and group [282], which identifies various types of interactions likes hydrophobic interaction, stable hydrogen bonding and metal-ligation interactions, and restricts any type of steric interactions. This hierarchical screening approach includes high throughput virtual screening (HTVS), standard precision docking (SP) and extra precision docking (XP).

Finally Energy-based pharmacophoric features are generated using the default chemical features: hydrogen bond acceptor, donor, hydrophobic (H), negative, positive, and aromatic ring, wherein Glide scoring terms are computed and energies are mapped onto the atom [283]. Each pharmacophoric feature is first assigned with an energetic value which is equal to the sum of the Glide XP contributions made by all the atoms present in that site. Based on the energetic terms used in the Glide XP descriptors, the sites are quantified and ranked. The e-pharmacophore model generation is explained in detail in Chapter 9, section 9.3.3.