Saarland University

Center for Bioinformatics

Masters program in Bioinformatics

Masters Thesis

Properties of Pore lining residues of transporter families in different staphylococcus Aureus strains

Submitted by

Jacob Marcus Ambat

15th May, 2016

Suprevisor

Prof. Dr. Volkhard Helms

Advisor

MSc. Duy Nguyen

Co-advisor

MSc. Rahmad Akbar

Reviewers

first Reviewer: Prof. Dr. Volkhard Helms

Second Reviewer: PD Dr. Micheal Hutter

Abstract

The central hypothesis in biology is that functionally important residues are conserved. The aim of our project is to expoit this hypothesis in identifying the pore lining residues (PLRs) of transmembrane proteins with unknown 3D structures. There are different ways in which the substrate molecules are transported through transporter proteins and channels. The PLRs are residues which are associated with the transport mechanism of the substrates. In our project we have successfully determined the conservation of these PLRs in the respective transporter families in SAN315 AND SACOL staphylococcus aureus subspecies. We have mapped the exact position of the PLRs and the conserved residues on-the primary sequence of the protein hits. From the mapped conserved PLRs for each protein family group, we then compare the degree of conservation for the staphylococcus aureus subspecies and did the hypothesis testing of the conserved residues and PLRs.

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides Statt, dass ich die vorliegende Arbeit selbstaständig verfasst und keine anderen als die angegeben Quellen und Hilfmittel verwendet habe.

Statement under Oath

I confirm under oath that i have written this thesis on my own and that i have not used any other media or materials than ones referred to this thesis.

Einverständniserklärung

Ich bin damit einverstanden, dass meine (bestandene) Arbeit in beiden Versionen in die Bibliothek der informatik aufgenommen und damit veröffentlicht wird.

Declaration of Consent

I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department

To My Family

Acknowledgements

I would like to gratefully and sincerely thank my suprevisor Prof. Dr. Volkhard helms for his guidance, understanding and patience during my studies at Universitat des Saarlandes. I especially appreciate my advisors MSc. Duy Nguyen and MSc. Rahmad Akbar for their guidance, explanations, discussions and patience.

I would like to thank all the members of Helms research group for their encouragement, comments and hard questions. I would also like to thank Ms. Kerstin Gronov pudelek , secretary of Prof. Dr. Volkhard Helms for her help during my studies.

Besides, I would also like to thank PD Dr. Micheal Hutter for reviewing my thesis.

Last but not the least, I would like to thank my friends in saarbrucken especially Daria, Taner, Pathmanaban, Anshika, Zhao, Sneha, Treesa, Chaithin and Vijay for their encouragement during my studies.

I would like to thank my family and colleagues for their extended support and endless encouragement for my studying.

Contents

List of Figures

List of Tables

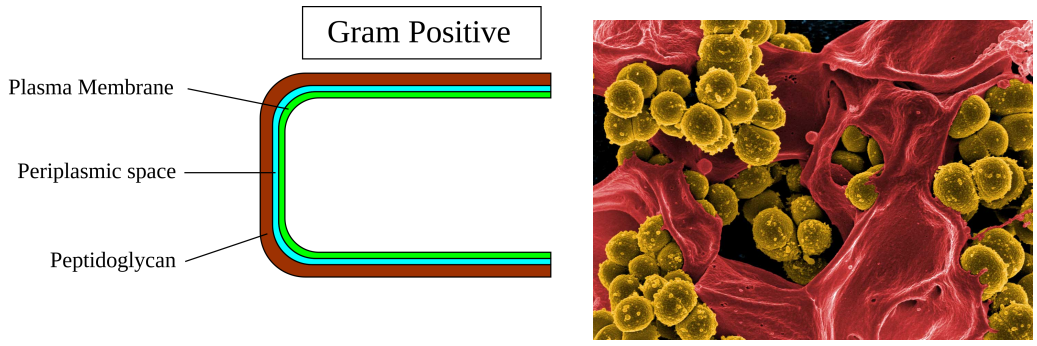
Nomenclature

ABC	ATP-binding cassette
BLAST	Basic Local Alignment Search Tool
COM	Centre Of Mass
DAACS	The Dicarboxylate/Amino Acid:Cation (Na+ or H+) Symporter
DOC	Degree Of Conservation
F-ATPase	The H+- or Na+-translocating F-type, V-type and A-type ATPase
FPRAT	Functional Pore Residue Annotation Tool
MFS	The Major Facilitator Superfamily
MSA	Multiple Sequence Alignment
OPM	Orientations of Proteins in Membranes
P-ATPase	The P-type ATPase
PLRs	Pore Lining Residues
PROPORES	PROtein PORE identification toolS
PTS	Phosphotransferase System
SSPTS	Sugar Specific Phosphotransferase System
TC	Taxonomy Classification
TM	Transmembrane

Chapter 1

Introduction

In our study, we have considered *Staphylococcus aureus*, which is a gram positive *coccal* bacterium (see figure 1.1a).It is mainly found in the respiratory tract and skin. It is also known by the name *Golden Staph* or *Oro Saphira*, as it is mainly seen as golden grape like colonies when seen under the microscope (see figure 1.1b) . *Staphylococcus aureus* is mainly associated with a wide range of diseases which are nosocomical and community acquired.



(a) (b)

Figure 1.1: (a) Gram positive bacteria cell wall structure (b) *Staphyllococcus Aureus.* Figure from[2]

Transporter proteins that are intergral trans-membrane proteins, are involved in the transport of materials within an organism. They contain pores or cavities for transporting essential molecules inward and outward through the membranes of biological cells and compartments. These pores are lined by Pore Lining Residues (PLRs), which have a direct contact with the pores. Identifying these PLRs will be useful for understanding the function of the particular transporter protein family. There are many methods available which can identify the pores/PLRs such as PROPORES[27], Pore Walker[21], Mole[25] which rely on the crystal structure of the proteins. This is the main drawback for these existing methodologies. This leads to the main aim of my thesis which is to develop a tool

to predict and determine the conservation of PLRs from the protein sequences. In the scope of this thesis, we focus mainly on prediction of PLRs for helical membrane transporters.

In this introductory chapter, a biological background on transporter proteins is being presented followed by information on the protein sequence datasets analysed in this work and an overview over related works which were done previously.

1.1 Membrane and Transporter Proteins

1.1.1 Plasma membrane and membrane bound proteins

Both prokaryotes and eukaryotic cells contain membranes. The cellular membrane is made up of a phospholipid bilayer with two sets of phospholipids with a hydrophilic/polar head facing outward a hydrophobic tail that is buried deep inside the membrane. The cellular membranes or the plasma membrane transport a wide variety of materials such as water, ions, metabolites and even entire protein chains across the membrane.

The main role of the plasma membrane is to protect, which facilitates them as barriers and also organises the cell. Both prokaryotes and eukaryotes have a cell membrane which controls what substances can enter and also the quantity of the substances that enters. Unlike prokaryotic cells, the eukaryotic cells possess also internal membranes that can enclose their organelle controlling the exchange of essential components which gives them a specialised structure and also supports their gate keeping function. The cell membranes are semi-permeable which means that they allow only certain molecules to pass through them. Different membrane proteins have different functions/activities occuring on their membrane. Small hydrophobic molecules and gases like CO_2 and O_2 can cross the membrane rapidly and easily while small polar molecules like water and ethanol pass through them very slowly. The non-facilitated passage of charged molecules, such as ions and of large molecules, such as sugars and amino acids through the membrane is energitically strongly disfavoured. Only certain specific transporter proteins spanning through the membrane can transport these molecules at the expense of energy. By doing so, these membrane proteins help in the transmission of electric impulses, catalyse enzymatic reactions, connect neighbouring cells and keeps proteins to at specific location (anchoring)....

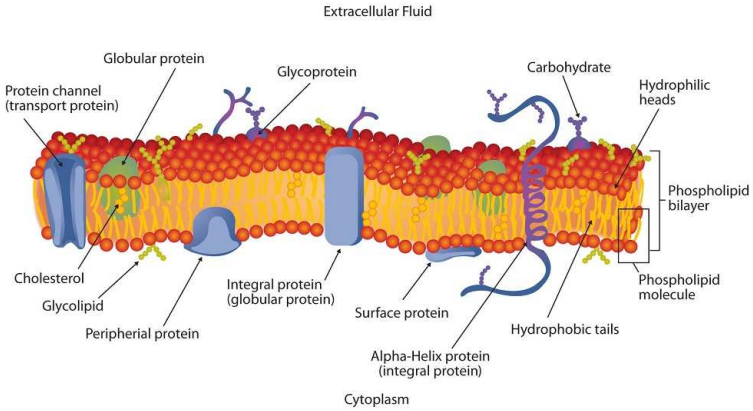


Figure 1.2: A lipid bilayer membrane including peripheral and transmembrane proteins . Figure from[6]

Membrane proteins can be classified into two large categories: Intergral (intrinsic) and Peripheral (extrinsic) based on their interactions(see figure 1.2). Most of the biomembranes have both sorts of these proteins. One part of the integral proteins is embedded in the phospholipid bilayer. On the other hand the peripheral proteins are located on the surface of the phospholipid bilayer and are bound to the membrane either indirectly by interacting with integral proteins or directly to the lipid section of the membrane. Our study is on membrane proteins of gram-positive bacteria, *Staphylococcus Aureu*s, We will focus on the integral transmembrane proteins for now[10].

There are different types of integral membrane transporter proteins assisting the movement of substances by active transport: Channels/Pores (either in open or closed state), electrochemical potential-driven transporters, primary active transporters (ATP-binding cassettes, (V,P,F)-type ATPase ...), group translocators (PEP group translocation) and electron carriers[30]. Most intergral proteins contain multiple transmembrane *α-helices*. They are embedded within the transmembrane region that is hydrohobic in nature. Thus they can form only van der waals interactions with the fatty acyl chains and need to sheild the polar (C=O) and (N-H) groups of the peptide bonds. This means that the amino acid residues can interact only within themselves through hydrogen bonds. Hence, only two structural motifs are possible for the membrane spanning domains of transmembrane proteins: *α-helices and β -sheets*(see figure 1.4). These are connected by external loops. α-helices are commonly found in all membrane proteins and mainly occur on the plasma membrane and endoplasmic reticulum membranes. The beta barrels are restricted to the outer surface of gram-negative bacteria and in the mitochondrial/chloroplast membranes allowing the passive diffusion of small molecules and toxins [13].

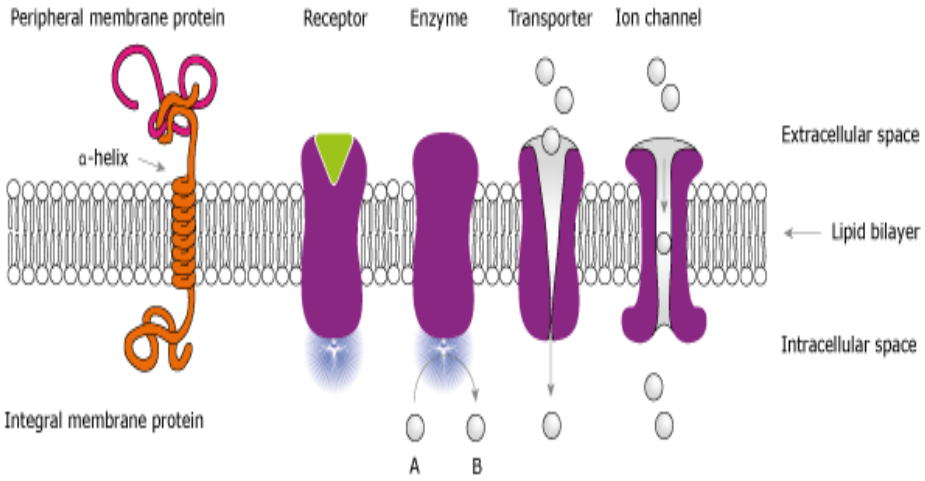


Figure 1.3: Membrane protein functions .Figure from[1]

In our study, we considered bacterial transmembrane proteins from *Staphylococcus Aureus* because this is important human pathogen is of interest in funded project of our group. Being a gram-positive bacteria, we need to consider the α-helical proteins which contain the functional residues embedded within them. Here, we were able to design a tool that can map the PLRs of the transmembrane proteins with the unknown 3-D structures by exploiting the central hypothesis in biology, i.e; functionally important residues are conserved. The next part provides insight into the transporter proteins.

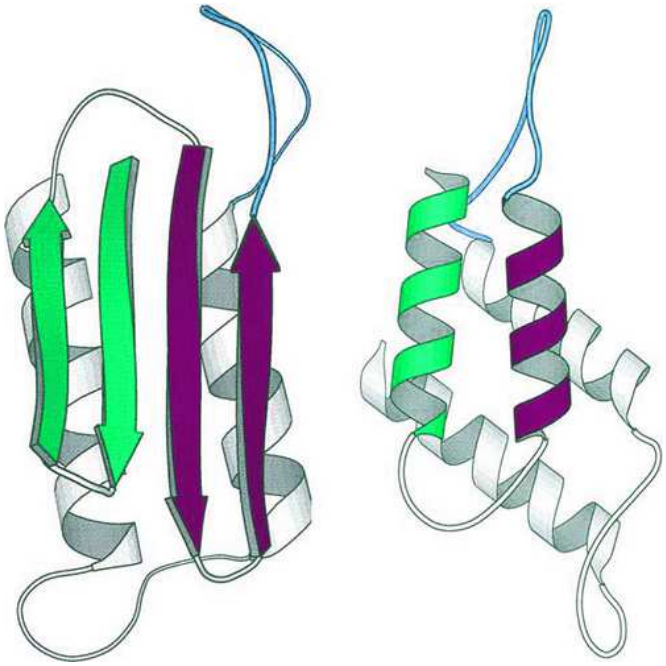


Figure 1.4: α-helical and β-sheets

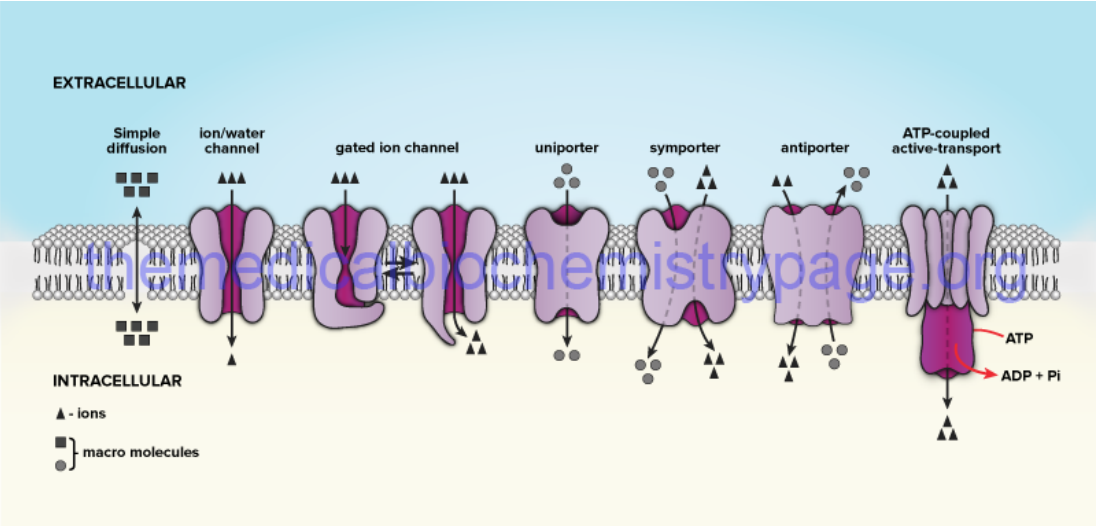


Figure 1.5: Types of transport proteins . Figure from[7]

1.1.2 Membrane transport proteins

As mentioned before, only few small molecules and gases may enter or leave organellar membranes, unaided by proteins. Their movement is mainly facilitated by concentration gradients across the membrane. Other macromolecules and molecules like water and urea need the help of active transporter proteins to cross the phospholipid bilayer.

There are three major classes of membrane transport proteins exhibiting high specificity on the substances transported: *ATP-powered pumps* (called as *pumps*), *Channels* and *Transporters*. They differ considerably in the rate of transport what reflects the difference in tthe source of activation energy (see figure1.5).

- ATP-powered pumps : They use energy from ATP hydrolysis for movement of the ions and small molecules across the membrane. This type of transport is otherwise called as *active transport*.

- Channels : They let water and/or specific ions permeate along their chemical gradient. When open, ions can translocate at a very high rate up to 10^8 per second per channel.

- Transporters : As a group they transport a wide variety of molecules and ions across the membrane. However, most transporters transport only a very specific class of substrates, sometimes only a single molecule.Unlike the channels, they transport only fewer molecules at a time. They undergo conformational changes once the sustrate binds to them, and only the bound substrate molecules get transported. Based on this mechanism, transporters are classified into three classes.

- Uniporters : They are able to transport only one molecule at a time down the concentration gradient. They transport molecules such as glucose and amino acids across the plasma membrane into mammalian cell.
- Symporters : They can accelerate the movement of multiple molecules at the same time, one towards lower gradient across the cell membrane and in the same direction. They are also called cotransporters.
- Antiporters : In contrast to the symporters, this class of transporters can transport ion(s) downhill the electrochemical gradient, allowing other molecules, to move against the concentration gradient[14].

In our study, we considered all the above membrane transporter proteins for the antibiotic resistant bacteria *Staphyllococcus Aureus* strains COL and N315. Now, we shall come to the point why these transporter proteins are important ?

1.1.3 Transport proteins in bacteria: Similarity in their structural makeup

Understanding membrane proteins is a very difficult task. Trying to extract, purify, homogenize or remove them from the membrane into which they are embedded results in a huge loss of information and also protein unfolding . In fact there is a high percentage of non-polar aminoacids and this has in fact resulted in that only a fairly small number of crystallised structures could be determined through x-ray crystallography.

Now, understanding the bacterial membrane proteins has led to very interesting findings. Through recent developments, scientists are trying for single model that can unify among the membrane transporters. So common structural motifs and evolutionary origins among these diverse energy coupling transporters suggests that a central module forming a transmembrane channel through which the solute passes. This would be a proof for the central hypothesis in biology: Functionally important residues are conserved. Pharmaceutical companies are trying to exploit this idea in order to develop drugs which can inhibit the actions of these transporter proteins embedded in the bacterial membrane region.

Being said that, the efflux pumps1.6 present on the anibiotic resistant bacteria become a serious issue. These bacteria are capable of developing their defenses by mutations in efflux pumps to flush the toxins (drugs) out of them. Thus millions of dollars get wasted on developing drugs which ultimately become ineffective[24].

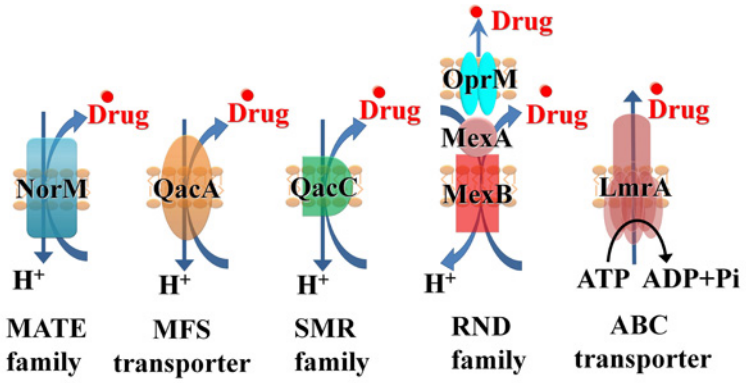


Figure 1.6: Drug efflux pumps in bacteria . Figure from[8]

Therefore, to solve this problem, an in depth understanding of the mechanism of function of this transporter family of proteins is imprtant. Sufficient information on the pores within these proteins and PLRs is the main focus of this thesis. As mentioned initially in the first part, the crystal structures of the pores/channels are important for pore detection tools and methods existing. Such structures are primarily provided by the Protein Data Bank (http://www.rcsb.org/pdb/home/home.do).

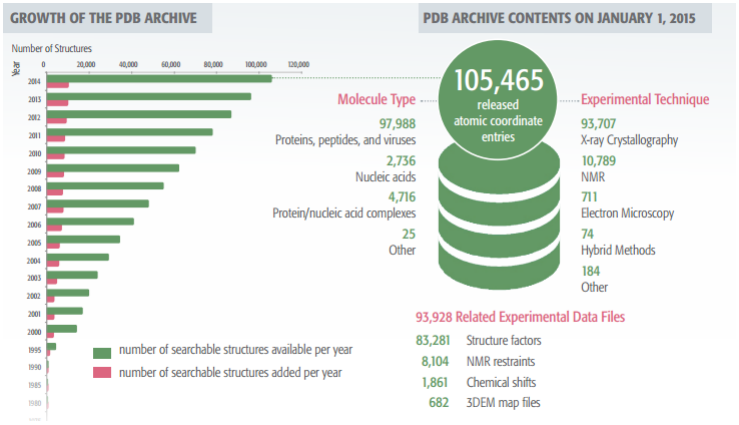
1.1.4 Available data

Originally established in 1996 as a web portal and then 2002 as a fully functional database, *TransportDB* (http://www.membranetransport.org/) maintains information on the cytoplasmic membrane transporters and the outer membrane channels of nearly 365 organisms, whose complete genome sequences are available. We have taken into account the *SACOL* and *SAN315* strains of *Staphyllococcus Aureus* which contain a total of 192 annotated *SACOL* transporters and 314 *SAN315* annotated transporters.

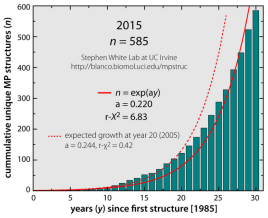
The PDB, established in 1971, provides a systematic database for protein structures (see figure 1.7). There are around 80000 enteries of protein structures available but only about 1572 structures of alpha-helical membrane proteins - data obtained from (http://blanco.biomol.uci.edu/mpstruc/) due to the limitation said before. But, with the available protein primary sequence data, new computaional methods can be developed to predict the pores/channels information.

TransportDB		Stephen White Lab
SACOL	SAN315	Transporter Protein: Alpha-Helical
192	314	1572

Table 1.1: Data available from databases



(a)



(b)

Figure 1.7: (a) Data representing the yearly growth of protein structures.(b) Graph representing the cumulative growth of membrane protein structures obtained from http://blanco.biomol.uci.edu/mpstruc/ .

1.2 Related Works

There are some sequence-based PLR prediction methods available already. Most of my references came from the work of my advisor Duy Nguyen and my suprevisor Prof. Dr. Volkhard Helms. They developed a tool named *PRIMSIPLR*, which predicts the inner-membrane PLRs for α-helical transmembrane proteins[23]. They were successful in developing a Support Vector Machine that distinguishes PLRs from other residues using the protein sequence alone. They were able to acheive a Matthews correlation of 0.41, accuracy of 0.86, sensitivity of 0.61 and specificity of 0.89 which seemed to indicate the predictor is fairly good.

PROPORES, is yet another tool developed by Po-Hsien Lee, which came in handy, where PLRs in small proteins were identified[19]. In this thesis, we developed a tool (in its initial phase) which can map the conserved PLRs on the primary sequence of various transporter proteins of the *Staphylococcus Aureus* bacteria. The methods and results will be discussed in following chapters.

Chapter 2

Materials

2.1 Sequence properties and derived information

Mapping the conserved PLRs and computing the degree of conservation (DOC) of those PLR set within different transporter proteins is the main aim of this thesis. In order to achieve this, it is necessary to extract biological and physicochemical information of residues from protein primary sequence such as E-value, Identity, coverage, conservation score, hydrophobicity ...

2.1.1 Protein Blast - BLASTP

BLAST is a collection of algorithms with variants that finds functional domains or shorter stretches of sequence similarity (protein and nucleotide). Of this set of algorithms, BLASTP compares protein sequences against a protein database. There are four stages of *BLASTP* programs: Hit detection, Ungapped extension, Gapped alignment, Gapped alignment with traceback[16]. Here, the users have the option of using either the *BLAST* standalone program (BLASTP 2.2.29+) which is downloaded and installed from the website ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/LATEST/ or by using a web-based tool (http://blast.ncbi.nlm.nih.gov/Blast.cgi), with default setting set according to requirements. For this thesis we used the standalone program.

2.1.2 Clustal Omega

Estimation of conservation is done with the help of Multiple Sequence Alignment (MSA) programs. There are various programs being listed in EBI website(http://www.ebi.ac.uk/Tools/msa/). For this thesis, the latest version of Clustal Omega (ClustalO) was used. ClustalO is an accurate program for pro-

teins of divergent organisms. It outperforms other programs in its execution time and quality[12]. Another important feature includes the feature of adding newer sequences and using the precomputed alignment information. The standalone version can be downloaded and installed from (http://www.clustal.org/omega/) and the web-based tool is available in http://www.ebi.ac.uk/Tools/msa/clustalo/. The input for ClustalO is a fasta input of the amino acid sequences (see figure2.1).

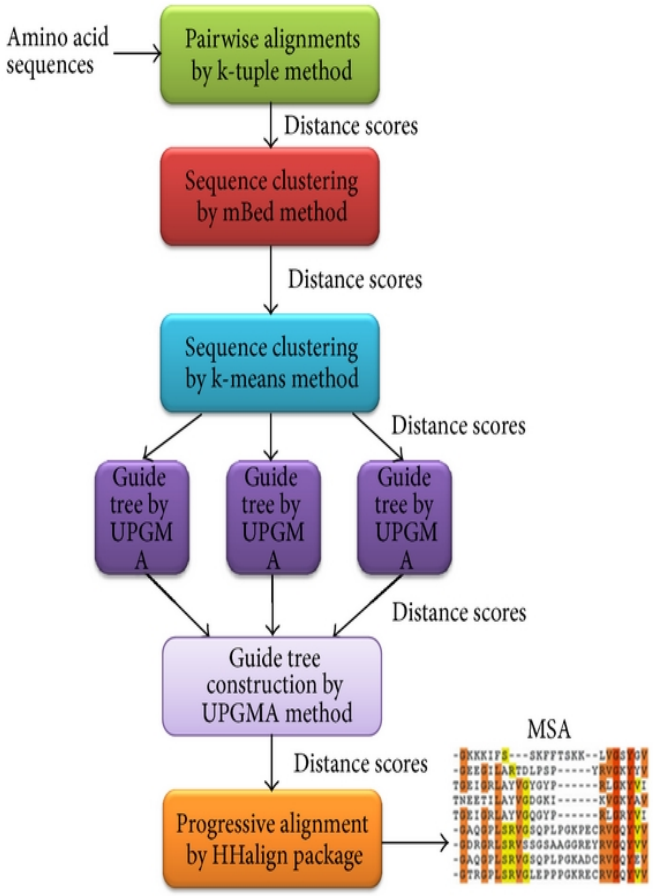


Figure 2.1: ClustalO algorithm . Figure from[17]

The alignment output contains different colours which indicates their physicochemical properties. They include: *RED*: Small (small + hydrophobic (including aromatic -Y)), *BLUE*: acidic, *MAGENTA*: Basic -H, *GREEN*: Hydroxyl + sulfhydryl + amine + G, *GREY*: Unusual amino/imino acids.

2.1.3 Rate4Site

Many proteins have a known 3D structure, yet their function is unknown. Rate4Site is an established method which maps the rate of evolution of homologous proteins to a known 3D structure[28]. Maximum likelihood principle is

calculated at each position of the protein sequence and evolutionary conservation is calculated. The input for Rate4site is a MSA e.g. the one from ClustalO which contains the aligned protein sequences with known and unknown 3D structures(see figure2.2).

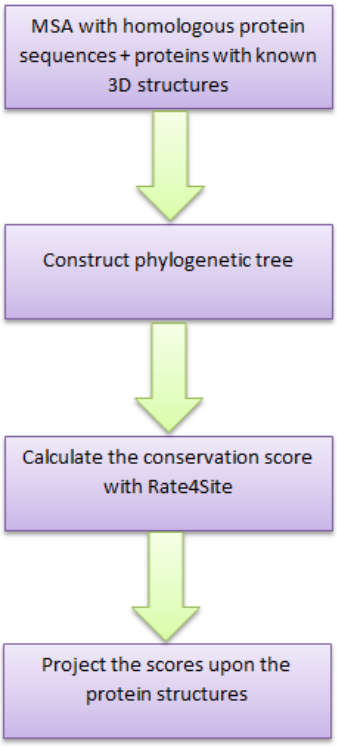


Figure 2.2: Rate4Site Flowchart

If we consider two sequences of M positions, we assume that the average distance betwen both these sequences is 'l'. Then the number of replacements expected at each site is :

$$l * r[j]$$

, where $r = r$[j] is the rate parameter at that position. Rate4Site finds the maximum likelihood estimate of this rate. The rate at a specific site is given as:

$$P(\{X, Y\}, data|r) = \Pi_x * P_{(x,y)}(r * t)$$

,where 'X' and 'Y' are two amino acids. Π_x is the frequency of amino acid 'X', P(x,y) is the frequency of amino acid 'X' replaced by amino acid 'Y' along the branch 't' given the rate is 'r'[28]. For the purpose of this thesis the Rate4Site (version 2.01) standalone program was used which can be downloaded and installed from http://www.tau.ac.il/∼itaymay/cp/rate4site.html.

2.1.4 Kyte-Doolittle Hydropathy plot

Since this thesis deals only with transmembrane proteins, there is a requirement of classifying the whole transmembrane protein into hydrophobic and hydrophilic regions. For this, in this thesis we used the Kyte-Doolittle Hydropathy scoring method[18]. It uses the sliding segment approach thereby computing the average hydropathy within a segment (e.g. a stretch of 20 amino acid residues) of the protein sequence. The scores are then plotted from the amino to the carboxyl terminus in accordance to the scores. The following table shows the hydrophobicity scores of the amino acid residues(see table2.1). A positive score suggests that a residue is a part of the α-helix spanning the lipid bilayer.

Amino Acid	Hydrophobicity score
ALA	1.800
ARG	-4.500
ASN	-3.500
ASP	-3.500
CYS	2.500
GLN	-3.500
GLU	-3.500
GLY	-0.400
HIS	-3.200
ILE	4.500
LEU	3.800
LYS	-3.900
MET	1.900
PHE	2.800
PRO	-1.600
SER	-0.800
THR	-0.700
TRP	-0.900
TYR	-1.300
VAL	4.200

Table 2.1: Hydrophobicity Scores

A midpoint line (normally a red line) in the plot separates the hydrophobic from the hydrophobic regions in the protein[28]. The web-based tool is available at http://web.expasy.org/protscale/.

2.2 Transmembrane Protein Databases

The transmembrane (TM) protein data in our study were downloaded from three membrane protein databases: (i) All transporter proteins

for *SACOL* and *SAN315* strains of *Staphylococcus Aureus* compiled in *TransportDB* (http://www.membranetransport.org/), (ii) All α-helical proteins list present at *Stephen White (SW) laboratory* at *UC Irvine* (http://blanco.biomol.uci.edu/mpstruc/), and (iii) Orientation of the proteins in the membrane region from (OPM) database (http://opm.phar.umich.edu/) [22].

2.2.1 TransportDB

TransportDB describes the predicted membrane transport part of those organisms whose whole genome is available. Here , the membrane transport region is identified and categorised into protein families in accordance with the Transporter Classification (TC) system. For this thesis, we used Staphylococcus Aureus strains, SACOL and SAN315. Both these strains have the following transporter families in them (see table2.2).

Transport Type	Transporter family	Number of proteins
ATP-Dependent		
	ABC	51
	F-ATPase	1
	P-ATPase	2
Phosphotransferase system (PTS)		
	SSPTS	16
Secondary Transporters		
	DAACS	2
	MFS	25
	POT	1
	Trk	2

Table 2.2: *Staphylococcus Aureus* Transporters. The coloured families are considered for this study. *Red* ones represent the larger transporter families, while the *Green* ones represent the smaller transporter families.The ATP-binding Cassette (ABC),The H+- or Na+-translocating F-type, V-type and A-type ATPase (F-ATPase), The P-type ATPase (P-ATPase), Sugar Specific Phosphotransferase System (SSPTS) and The K+ Transporter (Trk) are the larger family transporters. The Dicarboxylate/Amino Acid:Cation (Na+ or H+) Symporter (DAACS), The Major Facilitator Superfamily (MFS) and The Proton-dependent Oligopeptide Transporter (POT) are the smaller family transporters

2.2.2 Stephen White's Laboratory Database

The membrane proteins of known 3D structure (http://blanco.biomol.uci.edu/mpstruc/) is a database that was created and maintained at the laboratory of Prof. Stephen White at University of California, Irvine. This database contains information on the available protein

structures. It is mainly divided into 3 sections: Monotopic membrane proteins, Transmembrane proteins: Beta-barrel, Transmembrane proteins: Alpha-helical. The membrane proteins are classified differently in each of the sections based on the function of the protein. For the scope of this study, only α-helical TM proteins were taken into consideration.

2.2.3 OPM Database

The OPM database predicts the spatial arrangement of the membrane proteins which are available in PDB. Introduced in 2006, this database has a number of features like sorting, analysis and searching the membrane proteins based on :

- Structural classification, species, destination membrane.
- Total number of transmembrane segments.
- Number of secondary structure elements.
- Calculation of the hydrophobic thickness.

The membrane proteins submitted into this database were aligned along the z-coordinate which is perpendicular to the lipid bilayer using the method developed by Lomize[22]. This is the most important feature of the OPM that is used in this study. For this study , all Alpha-helical TM proteins (biotpic TMproteins and polytopic TM proteins) were downloaded. The web server is available at http://opm.phar.umich.edu/server.php.

2.3 Pore Detection Tools

Identifying the pores/channels in the TM proteins is essential to derive the PLRs. Also the PLRs embedded in the helical TM proteins which are facing towards the pores/channels are essential for this study. This can be achieved through computational approaches. PROPORES, developed by Po-Hsien Lee[27] served the best purpose of identifying the PLRs for smaller transporter proteins, while another tool named PoreWalker, developed by Pallegrini-calace[21] was used for detecting PLRs for larger transporter proteins.

2.3.1 PROtein PORE identification toolS(PROPORES)

PROPORES is a toolkit for identifying pockets, channels and cavities in a protein structure. This feature of PROPORES make it different compared to PoreWalker, which can identify only the channels. One advantage is that methods

like POCKET[20], LIGSITE[15] and dxTuber[29] like PROPORES are grid-based methods but give an undesireble orientation dependency, which is completely avoided in the case of PROPORES. Another advantage of PROPORES is that it automatically detects the pores and doesn't require specific information on the geometry and residue position as is the case with methods such as PASS (for detecting pockets) or CAVER[26], CHUNNEL[9], MOLE[25]...(for detecting channels). The only requirement is to input the target protein in PDB format. For this study, the PDB output from OPM is given for PROPORES.

PROPRES toolkit comprises of 3 modules:

- PORE_ID for identifying the pores.
- PORE_Trace for pore axes determination.
- Gate_Open for opening the gate between neighbouring pores.

For this study, the results of PORE_ID which is the pore volume, PLRs, grid coordinates is taken into consideration for further analysis and mapping which will be discussed in the coming chapters. The PROPORES package written in Perl can be downloaded and installed from http://gepard.bioinformatik.uni-saarland.de/software/propores/propores-page.

2.3.2 Pore Walker

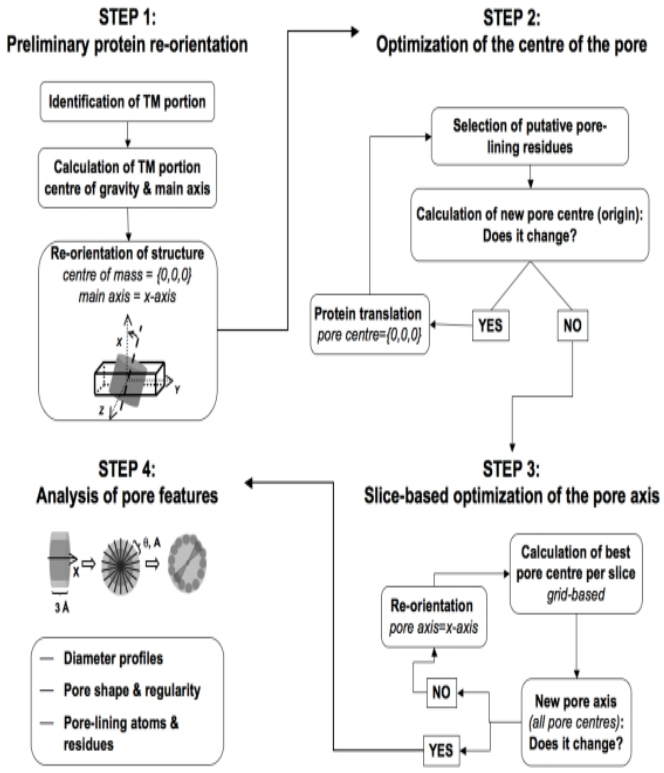


Figure 2.3: Flowchart of PoreWalker algorithm .Figure from [21]

PoreWalker is an automatic method with the goal of identifying the channels in TM protein through which small molecules or ligand may pass. This is a heuristic and iterative algorithm as explained in the diagram (see figure 2.3). PoreWalker was in principle an ideal method for this thesis, but since its available only as a web tool (http://www.ebi.ac.uk/thornton-srv/software/PoreWalker/), we generally used PROPORES. PoreWalker was used for identifying the PLRs in larger TM protein families.

2.4 FPRA:Functional PoreResidue Annotation

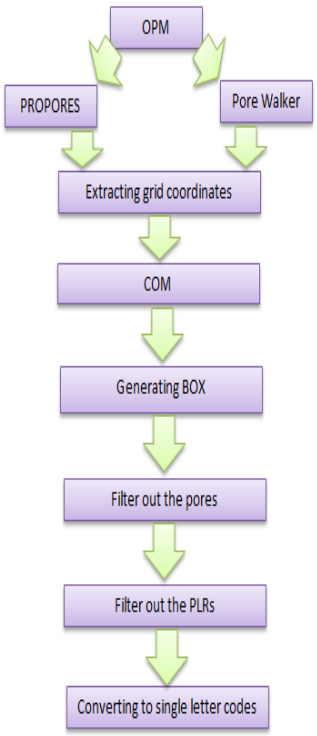


Figure 2.4: FPRA workflow

Identification of PLRs in the hydrophobic region is the key assumption in this thesis. It requires a computational method that can map the conservation of these PLRs on the protein sequence of the TM protein with unknown 3D structure. This is the main role of the tool FPRAT which is developed here. The key features of this tool include (see figure 2.4):

- Extract the coordinates of the residues to a text file for each OPM output as "PDBname_ATOM".
- Calculate the Center Of Mass (COM) of the coordinate file of the z-oriented OPM file (this that the axis normal to the assumed plane of the lipid bilayer

surrounding the membrane protein is aligned with the z-axis of the cartesian coordinate system).

- Create a box with COM as its center and spanning:$-15.0\mathring{A} <= z <= 15.0\mathring{A}, -1.0\mathring{A} <= x <= 1.0\mathring{A}, -1.0\mathring{A} <= y <= 1.0\mathring{A}$. This is par with the OPM definition of hydrophobic bilayer thickness.
- Output the coordinates of the box as a text file "PDBnameout_box".
- Filter out the pores identified using PROPORES.
- Parse out the PLRs and convert them to single letter codes.

Chapter 3

Methods

3.1 FPRAT Pipeline

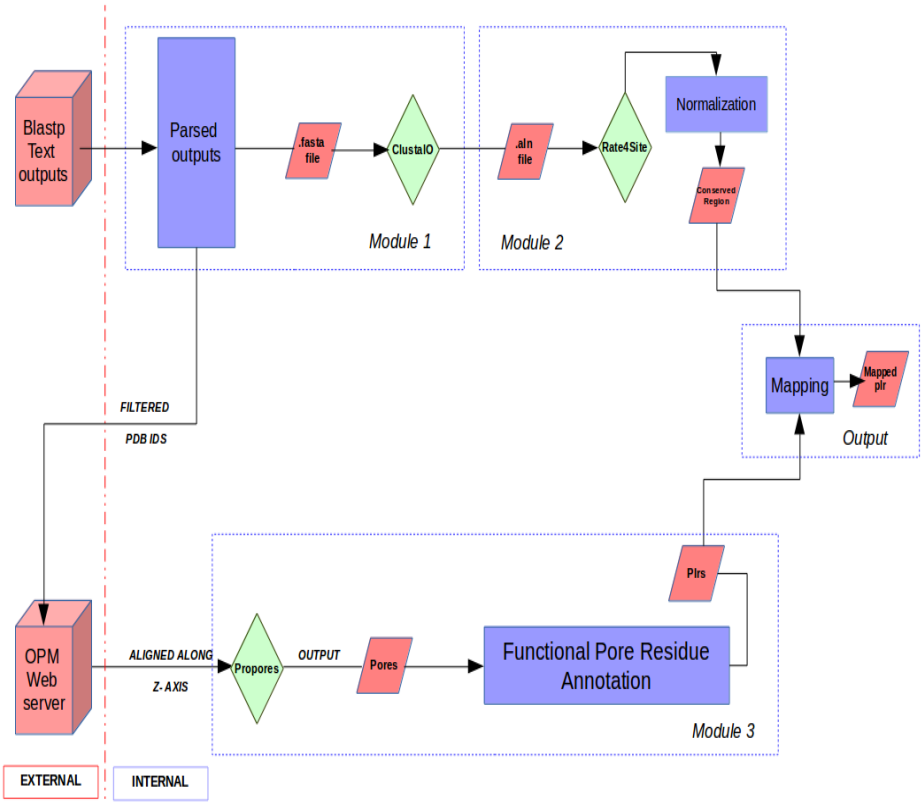


Figure 3.1: FPRAT pipeline

FPRAT is a tool (see figure 3.1) that can map the conserved PLRs on the protein sequence. The conserved PLRs are obtained from data about the pore lining residues obtained through the functional modules described below:

1. Module 1: Firstly, the data collection is done. The SACOL and SAN315 transporter sequences, along with the X-ray crystallographic structures of

helical membrane proteins are collected from the databases transportDB and SW's laboratory, respectively. Both these datasets are then subjected to pairwise sequence alignment using the BLAST standalone program. MSA is carried out using ClustalO on the filtered favourable hits for further processing in module 2.

2. Module 2: The MSA files obtained from Module1 are then given as input to the Rate4Site program with the base sequence as the *Staphylococcus Aureus* transporter sequence. The rate scores obtained from Rate4Site are normalized using Feature scaling/Unity-based normalization. All the scores which are generated are rescaled to the range [0,1]. The lower scores means higher conservation whereas the higher scores means lower conservation. These scores are then assigned to a colour grade according to the colour scheme 1-9 as shown below(see figure3.2). If the interval in the specific position falls in the first three bins, then those residues are considered to be conserved. Those that fall into four or higher bins are considered not conserved.

3. Module 3: OPM data processed by PROPORES and PoreWalker were used to filter out the pores and PLRs using the FPRA progam.

4. Mapping: The conserved set of residues obtained from Module2 is then compared with PLRs set obtained from Module3 using specific position information and mapped onto the base sequence. In this way the FPRAT tool helps to find the conserved PLRs on the transporter sequence for future analysis.

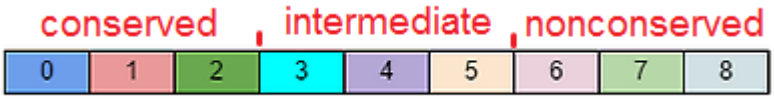


Figure 3.2: Colour Scheme

3.2 Data Collecting

The list of *Staphylococcus Aureus* transporter protein sequences in the fasta format is downloaded from the transportDB website. Also the list of alpha-helical transporter proteins are downloaded from SW's website. Both lists are then subjected to sequence homology using BLAST 2.2.29+ program against the NCBI repository(nr + pdbaa) (see table 3.1). The favourable hits from both the lists are then processed through a series of python scripts in the modules explained

before. There is a python script to search all the favourable hits in the OPM entries. The list comprises of larger and smaller transporters. These are then used for the next phase: Data processing.

transportDB		StephenWhite Lab
SACOL	SAN315	Transmembrane Protein: Alpha-Helical
192	314	1572

Table 3.1: Number of membrane transporters used from different sources

3.3 Data Processing

3.3.1 Module 1: Filtering hits

Data Parsing

Data parsing is done in each functional module by separate python scripts. Parsing is important at every step as the results of each step are carried for processing to the next one which leads to the fetching of the desired mapping result using FPRAT. The different parsers used at each of the modules are explained below:

Firstly, in Module 1 the Blastp parser does a two level parsing step to filter the hits:

- Level 1 parsing extracts $<=$ 500 homologues from all organisms.
- Level 2 parsing extracts homologues based on fixed thresholds being set in the parser (see algorithm 3.1):

    1. Expect - value (E-value): The expect value (E) is a parameter that describes the number of hits of the same or higher score, one can expect to see by chance when searching a database of given size. For this thesis, E-value was set to $<=$ 1e - 10, which means it has a very low probability of occuring ramdomly. This in fact helps in finding highly similar sequences in closely related species.
    2. Identity: Identity is defined as the extend to which two (nucleotide/amino acid) sequences have the same residues at the same positions in an alignment , often expressed as percentage. For this thesis, Identity $>=$ 35%, which means that there are 35 percent or more residues that are identitical witht he query sequence.
    3. Coverage (cov) : Coverage is defined as the percentage of query sequence that overlaps the subject sequence. it is given as, cov = (length

of alignment / length of query sequence). For this thesis, cov was set to > 70 %.

Algorithm 3.1 Parsing the data

```
#Blast_parser to parse out the favourable hits
file = #read the Blast2.2.2+ output text files
loop over file[:]:
   hit = file[column with filtered PDBIDs]
   e = file[column with E-value]
   ids = file[column with Identity percentage]
   cov = file[column with Coverage percentage]
   #The threshold values are being set as follows
   if e <=1e-10 and ids >= 35 and cov >= 75:
      #store the hits for further processing
      if len(hits) < 10: #make sure there are at least 10 hits
         print 'not match'
#MSA_parser to prepare the input for MSA
input = [Blast parser output files]
def seqfilter():
   files = open(input)
   loop over files[:]:
      content = #read the files
def blancofilter():
   blanco = open(blancoDB proteins fasta files)
   loop over blanco[:]:
   with open(write the blancoDB PDB ids) as outfile:
      outfile.write(pdbs)
def msafileprep():
   new_file=["S.AureusCOL_Prot_Blast.txt","S.AureusN315_Prot_Blast.txt"]
   loop over new_file[:]:
      new_files = open(new_file)
      loop over new_files[:]:
         values = #the input files in fasta format being prepared
         prot_values = [values'.fasta']
   fasta_files.write(write fasta files to the parent directory)
```

Secondly the MSA parser fetches the transporter sequence (query sequence) along with blanco sequences (Structure list) and NCBI sequence (sequence list) as a single fasta format file. This fasta file is then used as input for the ClustalO program.

3.3.2 Module 2: Generate list of Conserved PLRs

transportDB list extraction

After the MSA files are generated we proceed to Module 2. Firstly in this module, we extract those transporter families of interest and generate

the subfolders in their respective names using the transportDB parser program. The parser helps in extracting the transportDB list from its url :http://www.membranetransport.org/all_type.php?oOID=saur2 and copies the respective MSA file into the subfolders for further processing by the Rate4Site program. The input for Rate4Site has to be generated in a specific format with the query sequence (the transportDB sequence) at the beginning (index 0) of the MSA file. This query sequence will be the base sequence on which the Rate4Site program caluclates the rate scores.

Rate4Site processing

This is the main step of Module 2 of FPRAT. After the Rate4Site standalone program is run and the scores are generated upon the base sequence, the parser described ectracts informations (sequence position as per Rate4Site program (see algorithm3.2), and the conservation scores, the confidence interval (QQ)..) which is essential for understanding the evolutionary relation among the homologues in the MSA. This is an important step for normalization of the scores and extracting the conserved set of residues from the query sequence.

Algorithm 3.2 Rate4Site parser

```
def rate4site():
#Section 1 : Rate4Site parsing
   loop over rate_files in parent directory[:]:
      condition 'rate files' in file:
         query = query sequence
         loop over rate_files[:]:
            sequence = rate_files[Column for residue id information]
            residues = rate_files[Column for residue alphabet]
            score = rate_files[Column for rate4site scores]
            #store the rate scores for each residue of the query
                sequence
            range = [list of confidence interval which is (-QQ,QQ)]
            range_diff = -QQ + QQ
            data_reqs.append([score,range_diff])
#Section 2 : Normalization step.
loop over the scores[:]:
   rescale the scores to unit normalization [0,1]
   convert the normalized score to 1-9 color bins and drop the scores
       to each bin
   bin1-3 = most conserved
   bin4-6 = intermediate conserved
   bin7-9 = least conserved
rate4site()
```

Normalisation

What is normalization? When approaching the data for modelling, we need some standard procedure that prepares the data. This approach is normalization or standardisation bringing all the variables in proportion with one another. Traditionally, data normalization and data standardisation means to fit the data within unity (1), so that the data values (in this case rate scores) will take a value from 0 to 1.

Normalization is very important when there are parameters of different units and sclaes. The rescaling of data is done by the formula:

$$X_{new} = X - X_{min}/X_{max} - X_{min},$$

where X is the data values, X_{min} is the minimum of the data values and X_{max} is the maximum of the data values. Once the normalized scores are generated, we give colour schemes to them as explained previously. The normalized conservation scores are then divided into a discrete scale of nine bins for visualization, from the most conserved bin (grade 0 to grade 2) colored blue, red and green, through intermediately conserved bin (grade 3 - grade 5) colored cyan, purple and the least conserved bin (grade 6 - grade 8) colored violet, lime, aqua. This is the best way to determine the conservation of residues in the query sequence.

3.3.3 Module 3 : PLR prediction

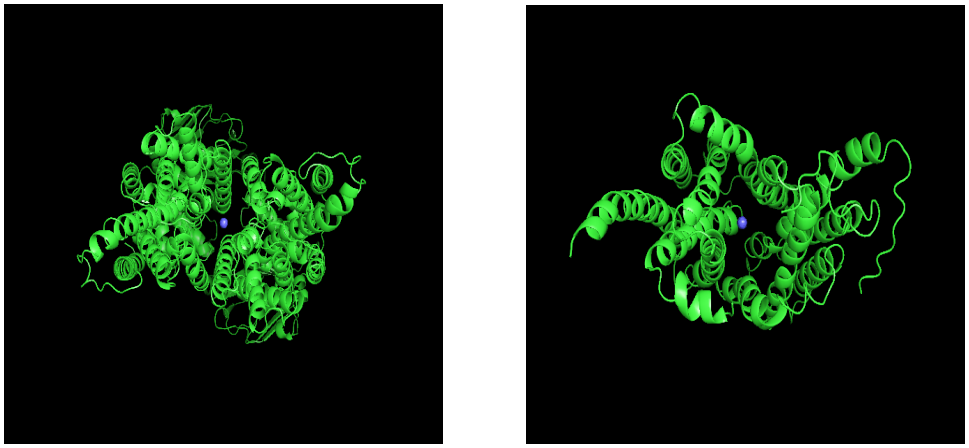
Transport Families		Function	Total	
Processed by PRO-PORES(Smaller Proteins)	DAACS	Homotrimer	6	9
	MFS	Monomer	1	
	POT	Homodimer	2	
Processed by Pore-Walker(Larger proteins)	ABC	Homodimer/Heterodimer	68	101
	F-ATPase	Homodimer	29	
	P-ATPase	Enzyme Complex	2	
	SSPTS	Enzyme Complex	1	
	Trk	Homotetramer	1	
Total				110

Table 3.2: Transporter Protein Families

The total of 110 helical TM proteins obtained from SWs and OPM are then processed by the tools PROPORES (for smaller protein structures) and PoreWalker (for larger protein structures) to identify the pores and PLRs (see table 3.2). From Pore Walker results it was clear that they filtered the beta-carbon (CB) for indentifying the PLRs. This idea was used in this thesis to filter the CB from

the PDB file and reduce the size of the PDB file for PROPORES to process fast. The reason behind filtering out the beta carbon atoms is that these atoms are in direct contact with the pore region when present in the side chains. This step makes it very easy for PROPORES to obtain the pore information for larger protein families. Still after reducing the file size , there were certain protein families which could not be processed using PROPORES as listed in table 4.4.

Here the proteins under consideration have different structural organisation ranging from monomers to homodimers and this defines them as a single functional unit. This concludes the fact that the functional pore is at the centre region of TM. Now in this thesis we come up with a method to identify this functional region. Firstly, after the pores are identified by PROPORES program, COM program and Box programs of Module 3 computes the COM of the respective protein (meaning the centre of the coordinates. See figure 3.3) and generates a box with COM as its centre, respectively. The box spans $\pm 15\mathring{A}$ along the z-axis, while $\pm 1\mathring{A}$ along the x-, y - axes. This will be considered as the function region as far as the TM protein is concerned.



(a) ABC (b) MFS

Figure 3.3: Location of functional pore (blue dot)

Identifying Functional Pores

Since in this thesis, the functional pores and PLRs of protein are the only concern, the conserved residues identified in Module 2 don't solve the problem. Also, there are many pores within the transporter protein which are identified and indexed according to decreasing order of their volume, by PROPORES. From figure 3.4, it is likely that the pink and orange colored pores are a part of the channel as they expand through the structure. On the other hand, the grey colored pore is located on the outside of the membrane region, meaning that it is not a functional pore. Now we propose a methodology to identify functional pores as well as to

identify the PLRs .

From understanding the nature of the TM proteins, it was clear that the functional pore is likely to be located at the center of the protein spanning perpendicular to either directions. From the observation, we constructed a box/pseudo-pore for each of the proteins. This pseudo-pore is a cuboidal box which expands from the COM of the protein. The total height of this box is taken as 30Å. This value comes from the fact that the width of the lipid bilayer is about 28Å to 30Å[11] and the functional pore spans the membrane completely. The width of this box was set to 1Å, so that the exterior pores donot overlap with the pseudo-pore. If a particular pore overlaps the pseudo-pore, that pore is considered as a functional pore. The length of the pore is defined as the length of the overlapped part. As representatives for each protein, those pores that overlap this pseudo-pore are taken into account. We were able to remove many unwanted pores through our method and obtained a total of 59 representatives as shown in 4.2 and 4.3.

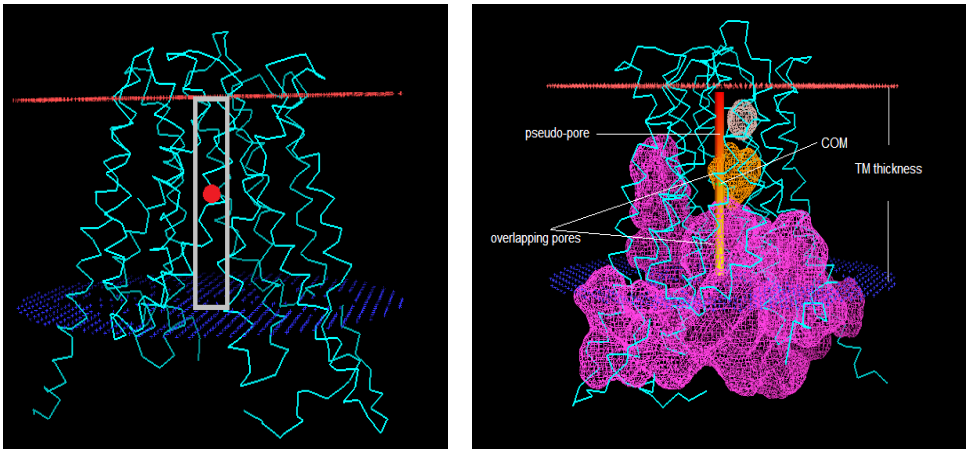


Figure 3.4: Three pores (pink, orange and grey) detected by PROPORES on 1PW4 with pseudopore.

Problematic cases

- Normally PLRs and non-PLRs appear alternately in a helical turn with respect to the pore/channel, due to the periodicity of 3.6 residues per helical turn. In our results, we see continuous stretches of PLRs. The only explanation to draw from this that some of the helices are completely engulfed by the pores. This is one of the drawbacks with PROPORES calculations concluding that it is not fully optimized to detect channels in TM transporters and pores (see figure 3.5).

- Another major drawback with PROPORES is that the running time is very high and requires efficient processors and RAM, especially if the size of the protein is large and complex.

– As mentioned earlier, FPRAT works with PROPORES for identifying the PLRs in a protein. But the major drawback with PROPORES is that it doesn't produce results for larger proteins (>= 15000 atoms). This causes a problem for finding PLRs using PROPORES on larger protein families. In such cases, we used PoreWalker to identify PLRs on the larger ABC protein 1l7v. Here we found that PoreWalker actually extracts the beta carbon atoms of the residues embedded on the helix that faces the pore. This gave us the idea to extract only the beta carbon ATOMS in OPM pdb file. By doing so, the size of the pdb file is reduced so much and made it easy for processing using PROPORES. But again because of the slowness and complexity of PROPORES, we still found difficulty with certain families like Trk, SSPTS to process. The table 4.4 shows the proteins of the larger family set that we couldn't process.

3.3.4 Mapping

Algorithm 3.3 Mapping the conserved positions and PLRs on the protein sequence

```
#start module 4: mapping the PLRs and Conserved residues on the
    primary sequence of protein"
def mapping()
   loop over files[:]:
      fetch 'plr file' in files
      fetch 'alignment file' in files
      fetch 'PDB file' in files:
      fetch 'MSA_input file' in files:
      fetch 'conserved file' in files:
      loop over each of the fetched files:
         #Convert to single letter codes for the residues
         amino_name =['three letter codes']
         code = [single letter codes]
         make the indexing correct for each of the residues for mapping
         loop residues in conserved file:
            line1 += add the fist line for conserved residues
         line += next line
         loop over the residues in plr file:
            line2 += add the second line for plr residues
         line += next line
         loop over residues in query :
            line3 += add the third line of the protein sequence
         write the final file with all the three lines
mapping.close()
print "End of Module 4"
```

In Module2, the rate scores are generated for the transportDB sequence which are the primary protein sequences without any indexing for itself. After running the rate4site program, a pseudoindex is given to the residues. Now if we see the Module3, the PDB files generated from OPM as index of its own. Now our aim is to match the index of the PLRs filtered out using FPRA to that of the primary sequence. This involved series of computational tasks and we were successful in mapping the PLRs and the conserved residues on the primary sequence of the transporter proteins. (see algorithm 3.3).

3.4 Degree Of Conservation(DOC)

Most of the soluble proteins (in this case the transporter proteins) are active and structurally small oligomers. Statistical surveys of oligomeric proteins have defined their roles of hydrophobicity and complementarity in the stability of proteins interfaces [31]. Oligomeric proteins have complex, convoluted structures with extended arms, deep groves and loops, surrounding the neighbouring chains. These chains join together and form globular domains instead of individual domains. These chains fold together to form stable homomers.

These interwined homomers comprises of identical protein subunits. The surfaces of the contact are highly hydrophobic. These domains are conserved across the protein families we studied.

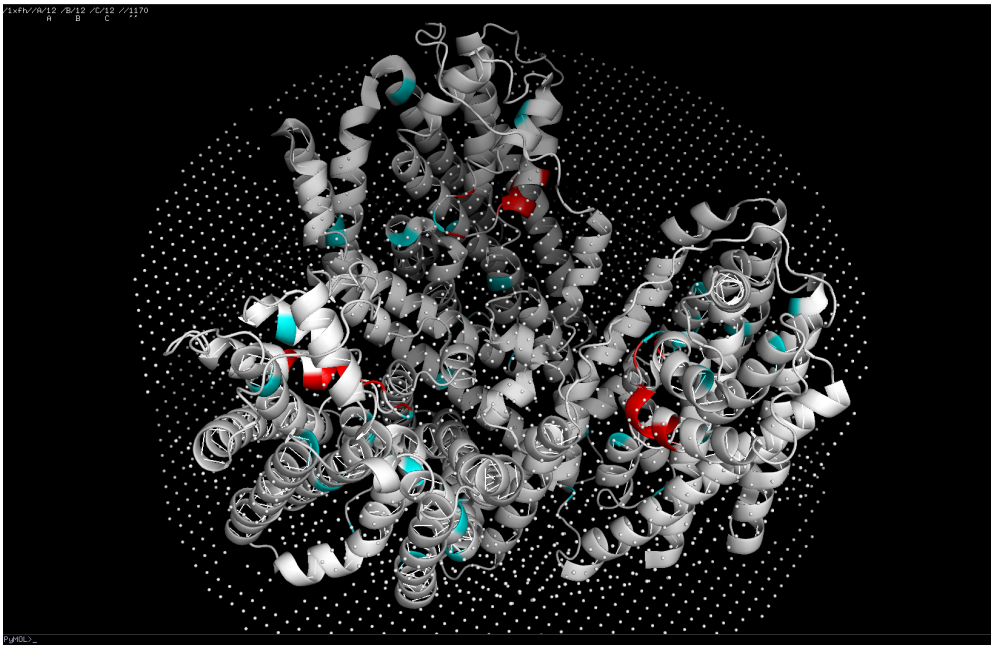


Figure 3.5: Mapped conserved PLRs in cyan and PROPORES anomaly of 13 residues at a stretch as PLRS in red

3.4.1 Conserved PLRs vs Conserved Non-PLRs

Now in this thesis, we performed DOC on the conserved set and gave a detailed statistical insight to decipher the evolutionary constraints at the functionally important set. Initially we compared the DOC of the conserved PLRs to that of the conserved non-PLRs. In our thesis, we have taken the transporter families of *Staphylococcus Aureus*. These transporters have the central domain in their pore which has to be conserved as per our hypothesis. This means that PLRs in the pore are mostly conserved or >= 50% conservation. From our observations most of the proteins tend to follow with some exceptions like 3WME, 4F4C,3RFC... where none of the PLRs are conserved and hence DOC is null.

3.5 Hypothesis testing

Hypothesis testing is a common method which uses the statistical evidence from a sample in order to draw a conclusion on the samples. Here the hypothesis statement asserted is call null hypothesis, H_0. In this thesis H_0 is the PLRs and Non-PLRs which do not display any disparity in their conservation. By default $H_0 \longrightarrow \mu_0 = \mu_1$

,where μ_0 and μ_1 are the mean of both the normal distributions. The argument statement is given as $H_1 \rightarrow \mu_0 \neq \mu_1$.

3.5.1 P-value Assumption

A P-value is used for quantification of the evidence in a null hypothesis (reductio ad absurdum) . A statistically significant evidence is found by rejecting the null hypothesis. The null hypothesis is assumed to be a standard normal distribution (0, 1). The rejection is either by showing (i) that the mean is not zero, (ii) the variance is not unity, or (iii) the distribution is not normal, depending on the test performed. The P-value is defined as the probability under the assumption of hypothesis H, of obtaining a result equal to or more extreme than what is actually observed. The most extreme condition being $\{X \geq x\}$(right-tailed event) or $\{X \leq x\}$(left-tailed event). Smaller condition $\{X \leq x\}$or $\{X \geq x\}$(double-tailed event). Thus the P-value is given as (see figure 3.6):

- $\Pr(X \geq x|H)$ for right-tailed event.
- $\Pr(X \leq x|H)$ for left-tailed event.
- 2 $\min\{Pr\,(X \leq x|H)\}\,, \{Pr\,(X \geq x|H)\}$ for double-tailed event.

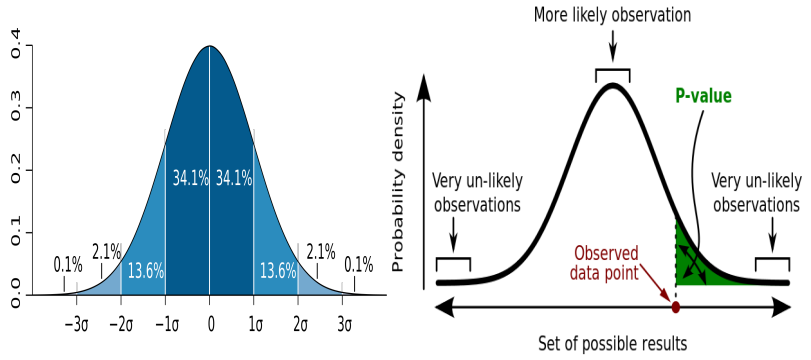


Figure 3.6: Standanrd deviation and P-value . Figure from [4][5]

Wilcoxon signed-rank test

Wilcoxon signed-ranked test is a hypothesis test used when we compare two related/matched samples (in our case the conserved PLRs and conserved non-PLRs) in order to asses if their population mean ranks differ. In other words, it is also called a paired difference test. This test is used as an alternative to the paired student's test, t-test on matched pairs or the t-test on dependent samples (where the population is not normally distributed). In order to use this test, we assume the following assumptions:

- The data is paired and come from the same population.
- Each of the pair is randomly taken and is independent to each other.
- The data are measured on an ordinal scale.

If N is the sample size, i.e, the number of pairs, Then there are total of 2N data points. For i =1,...,N, let $x_{1,i}$and $x_{2,i}$ denotes the measurements.

- For i = 1,.....,N, we calculate $\mid x_{2,i} - x_{1,i} \mid$ and sgn$(x_{2,i} - x_{1,i})$, where sgn is the sign function.
- Exclude the pairs with $\mid x_{2,i} - x_{1,i} \mid = 0$. Let $N_r$be the reduced size.
- Order the remaining N_r pairs from smallest absolute difference to largest absolute difference $\mid x_{2,i} - x_{1,i} \mid$.
- Rank the pairs, starting from the smallest (given a value 1). Ties get a rank which is equal to the average of the ranks they span. Let $R_i$denote the rank.
- The test statistic is calculated as :

$$W = \sum_{i-1}^{Nr} [sgn\,(x_{2,i} - x_{1,i}) * R_i], \text{ the sum of sumed ranks}$$

- Under the null hypothesis, W follows a specific distribution with no simple expression. The following distribution has an expected value 0 and variance of $N_r\left(N_r+1\right)\left(2N_r+1\right)/6$. W can be compared with a critical value from a reference table. The two-sided test consists in rejecting H_0, if $\mid W \mid \geq W_{critical,N_r}$.

- As $N_r$increases, the sampling distribution of W converges to a normal distribution. Thus,

  For $N_r \geq 10$, a z-score can be calculated as $z = W/\sigma w$,

  $$\sigma_w = \sqrt{N_r\left(N_r+1\right)\left(2N_r+1\right)/6}$$

  If $\mid z \mid > z_{critical}$then reject H_0(two-sided test). Alternatively one-sided tests can be realised with exact or the approximate distribution and P-value can be calculated.

Hypergeometric testing

In probability theory and statistics, hypergeometric distribution is a discrete probability distribution which decsribes the prbability k successes in n draws from a finite population of size N that contains K successes, wherein each draw is either a success or a faiture.

A random variable X follows the hypergeometric distribution if its probabilitymass function (pmf) is given by:

$$P(X=k) = \binom{K}{k}\binom{N-K}{n-k} / \binom{N}{n}$$

, where N is the population size, K is the number of success states in population, n is the number of draws, k is the number of observed successes and $\binom{a}{b}$is a binomial distribution. The pmf is positive when max(0, n+K-N)≤k≤min(K,n).

Chapter 4

Results and Discussions

4.1 FPRAT

4.1.1 Sequence and structure homology profiles

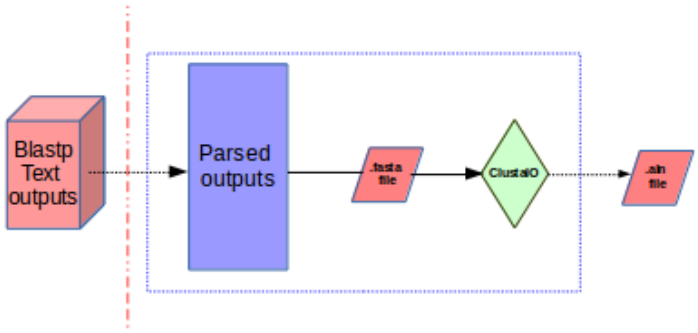


Figure 4.1: Module 1 workflow

The module1 (see figure 4.1) of FPRAT does two-level blast parsing to filter out the hits. Table 4.1 shows the outcome of the two-level parsing. This way the redundancies are brought to a minimum and further processing becomes less constrain. The threshold values set for parsing (E-value < = 1e-10, identity% >= 35% and Coverage% >=75%) gives a better hit percentage for the query sequences. After the parsing we still have a good number of samples for bth sequence (Seq) list and structure(Str)list.

Parsing level	SACOL		SAN315	
	Seq List	Str List	Seq List	Str List
Level 1	30896	6421	49616	8449
Level 2	28010	352	45032	436

Table 4.1: Blastp result

The alignment file prepared as input for the MSA has the query sequence, the sequence list and structure list in fasta format. This gives a better conservation of the residues. The figure 4.12 is the output from the Blastp parser.

4.1.2 Conserved region estimation

From the following figure of module 2 workflow (see figure 4.2), the focus is to find the conserved residues through applying the Rate4Site program to the query sequence. The red line in 4.3 depicts the distribution of Rate4Site scores of the query sequence from the alignment file. The categories of conserved and non conserved requires a certain cutoff on the scores. The 4.4 hence gives this cutoff. by normalizing the original scores to [0,1] and by picking the bin 1-3 as the most conserved list of residues , while others (bins 4-9) as non conserved list. The output conserved list from module 2 processing is shown in figure 4.14.

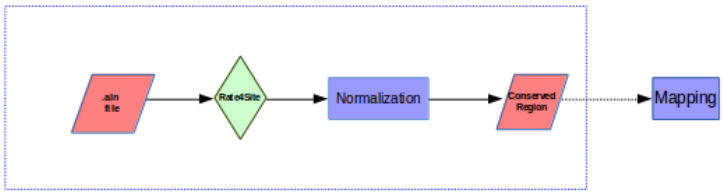


Figure 4.2: Module 2 workflow

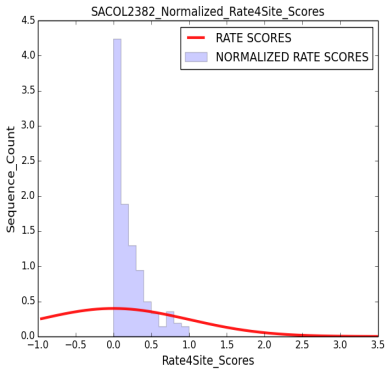


Figure 4.3: Rate4Site result

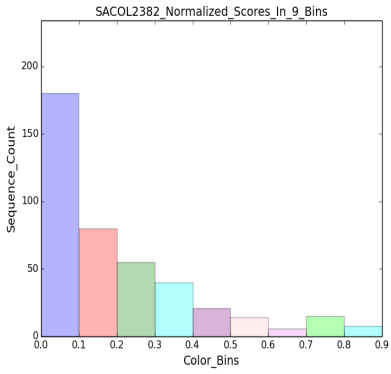


Figure 4.4: Color bins

4.1.3 FPRA

The following workflow (see figure 4.5) shows the functional module3 which comprises of the FPRA program. Here the structure homologous hits are subjected to the PROPORES program in order to find all the pores present in each of the

proteins. A default setting of PROPORES include [-r 1.0 -s 1.2 -c1.4], where r is the side length of the grid voxel, -s is the probe radius, -c is the trimming depth for shallow pore regions on the protein surface. This setting can be changed according to the size of the proteins. The results from PROPORES include 3 files namely, '.PTin','.pdb' and '.list' which are indexed from 0. The OPM files include the z-oriented PDB files which allows us to span over the TM region while running the FPRA program and filter the PLRs for each of the protein.

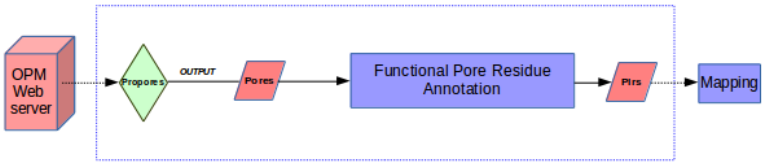
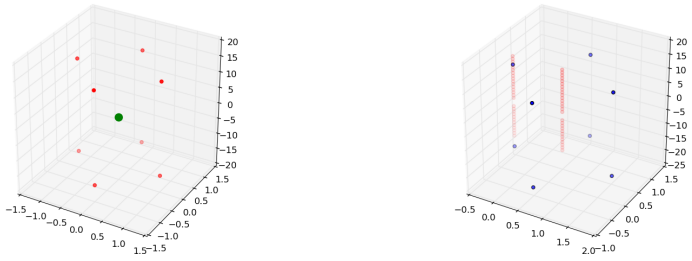


Figure 4.5: Module 3 workflow

PLRs prediction

After PROPORES program has been run over all the OPM files and the pore information has been generated, the FPRA program is run. FPRA has 3 parts. The task of part_1 is to generate the COM for the protein at the TM region. After COM is generated, the program spans two planes along z-axis, that is +15Å and -15Å (above and below) the COM and ±2Å along the x-,y-axis as shown in figure 4.6a. The imaginary box that is generated is used to filter out the pores. This is the task of part_2. Here the box scan through the pore information generated from PROPORES and filters out the pores that overlap the box (see figure 4.6b).



(a) Box Generated (b) Pore Cordinates Overlaping

Figure 4.6: PLRs prediction

The red dots in the figure indicate the pore coordinates that overlap the box.

Since the pore information is indexed, part_2 filters out the '.list' indexed file that contains the information about the PLRs. Now part_3 of the FPRA converts the residue information from three letter codes to single letter codes for mapping.

4.1.4 Mapping

This the final module of the FPRAT. As shown in the module workflow (see figure 4.7), we need the information from both module2 and module3 for mapping. Figure 4.8, shows the mapping of conserved list in row1 (annotated as 'c'), the PLRs list in row 2 (annotated as 'p') and the query sequence in row 3 under the PDB id for the protein 1XFH. Annotation of this sort helps in the easy identification of the conserved PLR and nonPLRs. The red boxes in the figure indicate the conserved PLRs which is the main purpose of this thesis.

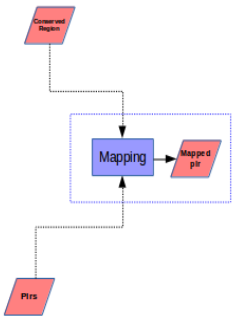


Figure 4.7: Mapping module

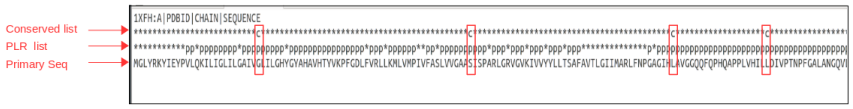
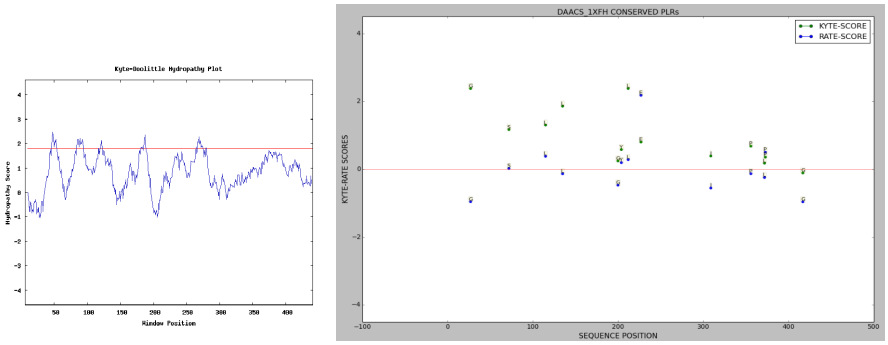


Figure 4.8: Mapped result

Once the conserved PLRs are identified through mapping, the hydrophobicity scores for these residues are checked using Kyte-Doolittle hydropathy scoring. This gives a better understanding on the position of these residues in the TM region of the transporter proteins. Kyte-Doolittle scoring givea a quantitative measure of the degree of hydrophobicity and is most useful in identifying possible domains of a protein structure. The figure4.9a shows the blue peaks, some of which are above the red line indicating the hydrohobic regions. A stretch of 20 amino acid residues in the plot with positive score indicates that they are a part of the alpha-helix spanning the lipid region.



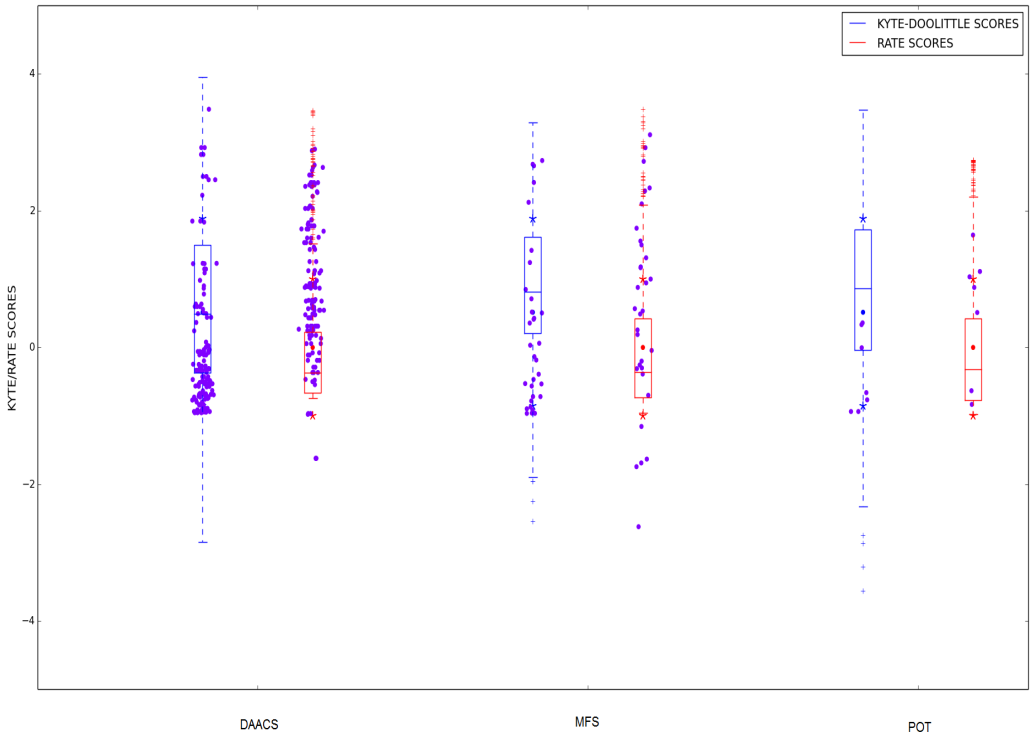
(a) Kyte-Doolittle hydrophobicity scoring

(b) Kyte-Rate scores scatter plot

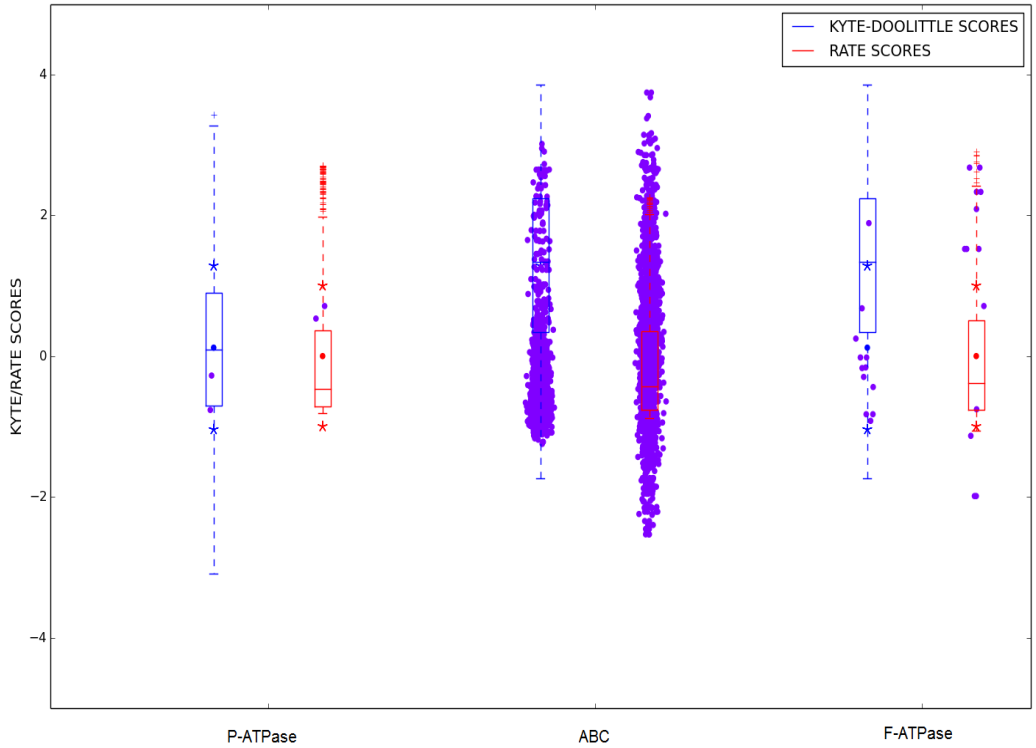
Figure 4.9: Kyte-Rate plots

The figure 4.9b is a scatter plot of the rate and Kyte-Doolittle scores for the conserved residues. The range of Kyte-Doolittle scores is from [-4.5,4.5], 4.5 being the most hydrophobic residue and -4.5 being the most hydrophilic residue. In the case of 1XFH protein, we can see that almost all the green dots (indicating the Kyte-Doolittle scores) fall above the red line and are positive meaning they are hydrophobic in nature.

From figure 4.10, we can see two different boxplots, the red boxplot represents the Rate4Site scores while the blue boxplot represents the Kyte-Doolittle scores for each of the query sequences. Now it is expected that the PLRs generally behave to have lower rate scores, meaning these residues are more conserved and also have a larger hydrophobicity score meaning that they are nonpolar in nature. If we see the distribution of the conserved PLRs for the larger protein family ABC , the cloud of mapped conserved PLRs tend to be in the range of -1 to 1 for hydrohobicity. These residues can form the part of the alpha helical region and are hence hydrophobic. Now if we consider the rate scores, we see a considerable cloud in the lower scores, which means that the conservation is high for these residues. This is in par with our argument. Also, for other families the sample size of the conserved PLRs are very small and its less significant to draw a conclusion from the graph. In the case of families like POT, P-ATPase and F-ATPase, the number of samples are too low for supporting our expectation. We also see that the rate scores for some of the PLRs are very high meaning that they might not be conserved as we expect. This anomaly can be reasoned with the initial setting for PROPORES in identifying the Pores and PLRs list. These residues might be located on the peripheral region which are usually not conserved in the case of transporter proteins. In the case for DAACS family, the cloud of PLRs tend to be in negative score for hydrobobicity and towards lower rate scores. Hence from the two qualitative graphs we can draw a conclusion that there are fewer conserved PLRs which actually are nonpolar and more conserved.



(a) Smaller protein families



(b) Larger protein families

Figure 4.10: Hydrophobicity Vs Conservation

4.2 Degree Of Conservation(DOC)

The evolutionarily conserved amino acids are often important and associated with the structure and function of the protein. The DOC gives this measure for each of the protein families. Hence, through conservation analysis of amino acid positions among transporter family members , we can often reveal the importance of each position for the protein structure or function.

The DOC for the family of transporter families was calculated as follows:

$$DOC = \sum conservedPLRs / \sum conserved$$

Now from both the tables 4.2 and 4.3, we see that most of the protein families have more than 50% DOC values for conserved PLRs. If we see the MFS family, the DOC is no the lesser side. But this means that the helical region that surrounds the functional domain of the transporter proteins contains both conserved PLRs and non-PLRs, with the region of helix with non-PLRs more conserved. Certain proteins in the ABC family like 3DHW, 3D31,4HUQ,4HZU.. have 100% DOC while other proteins like 3WME and 4F4C have 0% DOC. The mean of DOC for each family is computed and hypothesis testing is done to see if the functional regions are conserved or not. A comparison between the mean DOC for both the sets was done through the families (see figure 4.15).

No	Family	trans_ID	Protein Name	Frac of Cons_Ps	Frac of Cons_NPs	Mean DOC Cons_ Ps	Mean DOC Cons_ NPs
1	DAACS	SA2172	1XFH	0.64	0.36	0.55	0.45
			2NWL	0.21	0.79		
			3KBC	0.54	0.46		
		SACOL2382	3V8F	0.54	0.46		
			4KY0	0.81	0.19		
			4P19	0.55	0.45		
2	MFS	SA0325	1PW4	0.33	0.67	0.33	0.67
		SACOL0407					
3	POT	SA0682	4APS	0.57	0.43	0.51	0.49
		SACOL0788	4IKV	0.44	0.56		

Table 4.2: DOC for smaller protein families

No	Family	trans_ID	Protein Name	Frac of Cons_Ps	Frac of Cons_NPs	Mean DOC Cons_Ps	Mean DOC Cons_NPs
1	ABC	SA0110	1L7V	0.82	0.08	0.08	0.82
		SA0137	3DHW	1	0	0	1
		SA0166	3D31	0.93	0.07	0.085	0.915
			3FH6	0.9	0.1		
		SA0192	3DHW	0.76	0.24	0.29	0.29
			3FH6	0.66	0.34		
		SA0206	3FH6	0.93	0.07	0.07	0.93
		SA0209	3FH6	0.73	0.27	0.27	0.73
		SA0297	3DHW	0.78	0.22	0.22	0.78
		SA0420	3DHW	0.91	0.09	0.09	0.91
		SA0421	3DHW	0.72	0.28	0.28	0.72
		SA0599	2HYD	0.86	0.14	0.14	0.86
		SA0603	4G1U	0.89	0.11	0.11	0.89
		SA0769	3DHW	0.91	0.09	0.09	0.91
		SA0770	3DHW	0.8	0.2	0.2	0.8
		SA0888	2ONK	0.89	0.11	0.13	0.87
			3D31	1	0		
			3FH6	0.72	0.28		
		SA0981	4G1U	0.89	0.11	0.11	0.89
		SA1674	3DHW	0.93	0.07		
			3FH6	0.85	0.15	0.31	0.69
			3G5U	0.17	0.83		
			4HUQ	1	0		
			4HZU	1	0		
			4M1M	0.17	0.83		
		SA1683	2HYD	0.93	0.07	0.36	0.64
			3B60	0.91	0.09		
			3QF4	0.09	0.91		
		SA1977	4G1U	0.73	0.27	0.27	0.73
		SA1978	4G1U	0.78	0.22	0.22	0.78
		SA2019	4HUQ	0.38	0.62	0.37	0.63
			4HZU	0.88	0.12		
		SA2020	4HUQ	0.89	0.11	0.1	0.9
			4HZU	0.91	0.09		
		SA2021	3DHW	0.95	0.05	0.408	0.60
			3G5U	0.13	0.87		
			4HUQ	0.83	0.17		
			4HZU	0.92	0.08		
			4M1M	0.13	0.87		
		SA2072	2ONK	0.89	0.11	0.12	0.88
			3D31	0.86	0.14		

No	Family	trans_ID	Protein Name	Frac of Cons_Ps	Frac of Cons_NPs	Mean DOC Cons_ Ps	Mean DOC Cons_ NPs
1	ABC	SA2200	3DHW	1	0	0.89	0.11
			3FH6	0.94	0.06		
			4HUQ	0.8	0.2		
			4HZU	0.82	0.18		
		SA2251	3DHW	0.87	0.13	0.87	0.13
		SA2416	3DHW	0.95	0.05	0.89	0.11
			3FH6	0.87	0.13		
			4HUQ	0.86	0.14		
			4HZU	0.87	0.13		
		SAP022	3FH6	0.9	0.1	0.9	0.1
		SACOL0098	1L7V	0.9	0.1	0.9	0.1
		SACOL0127	3DHW	1	0	1	0
		SACOL0192	3FH6	0.93	0.07	0.93	0.07
		SACOL0306	3DHW	0.78	0.22	0.78	0.22
		SACOL0504	3DHW	0.91	0.09	0.91	0.09
		SACOL0882	3DHW	0.91	0.09	0.91	0.09
		SACOL1040	2ONK	0.9	0.1	0.81	0.19
			3FH6	0.71	0.29		
		SACOL1040	3DHW	0.93	0.07	0.59	0.41
			3FH6	0.85	0.15		
			3G5U	0.17	0.83		
			3WME	0	1		
			4HUQ	1	0		
			4HZU	1	0		
			4M1M	0.17	0.83		
		SACOL2210	4HUQ	0.9	0.1	0.9	0.1
			4HZU	0.9	0.1		
		SACOL2211	3DHW	0.95	0.05	0.47	0.53
			3G5U	0.11	0.89		
			4F4C	0	1		
			4HUQ	0.83	0.17		
			4HZU	0.83	0.17		
			4M1M	0.11	0.89		
		SACOL2270	2ONK	0.89	0.11	0.88	0.12
			3D31	0.86	0.14		
		SACOL2410	3DHW	1	0	0.89	0.11
			3FH6	0.94	0.06		
			4HUQ	0.8	0.2		
			4HZU	0.82	0.18		
		SACOL2462	3D31	0.92	0.08	0.92	0.08
		SACOL2472	3DHW	0.87	0.13	0.87	0.13

No	Family	trans_ID	Protein Name	Frac of Cons_Ps	Frac of Cons_NPs	Mean DOC Cons_Ps	Mean DOC Cons_NPs
1	ABC	SACOL2644	3DHW	0.86	0.14	0.92	0.08
			3FH6	1	0		
			4HUQ	0.91	0.09		
			4HZU	0.9	0.1		
2	F-ATPase	SA1910	2W5J	0.7	0.3	0.80	0.20
		SACOL2100	2WIE	1	0		
			3V3C	0.7	0.3		
3	P-ATPase	SA2344	3RFU	0	1	0.03	0.97
		SACOL2572	4BBJ	0.06	0.94		

Table 4.3: DOC for larger families

4.3 Discussions

4.3.1 Hypothesis tesisng: P-value Assumption

Our hypothesis : PLRs are functionally important. Here we expect that they are more conserved than the rest. Here, we focus only on residues which are conserved. Expect that the PLRs have a higher fraction of conserved residues than their proporetion in the sequence. The P-value for conserved PLRs and non-PLRs are computed byWilcoxcon signed-ranked test . The P-value for conserved PLRs is 3.45001e-10 which is much lower than the expected 0.05 significance. This shows that both the sets are different.

4.4 Hypergeometric testing

Here we set N as the total length of the protein sequence, K as the number of PLRs, n as the number of conserved residues through the length of the protein sequence and k as the number of conserved PLRs throught he length of the protein sequence. From figure 4.11, we see that the P-values for each of the proteins are very high proving less significance in the test as p-value need to be a low value.

This is insufficient to support the hypothesis. This can mainly due to the following reasons:

- very low balance between the conserved data set and PLR data set.
- The initial setting on PROPORES for the PLR prediction. The grid settings have to be proper in order to predict the PLRs in a much effective way.
- No difference in the protein sequence of the transporters of both the strains SN315 and SACOL obtained from the transportDB database.

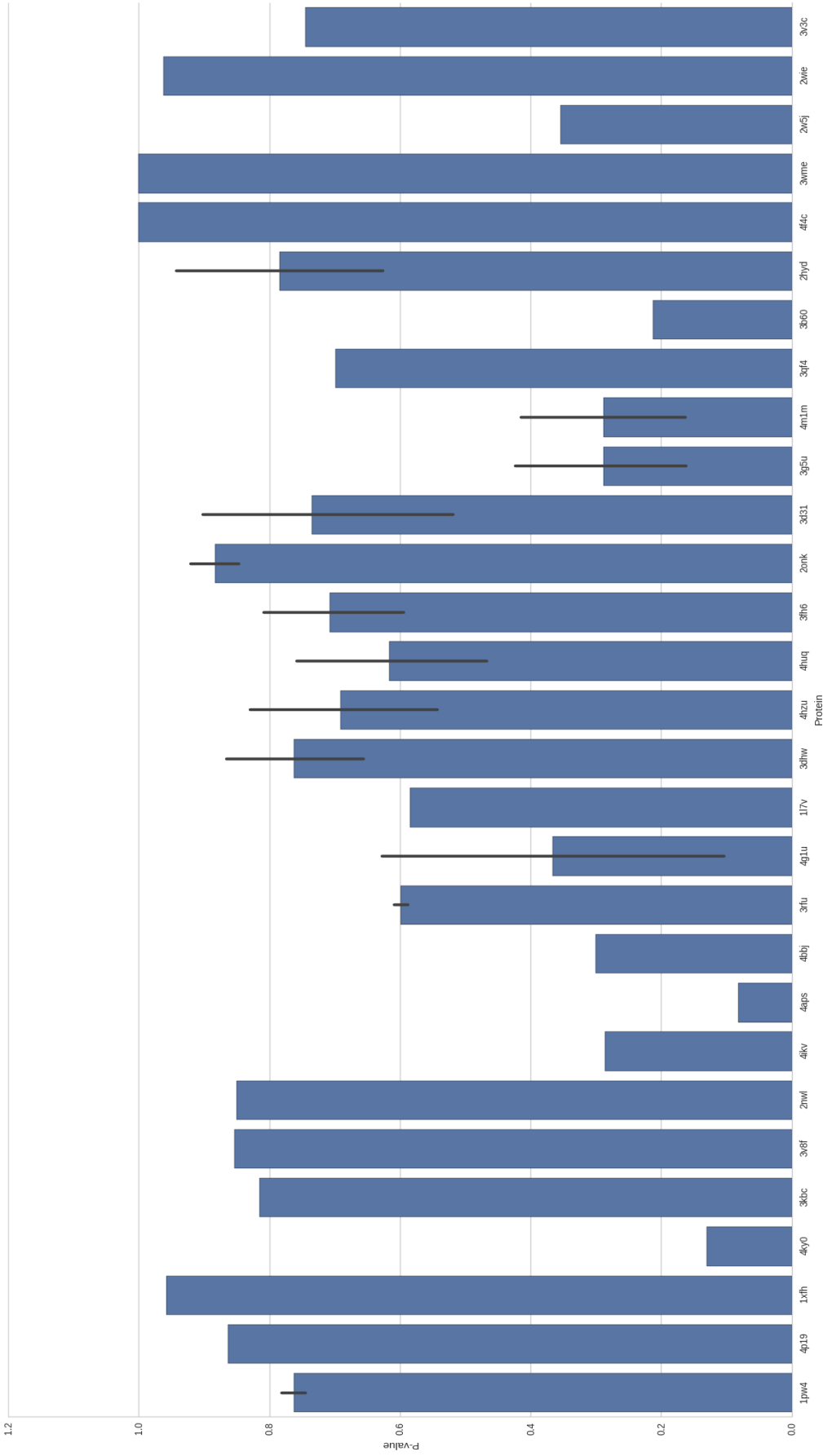


Figure 4.11: Hypergeometric test

Appendix

This section provides detailed information of the dataset and some background knowledge regarding amino acids and substrates mentioned in previous chapters.

transportDB_id	PDB	Bit Score	E-value	Identity%	Positives%	Gaps%	Gap number	Alignment len	cov %
SACOL2644	3DHW:G	150.0	8e-44	35	59	4	9	249	94.49
SACOL2644	3DHW:D	150.0	8e-44	35	59	4	9	249	94.49
SACOL2644	3DHW:C	150.0	8e-44	35	59	4	9	249	94.49
SACOL2644	4KHZ:B	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	4KHZ:A	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	4JBW:D	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	4JBW:C	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	4JBW:B	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	4JBW:A	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	3RLF:B	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	3RLF:A	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	3PV0:B	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	3PV0:A	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	3FH6:A	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	3FH6:B	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	3FH6:C	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	3FH6:D	141.0	2e-40	35	58	5	12	219	81.5
SACOL2644	2R6G:B	140.0	6e-40	35	58	5	12	219	81.5
SACOL2644	2R6G:A	140.0	6e-40	35	58	5	12	219	81.5
SACOL2644	4HZU:B	101.0	2e-26	35	52	3	7	225	85.83
SACOL2644	4HUQ:A	101.0	2e-26	35	52	3	7	225	85.83
SACOL2462	3D31:B	105.0	9e-28	35	52	5	12	221	95.0
SACOL2462	3D31:A	105.0	9e-28	35	52	5	12	221	95.0
SACOL2462	4AYT:A	102.0	6e-26	35	52	7	13	200	85.0
SACOL1040	4KHZ:B	98.2	2e-25	35	51	4	9	212	95.31
SACOL1040	4KHZ:A	98.2	2e-25	35	51	4	9	212	95.31
SACOL1040	4JBW:D	98.2	2e-25	35	51	4	9	212	95.31
SACOL1040	4JBW:C	98.2	2e-25	35	51	4	9	212	95.31
SACOL1040	4JBW:B	98.2	2e-25	35	51	4	9	212	95.31
SACOL1040	4JBW:A	98.2	2e-25	35	51	4	9	212	95.31
SACOL1040	3RLF:B	98.2	2e-25	35	51	4	9	212	95.31
SACOL1040	3RLF:A	98.2	2e-25	35	51	4	9	212	95.31
SACOL1040	3PV0:B	98.2	2e-25	35	51	4	9	212	95.31
SACOL1040	3PV0:A	98.2	2e-25	35	51	4	9	212	95.31

Figure 4.12: Blast Parser output

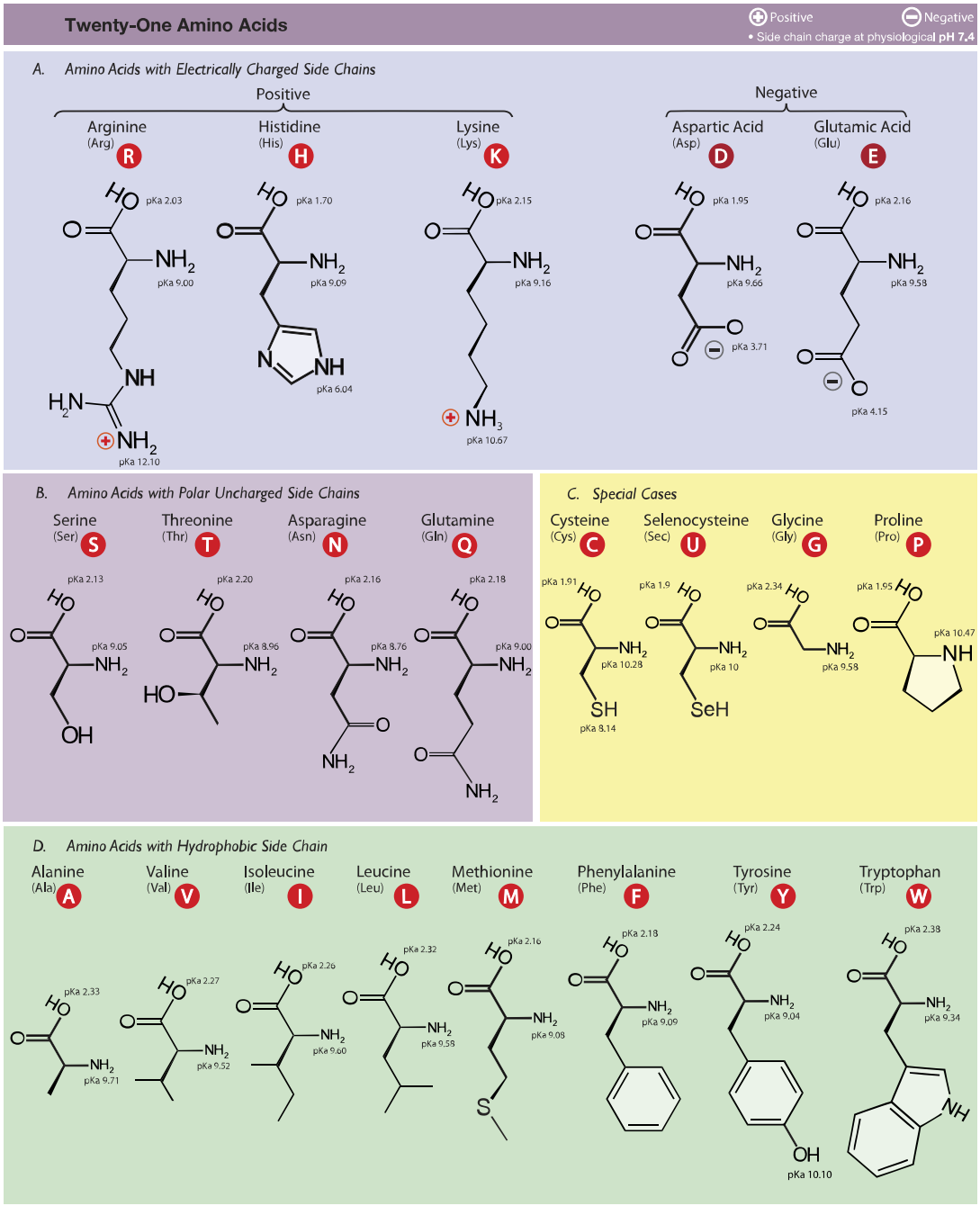


Figure 4.13: Amino acid chart. Figure taken from[3]

```
A    ['70', '413', '233', '36', '358', '147', '24', '353', '381', '116', '218', '71', '360', '384', '94', '145', '211', '127', '38', '214', '391',
'229', '164', '364', '328', '307', '410', '384', '218', '71', '381', '145', '164', '205', '391', '214', '38', '410', '364', '70', '358', '233', '147',
'380', '360', '116', '353', '127', '211', '94', '229', '328', '307', '413', '36', '24', '211', '127', '145', '94', '218', '71', '360', '384', '381',
'116', '353', '410', '364', '328', '307', '391', '164', '214', '38', '358', '36', '348', '413', '70', '380', '147', '186', '64', '147', '147', '64',
'186', '186', '147', '145']
E    ['227', '291', '296', '181', '219', '192', '172', '172', '227', '291', '174', '219', '192', '181', '296', '192', '219', '172', '291', '227',
'296', '181']
D    ['390', '267', '257', '394', '48', '185', '136', '48', '136', '185', '267', '257', '394', '390', '394', '257', '136', '185', '48', '267',
'390']
G    ['354', '27', '297', '34', '82', '351', '144', '417', '357', '118', '79', '293', '119', '396', '47', '200', '112', '280', '226', '221', '69',
'388', '149', '31', '110', '200', '47', '19', '221', '69', '110', '27', '354', '82', '118', '119', '112', '396', '226', '388', '31', '149', '297',
'357', '144', '351', '417', '293', '79', '226', '162', '280', '47', '200', '112', '19', '110', '31', '149', '388', '69', '82', '34', '354', '297',
'119', '79', '293', '118', '144', '351', '417', '357']
F    ['248', '46', '301', '210', '273', '156', '122', '324', '261', '156', '273', '95', '261', '324', '301', '210', '122', '248', '46', '122',
'156', '95', '261', '324', '46', '210', '301', '63', '143', '157', '157', '143', '63', '63', '157', '143']
I    ['21', '361', '25', '160', '207', '133', '309', '389', '243', '157', '113', '159', '155', '163', '256', '197', '16', '294', '262', '73', '213',
'137', '350', '392', '298', '159', '155', '18', '243', '294', '16', '73', '392', '25', '21', '207', '160', '29', '85', '256', '197', '163', '113',
'262', '350', '137', '213', '298', '361', '133', '389', '309', '294', '16', '85', '256', '197', '163', '155', '159', '113', '298', '350', '73', '137',
'160', '207', '361', '165', '309', '133', '187', '187', '187']
H    ['37', '114', '40', '368', '32', '132', '264', '223', '125', '223', '37', '32', '125', '264', '132', '114', '40', '368', '223', '264', '125',
'132', '40', '114', '37', '368']
K    ['290', '55', '44', '263', '175', '414', '266', '15', '196', '15', '414', '263', '175', '178', '266', '196', '55', '290', '44', '175', '196',
'15', '414', '55', '290', '263', '44']
M    ['362', '292', '399', '395', '269', '365', '59', '385', '202', '217', '169', '56', '365', '399', '292', '269', '395', '385', '59', '202',
'217', '56', '169', '362', '202', '385', '59', '365', '169', '56', '292', '362']
L    ['372', '347', '91', '13', '367', '57', '168', '374', '305', '28', '53', '115', '152', '212', '244', '90', '239', '184', '270', '135', '78',
'130', '282', '17', '343', '30', '20', '134', '228', '387', '49', '244', '90', '78', '228', '30', '91', '13', '347', '57', '367', '305', '374', '168',
'152', '212', '115', '28', '53', '406', '239', '130', '135', '270', '184', '49', '387', '20', '134', '343', '17', '282', '22', '90', '212', '152',
'115', '53', '228', '387', '363', '66', '49', '343', '30', '282', '134', '130', '135', '78', '184', '91', '13', '347', '22', '372', '374', '168',
'305', '57', '367', '161', '183', '146', '190', '146', '190', '161', '183', '183', '161', '190', '146']
N    ['173', '148', '188', '170', '199', '141', '141', '170', '188', '148', '199', '173', '141', '199', '173', '188', '170', '148', '310']
Q    ['120', '220', '123', '126', '150', '121', '203', '203', '150', '121', '120', '220', '126', '123', '121', '203', '220', '120', '150', '126',
'123']
```

Figure 4.14: Conserved list

No:	Family	transportDB	PDBid
1	ABC	SA1674	3WME
		SA1683	4Q4H
		SA0166	2R6G
		SA0206	
		SA0209	
		SA0888	
		SA1674	
		SA2200	
		SA2416	
		SACOL0192	
		SACOL1040	
		SACOL1915	
		SACOL2410	
		SACOL2644	
		SA0166	3PV0
		SA0192	
		SA0206	
		SA0209	
		SA0888	
		SA1674	
		SA2200	
		SA2416	
		SACOL0192	
		SACOL1040	
		SACOL1915	
		SACOL2410	
		SACOL2644	
		SAP022	

No:	Family	transportDB	PDBid
1	ABC	SA0192	3RLF
		SA0166	
		SA0206	
		SA0209	
		SA0888	
		SA1674	
		SA2200	
		SA2416	
		SACOL0192	
		SACOL1040	
		SACOL1915	
		SACOL2410	
		SACOL2644	
		SAP022	
		SA0166	4JBW
		SA0192	
		SA0206	
		SA0209	
		SA0888	
		SA1674	
		SA2200	
		SA2416	
		SACOL0192	
		SACOL1040	
		SACOL1915	
		SACOL2410	
		SACOL2644	
		SAP022	

No:	Family	transportDB	PDBid
		SA0166	4KHZ
		SA0192	
		SA0206	
		SA0209	
		SA0888	
		SA1674	
		SA2200	
		SA2416	
		SACOL0192	
		SACOL1040	
		SACOL1915	
		SACOL2410	
		SACOL2644	
		SAP022	
		SA0950	3FVQ
		SA1674	
		SA2072	
		SA2200	
		SACOL1040	
		SACOL1108	
		SACOL1915	
		SACOL2270	
		SACOL2410	
		SACOL2644	
2	SSPTS	SA1255	4JBW
		SACOL1457	
3	TrK	SA1815	4J7C
		SACOL2011	

No:	Family	transportDB	PDBid
1	F-ATPase	SA1907	4DLO
		SACOL2097	
		SA1905	1BMF
		SA1907	
		SACOL2095	
		SACOL2097	
		SA1905	1COW
		SA1907	
		SACOL2095	
		SACOL2097	
		SA1905	1ERF
		SA1907	
		SACOL2095	
		SACOL2097	
		SA1905	1Q01
		SA1907	
		SACOL2095	
		SACOL2097	

No:	Family	transportDB	PDBid
1	F-ATPase	SA1905	2CK3
		SA1907	
		SACOL2095	
		SACOL2097	
		SA1905	2HLD
		SA1907	
		SACOL2095	
		SACOL2097	
		SA1905	2JDI
		SA1907	
		SACOL2095	
		SACOL2097	
		SA1905	2WPD
		SA1907	
		SACOL2095	
		SACOL2097	

No:	Family	transportDB	PDBid
1	F-ATPase	SA1905	2WSS
		SA1907	
		SACOL2095	
		SACOL2097	
		SA1905	2XND
		SA1907	
		SACOL2095	
		SACOL2097	
		SA1905	3OAA
		SA1906	
		SA1907	
		SACOL2095	
		SACOL2096	
		SACOL2097	
		SA1905	3ZIA
		SA1907	
		SACOL2095	
		SACOL2097	

Table 4.4: Unprocessed Larger proteins

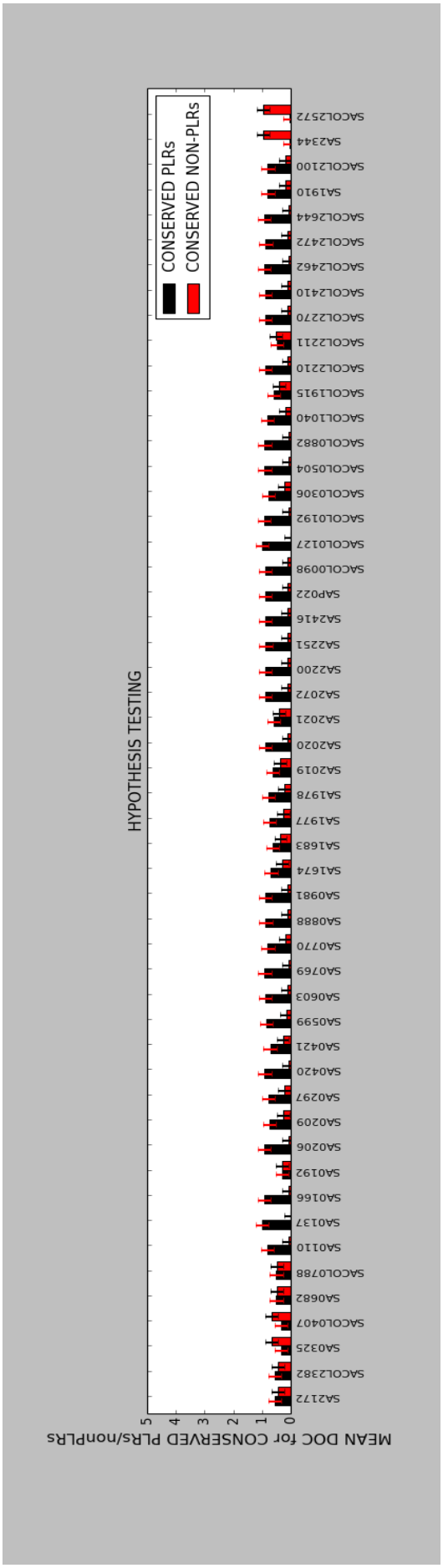


Figure 4.15: Comparison of mean DOC between families

[1] URL http://www.proteinatlas.org/humanproteome/secretome.

[2] URL http://www.sci-news.com/.

[3] URL https://en.wikipedia.org/wiki/Amino_acid.

[4] URL https://en.wikipedia.org/wiki/P-value#/media/File:P-value_in_statistical_significance_testing.svg.

[5] URL http://image.slidesharecdn.com/hypothesistestingandp-valueeyenirvaan-195/hypothesis-testing-and-pvalue-wwweyenirvaancom-15-638.jpg?cb=1387888274.

[6] URL https://en.wikipedia.org/wiki/Cell_membrane.

[7] URL http://themedicalbiochemistrypage.org/membranes.php.

[8] Ying-Ying Huang Sanjay M Jachak D Mariano A Vera Proma Khondkar Simon Gibbons Michael R Hamblin George P Tegos Christina Kourtesi, Anthony R Ball. Microbial efflux systems and inhibitors: Approaches to drug discovery and the challenge of clinical implementation. 2013.

[9] Sharp KA. Coleman RG. Finding and characterizing tunnels in macromolecules with application to ion channels and pores. 2009.

[10] Rhonda J. S. David W. B, John W. H. Introduction to chemistry: General, organic, and biological. 2012.

[11] Thornton JM. Eyre TA1, Partridge L. Computational analysis of alpha-helical membrane protein structure: implications for the prediction of 3d structural models. 2004.

[12] David Dineen Toby J Gibson Kevin Karplus Weizhong Li Rodrigo Lopez Hamish McWilliam Michael Remmert Johannes So ding Julie D Thompson Fabian Sievers, Andreas Wilm and Desmond G Higgins. Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. 2011.

[13] David B James D et al. Harvey L, Arnold B. Molecular cell biology. 4th edition. 2000.

[14] S Lawrence Zipursky Paul Matsudaira David Baltimore Harvey Lodish, Arnold Berk and James Darnell. *Molecular Cell Biology, 4th edition.*, volume ISBN-10: 0-7167-3136-3. New York: W. H. Freeman;, 2000.

[15] Barnickel G. Hendlich M, Rippmann F. Ligsite: automatic and efficient detection of potential small molecule-binding sites in proteins. 1997.

[16] Accelerating Protein Sequence Search in a Heterogeneous Computing System. Shucai xiao, heshan lin, wu-chun feng. Master's thesis.

[17] Aisling O' Driscoll Jurate Daugelaite and Roy D. Sleator. An overview of multiple sequence alignments and cloud computing in bioinformatics. 2013.

[18] Doolittle RF Kyte J. A simple method for displaying the hydropathic character of a protein. 1982.

[19] Helms V. Lee PH1. Identifying continuous pores in protein structures with propores by computational repositioning of gating residues. 2011.

[20] Banaszak LJ. Levitt DG. Pocket: a computer-graphics method for identifying and displaying protein cavities and their surrounding amino-acids. 1992.

[21] Janet M. Thornton. Marialuisa Pellegrini-Calace, Tim Maiwald. Porewalker: a novel tool for the identification and charecterization of channels in transmembrane pprotein from their three-dimensional structure. *PLoS Computational Biology.*, 2009.

[22] Irina D. Pogozheva2 Mikhail A. Lomize1, Andrei L. Lomize2 and Henry I. Mosberg. Opm: Orientations of proteins in membranes database. 2006.

[23] Lee PH Nguyen D1, Helms V. Primsiplr: prediction of inner-membrane situated pore-lining residues for alpha-helical transmembrane proteins. 2014.

[24] Saier MH Jr. Nikaido H1. Transport proteins in bacteria: common themes in their design. 1992.

[25] Koca J Otyepka M et al. Patrek M, Kosinova P. Mole: a voronoi diagram-based explorer of molecular chanels, pores and tunnel. *Structure: 15:1357-1363.*, 2007.

[26] Banas P Kosinova P Koca J Damborsky J. Petrek M, Otyepka M. Caver: a new tool to explore routes from protein clefts, pockets and cavities. 2006.

[27] Volkhard Helms. Po-Hsien Lee. Identifying continuous pores in protein structures with propores by computational repositoring of gating residues. *Proteins: Structure, Function, and Bioinformatics, 80:421-432.*, 2012.

[28] Mayrose I Glaser F Ben-Tal N. Pupko T1, Bell RE. Rate4site: an algorithmic tool for the identification of functional regions in proteins by surface mapping of evolutionary determinants within their homologues. 2002.

[29] Kandt C. Raunest M. dxtuber: detecting protein cavities, tunnels and clefts based on protein and solvent dynamics. 2011.

[30] C. Pedone. S. Galdiero, M. Galdiero. beta-barrel membrane bacterial proteins: structure, function, assembly and interaction with lipids. *Department of Biological Sciences, Division of Biostructures, University of Naples Federico II, CNR, Naples, Italy. Curr Protein Pept Sci. 8(1):63-82.*, 2007.

[31] David S Goodsell Teresa A Larsen, Arthur J Olson. Morphology of protein-protein interfaces. 1998.