**Protein Data Bank**

**.1Introduction**

They are the structural databases for the large molecules like proteins [1]. Proteins are very tiny and are of microscopic size, determining their size and shape can be done using many modern day technologies, few of them are listed below.

1. X-ray Crystallography [2]: Is a method used to determine the arrangement of atoms in three dimensional spaces. This technique takes advantage of the interatomic spacing of most crystalline solids by employing them as a diffraction gradient for x-ray light, which has wavelengths on the order of 1 angstrom.
2. NMR Spectroscopy: Nuclear magnetic resonance spectroscopy exploits the magnetic properties of atoms to determine the physical and chemical properties of atoms. NMR can either be used to match against spectral libraries or the basic structure of the protein can be directly inferred [3].
3. Cryo-electron microscopy [4]: Cryo EM is becoming the primary technology in structural biology at a molecular resolution. In the past few years, Cryo EM has been used in a broad range of experimental methods. At the core, each of these is based upon the principle of imaging radiation-sensitive specimens in a transmission electron microscope under cryogenic conditions.

**.2 Standard PDB format**

The PDB’s are populated with the data generated by biochemists after performing the above experiments on the protein structures. The crystallographic data of the proteins are stored in .pdb file format which are made available freely over the internet. The .pdb files can be downloaded from the websites like RCSB [5]. Every protein structure uploaded to any member organizations like RCSB, PDBe, PDBj is given a unique PDB ID to recognize the protein. Almost everywhere, the protein data is shared through a PDB (.pdb file).

A standard PDB file from RCSB is eighty columns wide and is always terminated by a end of the line indicator. The first six columns of every line contain a "record name" [7] and any PDB would have the following information in them:

Title and date: Name of the protein and the date when it was submitted and the PDB ID.

Experiment data, author information and remarks: The experiment through which the PDB data was generated followed by the name of the author and the journals it was published in and any remarks that the author wished to include.

Physical structure: SEQRES, HELIX, SHEET etc.

SEQRES is the list of the primary sequence of the polymeric molecules present in the entry [6].

HELIX and SHEET: the data here corresponds to parts of the structure of the protein which forms the helical structures and the sheets in the protein. Visualizing softwares use data in this section to draw corresponding structures.

Atom data:

Every atom of every amino acid in the protein will have a row in this file which contains data like atom number, residue type, chain and most importantly its structural data(x, y, z co-ordinates). The list of ATOM records for each chain starts with an N terminal and ends with the C terminal. Every polymer chain must be terminated with a TER record which defines the end of the particular chain. All the PDB files end with the line containing only the word END [7].

**.3 Role of PDB in knotfind and slipknotfind**

In our algorithm, we consider only the cα atoms from the atom data in the PDB file. Cα’s are alpha carbon atoms in the protein chain which form the backbone of the structure. Only the rows with cα atoms are filtered out of the protein chain and are stored in a Java ArrayList [8]. Each row of cα corresponds to one atom row of the PDB and will have the corresponding cα’s atom type, atom number, residue number, x-y-z co-ordinates and a few other columns of data. We save data from selected columns in the arraylist and discard everything else.

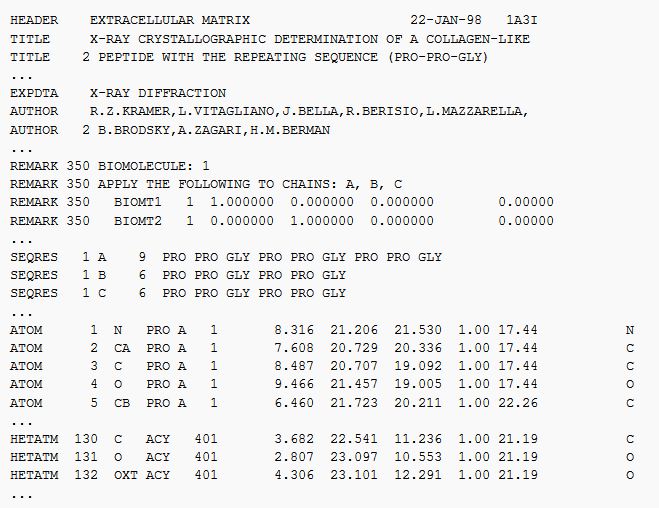


Figure 1: An example of how a protein data bank would look like is presented in the above image. Image courtesy: https://en.wikipedia.org/wiki/Protein\_Data\_Bank\_(file\_format).

**.4 Visualize PDB’s:**

PDB’s are plain text files which contain a lot of data. Given the increasing number of protein structures discoveries using faster mechanisms like Cryo EM, it gets very difficult and time consuming to visualize each and every protein manually and given the limited number of operations that we can perform on them, we need special and more sophisticated protein visualizing tools to visualize proteins effectively. And there are a lot of them available today, We use PyMol [9] and Jmol/JSmol [10] to visualize the proteins that we simplify using our algorithm. These tools also allow us to write our own custom scripts which will help any user to visualize structures as per their needs. PyMol comes with an API of its own and allows scripting in Python. Whereas Jmol/JSmol can be scripted using Java and JavaScript, using these tools and how we implement our scripts using them are explained in detail in the later sections.